

## Arvados - Bug #15928

### Deadlock in crunch-run tests

12/11/2019 02:57 PM - Tom Clegg

<b>Status:</b>	Resolved	<b>Start date:</b>	12/11/2019
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assigned To:</b>	Tom Clegg	<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	2020-01-02 Sprint		
<b>Description</b>			
TestSuite.TestStderrMount hangs (possibly the first test to write multiple outputs?)			
Almost certainly introduced in <a href="#">#15910</a>			
Example: <a href="https://ci.curoverse.com/job/developer-run-tests-remainder/1756/console">https://ci.curoverse.com/job/developer-run-tests-remainder/1756/console</a>			
<b>Subtasks:</b>			
Task # 15929: Review 15928-fs-deadlock			<b>Resolved</b>
<b>Related issues:</b>			
Related to Arvados - Bug #15946: [crunch-run] [collectionfs] Deadlock while w...		<b>Resolved</b>	<b>12/23/2019</b>

### Associated revisions

#### Revision b49229f9 - 12/12/2019 06:23 AM - Tom Clegg

Merge branch '15928-fs-deadlock'

refs #15928

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <[tom@tomclegg.ca](mailto:tom@tomclegg.ca)>

### History

#### #1 - 12/11/2019 02:58 PM - Tom Clegg

```
goroutine 692 [chan receive, 1 minutes]:
git.arvados.org/arvados.git/sdk/go/arvados.(*fileNode).waitPrune(0xc0003ca500)
    /home/tom/arvados/sdk/go/arvados/fs_collection.go:606 +0xf8
git.arvados.org/arvados.git/sdk/go/arvados.(*dirNode).marshalManifest(0xc000468ab0, 0xdc18a0, 0xc00004b880, 0xc00081925c, 0x3, 0x0, 0x0, 0x0, 0x0)
    /home/tom/arvados/sdk/go/arvados/fs_collection.go:903 +0x987
git.arvados.org/arvados.git/sdk/go/arvados.(*dirNode).marshalManifest.func1(0xc0004cf778, 0x0)
    /home/tom/arvados/sdk/go/arvados/fs_collection.go:928 +0x147
git.arvados.org/arvados.git/sdk/go/arvados.(*contextGroup).Go.func1(0xc00004b8c0, 0xc00076a780)
    /home/tom/arvados/sdk/go/arvados/contextgroup.go:69 +0x7c
created by git.arvados.org/arvados.git/sdk/go/arvados.(*contextGroup).Go
    /home/tom/arvados/sdk/go/arvados/contextgroup.go:67 +0xca

goroutine 634 [semacquire, 1 minutes]:
sync.runtime_Semacquire(0xc00004b8e0)
    /usr/local/go/src/runtime/sema.go:56 +0x42
sync.(*WaitGroup).Wait(0xc00004b8d8)
    /usr/local/go/src/sync/waitgroup.go:130 +0x64
git.arvados.org/arvados.git/sdk/go/arvados.(*contextGroup).Wait(0xc00004b8c0, 0x0, 0x0)
    /home/tom/arvados/sdk/go/arvados/contextgroup.go:88 +0x3d
git.arvados.org/arvados.git/sdk/go/arvados.(*dirNode).marshalManifest(0xc000468630, 0xdc18e0, 0xc0000c4018, 0xca46e5, 0x1, 0x0, 0x0, 0x0, 0x0)
    /home/tom/arvados/sdk/go/arvados/fs_collection.go:994 +0x5f4
git.arvados.org/arvados.git/sdk/go/arvados.(*collectionFileSystem).MarshalManifest(0xc00004b300, 0xca46e5, 0x1, 0x0, 0x0, 0x0)
    /home/tom/arvados/sdk/go/arvados/fs_collection.go:189 +0xf2
git.arvados.org/arvados.git/services/crunch-run.(*copier).Copy(0xc0006cd4b8, 0x0, 0xdadb00, 0xcdbc98, 0x0)
    /home/tom/arvados/services/crunch-run/copier.go:108 +0x58d
git.arvados.org/arvados.git/services/crunch-run.(*ContainerRunner).CaptureOutput(0xc0007de900, 0x0, 0x0)
    /home/tom/arvados/services/crunch-run/crunchrun.go:1294 +0x19e
git.arvados.org/arvados.git/services/crunch-run.(*ContainerRunner).Run.func2(0xc0007de900, 0xc000227820)
    /home/tom/arvados/services/crunch-run/crunchrun.go:1591 +0x113
git.arvados.org/arvados.git/services/crunch-run.(*ContainerRunner).Run(0xc0007de900, 0x0, 0x0)
```

```
    /home/tom/arvados/services/crunch-run/crunchrun.go:1665 +0x93a
git.arvados.org/arvados.git/services/crunch-run.(*TestSuite).fullRunHelper(0xc0002f1c40, 0xc0008a82d0, 0xcd1fba, 0x1b6, 0x0, 0x0, 0x0, 0x1, 0xcd9de8, 0xc000791520, ...)
    /home/tom/arvados/services/crunch-run/crunchrun_test.go:822 +0x899
git.arvados.org/arvados.git/services/crunch-run.(*TestSuite).TestStderrMount(0xc0002f1c40, 0xc0008a82d0)
    /home/tom/arvados/services/crunch-run/crunchrun_test.go:1956 +0x88
reflect.Value.Call(0xc9bf00, 0xc0002f1c40, 0x7e13, 0xca4c8c, 0x4, 0xc00051af30, 0x1, 0x1, 0x15080a0, 0xc00051add0, ...)
    /usr/local/go/src/reflect/value.go:460 +0x5f6
reflect.Value.Call(0xc9bf00, 0xc0002f1c40, 0x7e13, 0xc00051af30, 0x1, 0x1, 0xc0006271d0, 0x0, 0x0)
    /usr/local/go/src/reflect/value.go:321 +0xb4
gopkg.in/check%2ev1.(*suiteRunner).forkTest.func1(0xc0008a82d0)
    /home/tom/arvados/tmp/GOPATH/pkg/mod/gopkg.in/check.v1@v1.0.0-20161208181325-20d25e280405/check.go:772 +0x5e4
gopkg.in/check%2ev1.(*suiteRunner).forkCall.func1(0xc0002ef680, 0xc0008a82d0, 0xc0000d3340)
    /home/tom/arvados/tmp/GOPATH/pkg/mod/gopkg.in/check.v1@v1.0.0-20161208181325-20d25e280405/check.go:666 +0xa2
created by gopkg.in/check%2ev1.(*suiteRunner).forkCall
    /home/tom/arvados/tmp/GOPATH/pkg/mod/gopkg.in/check.v1@v1.0.0-20161208181325-20d25e280405/check.go:663 +0x1fb

goroutine 691 [semacquire, 1 minutes]:
sync.runtime_SemacquireMutex(0xc000468b1c, 0xc00065bc00, 0x1)
    /usr/local/go/src/runtime/sema.go:71 +0x47
sync.(*Mutex).lockSlow(0xc000468b18)
    /usr/local/go/src/sync/mutex.go:138 +0xfc
sync.(*Mutex).Lock(...)
    /usr/local/go/src/sync/mutex.go:81
sync.(*RWMutex).Lock(0xc000468b18)
    /usr/local/go/src/sync/rwmutex.go:98 +0x97
git.arvados.org/arvados.git/sdk/go/arvados.(*dirnode).commitBlock.func1(0xc00034bb60, 0xc00076a6c0, 0xc000468ab0, 0xc000173400, 0x6, 0x400, 0xc0004cf600, 0xc00082e160, 0x1, 0x1, ...)
    /home/tom/arvados/sdk/go/arvados/fs_collection.go:706 +0x5ee
created by git.arvados.org/arvados.git/sdk/go/arvados.(*dirnode).commitBlock
    /home/tom/arvados/sdk/go/arvados/fs_collection.go:698 +0x3ff
```

## #2 - 12/11/2019 03:02 PM - Tom Clegg

- Target version set to 2020-01-02 Sprint

## #3 - 12/11/2019 03:48 PM - Tom Clegg

15928-fs-deadlock @ [4257184a0fd276af7e1741dda8a7468a30b4a9c6](https://ci.curoverse.com/view/Developer/job/developer-run-tests/1694/) -- <https://ci.curoverse.com/view/Developer/job/developer-run-tests/1694/>

## #4 - 12/11/2019 06:58 PM - Peter Amstutz

Tom Clegg wrote:

15928-fs-deadlock @ [4257184a0fd276af7e1741dda8a7468a30b4a9c6](https://ci.curoverse.com/view/Developer/job/developer-run-tests/1694/) -- <https://ci.curoverse.com/view/Developer/job/developer-run-tests/1694/>

If I'm understanding correctly, the rationale is that even if the file has been deleted, renamed or moved, it is safe to replace the memSegment with a storedSegment, so tracking possible renames is counterproductive.

LGTM, but having trouble running the test in arvbox (for unrelated reason the API server won't start, see gitter).

## #5 - 12/12/2019 03:10 PM - Tom Clegg

- Status changed from In Progress to Resolved

Peter Amstutz wrote:

If I'm understanding correctly, the rationale is that even if the file has been deleted, renamed or moved, it is safe to replace the memSegment with a storedSegment, so tracking possible renames is counterproductive.

Yes. The previous code avoided deadlock using the "lock all files in this dir, sorted by name" strategy, which can't lock (and therefore can't update) files that are no longer in the same dir. The new code avoids deadlock by locking only one file at a time instead, so the file's name and parent dir are irrelevant.

## #6 - 12/22/2019 12:34 AM - Tom Clegg

- Related to Bug #15946: [crunch-run] [collectionfs] Deadlock while writing output collection added