

Arvados - Bug #10587

All Python CLI utilities should report --version

11/22/2016 05:06 PM - Tom Morris

Status:	Resolved	Start date:	11/22/2016
Priority:	Normal	Due date:	
Assigned To:	Lucas Di Pentima	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	2016-12-14 sprint		
Description			
Subtasks:			
Task # 10624: Review 10587-python-cli-version			Resolved
Related issues:			
Related to Arvados - Feature #10666: All Arvados components should report the...			Resolved 12/05/2016

Associated revisions

Revision 0514b290 - 12/09/2016 05:10 PM - Lucas Di Pentima

Merge branch '10587-python-cli-version'
Closes #10587

Revision 3265a76b - 12/12/2016 04:29 PM - Peter Amstutz

Merge branch '10587-node-manager-report-version' refs #10587

Revision d070d1e5 - 12/12/2016 05:46 PM - Peter Amstutz

Nodemanager also logs libcloud package version, refs #10587

History

#1 - 11/23/2016 08:11 PM - Tom Morris

- Subject changed from All CLI utilities should report --version to All Python CLI utilities should report --version
- Assigned To set to Lucas Di Pentima
- Story points set to 1.0

#2 - 11/29/2016 05:10 PM - Lucas Di Pentima

- Status changed from New to In Progress

#3 - 12/06/2016 11:18 PM - Lucas Di Pentima

Test run at: <https://ci.curoverse.com/job/developer-run-tests/87/>
Updates: [fee8873](#)

Added --version to the following commands:

- arv-ls
- arv-put
- arv-ws
- arv-get
- arv-copy
- arv-run
- arv-normalize
- arv-keepdocker
- arv-mount
- arvados-node-manager

Also added tests to confirm the new argument.

#4 - 12/07/2016 04:37 PM - Tom Clegg

Looks like we need to add setuputils dependency to our deb/rpm packages, like we did for arvados-cwl-runner, in order to use this mechanism.

I guess it also makes sense to add setuptools to install_requires in setup.py.

Tried in wheezy, and found that the wheezy version of python-setuptools (0.6.24-1) does work, so at least we don't have to build another new package.

I'm very wary of using multiprocessing in tests, especially arv-put which definitely uses threading. Combining multiprocessing and threading in Python seems to be a recipe for disaster (or at least unbounded debugging efforts). Can we get what we need just by catching SystemExit? In [source:services/fuse/tests/test_command_args.py](https://source.services/fuse/tests/test_command_args.py) it looks like this works for argparse. Failing that, maybe subprocess?

#5 - 12/07/2016 09:01 PM - Lucas Di Pentima

Updates at: [ceae525](#)

Test suite run at: <https://ci.curoverse.com/job/developer-run-tests/88/>

- Changed those tests depending on multiprocessing to use a context manager stderr/stdout redirector and a SystemExit catching assertion as requested.
- Added setuptools dependencies on package building script and setup.py install_require definitions.

#6 - 12/09/2016 04:09 PM - Lucas Di Pentima

Merged master into branch at [feb463839301b0b596089e48a981660365d2c4a7](#)

Tests running: <https://ci.curoverse.com/job/developer-run-tests/89/>

#7 - 12/09/2016 04:45 PM - Tom Clegg

LGTM thanks!

#8 - 12/09/2016 05:15 PM - Lucas Di Pentima

- Status changed from *In Progress* to *Resolved*

- % Done changed from 0 to 100

Applied in changeset arvados|commit:0514b290f5ff9a2700b599bf6fb19a468a73c3fb.