

Arvados - Bug #11226

[SDK] Python client tries to cache discovery document at the same local path regardless of user

03/08/2017 07:31 PM - Peter Amstutz

Status:	Resolved	Start date:	03/10/2017
Priority:	Normal	Due date:	
Assigned To:	Peter Amstutz	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	2017-03-15 sprint		
Description			
<p>The Python SDK instructs httplib2 to cache the discovery document in ~/.cache/arvados/discovery. However, the google API client code also has its own cache logic, which caches the discovery document in /tmp/google-api-python-client-discovery-doc.cache.</p> <p>This is a problem particularly on a multi-user system such as a shell node. The file gets created in a global location like /tmp, then only the original user can update the file.</p> <p>From looking at the google-api-python-client code, the destination is chosen as os.path.join(tempfile.gettempdir(), FILENAME) where FILENAME = 'google-api-python-client-discovery-doc.cache'</p> <p>https://github.com/google/google-api-python-client/blob/dc6c1efea8e6c89afa94df984d9291359cb04797/googleapiclient/discovery_cache/file_cache.py#L87</p>			
Proposed fix			
See note-6.			
Subtasks:			
Task # 11233: Review			Resolved
Related issues:			
Related to Arvados - Bug #10669: googleapiclient.errors.InvalidJsonError retr...		Resolved	12/06/2016

Associated revisions

Revision 82697fea - 03/09/2017 07:44 PM - Peter Amstutz

Merge branch '11226-discovery-doc-cache' closes #11226

History

#1 - 03/08/2017 07:32 PM - Peter Amstutz

- Description updated

#2 - 03/08/2017 07:32 PM - Peter Amstutz

- Description updated

#3 - 03/08/2017 07:41 PM - Peter Amstutz

- Description updated

#4 - 03/09/2017 03:24 PM - Tom Clegg

- Subject changed from [SDK] Not using ~/.cache/arvados/discovery to [SDK] Python client tries to cache discovery document at the same local path regardless of user

- Description updated

#5 - 03/09/2017 03:26 PM - Tom Clegg

- Description updated

#6 - 03/09/2017 03:38 PM - Peter Amstutz

<https://github.com/google/google-api-python-client/pull/127>

I think we can use the `cache_discovery` option to `build()` to disable the Google API client caching.

#7 - 03/09/2017 03:39 PM - Tom Clegg

- *Description updated*

#8 - 03/09/2017 03:45 PM - Peter Amstutz

Also want to do one of:

a) continue to use `httplib2` caching, which is apparently not multithread/multiprocess safe

b) provide our own `httplib2` cache which is multithread/multiprocess safe

c) provide a google api client cache object (with `build(cache=XXX)`) which is multithread/multiprocess safe:

https://github.com/google/google-api-python-client/blob/v1.4.2/googleapiclient/discovery_cache/base.py

#9 - 03/09/2017 03:48 PM - Tom Clegg

Suggest we

1. pass `cache_discovery=False` to `google-api-client` so we get back to what we thought we were doing, i.e., sensible http caching
2. also fix [#10669](#) so the http cache is safe for multiple processes to use

#10 - 03/09/2017 03:49 PM - Peter Amstutz

+1

#11 - 03/09/2017 03:51 PM - Peter Amstutz

- *Assigned To set to Peter Amstutz*

- *Story points set to 0.5*

#12 - 03/09/2017 03:54 PM - Peter Amstutz

- *Target version set to 2017-03-15 sprint*

#13 - 03/09/2017 07:50 PM - Peter Amstutz

- *Status changed from New to Resolved*

- *% Done changed from 0 to 100*

Applied in changeset `arvados|commit:82697fea93b1c87cdee27d2b9a76c1b7ac07497e`.

#14 - 03/13/2017 04:19 PM - Tom Morris

I take it this got instagroomed while I was gone.

Does this mean that [#10669](#) is fixed (and can be closed) as well?

#15 - 03/13/2017 04:55 PM - Peter Amstutz

Tom Morris wrote:

I take it this got instagroomed while I was gone.

Does this mean that [#10669](#) is fixed (and can be closed) as well?

No, this only disables google-api-client caching so that it uses the httplib2 cache as originally intended. The bug in [#10669](#) is that the httplib2 cache is not multi-thread/multi-process safe, and which makes it possible to get corrupted. The google-api-client cache actually does do file locking, but puts the cache file in the global /tmp and doesn't provide a way to change that except for changing the global tempfile path (which has side effects for everything using the tempfile module).