Recently we found 'rbd map' hang forever but rbd device can work fine after terminating 'rbd map' or waiting it timed-out by using this change: http://tracker.ceph.com/issues/38792.

We tried to dig into it by adding some logs into krbd.cc and found that the order of subsystem returned by udev_device_get_subsystem might not be same order as adding subsystem by udev_monitor_filter_add_match_subsystem_devtype. So if subsystem 'block' is returned first and subsystem 'rbd' is returned next, then further poll will get nothing back and hang there.

So our fix is no matter which subsystem is returned first, we will deal with it properly. When both subsystems are returned, we will do the final check and compare. If something is wrong, it will still fall into next poll to be timed-out.

Related issues:
Related to Linux kernel client - Bug #14737: libkrbd vs udev event ordering
Copied to rbd - Backport #39314: luminous: krbd: fix rbd map hang due to udev...
Copied to rbd - Backport #39315: nautilus: krbd: fix rbd map hang due to udev...
Copied to rbd - Backport #39316: mimic: krbd: fix rbd map hang due to udev re...

History
#1 - 04/03/2019 09:50 AM - Zhi Zhang
- Pull request ID set to 27339

https://github.com/ceph/ceph/pull/27339

#2 - 04/03/2019 01:20 PM - Ilya Dryomov
- Related to Bug #14737: libkrbd vs udev event ordering added

#3 - 04/15/2019 08:41 AM - Ilya Dryomov
- Status changed from New to Pending Backport
- Assignee set to Zhi Zhang
- Backport set to luminous,mimic,nautilus

#4 - 04/16/2019 08:01 AM - Nathan Cutler
- Copied to Backport #39314: luminous: krbd: fix rbd map hang due to udev return subsystem unordered added

#5 - 04/16/2019 08:01 AM - Nathan Cutler
While running with --resolve-parent, the script "backport-create-issue" noticed that all backports of this issue are in status "Resolved" or "Rejected".