

Arvados - Bug #7026

[Node Manager] Mishandles stop signals

08/19/2015 03:43 PM - Brett Smith

Status:	New	Start date:	08/19/2015
Priority:	Normal	Due date:	
Assigned To:		% Done:	0%
Category:	Node Manager	Estimated time:	0.00 hour
Target version:	Arvados Future Sprints		
Description			
There are two issues here. They might have the same root cause, so for now, I'm filing one issue. If it turns out they don't, it's more important to address the primary bug, and we can split the secondary one off to be dealt with separately.			
Primary bug			
When Node Manager gets a stop signal like SIGINT, the daemon shuts down, along with all the monitors it's created. However, actors started in launcher.main (TimedCallbackActor, the list pollers, the node update actor) never stop, causing the Node Manager process to stay alive.			
launcher.main() ends with this code:			
<pre>signal.pause() daemon_stopped = node_daemon.actor_ref.actor_stopped.is_set while not daemon_stopped(): time.sleep(1) pykka.ActorRegistry.stop_all()</pre>			
Logs show that the daemon actor stops in Pykka, then many monitor actors shut down after. The daemon actor has no logic to shut down monitor actors. All this makes it <i>look</i> like we're getting to the stop_all() line. Otherwise, how else would the monitor actors be getting stopped? But other actors apparently aren't being stopped by this method call.			
Secondary bug			
Node Manager is supposed to implement escalating shutdown processes when it gets a stop signal repeatedly, until it eventually forces an exit. See launcher.shutdown_signal(). However, subsequent signals seem to have no effect.			
Related issues:			
Related to Arvados - Story #8543: [NodeManager] Don't use Futures when not ex...		Resolved	03/04/2016

History

#1 - 08/19/2015 03:55 PM - Brett Smith

One possibility: the [docs for stop_all](#) say actors are stopped in LIFO order. If that's true, there are two reasons to believe ComputeNodeUpdateActor is the culprit: it sleeps, and it's the last actor started in launcher.main(). If it has a large queue of sync_node requests that all start by sleeping, it could be some time before it finally gets and processes its stop message.

The logs we have are consistent with this theory. After the daemon shuts down, the logs show exceptions like #6225 being raised by the update actor, every 3 minutes (the default maximum wait time between requests in the actor).

If this is the cause, we might need to re-think the implementation of ComputeNodeUpdateActor. We could potentially fix this issue, plus things like #6225 in one fell swoop...

#2 - 01/14/2016 11:52 AM - Nico César

I don't know if this is the bug. But I guess it is.

here is some actions and symptoms

it doesn't want to die:

```
manage.qr2hi:/etc/sv# sv stop arvados-node-manager/
timeout: run: arvados-node-manager/: (pid 17243) 240431s, want down, got TERM
```

in the logs

```
2016-01-14_11:48:05.60399 2016-01-14 11:48:05 arvnodeman.cloud_nodes[17243] WARNING: CloudNodeListMonitorActor  
(at 42728272) got error: The read operation timed out - waiting 20 seconds
```

and node-manager continues spitting logs as usual, but (partially) running- I did a kill -9 and everything is back to normal.