

Arvados - Feature #14930

Add flag to arv-put to set a trash_at date

03/07/2019 07:36 PM - Bryan Cosca

Status:	Resolved	Start date:	06/04/2019
Priority:	Normal	Due date:	
Assigned To:	Lucas Di Pentima	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	2019-06-19 Sprint		
Description			
The use case would be for when sequencers finish uploading to arvados, we can set the timer to delete that data after some designated time period.			
Trash times could be specified in 2 ways:			
1. Absolute datetime: Could be accepted from a param like --trash_at "YYYY-MM-DD HH:MM"			
2. Relative times: Could be accepted from a param like --trash_after XX (where XX is number of days)			
Both parameters would be mutually exclusive.			
The accepted format for absolute datetimes would be the one described at https://en.wikipedia.org/wiki/ISO_8601			
The accepted relative time parameter unit will be number of days (to be converted to amount of seconds) and it should take into account possible timezone changes.			
The relative trash time should take note of the upload finish process datetime. When uploading using the "rsync mode", it should update the trash_at value when checkpointing.			
Subtasks:			
Task # 15267: Review 14930-arvput-trash-at			Resolved
Related issues:			
Related to Arvados - Feature #12980: Add --trash-at flag to arv-put			Resolved

Associated revisions

Revision 2126d3a9 - 06/11/2019 07:13 PM - Lucas Di Pentima

Merge branch '14930-arvput-trash-at'
Closes #14930

Arvados-DCO-1.1-Signed-off-by: Lucas Di Pentima <ldipentima@veritasgenetics.com>

History

#1 - 03/13/2019 04:04 PM - Tom Morris

- Target version set to To Be Groomed

#2 - 03/20/2019 03:15 PM - Lucas Di Pentima

- Description updated

#3 - 03/20/2019 03:16 PM - Lucas Di Pentima

- Description updated

#4 - 03/20/2019 04:04 PM - Lucas Di Pentima

- Subject changed from Add flag to arv-put to set a trash_at / delete_at date to Add flag to arv-put to set a trash_at date

- Description updated

- Target version changed from To Be Groomed to Arvados Future Sprints

- Story points set to 1.0

#5 - 05/22/2019 05:44 PM - Tom Morris

- Target version changed from Arvados Future Sprints to 2019-06-05 Sprint

#6 - 05/22/2019 05:49 PM - Lucas Di Pentima

- Assigned To set to Lucas Di Pentima

#7 - 05/31/2019 02:12 PM - Lucas Di Pentima

- Status changed from New to In Progress

#8 - 06/04/2019 01:46 PM - Lucas Di Pentima

Updates at [56e6ffd4a](#) - branch 14930-arvput-trash-at

Test run: <https://ci.curoverse.com/job/developer-run-tests/1276/>

- arv-put accepts --trash-at DATE where DATE's format is the subset of ISO8601 supported by ciso8601 python module.
- arv-put also accepts --trash-after N, being N the number of days in the future from the upload finish datetime.
- When not passing timezone information to --trash-at, it will assume the provided date is expressed in the local system's timezone configuration.
- When using --update-collection UUID and --trash-after N, it periodically updates the updated collection's trash_at attribute.

#9 - 06/05/2019 02:51 PM - Lucas Di Pentima

- Target version changed from 2019-06-05 Sprint to 2019-06-19 Sprint

#10 - 06/07/2019 01:30 PM - Peter Amstutz

```
+         if trash_at.tzinfo is not None:
+             # Timezone aware datetime provided.
+             utcoffset = trash_at.utcoffset()
+         else:
+             # Timezone naive datetime provided. Assume is local.
+             utcoffset = datetime.timedelta(hours=-time.timezone/3600)
+             # Convert to UTC timezone naive datetime.
+         trash_at = trash_at.replace(tzinfo=None) - utcoffset
```

Not sure why you're rounding off to hours if the timezone struct has the offset in seconds? Also why calculate a negative offset and then subtract it when the source value is meant to be added?

```
+         logger.error("--trash-at argument should be set in the future")
+         sys.exit(1)
+     if args.trash_after is not None:
+         if args.trash_after < 1:
+             logger.error("--trash-after argument should be >= 1")
```

"should" → "must" since it is being enforced.

What happens if the user gives YYYY-MM-DD with no time? My guess is that you probably get 00:00 (midnight) at the start of the day as the expire date, but we should confirm. We should also consider if it would be more user friendly to make the expire date 23:59 at the end of the day. Does it accept YYYY-MM or YYYY? ¿En Español se escribe "AAAA-MM-DD"?

#11 - 06/07/2019 05:05 PM - Lucas Di Pentima

Updates at [64eadab02](#)

Test run: <https://ci.curoverse.com/job/developer-run-tests/1291/>

- Don't round off to hours the timezone
- Don't double subtract the utc offset on the code
- Error message corrections
- Don't accept day-less dates (eg: 2020-01) on --trash-at
- If the user doesn't provide HH:MM on --trash-at, assume it to be at the end of the specified day (local time, of course) instead of at 00:00
- Log the expiration date after saving the collection.
- Tests additions

#12 - 06/10/2019 02:14 PM - Ward Vandewege

- Release set to 22

#13 - 06/10/2019 03:09 PM - Peter Amstutz

I think you have a daylight savings time bug (time zones are awful, and daylight savings is the worst):

```
$ date
Mon Jun 10 10:57:07 EDT 2019

$ arv-put --trash-after 2 lightning-2.6.3-sm+tb-linux.xpi
2019-06-10 10:57:39 arvados.arv_put[6212] INFO: Creating new cache file at /home/peter/.cache/arvados/arv-put/
5b9195b264c8dda29afac74006b05663
3M / 3M 100.0% 2019-06-10 10:57:42 arvados.arv_put[6212] INFO:
```

2019-06-10 10:57:42 arvados.arv_put[6212] INFO: Collection saved as 'Saved at 2019-06-10 14:57:39 UTC by peter@peteravg'. It will expire on 2019-06-12 09:57:40 -0400.
[c97gk-4zz18-rj4m8nvzt0lokcv](#)

I would have expected now + 2 days to be 2019-06-12 10:57:40 -0400

#14 - 06/11/2019 01:25 PM - Lucas Di Pentima

Updates at [4aee7d57faff02fc6b7b6f750dc22a29e58bb963](#)

Test run: <https://ci.curoverse.com/job/developer-run-tests/1300/>

Was able to handle DST without including any additional dependencies. Not sure how to write a test covering different DST scenarios. I tested it manually, though.

#15 - 06/11/2019 07:05 PM - Peter Amstutz

Daylight savings time works.

I don't want to go back and forth on this one any more but there was one more thing I noticed with manual testing.

```
$ arv-put --trash-at '2019-06-12 4:51:00' arvados_version.py
2019-06-11 14:40:55 arvados.arv_put[15306] ERROR: --trash-at argument format invalid, use --help to see examples. (X-Request-Id: req-fl7ankvzww2h14tusx71)
(venv) peter@peteravg:[pts/3]:~/arvbox/arvbox/arvados/sdk/python [(HEAD detached at origin/14930-arvput-trash-at)]
```

```
$ arv-put --trash-at '2019-06-12 04:51:00' arvados_version.py
2019-06-11 14:41:00 arvados.arv_put[15334] INFO: Resuming upload from cache file /home/peter/.cache/arvados/arv-put/4eaf371b61459e2188bc4e4cbeaa497e
0M / 0M 100.0% 2019-06-11 14:41:00 arvados.arv_put[15334] INFO:
```

2019-06-11 14:41:00 arvados.arv_put[15334] INFO: Collection saved as 'Saved at 2019-06-11 18:41:00 UTC by peter@peteravg'. It will expire on 2019-06-12 04:51:00 -0400.
[4xphq-4zz18-yps644jfbjtlv8f](#)

So '2019-06-12 4:51:00' fails but '2019-06-12 04:51:00' works, it looks like the parser is very particular about field widths. (Also noticed that it is documented to have a 'T' in between the date and time but it also accepts a space).

I'll leave this one up to you if you want to do anything about it, otherwise LGTM.

#16 - 06/11/2019 09:34 PM - Lucas Di Pentima

- Status changed from In Progress to Resolved

- % Done changed from 0 to 100

Applied in changeset [arvados|2126d3a9115620040260847a9e29fd5c73b466ab](#).

#17 - 02/25/2020 09:00 PM - Peter Amstutz

- Related to Feature #12980: Add --trash-at flag to arv-put added