

Arvados - Story #7475

[Node manager] Better communication when job is unsatisfiable

10/07/2015 06:58 PM - Peter Amstutz

Status:	Resolved	Start date:	07/05/2017
Priority:	Normal	Due date:	
Assigned To:	Lucas Di Pentima	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	2017-08-02 sprint		
Description			
When a job cannot be satisfied by node manager, it will be queued forever with no feedback to the user (and almost no feedback to the admin, either). There are two distinct cases: 1) A job's min_nodes request is greater than node manager's configured max_nodes. In this case, node manager silently skips over the job with no feedback as to why no nodes are being started. 2) A job's resource requirements for a single node exceed the available cloud node size. In this case, the only indication this is a problem is a message of "job XXX not satisfiable" in the node manager log (and even then only if debug logging is turned on). If a job request cannot be satisfied under its current configuration, node manager should have some way of signaling this to the user.			
Subtasks:			
Task # 11759: Review 7475-nodemgr-unsatisfiable-job-comms			Resolved
Related issues:			
Related to Arvados - Bug #9354: [workbench] make quota errors available to th...		Closed	06/06/2016
Has duplicate Arvados - Feature #9023: If a component is not satisfiable, rep...		Duplicate	04/20/2016

Associated revisions

Revision c0e203e7 - 07/31/2017 02:52 PM - Lucas Di Pentima

Merge branch '7475-nodemgr-unsatisfiable-job-comms'
Closes #7475

Arvados-DCO-1.1-Signed-off-by: Lucas Di Pentima <lucas@curoverse.com>

History

#1 - 10/07/2015 07:08 PM - Peter Amstutz

- Tracker changed from Bug to Story
- Description updated

#2 - 10/07/2015 07:15 PM - Brett Smith

This can't just be Node Manager's job though, right? The system needs to know what Node Manager is willing to do, but any of these problems can also arise on static clusters that aren't even running Node Manager.

#3 - 10/07/2015 07:23 PM - Peter Amstutz

Yes, that's true. I think the right long term solution is for crunch v2 to combine the jobs of crunch-dispatch and node manager into one process, because otherwise neither process has quite enough information to be able to tell the user what's actually going on.

In the short term, there's still benefit in making incremental improvements to node manager for cloud installs.

#4 - 02/02/2016 05:07 PM - Tom Clegg

It seems like Nodemanager should emit a log (with object_uuid == job uuid) and cancel the job.

If we start telling crunch-dispatch whether nodemanager is running, in cases where nodemanager *isn't* running, crunch-dispatch could emit a log and cancel the job if it's unsatisfiable with the current set of (alive?) slurm nodes.

Short of running nodemanager on static clusters (add a slurm driver?) it seems like we need the logic in both places if we want to fix the bug in both types of install.

#5 - 04/26/2017 08:38 PM - Tom Morris

- Target version set to 2017-05-24 sprint

#6 - 05/09/2017 07:07 PM - Tom Morris

- Story points set to 1.0

#7 - 05/09/2017 07:12 PM - Tom Clegg

For crunch2, when node manager is not in use, sbatch rejects unsatisfiable jobs and the user gets an error -- however, crunch-dispatch-slurm will keep retrying forever. This infinite-retry problem will be mostly addressed by [#9688](#), but ideally crunch-dispatch-slurm should also recognize the "unsatisfiable job" error as a non-retryable error, and tell the API server that it won't be re-attempted (if crunch-dispatch-slurm assumes/knows that it is the only dispatcher, it can indicate this by cancelling the container).

#8 - 05/10/2017 07:44 PM - Tom Morris

- Target version changed from 2017-05-24 sprint to 2017-06-07 sprint

#9 - 05/24/2017 06:59 PM - Lucas Di Pentima

- Assigned To set to Lucas Di Pentima

#10 - 06/07/2017 06:21 PM - Lucas Di Pentima

- Target version changed from 2017-06-07 sprint to 2017-06-21 sprint

#11 - 06/19/2017 07:52 PM - Lucas Di Pentima

- Status changed from New to In Progress

#12 - 06/21/2017 06:25 PM - Lucas Di Pentima

- Target version changed from 2017-06-21 sprint to 2017-07-05 sprint

#13 - 07/03/2017 08:14 PM - Lucas Di Pentima

Updates @ [3dad67f27](#)

Test run: <https://ci.curoverse.com/job/developer-run-tests/376/>

Modified ServerCalculator.servers_for_queue() so that it also returns a dict with information about unsatisfiable jobs that should be cancelled by its caller.

Updated some tests that started failing because of this change.

New tests pending.

#14 - 07/05/2017 02:31 PM - Lucas Di Pentima

New updates at [f77d08dd5](#)

Test run: <https://ci.curoverse.com/job/developer-run-tests/377/>

- Enhanced error checking when trying to emit a log and cancel an unsatisfiable job.
- Added test cases.

#15 - 07/05/2017 03:56 PM - Peter Amstutz

7475-nodemgr-unsatisfiable-job-comms @ [f77d08dd57a1021525717c8669296eb3e463c5f7](#)

- In `_got_response`, the `uuid` can be either a job or a container. It needs to look at the `type` field of the `uuid`. This is only valid if the `uuid` is for a job:

```
self._client.jobs().cancel(uuid=job_uuid).execute()
```

If the `uuid` is for a container and `self.slurm_queue` is true, it should do this:

```
subprocess.check_call(['scancel', '--name='+uuid])
```

This may require a stub to ensure that tests don't try to call the real `scancel`.

I'd like to see an integration test, if it isn't too much work. Upon seeing the log message about an unsatisfiable job/container, it should check that (a) the expected log message was added and (b) the job was cancelled/scancel was called.

#16 - 07/05/2017 06:43 PM - Lucas Di Pentima

- Target version changed from 2017-07-05 sprint to 2017-07-19 sprint

#17 - 07/06/2017 09:14 PM - Lucas Di Pentima

Updates at [f507162f3](#)

Test run: <https://ci.curoverse.com/job/developer-run-tests/378/>

