### RADOS - Bug #38892

**/ceph/src/tools/kvstore_tool.cc:266:1: internal compiler error: Segmentation fault**

03/22/2019 04:46 PM - Sebastian Wagner

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
<th>% Done:</th>
<th>0%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Low</td>
<td>Spent time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Assignee:</td>
<td>Brad Hubbard</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Category:</td>
<td>Code Hygiene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Source:</td>
<td>Development</td>
<td>Affected Versions:</td>
<td></td>
</tr>
<tr>
<td>Tags:</td>
<td></td>
<td>ceph-qa-suite:</td>
<td></td>
</tr>
<tr>
<td>Backport:</td>
<td></td>
<td>Component(RADOS):</td>
<td></td>
</tr>
<tr>
<td>Regression:</td>
<td>No</td>
<td>Pull request ID:</td>
<td>26828</td>
</tr>
<tr>
<td>Severity:</td>
<td>3 - minor</td>
<td>Crash signature (v1):</td>
<td></td>
</tr>
<tr>
<td>Reviewed:</td>
<td></td>
<td>Crash signature (v2):</td>
<td></td>
</tr>
</tbody>
</table>

**Description**

```
[ 79%] Building CXX object src/tools/CMakeFiles/ceph-kvstore-tool.dir/kvstore_tool.cc.o
/home/jenkins-build/build/workspace/ceph-pull-requests/src/tools/kvstore_tool.cc:266:1: internal compiler error: Segmentation fault
```

Please submit a full bug report, with preprocessed source if appropriate.
Preprocessed source stored into /tmp/cc51Ovir.out file, please attach this to your bugreport.

- [https://jenkins.ceph.com/job/ceph-pull-requests/16170/consoleFull#14359466752a811ea2-3e7b-466b-84b4-d13df7e35809](https://jenkins.ceph.com/job/ceph-pull-requests/16170/consoleFull#14359466752a811ea2-3e7b-466b-84b4-d13df7e35809)

This is line 266 this time.

**Related issues:**

| Copied from RADOS - Bug #38633: /ceph/src/tools/kvstore_tool.cc:39:1: internal... | Resolved | 03/08/2019 |

**History**

**#1 - 03/22/2019 04:46 PM - Sebastian Wagner**

- Copied from Bug #38633: /ceph/src/tools/kvstore_tool.cc:39:1: internal compiler error: Segmentation fault on Bionic added

**#2 - 03/22/2019 04:48 PM - Sebastian Wagner**

- Description updated

**#3 - 03/22/2019 09:05 PM - Brad Hubbard**

While I was looking into this I noticed this warning in the Jenkins output.

```
JENKINS-19022: warning: possible memory leak due to Git plugin usage; see: https://wiki.jenkins-ci.org/display/JENKINS/Remove+Git+Plugin+BuildsByBranch+BuildData
```

I wonder whether this is a contributing factor to some of the issues we've seen?

This specific issue appears to be transient as telling Jenkins to try again appears to resolve it in most cases. This may indicate some form of resource contention as the cause.
If we see this again we could try temporarily adding "`--param ggc-min-expand=1 --param ggc-min-heapsize=0` to the compiler command line as that seems to prevent the segfault, at least in my testing.

- ggc-min-expand is the percentage of the gc heap expansion before reaching which, gc will perform collection. Its default value is $\min(30\% + 70\% \times (\text{RAM}/1\text{GB}), 100\%)$. Changing it to 1 will force gc to try to do gc very often -- it will try to do gc every time its heap size increases by 1%.

- ggc-min-heapsize is the absolute size of gc before it tries to perform collection. Its default value is $\min(\text{4MiB}, \text{RAM}/8, 128\text{MiB})$, where RAM is $\min(\text{prlimit} \rightarrow \text{--data} \rightarrow \text{rss})$. 0 will force GCC's gc to collect whenever possible no matter how small the heap is.

See https://github.com/ceph/ceph/pull/27479 for a viable workaround. Note that this is a bug in gcc7 [1] and the preferred fix is to move to gcc8 however I wanted to add that PR so people had access to the workaround for this, and any future occurrences.

[1] https://bugzilla.redhat.com/show_bug.cgi?id=1531154

- Status changed from New to 15
- Priority changed from High to Low

- Status changed from Fix Under Review to Closed