**Arvados - Story #11349**

[Node Manager] Add status URL for node manager

03/28/2017 06:33 PM - Tom Morris

<table>
<thead>
<tr>
<th>Status:</th>
<th>Resolved</th>
<th>Start date:</th>
<th>04/10/2017</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>Assigned To:</td>
<td>Tom Clegg</td>
<td>% Done:</td>
<td>100%</td>
</tr>
<tr>
<td>Category:</td>
<td></td>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Target version:</td>
<td>2017-04-12 sprint</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Description**

Implemented HTTP server which serves status URL with JSON format output

Configurable port #

Start with the data which is currently being logged:

- List of nodes sizes
- Number of nodes in each state
- State of each node

**Subtasks:**

- Task # 11375: Review 11349-nodemanager-status-api
- Task # 11447: update wiki

**Related issues:**

- Related to Arvados - Feature #11799: [Node manager] Publish status.json (Duplicate)
- Related to Arvados - Story #11836: [Nodemanager] Improve status.json for moni... (Rejected 05/23/2018)

**Associated revisions**

- Revision 2c094e28 - 04/11/2017 03:33 PM - Tom Clegg
  Merge branch '11349-nodemanager-status-api'
  refs #11349

- Revision a6be53f6 - 04/11/2017 06:52 PM - Tom Clegg
  Build packages for python "future" module.
  refs #11349
  refs #11308

- Revision 1b290e51 - 04/12/2017 03:39 PM - Tom Clegg
  11349: Fix section name in example configs.
  refs #11349

**History**

#1 - 03/28/2017 06:40 PM - Tom Morris
- Description updated
- Story points set to 2.0

#2 - 03/28/2017 06:43 PM - Tom Clegg
See source/sdk/python/tests/keepstub.py and source/sdk/python/tests/test_keep_client.py for example of starting up a multithreaded http server.

Suggest maintaining a global status variable, protected by a mutex, and just dumping its content in the status.json handler.

#3 - 03/29/2017 07:58 PM - Tom Clegg
- Assigned To set to Tom Clegg
- Target version changed from Arvados Future Sprints to 2017-04-12 sprint
#4 - 04/07/2017 05:21 PM - Tom Clegg
- Status changed from New to In Progress

#5 - 04/07/2017 07:31 PM - Tom Clegg
details (proposed):

New config section "[Manage]" with "port" (127.0.0.1) and "address" (default -1, which disables management server)

status.json response

```
{
  "nodes_up": 3,
  "nodes_shutdown": 1,
  "nodes_booting": 2,
  "nodes_wish": 4
}
```

#6 - 04/07/2017 09:29 PM - Tom Clegg
11349-nodemanager-status-api @ ab9a73d2c0b567d3c05d1d4d846363a69eafda2

#7 - 04/07/2017 09:36 PM - Nico César

I see 2 clusters of questions I get often, one group is about "orchestration-related" or "pipeline-wide" and the other group of questions is about the resources inside a node when a job is running.

From the first group I usually get question like this (which this should help):

- "why my job is queued for X hours?" -> having a historical # nodes in wishlist could potentially give a clue.
- "my pipeline ran for 24 hours, which nodes did it use? " -> having a correlation of node with the pipeline helps.

From the second group:

- "is my node actually doing something?" -> having a "node38: up" doesn't say much, I think that's a question to answer with logs
- "how many cores/ram/big should my nodes have/be?" -> this is an analysis with the resources inside the node

so I think we can pull information from node manager to respond to the first group, usually this implies that the "node size" isn't as important as "how long has it been up and in which state"... so uniquely identifying the node than been able to plot that is good. But I have to admit that too much detail could turn this in to an Logstash nightmare-adventure I don't want to go, so some summarized state values as a first step is good.

the proposal is good:

```
{
  "nodes_up": 3,
  "nodes_shutdown": 1,
  "nodes_booting": 2,
  "nodes_wish": 4
}
```

later will be good to have unique node names and a way to report them over time (which makes it very difficult when they weren't born yet and in the "nodes_wish" pile)

#8 - 04/10/2017 02:24 PM - Tom Clegg
11349-nodemanager-status-api @ e7876a3ac520b128be7836e39172079ab2a5e45

#9 - 04/11/2017 02:32 PM - Lucas Di Pentima

Local test run was successful

Questions:

- services/nodemanager/arvnodeman/status.py - Do you think it would be good idea to log messages indicating when no management server is started (and maybe the reason?)
- services/nodemanager/tests/test_status.py:43 - Is that assertion superfluous given the following one? if it's to prove that old values remain, can it be checked outside the loop?
- Is the state of each node going to be included? (asking because it's mentioned on the story description)

#10 - 04/11/2017 03:18 PM - Tom Clegg

Lucas Di Pentima wrote:

Local test run was successful
- services/nodemanager/arvnodeman/status.py - Do you think it would be good idea to log messages indicating when no management server is started (and maybe the reason?)

Yes, added.

```python
if not self.enabled:
    _logger.warning("Management server disabled. " +
                          "Use [Manage] config section to enable.")
    return
```

- services/nodemanager/tests/test_status.py:43 - Is that assertion superfluous given the following one? if it's to prove that old values remain, can it be checked outside the loop?

Yes, moved it outside the loop.

- Is the state of each node going to be included? (asking because it's mentioned on the story description)

Indeed, we seem to have changed our minds about that: for now we just want a summary that we can graph easily.

Suggest adding "/nodes.json" with info about each node. (Not sure if we should keep this issue open for it or make a new one.)

11349-nodemanager-status-api @ a779382603d2da2ec38ceb8a21262cc4f15f077

#11 - 04/11/2017 03:32 PM - Lucas Di Pentima
LGTM, thanks!

#12 - 04/12/2017 05:35 PM - Tom Clegg
- Status changed from In Progress to Resolved