Arvados - Feature #10181

Crunch job output logging improvement stories

10/03/2016 11:16 PM - Joshua Randall

Status:	Resolved	**Start date:**	02/16/2017
Priority:	Normal	**Due date:**	
Assigned To:	Tom Clegg	**% Done:**	100%
Category:	API	**Estimated time:**	0.00 hour
Target version:	2018-10-03 Sprint		

Description

Story: job output logged to keep while job is running

As a user, I would like to be able to retrieve complete (i.e. starting at the beginning and without being silenced by rate limiting) output of any running jobs. My strong preference would be to use keep for this, since (a) the full job log will be in keep anyway, so I'm already used to using that as an interface and (b) the bulk storage available to keep is generally going to be much greater than other places the logs could be kept (such as in the database). I would be ok with not necessarily always having up-to-the minute job logs available in keep, as long as the logs that are there are complete up to the point where they are truncated. Perhaps the final line of a truncated log entry could note that the job is still running and more output will arrive soon, along with stating the timestamp of the point at which the logs were flushed to keep (i.e. I would then know to expect that the next line would be timestamped after that time).

As a sysadmin, I would like to be able to adjust settings for flushing job logs to keep. I assume that any time a crunch job has a full block (i.e. 64MB) of output that it would be immediately written to keep and that the job's log collection would be updated to point to a new portable data hash which includes the new block. However, it would also be good to have a setting for flushing smaller amounts of log data to keep, so that logs from jobs that haven't output very much in some time can nonetheless be available. For example, I might configure a setting such that job output would be written to keep and the collection portable data hash updated every 15m regardless of how much output has been produced. That configuration option would be a tradeoff between creating a potentially large number of partially used keep blocks (although they would end up being cleaned up by keep-balance once a collection no longer points to them) and having a wait a long time for job output to appear in keep.

(remainder moved to #14284)

~~Story: job output does not belong in the database logs table and should be able to be directed to non-Arvados logging systems~~

~~As a sysadmin, I'd rather my postgres database not fill up with hundreds of GB of job output logs. In addition to requiring a large amount of storage on the volume where the postgres database lives, this also tends to make queries to the logs table that have nothing to do with job output logging (i.e. fulfilling its role as more of an audit-log, such as checking for recent changes to collections) take ridiculously long. I think it would be best if no job output at all was stored in the central postgres database. In conjunction with the above story regarding storing in-progress job logs to keep, it would be great if some other system which is better suited to the task of buffering and distributing recent job output in order to make real-time job output available. It would be great if it could be sent via an existing log broker system such as logstash or fluentd such that it would be possible to not only direct the logs to whatever component Arvados uses to buffer and deliver the logs to consumers (such as via the existing websockets interface) but also to other non-Arvados logging systems (where we may be running the rest of the ELK/EFK stack for search and visualisation).~~

Subtasks:

Task # 11122: log partial job output to keep while job is running	**Resolved**
Task # 14241: Review 10181-incremental-log	**Resolved**

Related issues:

Related to Arvados - Feature #12996: [SDKs] CollectionFS should repack highly...	**New**
Related to Arvados - Story #13048: Refactor crunch2 logging	**New**
Related to Arvados - Story #14284: Send real time container logs to a suitabl...	**Closed**

Associated revisions

Revision c25f36a0 - 10/01/2018 07:43 PM - Tom Clegg

Merge branch '10181-incremental-log'

refs #10181

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <tclegg@veritasgenetics.com>

History

#1 - 08/29/2017 01:47 PM - Tom Morris

- Target version set to Arvados Future Sprints

#2 - 01/22/2018 04:45 PM - Tom Clegg

Implementation notes

- Containers API needs to permit updating to a non-null "log" field when state=Running.
- This is an opportune time for crunch-run to drop its custom collection-writing code and use "collection filesystem" instead (if that hasn't happened already). CollectionFS makes it easy to get a goroutine-safe manifest snapshot.
- The "flushing logs" and "flushed logs, pdh is X" log entries can go in crunch-run.txt.

Good times to flush:

- Just before starting the container (i.e., immediately after updating state to Running) -- this ensures node-info etc. are stored.
- Every hour (configurable)
- SIGUSR1
- 32 MiB of logs have been written since last flush (or some other threshold that's large but less than 64MiB)

#3 - 01/24/2018 07:13 PM - Tom Morris

- Story points set to 2.0

#4 - 01/24/2018 08:33 PM - Tom Clegg

- Related to Feature #12996: [SDKs] CollectionFS should repack highly fragmented files added

#5 - 01/24/2018 08:37 PM - Tom Clegg

(from discussion offline) we can do this in three phases.

phase 1: save log snapshots periodically

(as described in note-2)

phase 2: improve block packing in CollectionFS

This is #12996.

phase 3: save log snapshots frequently

Once #12996 is done, it should be feasible to save a snapshot much more frequently -- e.g., every ~10 seconds or ~1KB. This would eliminate the motivation for a user-accessible "tell crunch-run to flush logs" API. Users could usefully run "tail -f" on log collections.

This will also make progress toward obsoleting the "send logs to API server, too" stuff: workbench/users can use "tail -f" instead.

#6 - 02/07/2018 08:04 PM - Tom Clegg

- Related to Story #13048: Refactor crunch2 logging added

#7 - 09/19/2018 03:46 PM - Tom Clegg

- Assigned To set to Tom Clegg

- Target version changed from Arvados Future Sprints to 2018-10-03 Sprint

#8 - 09/19/2018 09:10 PM - Tom Clegg

- Status changed from New to In Progress

#9 - 09/20/2018 06:16 PM - Peter Amstutz

Capturing from chat:

> Anyone have thoughts on how we should give the container-requesting users permission to read the log collection for a running container? extend the permission system to give access through the CR... or create a copy of the (container's own) log collection for each CR before it's finalized and keep their contents synchronized as the main one gets updated... or implement a special symlink/alias kind of collection... or...?

Since the container request has a specific log_uuid, I think we want to create a log output collection per container request, and when the container's log is updated, find all the linked container requests and updates their log collections in the same transaction. So your 2nd idea. If/when we create a separate table to deduplicate manifest text, then it becomes a trivial update

#10 - 09/21/2018 08:51 PM - Tom Clegg

Testing reveals a bug in collectionfs at e7b3a06c0:

```
panic: can't marshal segment type *arvados.memSegment

goroutine 21 [running]:
git.curoverse.com/arvados.git/sdk/go/arvados.(*dirnode).marshalManifest(0xc4203e6090, 0xb55055, 0x1, 0x0, 0x0, 0x0, 0x0)
        /home/tom/.cache/arvados-build/GOPATH/src/git.curoverse.com/arvados.git/sdk/go/arvados/fs_collection.go:689 +0x13e2
git.curoverse.com/arvados.git/sdk/go/arvados.(*collectionFileSystem).MarshalManifest(0xc42034a0c0, 0xb55055, 0x1, 0x0, 0x0, 0x0, 0x0)
        /home/tom/.cache/arvados-build/GOPATH/src/git.curoverse.com/arvados.git/sdk/go/arvados/fs_collection.go:141 +0xa8
git.curoverse.com/arvados.git/services/crunch-run.(*ContainerRunner).saveLogCollection(0xc4200fad00, 0x435, 0x118c940, 0xf4240, 0xbee173ef36b972a6, 0x79940efb, 0x118c940, 0xc42017e100, 0x0, 0x0, ...)
        /home/tom/.cache/arvados-build/GOPATH/src/git.curoverse.com/arvados.git/services/crunch-run/crunchrun.go:1373 +0xfa
git.curoverse.com/arvados.git/services/crunch-run.(*ContainerRunner).checkpointLogs(0xc4200fad00)
        /home/tom/.cache/arvados-build/GOPATH/src/git.curoverse.com/arvados.git/services/crunch-run/crunchrun.go:1200 +0x347
created by git.curoverse.com/arvados.git/services/crunch-run.NewContainerRunner
        /home/tom/.cache/arvados-build/GOPATH/src/git.curoverse.com/arvados.git/services/crunch-run/crunchrun.go:1693 +0x419
exit status 2
```

#11 - 09/25/2018 06:52 PM - Tom Clegg

10181-incremental-log @ 3596aff0954f405b06799814585d834502d0d76ahttps://ci.curoverse.com/view/Developer/job/developer-run-tests/904/

#12 - 09/25/2018 09:01 PM - Tom Clegg

10181-incremental-log @ d5aeb0c9a768004dfb806678573f11b5048e26f0https://ci.curoverse.com/view/Developer/job/developer-run-tests/905/

#13 - 09/26/2018 08:21 PM - Tom Clegg

10181-incremental-log @ 5d6a2c2e4b85434e9bae1dd0adc27c284cb9ea85

- update log on SIGUSR1
- rename configs to crunchLogUpdateSize/Period
- add configs to apiserver config and discovery doc so they're configurable in practice

#14 - 09/27/2018 08:00 PM - Lucas Di Pentima

I did a manual test using arvbox. Set up the update to be every 5 secs and made a workflow with a step that counted from 1 to 300 doing a 1s sleep in between.
The initial collection got created with the first 5 seconds of data, but that was it, below is a piece of the log streamed to workbench:

```
[...]
2018-09-27T19:45:52.828197877Z 1
2018-09-27T19:45:52.756017822Z notice: reading stats from /sys/fs/cgroup/cpuacct/docker/e657d1b181a64f54ccc515eea956c650a8fc7a10ba6e20525a2f37dffbc33c41/cgroup.procs
2018-09-27T19:45:52.756091771Z notice: monitoring temp dir /tmp/crunch-run.o967z-dz642-7sirqk1519vjhn5.660544265
2018-09-27T19:45:52.756151559Z notice: reading stats from /sys/fs/cgroup/memory/docker/e657d1b181a64f54ccc515eea956c650a8fc7a10ba6e20525a2f37dffbc33c41/memory.stat
2018-09-27T19:45:52.756614541Z mem 12288 cache 0 pgmajfault 724992 rss
2018-09-27T19:45:52.756685046Z notice: reading stats from /sys/fs/cgroup/cpuacct/docker/e657d1b181a64f54ccc515eea956c650a8fc7a10ba6e20525a2f37dffbc33c41/cpuacct.stat
2018-09-27T19:45:52.756732437Z notice: reading stats from /sys/fs/cgroup/cpuset/docker/e657d1b181a64f54ccc515eea956c650a8fc7a10ba6e20525a2f37dffbc33c41/cpuset.cpus
2018-09-27T19:45:52.756755175Z cpu 0.0000 user 0.0000 sys 4 cpus
2018-09-27T19:45:52.756777390Z notice: reading stats from /sys/fs/cgroup/blkio/docker/e657d1b181a64f54ccc515eea956c650a8fc7a10ba6e20525a2f37dffbc33c41/blkio.io_service_bytes
2018-09-27T19:45:52.756882819Z net:eth0 4869065 tx 33574709 rx
2018-09-27T19:45:52.756889551Z net:docker0 0 tx 0 rx
2018-09-27T19:45:52.756910270Z statfs 34100273152 available 24358596608 used 61612142592 total
2018-09-27T19:45:52.831408234Z Waiting for container to finish
2018-09-27T19:45:53.830168337Z 2
2018-09-27T19:45:54.830870671Z 3
2018-09-27T19:45:55.833690109Z 4
2018-09-27T19:45:56.835402401Z 5
2018-09-27T19:45:57.836645761Z 6
2018-09-27T19:45:58.838261194Z 7
2018-09-27T19:45:59.839813679Z 8
```

```
2018-09-27T19:46:00.842056232Z 9
2018-09-27T19:46:01.859913158Z error updating log collection: error saving log collection: arvados API server 
error: Path not found (404: 404 Not Found) returned by 172.17.0.2:8000
[...]
```

... then every 5 seconds it produced the same error message. When the count finished, the container was cancelled:

```
[...]
2018-09-27T19:50:49.517948845Z 297
2018-09-27T19:50:50.519363966Z 298
2018-09-27T19:50:51.520621547Z 299
2018-09-27T19:50:52.206410749Z error updating log collection: error saving log collection: arvados API server 
error: Path not found (404: 404 Not Found) returned by 172.17.0.2:8000
2018-09-27T19:50:51.925490170Z mem 243326976 cache 66074 pgmajfault 2351603712 rss
2018-09-27T19:50:51.925567230Z cpu 811.9800 user 273.9800 sys 4 cpus -- interval 9.9991 seconds 5.5200 user 1.
6900 sys
2018-09-27T19:50:51.925633581Z blkio:8:0 85962752 write 4860669952 read -- interval 9.9991 seconds 376832 writ
e 0 read
2018-09-27T19:50:51.925977021Z net:eth0 5654043 tx 33647339 rx -- interval 9.9991 seconds 10681 tx 1320 rx
2018-09-27T19:50:51.925983775Z net:docker0 0 tx 0 rx -- interval 9.9991 seconds 0 tx 0 rx
2018-09-27T19:50:52.484416864Z crunchstat: keepcalls 0 put 0 get -- interval 10.0000 seconds 0 put 0 get
2018-09-27T19:50:52.484416864Z crunchstat: net:keep0 0 tx 0 rx -- interval 10.0000 seconds 0 tx 0 rx
2018-09-27T19:50:52.484416864Z crunchstat: keepcache 0 hit 0 miss -- interval 10.0000 seconds 0 hit 0 miss
2018-09-27T19:50:52.484416864Z crunchstat: fuseops 0 write 0 read -- interval 10.0000 seconds 0 write 0 read
2018-09-27T19:50:52.484416864Z crunchstat: blkio:0:0 0 write 0 read -- interval 10.0000 seconds 0 write 0 read
2018-09-27T19:50:52.521507071Z 300
2018-09-27T19:50:52.757394562Z mem 585728 cache 1 pgmajfault 204800 rss
2018-09-27T19:50:52.757461149Z cpu 0.0800 user 0.1900 sys 4 cpus -- interval 10.0003 seconds 0.0000 user 0.020
0 sys
2018-09-27T19:50:52.757517542Z blkio:8:0 0 write 557056 read -- interval 10.0002 seconds 0 write 0 read
2018-09-27T19:50:52.757583839Z net:eth0 5655296 tx 33647471 rx -- interval 10.0002 seconds 10717 tx 1320 rx
2018-09-27T19:50:52.757588208Z net:docker0 0 tx 0 rx -- interval 10.0002 seconds 0 tx 0 rx
2018-09-27T19:50:52.757602619Z statfs 34089582592 available 24369287168 used 61612142592 total -- interval 10.
0001 seconds 532480 used
2018-09-27T19:50:53.664588241Z Container exited with code: 0
2018-09-27T19:50:53.734082394Z Complete
2018-09-27T19:50:53.896697Z Container o967z-dz642-7sirqk1519vjhn5 was cancelled
2018-09-27T19:50:53.833473865Z error in CommitLogs: error saving log collection: arvados API server error: Pat
h not found (404: 404 Not Found) returned by 172.17.0.2:8000
2018-09-27T19:50:53.961853816Z Running [arv-mount --unmount-timeout=8 --unmount /tmp/crunch-run.
o967z-dz642-7sirqk1519vjhn5.660544265/keep677697748]
2018-09-27T19:50:54.342359677Z crunch-run finished
```

#15 - 09/27/2018 09:09 PM - Lucas Di Pentima

Checked the API's log, and what I found related to 404s & collections is:

```
{"method":"PUT","path":"/arvados/v1/collections/o967z-4zz18-p8b0irn356trohm","format":"html","controller":"Arv
ados::V1::CollectionsController","action":"update","status":404,"duration":4.99,"view":0.29,"db":1.07,"request
_id":"req-1cawyeq4uxdgm8qhv84c","client_ipaddr":"127.0.0.1","client_auth":"o967z-gj3su-begs3z1v1wa1j4n
","params":{"collection":"{\"is_trashed\":true,\"manifest_text\":\". 93ee9c2c5dc149f843ce396ebac7f170+1305+A1f
ae52fac4c47a8999cf7111e2e635fbaacb0e6d@5bbfa5c3 455992e54d23717fe0c483d0a20d771e+524+Ae7f273d9776381dda6e08a55
f2efb4f50eec4201@5bbfa5c9 0:496:crunch-run.txt 496:1333:hoststat.txt\\n\",\"name\":\"logs for 
o967z-dz642-8e3ddcha87nujem
\"}"},"@timestamp":"2018-09-27T19:34:33.287219804Z","@version":"1","message":"[404] PUT /arvados/v1/collection
s/o967z-4zz18-p8b0irn356trohm (Arvados::V1::CollectionsController#update)"}
```

That collection is trashed so, that could be the reason it returns 404?

#16 - 09/28/2018 06:42 PM - Tom Clegg

Yes, that makes sense. Updated: 10181-incremental-log @ 5e41146b32e4afd9de4eefc22b3c9fef05c471dc

#17 - 09/28/2018 07:32 PM - Lucas Di Pentima

For some reason runtime_status is not being able to be updated when status==Running. The following appeared on stderr.log on the runner container request:

```
2018-09-28T19:01:05.115298403Z arvados.cwl-runner INFO: Couldn't update runtime_status: <HttpError 422 when re
questing https://172.17.0.2:8000/arvados/v1/containers/o967z-dz642-4hyignu6z3r43w8?alt=json returned "Runtime 
status cannot be modified in state 'Running' ({}, {"error"=>"arvados.cwl-runner: [container s1] (
o967z-xvhdp-e8j0rgpir5sgegr
) error log:", "errorDetail"=>"\n  2018-09-28T18:59:53.168544047Z crunch-run crunch-run dev started\n  2018-09
-28T18:59:53.168564083Z crunch-run Executing container 'o967z-dz642-xshagnvtfnpcjl2
'\n  2018-09-28T18:59:53.168619142Z crunch-run Executing on host '8210e5fb2838'\n  2018-09-28T18:59:53.2437062
65Z crunch-run Fetching Docker image from collection '7948107960ee430ccbe3c4b7351377bf+342'\n  2018-09-28T18:5
9:53.295959430Z crunch-run Using Docker image id 'sha256:a8254670f419e7271c387b0e66507e96704ea4ade001f67403f6c
4d5a1a14432'\n  2018-09-28T18:59:53.388418126Z crunch-run Loading Docker image from keep\n  2018-09-28T19:00:4
9.169760803Z crunch-run Docker response: {\"stream\":\"Loaded image ID: sha256:a8254670f419e7271c387b0e66507e9
6704ea4ade001f67403f6c4d5a1a14432\\n\"}\n  2018-09-28T19:00:49.170701530Z crunch-run Running [arv-mount --fore
ground --allow-other --read-write --crunchstat-interval=10 --file-cache 268435456 --mount-by-pdh by_id /tmp/cr
unch-run.o967z-dz642-xshagnvtfnpcjl2
.228296496/keep086461647]\n  2018-09-28T19:00:53.546324711Z crunch-run Creating Docker container\n  2018-09-28
T19:00:53.975811828Z crunch-run Attaching container streams\n  2018-09-28T19:00:54.713184622Z crunch-run Start
ing Docker container id 'd65bdd2adc677a0c07216bc31347d84280d1c661acbba9606e7fe782c33afa2c'\n  2018-09-28T19:00
:55.402550893Z crunch-run Waiting for container to finish\n  2018-09-28T19:01:01.043894451Z crunch-run Contain
er exited with code: 1\n  2018-09-28T19:01:01.146545138Z crunch-run Complete"})">
2018-09-28T19:01:05.189183798Z cwltool WARNING: [step s1] completed permanentFail
2018-09-28T19:01:05.380092702Z arvados.cwl-runner INFO: Couldn't update runtime_status: <HttpError 422 when re
questing https://172.17.0.2:8000/arvados/v1/containers/o967z-dz642-4hyignu6z3r43w8?alt=json returned "Runtime
status cannot be modified in state 'Running' ({}, {"warning"=>"cwltool: [step s1] completed permanentFail"})">
```

In this case step s1 is just an "sleep 5; exit 1" script, I was using the same example wf to demo #13373.

#18 - 09/28/2018 08:24 PM - Tom Clegg

Ah, we were only testing updating runtime_status using the dispatcher token, not using the container's own token. Fixed.

10181-incremental-log @ ced2d4772ac338f8f62e83dad1423194f2018c9b

#19 - 10/01/2018 01:33 PM - Lucas Di Pentima

This LGTM, thanks.

#20 - 10/03/2018 03:18 PM - Tom Clegg

- Status changed from In Progress to Resolved

#21 - 10/03/2018 07:24 PM - Tom Clegg

- Related to Story #14284: Send real time container logs to a suitable log distribution system (instead of adding rows to the postgres logs table) added

#22 - 10/03/2018 07:24 PM - Tom Clegg

- Description updated

#23 - 11/13/2018 08:49 PM - Tom Morris

- Release set to 14