

Base Language - Bug #2695

race conditions when starting an appserver/batch process

09/09/2015 06:44 AM - Constantin Asofiei

Status:	Closed	Start date:	09/09/2015
Priority:	Normal	Due date:	
Assignee:	Constantin Asofiei	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	Cleanup and Stablization for Server Features	case_num:	
billable:	No		
vendor_id:	GCD		
Description			

History

#1 - 09/09/2015 06:51 AM - Constantin Asofiei

When adding batch programs to be started via the schedulers, there are intermittent errors:

```
[09/09/2015 01:44:47 BST] (SessionManager.listen():INFO) {00000000:00000001:standard} Server ready
[09/09/2015 01:44:47 BST] (UpdateAccountTask.run():WARNING) {00000000:00000023:standard} Cannot change 7e691M1
1zN0cs5TQ account
Problem retrieving the spawn command, using these settings: port [3333] host [localhost] alias [standard] UUID
[fadd8254-bac4-4c66-8a08-9a483b44e489]
java.lang.Exception: Authentication failed
    at com.goldencode.p2j.net.SessionManager.createQueue(SessionManager.java:1042)
    at com.goldencode.p2j.net.LeafSessionManager.connectDirect(LeafSessionManager.java:201)
    at com.goldencode.p2j.main.NativeSecureConnection.command(NativeSecureConnection.java:82)
[09/09/2015 01:44:49 BST] (AppServerLauncher:INFO) Appserver 'app_server' was started successfully!
```

This is a result of the fact that each job is executed in a different thread, using the server's context; thus the DirectoryService instance is shared among the threads, and this class is not thread-safe. If a thread calls ds.unbind while another thread has a not-finished ds.openBatch bracket, it closes the batch and the thread with ds.openBatch usage can no longer continue operations on the batch.

I was thinking on synchronizing on the DirectoryService instance; currently I think there are only two places which require this:

1. UpdateAccountTask.run (where AdminServerImpl.setUser is called)
2. AppServerDefinition.refresh - where the appserver details are read

There was also another race condition for TemporaryAccountWorker.callerLatch - this was made null once the task was completed, but if the task is finished before the caller has a chance to start waiting on the latch, a NPE will result.

Branch 2695a was created for this task.

Revision 10933 contains fixes for the above issues - please review.

#2 - 09/09/2015 07:57 AM - Greg Shah

Code Review Task Branch 2695a Revision 10933

I'm OK with the changes.

I think the TemporaryAccountWorker change is hit by MAJIC regression testing, but I don't think the UpdateAccountTask or AppServerDefinition can be hit. How do you want to test this?

#3 - 09/09/2015 09:16 AM - Constantin Asofiei

Greg Shah wrote:

Code Review Task Branch 2695a Revision 10933

I'm OK with the changes.

I think the TemporaryAccountWorker change is hit by MAJIC regression testing, but I don't think the UpdateAccountTask or AppServerDefinition can be hit. How do you want to test this?

Actually TemporaryAccountWorker.run (where the change is) is not hit by MAJIC, unless we connect to it via a web client or use the spawner in another way. I'll double-check to confirm this is still working, and if it does, it can be merged to trunk.

#4 - 09/09/2015 09:36 AM - Greg Shah

I'll double-check to confirm this is still working, and if it does, it can be merged to trunk.

OK, go ahead with this plan.

#5 - 09/09/2015 09:53 AM - Constantin Asofiei

Greg Shah wrote:

I'll double-check to confirm this is still working, and if it does, it can be merged to trunk.

OK, go ahead with this plan.

Web clients in MAJIC are working properly.

#6 - 09/09/2015 09:57 AM - Greg Shah

Please merge to trunk.

#7 - 09/09/2015 10:06 AM - Constantin Asofiei

2695a was merged to trunk rev 10933 and archived.

#8 - 09/09/2015 10:09 AM - Greg Shah

- *Target version set to Milestone 11*
- *Status changed from WIP to Closed*

#9 - 11/16/2016 12:06 PM - Greg Shah

- *Target version changed from Milestone 11 to Cleanup and Stabilization for Server Features*