

## Arvados - Feature #8163

### [FUSE] arv-mount should detect and log any files/dirs that are still open after unmounting

01/07/2016 09:06 PM - Tom Clegg

<b>Status:</b>	New	<b>Start date:</b>	01/07/2016
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assigned To:</b>		<b>% Done:</b>	0%
<b>Category:</b>	FUSE	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			

#### Description

### Background

If you wrap a program in "arv-mount --exec" (e.g., by running a job on a compute node) and the wrapped program exits but some other process has started using the mount (e.g., by reading a file or having cwd in the mount), the FUSE mount will detach but arv-mount will stay alive until all open files/dirs are released.

In practice, this means a background process like "updatedb" can start while a crunch job task is running, and prevent arv-mount from exiting when the task container exits. This is difficult to debug: there's no mount point any more, so the usual "fuser" and "lsof" tools can't help you find the offending process even if you realize this is happening.

### Demonstration

terminal 1	terminal 2
<pre>\$ date; arv-mount MNT --exec sh -c 'sleep 5; date'; date Thu Jan 7 15:35:03 EST 2016 Thu Jan 7 15:35:08 EST 2016 Thu Jan 7 15:35:14 EST 2016</pre>	<pre>\$ date; (cd MNT/home; sleep 10); date Thu Jan 7 15:35:04 EST 2016 Thu Jan 7 15:35:14 EST 2016</pre>

### Proposed improvement 1

When unmounting or receiving SIGUSR1, print (on stderr) a list of processes that still have open files/directories.

- An llfuse.RequestContext has a "pid" field that (I hope) will make this information relatively easy to track and report.
- If it turns out to be much easier to print a message the first time a given PID does some operation after unmounting, or the first time any PID does some operation after we receive SIGUSR1, those options would be nearly as good.

This won't *fix* the problem but it will make it possible for a user/sysadmin to [a] figure out that this is why a job task isn't exiting even though its docker container has exited, and [b] track down which process is responsible for keeping arv-mount alive.

### Proposed improvement 2

When using --exec, after the child exits, return IO errors for all operations. At least in some cases (like updatedb), this will have the desired effect of causing the intruding process to give up reasonably quickly so arv-mount can exit. (It should be possible to control this behavior with a command line switch, though: there might be some use cases where the current behavior is actually desired.)

#### Related issues:

Related to Arvados - Bug #8288: arv-mount / crunchstat in a crunch job fails ...

**Resolved**

**01/23/2016**

#### History

##### #1 - 01/08/2016 02:54 PM - Joshua Randall

I can't reproduce the problem using your example (well, arv-mount has a problem but rather than getting hung it has an unhandled exception error):

Terminal 1:

```\n

```
1. mkdir -p MNT && date; arv-mount MNT --exec sh -c 'sleep 5; date'; date
Fri Jan 8 14:49:20 GMT 2016
Fri Jan 8 14:49:30 GMT 2016
```

```
Exception in thread WebSocketClient:
Traceback (most recent call last):
File "/usr/lib/python2.7/threading.py", line 551, in _bootstrap_inner
self.run()
File "/usr/lib/python2.7/threading.py", line 504, in run
self._target(*self._args, **self._kwargs)
File "/usr/local/lib/python2.7/dist-packages/ws4py/websocket.py", line 430, in run
self.terminate()
File "/usr/local/lib/python2.7/dist-packages/ws4py/websocket.py", line 327, in terminate
self.closed(1006, "Going away")
TypeError: 'bool' object is not callable
```

2016-01-08 14:49:54 arvados.arvados\_fuse<sup>12243</sup> ERROR: Unhandled exception during FUSE operation

```
Traceback (most recent call last):
File "/usr/local/lib/python2.7/dist-packages/arvados_fuse/__init__.py", line 277, in catch_exceptions_wrapper
return orig_func(self, *args, **kwargs)
File "/usr/local/lib/python2.7/dist-packages/arvados_fuse/__init__.py", line 467, in forget
ent = self.inodes[inode]
File "/usr/local/lib/python2.7/dist-packages/arvados_fuse/__init__.py", line 215, in getitem
return self._entries[item]
KeyError: 5L
Terminated
Fri Jan 8 14:49:54 GMT 2016
```

```
1. ps auxwww | grep arv-mount
root 12512 0.0 0.0 9388 932 pts/3 S+ 14:52 0:00 grep --color=auto arv-mount
...
```

Terminal 2:  
...

```
1. date; (cd MNT/home; sleep 10); date
Fri Jan 8 14:49:21 GMT 2016
Fri Jan 8 14:49:54 GMT 2016
...
```

I have python-arvados-fuse 0.1.20151119022705

## #2 - 01/08/2016 03:00 PM - Joshua Randall

Oh, I guess actually mine **is** getting hung until the other process finishes and is **also** having the unhandled exception error.

The issue on our machines is almost certainly not `updatedb`, as that appears to be correctly configured to ignore fuse filesystems (and anything under /tmp).

## #3 - 01/20/2016 06:40 PM - Brett Smith

- Target version set to Arvados Future Sprints

## #4 - 01/25/2016 07:57 PM - Tom Clegg

TypeError: 'bool' object is not callable → looks like the bug fixed in the Python SDK in [a85ea61](#) (4 days after your arvados-fuse version)

## #5 - 07/07/2021 06:27 PM - Ward Vandewege

- Target version deleted (Arvados Future Sprints)