Linux kernel client - Bug #1907

rbd: don't reuse device ids while they're still in use elsewhere

01/09/2012 11:23 AM - Josh Durgin

Status:	Resolved	**% Done:**	0%
Priority:	Normal	**Spent time:**	0.00 hour
Assignee:	Alex Elder		
Category:	rbd		
Target version:	v3.3		
Source:	Development	**Reviewed:**	
Tags:		**Affected Versions:**	
Backport:		**ceph-qa-suite:**	
Regression:	No	**Crash signature:**	
Severity:	3 - minor		

Description

If an FS on top of rbd is mounted, and the rbd device is unmapped, and another one is mapped, the old sysfs entry is still around.
See http://permalink.gmane.org/gmane.comp.file-systems.ceph.devel/4876

History

#1 - 01/09/2012 03:51 PM - Sage Weil

- Target version set to v3.3

#2 - 01/13/2012 03:53 PM - Sage Weil

- Assignee set to Alex Elder

#3 - 01/18/2012 09:51 AM - Alex Elder

From the linked message:

```
root <at> cephnode3:/# rbd unmap /dev/rbd0
```

< -> works without any error message (should this work with a mounted
filesystem to /mnt ?)

I think the answer to this should be "no." But I may be mistaken.

Or if it is allowed, it should somehow notify the upper layers that
the device has disappeared.

Either way, the problem of reusing a name would never occur.
I'm not sure how to avoid reusing the name; if the name were
cleaned up out of sysfs when it got unmapped, the problem would
be avoided as well.

So my gut reaction is that we shouldn't avoid reusing device id's,
we should avoid device id's from persisting when they are no longer
meaningful.

I have to learn a bit more about this though. Here goes...

#4 - 01/18/2012 10:40 AM - Josh Durgin

Alex Elder wrote:

> From the linked message:
>
>> root <at> cephnode3:/# rbd unmap /dev/rbd0
>
> < -> works without any error message (should this work with a mounted
> filesystem to /mnt ?)
>
> I think the answer to this should be "no." But I may be mistaken.

I agree, but I'm not sure how easy this is to detect if you manually
use the kernel interface to remove the device instead of using 'rbd unmap', i.e.

```
echo 0 > /sys/bus/rbd/remove
```

> Or if it is allowed, it should somehow notify the upper layers that
> the device has disappeared.
>
> Either way, the problem of reusing a name would never occur.
> I'm not sure how to avoid reusing the name; if the name were
> cleaned up out of sysfs when it got unmapped, the problem would
> be avoided as well.
>
> So my gut reaction is that we shouldn't avoid reusing device id's,
> we should avoid device id's from persisting when they are no longer
> meaningful.

I like this approach better as well.

> I have to learn a bit more about this though. Here goes...

#5 - 01/18/2012 11:27 AM - Sage Weil

my gut feeling is also that 'echo > /sys/bus/rbd/remove' should return EBUSY (along with rbd unmap). if you can't tear down the file system, i'm not sure ripping the bdev out from underneath it is a good solution. wanna ask on #fsdevel or something?

#6 - 01/19/2012 07:51 AM - Alex Elder

I can ask on fsdevel, but right now I feel the need to understand a
little better what's going in inside rbd in order to even form
the question. I've been slowed a bit by network problems on this
other problem I'm trying to chase in the background though.

#7 - 02/08/2012 02:23 PM - Alex Elder

After a few weeks of wandering around the code, figuring out how
things work and refactoring and fixing things as I encounter
problems, I realized that the underlying problem was sort of
the next one on my list to address...

The problem has to do with the way unique identifiers for
rbd devices are selected. Each one gets a new id when it
gets created. The id used is one more than the highest
id already in use. When an rbd device is removed, its
id is released/put back for later reuse.

The problem is that the id is put back when a remove request
is made, but the underlying rbd_device doesn't actually go
away until the final reference on it is dropped. In the
case that was originally reported, a mounted filesystem held
that last reference to an rbd_device.

When a new attempt to map an rbd device was made, a new
id was allocated, resulting in the just-released id being
selected for reuse. This failed, however, at the point
where an attempt was made to create the entry in
/sys/bus/rbd/devices for that id. The entry for the
persisting rbd_device that held that id was not yet
gone (it won't go away until the final close). And
it is an error to attempt to hook a duplicate entry
into the bus's namespace in sysfs.

So **that** was the error.

The fix is to hold off marking an rbd id as available
for reuse until the final reference on the rbd_device
is dropped. I have a fix under test and expect to
commit it today.

#8 - 02/08/2012 02:26 PM - Alex Elder

- Status changed from New to 7

#9 - 02/24/2012 05:57 AM - Alex Elder

- Status changed from 7 to Resolved

- Source set to Development

Committed a couple of weeks ago and has seen no bad effect during the intervening testing. So I'm marking this one resolved.

Commit ID: ceph-client eda84b58922928516e6e62af85430b7c9705b6cf