dockerd: failed to start daemon: Devices cgroup isn't mounted #2104   New issue
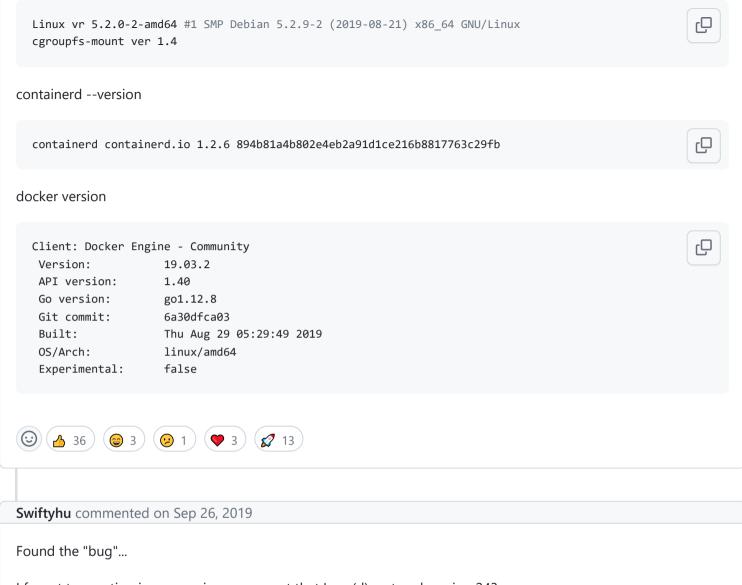
✓ Closed   **lyndalopez544** opened this issue on Sep 25, 2019 · 30 comments

lyndalopez544 commented on Sep 25, 2019 • edited ▾

Description

The problematic part of `dockerd` 's output:

```
WARN[2019-09-24T16:38:43.968826308Z] Your kernel does not support cgroup memory limit
WARN[2019-09-24T16:38:43.968902465Z] Unable to find cpu cgroup in mounts
WARN[2019-09-24T16:38:43.968935049Z] Unable to find blkio cgroup in mounts
WARN[2019-09-24T16:38:43.969017890Z] Unable to find cpuset cgroup in mounts
WARN[2019-09-24T16:38:43.969031008Z] mountpoint for pids not found
failed to start daemon: Devices cgroup isn't mounted
```

Output of `cgroupfs-mount` :

```
mount: /sys/fs/cgroup/cpu: cgroup already mounted on /sys/fs/cgroup/cpuset.
mount: /sys/fs/cgroup/blkio: cgroup already mounted on /sys/fs/cgroup/cpuset.
mount: /sys/fs/cgroup/memory: cgroup already mounted on /sys/fs/cgroup/cpuset.
mount: /sys/fs/cgroup/pids: cgroup already mounted on /sys/fs/cgroup/cpuset.
```

Output of `docker version` :

```
Client: Docker Engine - Community
Version:           19.03.2
API version:       1.40
```

```
Go version:        go1.12.8
Git commit:        6a30dfc
Built:             Thu Aug 29 05:29:29 2019
OS/Arch:           linux/amd64
Experimental:      false
Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?
```

Output of `docker info` :

```
Client:
 Debug Mode: false

Server:
 ERROR: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?
errors pretty printing info
```

Running physically on Debian 10 with kernel 5.3.0-rc5-amd64

😊 👍 31

Swiftyhu commented on Sep 26, 2019

I have the same issue.
However I found a workaround:

```
service docker stop
service containerd stop
cgroupfs-umount
cgroupfs-mount
service containerd start
service docker start
```

Sysinfo:

```
Linux vr 5.2.0-2-amd64 #1 SMP Debian 5.2.9-2 (2019-08-21) x86_64 GNU/Linux
cgroupfs-mount ver 1.4
```

containerd --version

```
containerd containerd.io 1.2.6 894b81a4b802e4eb2a91d1ce216b8817763c29fb
```

docker version

```
Client: Docker Engine - Community
 Version:           19.03.2
 API version:       1.40
 Go version:        go1.12.8
 Git commit:        6a30dfca03
 Built:             Thu Aug 29 05:29:49 2019
 OS/Arch:           linux/amd64
 Experimental:      false
```

👍 36   😄 3   🙁 1   ❤️ 3   🚀 13

Swiftyhu commented on Sep 26, 2019

Found the "bug"…

I forgot to mention in my previous comment that I use(d) systemd version 243.
With systemd 242 works everything flawlessly… :)

From the systemd changelog:

```
        * systemd now defaults to the "unified" cgroup hierarchy setup during
          build-time, i.e. -Ddefault-hierarchy=unified is now the build-time
          default. Previously, -Ddefault-hierarchy=hybrid was the default. This
          change reflects the fact that cgroupsv2 support has matured
          substantially in both systemd and in the kernel, and is clearly the
          way forward. Downstream production distributions might want to
```

```
continue to use -Ddefault-hierarchy=hybrid (or even =legacy) for
their builds as unfortunately the popular container managers have not
caught up with the kernel API changes.
```

Sooo... Houston, we have a problem:

1. systemd will (or already did) jump on the cgroupsv2 bandwagon...
2. cgroupfs-mount tools does not work with newer systemd setups.

"Same" issue in kubernetes

😊   👍 12   🙁 3

kj54321 commented on Oct 30, 2019 • edited ▾

Hi All,

```
    Seems issue still persist on latest Fedora 31, even with adding extra command
```

--exec-opt native.cgroupdriver=systemd

And per upstream workaround, which works temporarily with cgroup v1.

```
    However, the question lays on how do we address/fix this issue permanently as it's kernel API relate
```

Thanks

😊   👍 15   ❤️ 2

moughamir commented on Nov 15, 2019

@kj54321
Fixed by the workaround suggested, just add the argument `systemd.unified_cgroup_hierarchy=0` in grub

narfman0 commented on Dec 10, 2019

Very old fedora install here. I:

1. Added `GRUB_CMDLINE_LINUX="systemd.unified_cgroup_hierarchy=0"` to `/etc/default/grub`
2. Ran `grub2-mkconfig -o /boot/grub2/grub.cfg`
3. Ran `reboot`

and everything was happy

duckinator commented on Jan 1, 2020

Running this on Fedora 31, after an in-place upgrade from Fedora 30, works:

```
sudo grubby --args='systemd.unified_cgroup_hierarchy=0' --update-kernel /boot/vmlinuz-5.3.16-
300.fc31.x86_64
```

See a specific comment on the Fedora 31 bug report and the Fedora 31 documentation on using grubby.