Arvados - Bug #11626

[Crunch2] Still not propagating slurm errors to user

05/05/2017 12:27 PM - Peter Amstutz

Status:	Resolved	**Start date:**	05/08/2017
Priority:	Normal	**Due date:**	
Assigned To:	Peter Amstutz	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:	2017-05-24 sprint		

Description

If a user submits a job and slurm rejects it, the slurm error should be propagated to the container log so the user can see what is wrong.

Subtasks:

Task # 11633: Review 11626-crunch2-log-slurm-errors	**Resolved**

Related issues:

Related to Arvados - Bug #11669: [Crunch2] crunch-dispatch-slurm hits scancel...	**In Progress**	**05/10/2017**

Associated revisions

Revision a309f8c5 - 05/11/2017 02:19 PM - Peter Amstutz

Merge branch '11626-crunch2-log-slurm-errors' closes #11626

History

#1 - 05/05/2017 12:28 PM - Peter Amstutz

- Description updated

#2 - 05/08/2017 01:25 PM - Peter Amstutz

- Target version set to 2017-05-10 sprint

#3 - 05/08/2017 01:25 PM - Peter Amstutz

- Status changed from New to In Progress

#4 - 05/08/2017 01:26 PM - Peter Amstutz

- Assigned To set to Peter Amstutz

#5 - 05/08/2017 07:55 PM - Tom Clegg

I'm not sure why the HasUUID() check is added in run() in crunch-dispatch-slurm.go. ctx.Done() fires as soon as HasUUID() returns false, so it seems like it could only make a difference during a race between <-status and <-ctx.Done() -- the rest of the time it just introduces an unnecessary polling delay between dispatch telling crunch-dispatch-slurm to cancel and c-d-s actually running scancel.

Also not sure why we're adding the extra call to tracker.update(c) before tracker.close(). The dispatch library is supposed to reduce the amount of state/priority logic each dispatcher has to implement -- if the idea here is just to log every state change, perhaps it should go in dispatch.go instead?

event_type="dispatch" might be better than "crunch-dispatch-slurm" for filtering.

In TestIntegrationNormal, why is the *bool better than a bool?

#6 - 05/09/2017 08:43 PM - Peter Amstutz

However I've rebased out the HasUUID changes so we can discuss it separately.

Now at @ f8675ad473b45387b1286c6b7a41edf36148ebac

(note-5 refering to 26d7e1809376534b9060534e25399cbd462f38da)

Hhe problem I was trying to solve was that the logs are full of this:

```
2017-05-08_14:09:51.64626 2017/05/08 14:09:51 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:09:52.35643 2017/05/08 14:09:52 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
```

```
2017-05-08_14:09:53.35654 2017/05/08 14:09:53 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:09:54.35664 2017/05/08 14:09:54 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
2017-05-08_14:09:55.35680 2017/05/08 14:09:55 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:09:56.35509 2017/05/08 14:09:56 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
2017-05-08_14:09:57.35538 2017/05/08 14:09:57 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:09:58.35661 2017/05/08 14:09:58 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
2017-05-08_14:09:59.35681 2017/05/08 14:09:59 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:10:00.35669 2017/05/08 14:10:00 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
2017-05-08_14:10:01.35804 2017/05/08 14:10:01 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:10:02.35522 2017/05/08 14:10:02 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
2017-05-08_14:10:03.35664 2017/05/08 14:10:03 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:10:04.35882 2017/05/08 14:10:04 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
2017-05-08_14:10:05.35906 2017/05/08 14:10:05 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:10:06.36192 2017/05/08 14:10:06 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
2017-05-08_14:10:07.36220 2017/05/08 14:10:07 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:10:08.36188 2017/05/08 14:10:08 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
2017-05-08_14:10:09.36211 2017/05/08 14:10:09 Dispatcher says container qr1hi-dz642-tjezupwylh67k2b
 is done: cancel slurm job
2017-05-08_14:10:10.35595 2017/05/08 14:10:10 container qr1hi-dz642-tjezupwylh67k2b
 is still in squeue after scancel
```

This is because once the channel is closed, it just keeps hitting scancel() with no delay in between. So I thought it could at least wait for the next HasUUID() broadcast.

#7 - 05/10/2017 07:04 PM - Peter Amstutz

- Target version changed from 2017-05-10 sprint to 2017-05-24 sprint

#8 - 05/10/2017 07:56 PM - Tom Clegg

Peter Amstutz wrote:

> Hhe problem I was trying to solve was that the logs are full of this:
>
> [...]
>
> This is because once the channel is closed, it just keeps hitting scancel() with no delay in between. So I thought it could at least wait for the next HasUUID() broadcast.

It looks to me like there's a 1-second delay. This delay is inserted by scancel(). The reason it keeps hitting scancel() is that the mocked squeue keeps reporting that the slurm job is still running.

I think the correct way to get rid of those messages would be to fix the squeue mock, so it mocks the behavior we're hoping to test against, instead of a slurm failure mode that makes us (correctly) produce a bunch of debug logs.

#9 - 05/10/2017 08:04 PM - Tom Clegg

oops, just noticed

log.Printf(text)

should be

log.Print(text)

Other than that, LGTM @ f8675ad473b45387b1286c6b7a41edf36148ebac

(as tempting as it is to ask for an error check on disp.Arv.Create() so we can log "error logging error: ...")

#10 - 05/10/2017 09:21 PM - Tom Clegg

I missed that the example logs above are from a real site, not the test suite. So the mystery is why scancel wasn't effective.

AFAICT crunch-dispatch-slurm was doing the right thing in the circumstances: keep hitting scancel, with a ≥1 second delay between attempts, until the job goes away.

In any case it seems like a separate topic from "propagate slurm errors to user", because at this point the container is already done and the requesting user probably has little interest in (and ability to fix) slurm problems that happen later. So I think if this scancel behavior needs attention, we should ~~open a new issue.~~ discuss further on #11669.

#11 - 05/11/2017 02:25 PM - Peter Amstutz

- Status changed from In Progress to Resolved

- % Done changed from 0 to 100

Applied in changeset arvados|commit:a309f8c5b4842075d6c83f99a8f2a1e1016976f5.