Hadoop YARN / YARN-4148

When killing app, RM releases app's resource before they are released by NM

⌄ Details

Type:	🔲 Bug	Status:	**RESOLVED**
Priority:	⪢ Major	Resolution:	Fixed
Affects Version/s:	None	Fix Version/s:	2.8.0, 2.9.0, 3.0.0-alpha2
Component/s:	resourcemanager		
Labels:	None		
Target Version/s:	2.8.0		
Hadoop Flags:	Reviewed		

⌄ Description

When killing a app, RM scheduler releases app's resource as soon as possible, then it might allocate these resource for new requests. But NM have not released them at that time.

The problem was found when we supported GPU as a resource(YARN-4122). Test environment: a NM had 6 GPUs, app A used all 6 GPUs, app B was requesting 3 GPUs. Killed app A, then RM released A's 6 GPUs, and allocated 3 GPUs to B. But when B tried to start container on NM, NM found it didn't have 3 GPUs to allocate because it had not released A's GPUs.

I think the problem also exists for CPU/Memory. It might cause OOM when memory is overused.

⌄ Attachments

📄 free_in_scheduler_but_not_node_prototype-branch-2.7.patch	13 kB	23/Jun/16 15:07
📄 YARN-4148.001.patch	21 kB	16/Sep/15 16:54
📄 YARN-4148.002.patch	21 kB	21/Nov/16 22:14
📄 YARN-4148.003.patch	21 kB	12/Dec/16 21:09
📄 YARN-4148.wip.patch	18 kB	15/Sep/15 08:43
📄 YARN-4148-branch-2.8.003.patch	22 kB	10/Jan/17 22:06

⌄ Issue Links

is duplicated by

🔲 YARN-5290 ResourceManager can place more containers on a node than the node size allows	⪢ **RESOLVED**

relates to

🔲 YARN-5197 RM leaks containers if running container disappears from node update	🔖 **CLOSED**

⌄ Activity

↑

⌄ ◉ Jun Gong added a comment - 14/Sep/15 10:49

I have some thoughts.

Proposal A: NM records its total resource and available resource. When launching a container, NM checks available resource and waits until there is enough resource for container. But there might be a time gap from AM's perspective, AM thinks it has launched container, however container might be waiting for its resource.

Proposal B: RM does not release app's resource until containers actually finish and NM releases the resource. It seems a little complex.

I prefer proposal A. Any suggestion or feedback is greatly appreciated.

Jun Gong added a comment - 15/Sep/15 08:55

Attach a WIP patch.

The patch implements proposal B... Its logic is more reasonable.

When completing a container, RM does not release container's resource until it receive container's finished information from NM.

Jun Gong added a comment - 16/Sep/15 16:56

Attach a new patch.

Add a new config to specify whether permits RM release container's resource before NM.

Hadoop QA added a comment - 16/Sep/15 18:35

❌ -1 overall

Vote	Subsystem	Runtime	Comment
-1	pre-patch	17m 4s	Findbugs (version) appears to be broken on trunk.
+1	@author	0m 0s	The patch does not contain any @author tags.
-1	tests included	0m 0s	The patch doesn't appear to include any new or modified tests. Please justify why no new tests are needed for this patch. Also please list what manual steps were performed to verify this patch.
+1	javac	7m 48s	There were no new javac warning messages.
+1	javadoc	10m 1s	There were no new javadoc warning messages.
+1	release audit	0m 24s	The applied patch does not increase the total number of release audit warnings.
-1	checkstyle	1m 19s	The applied patch generated 1 new checkstyle issues (total was 211, now 211).
+1	whitespace	0m 2s	The patch has no lines that end in whitespace.
+1	install	1m 31s	mvn install still works.
+1	eclipse:eclipse	0m 34s	The patch built with eclipse:eclipse.
+1	findbugs	3m 6s	The patch does not introduce any new Findbugs (version 3.0.0) warnings.
-1	yarn tests	0m 21s	Tests failed in hadoop-yarn-api.
-1	yarn tests	54m 45s	Tests failed in hadoop-yarn-server-resourcemanager.
		97m 11s	

Reason	Tests
Failed unit tests	hadoop.yarn.conf.TestYarnConfigurationFields
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestLeafQueue
	hadoop.yarn.server.resourcemanager.resourcetracker.TestNMReconnect
	hadoop.yarn.server.resourcemanager.resourcetracker.TestNMExpiry
Timed out tests	org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.TestNodeLabelContainerAllocation

Subsystem	Report/Notes
Patch URL	http://issues.apache.org/jira/secure/attachment/12756300/YARN-4148.001.patch
Optional Tests	javadoc javac unit findbugs checkstyle

git revision	trunk / bf2f2b4
checkstyle	https://builds.apache.org/job/PreCommit-YARN-Build/9170/artifact/patchprocess/diffcheckstylehadoop-yarn-api.txt
hadoop-yarn-api test log	https://builds.apache.org/job/PreCommit-YARN-Build/9170/artifact/patchprocess/testrun_hadoop-yarn-api.txt
hadoop-yarn-server-resourcemanager test log	https://builds.apache.org/job/PreCommit-YARN-Build/9170/artifact/patchprocess/testrun_hadoop-yarn-server-resourcemanager.txt
Test Results	https://builds.apache.org/job/PreCommit-YARN-Build/9170/testReport/
Java	1.7.0_55
uname	Linux asf905.gq1.ygridcore.net 3.13.0-36-lowlatency #63-Ubuntu SMP PREEMPT Wed Sep 3 21:56:12 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux
Console output	https://builds.apache.org/job/PreCommit-YARN-Build/9170/console

This message was automatically generated.

Jason Darrell Lowe added a comment - 23/Jun/16 15:07

Sorry for joining the discussion late, as I missed this originally. As I mentioned in YARN-5290, having the RM wait until the NM confirms container release can unnecessarily slow down subsequent allocations on other nodes due to scheduler limits (user limit, queue limit, etc.). We could leverage some form of the NM queuing, but I agree it could be confusing when the AM launches a container and it doesn't appear to be active afterwards when querying the node.

We could have the RM wait until it receives hard confirmation from the NM before it releases the resources associated with a container, but that would needlessly slow down scheduling in some cases. For example, if a user is at the scheduler user limit but releases a container on node A, I don't see why we have to wait until that container is confirmed dead over two subsequent NM heartbeats (one to tell the NM to shoot it and another to confirm its dead) before allowing the user to allocate another container of the same size on node B. However I do think it's bad for us to allocate the new container on the same node as the released one since we can accidentally overwhelm the node if the old container isn't cleaned up fast enough.

Therefore I propose that we go ahead and let the scheduler queues and user limit computations update immediately so other nodes can be scheduled, but we don't release the resources in the SchedulerNode itself until the node confirms a previously running container is dead. IMHO if the RM ever sees a container in the RUNNING state on a node, it should never think that node has freed the resources for that container until the node itself says that container has completed.

Here's a prototype patch against branch-2.7 that is similar to what we're using internally to work around this issue. It goes ahead and releases the resources for running containers in the scheduler bookkeeping (i.e.: cluster resource, queues, user limits, etc.) but *not* in the SchedulerNode. So the RM could allocate those resources elsewhere but not on the current node until the node reports the container as completed.

NOTE: with any of these "wait until the node says the container is done" approaches it's important to get the fix for YARN-5197 or if the NM ever skips sending a container completion event the RM will leak those resources on the node.

There is an interesting corner case where the RM has handed out a container to an AM (i.e.: container is in the ACQUIRED state) but it hasn't seen it running on a node yet. If the container is killed by the RM or AM, there's still a chance where the container could appear on the node after the RM has considered those resources freed. We'll have to decide how to handle that race. One way to solve it is to assume the container resources could still be "used" until it has had a chance to tell the NM that the container token for that container is no longer valid and confirmed in a subsequent NM heartbeat that the container has not appeared since. Maybe there's a simpler/faster way to safely free the containers resources for that race condition?

Hadoop QA added a comment - 23/Jun/16 15:09

❌ **-1 overall**

Vote	Subsystem	Runtime	Comment
0	reexec	0m 0s	Docker mode activated.
-1	patch	0m 4s	YARN-4148 does not apply to trunk. Rebase required? Wrong Branch? See https://wiki.apache.org/hadoop/HowToContribute for help.

Subsystem	Report/Notes
JIRA Patch URL	https://issues.apache.org/jira/secure/attachment/12812850/free_in_scheduler_but_not_node_prototype-branch-2.7.patch
JIRA Issue	YARN-4148
Console output	https://builds.apache.org/job/PreCommit-YARN-Build/12119/console
Powered by	Apache Yetus 0.3.0 http://yetus.apache.org

This message was automatically generated.

Jun Gong added a comment - 28/Jun/16 10:26

Sorry for late. Thanks jlowe for your patch, the patch is more reasonable than mine. Assign it to you now.

> We could have the RM wait until it receives hard confirmation from the NM before it releases the resources associated with a container, but that would needlessly slow down scheduling in some cases.

The propose is very reasonable. Not sure whether it works well in your cluster? Killing app is not the only case that leads to mismatch of state between NMs and the RM. When app completed and did not clean some containers, RM needs wait those unfinished containers to finish too. These cases might make RM's available resource look like less than before because RM does not release them actually, will it affect the scheduling speed for a busy cluster?

> One way to solve it is to assume the container resources could still be "used" until it has had a chance to tell the NM that the container token for that container is no longer valid and confirmed in a subsequent NM heartbeat that the container has not appeared since.

How about this idea: RM consider it used until the container becomes RUNNING(then RM kills it) or becomes invalid? However it will makes the resource unavailable even it has been freed.

Gour Saha added a comment - 27/Oct/16 22:18

Hi jlowe do you have any further feedback on hex108's comments? Would you be providing a trunk patch also?

Jason Darrell Lowe added a comment - 27/Oct/16 22:33

I'll try to get a patch for trunk in the next few weeks.

Jason Darrell Lowe added a comment - 21/Nov/16 22:14

Sorry for the delay. I rebased the patch on trunk and added a unit test.

We've been running with this patch on our production clusters for quite some time now, and it works well for us. It simply tracks what the node has reported as running and does not allow the space on the node to be freed up until the node has reported the container as completed. It *does* free up the space in the scheduler queue sense, just not the specific node. Therefore if there is sufficient space in the cluster elsewhere for containers the user limit won't artificially slow down allocation.

This patch does not address the race condition discussed above, so there could still be a case where the RM could over-allocate a node if a container is released by the RM when it is in the ACQUIRED state. The node may be running the container but not yet heartbeated into the RM to let it know, and we will immediately free the space on the node since we never saw it running there. In practice this isn't a significant problem for us, so this patch is working well to fix the most common case where this occurs (i.e.: container is already running for a while then is released by the RM and quickly re-allocated to something else).

Hadoop QA added a comment - 29/Nov/16 15:59

❌ **-1 overall**

Vote	Subsystem	Runtime	Comment
0	reexec	0m 19s	Docker mode activated.
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 2 new or modified test files.

+1	mvninstall	6m 57s	trunk passed
+1	compile	0m 35s	trunk passed
+1	checkstyle	0m 24s	trunk passed
+1	mvnsite	0m 38s	trunk passed
+1	mvneclipse	0m 17s	trunk passed
+1	findbugs	0m 59s	trunk passed
+1	javadoc	0m 21s	trunk passed
+1	mvninstall	0m 32s	the patch passed
+1	compile	0m 30s	the patch passed
+1	javac	0m 30s	the patch passed
-0	checkstyle	0m 21s	hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager: The patch generated 3 new + 231 unchanged - 2 fixed = 234 total (was 233)
+1	mvnsite	0m 36s	the patch passed
+1	mvneclipse	0m 15s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	findbugs	1m 4s	the patch passed
-1	javadoc	0m 18s	hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager generated 1 new + 926 unchanged - 0 fixed = 927 total (was 926)
-1	unit	38m 48s	hadoop-yarn-server-resourcemanager in the patch failed.
+1	asflicense	0m 17s	The patch does not generate ASF License warnings.
		54m 32s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.security.TestDelegationTokenRenewer
	hadoop.yarn.server.resourcemanager.scheduler.TestAbstractYarnScheduler
	hadoop.yarn.server.resourcemanager.TestRMRestart

Subsystem	Report/Notes
Docker	Image:yetus/hadoop:a9ad5d6
JIRA Issue	YARN-4148
JIRA Patch URL	https://issues.apache.org/jira/secure/attachment/12839898/YARN-4148.002.patch
Optional Tests	asflicense compile javac javadoc mvninstall mvnsite unit findbugs checkstyle
uname	Linux b7568047b894 3.13.0-36-lowlatency #63-Ubuntu SMP PREEMPT Wed Sep 3 21:56:12 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/hadoop/patchprocess/precommit/personality/provided.sh
git revision	trunk / 00096dc
Default Java	1.8.0_111
findbugs	v3.0.0
checkstyle	https://builds.apache.org/job/PreCommit-YARN-Build/14098/artifact/patchprocess/diff-checkstyle-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt
javadoc	https://builds.apache.org/job/PreCommit-YARN-Build/14098/artifact/patchprocess/diff-javadoc-javadoc-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt

unit	https://builds.apache.org/job/PreCommit-YARN-Build/14098/artifact/patchprocess/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt
Test Results	https://builds.apache.org/job/PreCommit-YARN-Build/14098/testReport/
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager U: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager
Console output	https://builds.apache.org/job/PreCommit-YARN-Build/14098/console
Powered by	Apache Yetus 0.4.0-SNAPSHOT http://yetus.apache.org

This message was automatically generated.

Jason Darrell Lowe added a comment - 12/Dec/16 21:09

Updating the patch to cleanup the javadoc and checkstyle issues and fixed a race condition in the unit test.

Hadoop QA added a comment - 12/Dec/16 22:27

✅ **+1 overall**

Vote	Subsystem	Runtime	Comment
0	reexec	0m 17s	Docker mode activated.
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 2 new or modified test files.
+1	mvninstall	7m 10s	trunk passed
+1	compile	0m 33s	trunk passed
+1	checkstyle	0m 24s	trunk passed
+1	mvnsite	0m 39s	trunk passed
+1	mvneclipse	0m 19s	trunk passed
+1	findbugs	1m 2s	trunk passed
+1	javadoc	0m 21s	trunk passed
+1	mvninstall	0m 31s	the patch passed
+1	compile	0m 31s	the patch passed
+1	javac	0m 31s	the patch passed
-0	checkstyle	0m 21s	hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager: The patch generated 1 new + 226 unchanged - 2 fixed = 227 total (was 228)
+1	mvnsite	0m 36s	the patch passed
+1	mvneclipse	0m 15s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	findbugs	1m 4s	the patch passed
+1	javadoc	0m 19s	the patch passed
+1	unit	42m 24s	hadoop-yarn-server-resourcemanager in the patch passed.
+1	asflicense	0m 16s	The patch does not generate ASF License warnings.
		58m 19s	

Subsystem	Report/Notes
Docker	Image:yetus/hadoop:a9ad5d6
JIRA Issue	YARN-4148
JIRA Patch	https://issues.apache.org/jira/secure/attachment/12842873/YARN-4148.003.patch

URL	
Optional Tests	asflicense compile javac javadoc mvninstall mvnsite unit findbugs checkstyle
uname	Linux 58b63401ae30 3.13.0-95-generic #142-Ubuntu SMP Fri Aug 12 17:00:09 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/hadoop/patchprocess/precommit/personality/provided.sh
git revision	trunk / f66f618
Default Java	1.8.0_111
findbugs	v3.0.0
checkstyle	https://builds.apache.org/job/PreCommit-YARN-Build/14264/artifact/patchprocess/diff-checkstyle-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt
Test Results	https://builds.apache.org/job/PreCommit-YARN-Build/14264/testReport/
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager U: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager
Console output	https://builds.apache.org/job/PreCommit-YARN-Build/14264/console
Powered by	Apache Yetus 0.5.0-SNAPSHOT http://yetus.apache.org

This message was automatically generated.

Junping Du added a comment - 06/Jan/17 22:28

Latest patch LGTM. Kick off jenkins again given it is has been a while since last Jenkins report. +1 pending on jenkins result.

Hadoop QA added a comment - 07/Jan/17 01:08

❌ **-1 overall**

Vote	Subsystem	Runtime	Comment
0	reexec	0m 15s	Docker mode activated.
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 2 new or modified test files.
+1	mvninstall	14m 20s	trunk passed
+1	compile	0m 39s	trunk passed
+1	checkstyle	0m 25s	trunk passed
+1	mvnsite	0m 41s	trunk passed
+1	mvneclipse	0m 18s	trunk passed
+1	findbugs	1m 15s	trunk passed
+1	javadoc	0m 24s	trunk passed
+1	mvninstall	0m 39s	the patch passed
+1	compile	0m 37s	the patch passed
+1	javac	0m 37s	the patch passed
-0	checkstyle	0m 23s	hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager: The patch generated 1 new + 220 unchanged - 2 fixed = 221 total (was 222)
+1	mvnsite	0m 40s	the patch passed
+1	mvneclipse	0m 16s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	findbugs	1m 24s	the patch passed
+1	javadoc	0m 22s	the patch passed

-1	unit	39m 41s	hadoop-yarn-server-resourcemanager in the patch failed.
+1	asflicense	0m 17s	The patch does not generate ASF License warnings.
		63m 53s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.TestRMRestart
	hadoop.yarn.server.resourcemanager.security.TestDelegationTokenRenewer

Subsystem	Report/Notes
Docker	Image:yetus/hadoop:a9ad5d6
JIRA Issue	YARN-4148
JIRA Patch URL	https://issues.apache.org/jira/secure/attachment/12842873/YARN-4148.003.patch
Optional Tests	asflicense compile javac javadoc mvninstall mvnsite unit findbugs checkstyle
uname	Linux 5a5e206acb7a 3.13.0-95-generic #142-Ubuntu SMP Fri Aug 12 17:00:09 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/hadoop/patchprocess/precommit/personality/provided.sh
git revision	trunk / 71a4acf
Default Java	1.8.0_111
findbugs	v3.0.0
checkstyle	https://builds.apache.org/job/PreCommit-YARN-Build/14594/artifact/patchprocess/diff-checkstyle-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt
unit	https://builds.apache.org/job/PreCommit-YARN-Build/14594/artifact/patchprocess/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt
Test Results	https://builds.apache.org/job/PreCommit-YARN-Build/14594/testReport/
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager U: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager
Console output	https://builds.apache.org/job/PreCommit-YARN-Build/14594/console
Powered by	Apache Yetus 0.5.0-SNAPSHOT http://yetus.apache.org

This message was automatically generated.

⌄ ◉ Jason Darrell Lowe added a comment - 09/Jan/17 22:46 - edited

The unit test failures appear to be unrelated. They pass for me locally with the patch applied, and there are JIRAs that are tracking those failures. The TestDelegationTokenRenewer failure is being tracked by YARN-5816 and the TestRMRestart failure is tracked by YARN-5548.

Thanks for the review, junping_du! If you agree the failures are unrelated then feel free to commit, or I'll do so in a few days unless I hear otherwise.

⌄ ◉ Junping Du added a comment - 09/Jan/17 23:18

Yes, I agree that the two test failures are not related to the patch. Thanks jlowe for reminding me. Committing it now.

⌄ ◉ Junping Du added a comment - 10/Jan/17 02:51

I just commit the 003 patch to trunk and branch-2. For branch-2.8, there are several conflicts there.
Hi jlowe, do we want this commit get landed in branch-2.8? If so, can you put a patch here for branch-2.8? Thx!

⌄ ◯ **Hudson** added a comment - 10/Jan/17 02:56

SUCCESS: Integrated in Jenkins build Hadoop-trunk-Commit #11095 (See https://builds.apache.org/job/Hadoop-trunk-Commit/11095/)
YARN-4148. When killing app, RM releases app's resource before they are (junping_du: rev 945db55f2e6521d33d4f90bbb09179b0feba5e7a)

- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/common/fica/FiCaSchedulerNode.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/AbstractYarnScheduler.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestChildQueueOrder.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fifo/FifoScheduler.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/SchedulerNode.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/TestAbstractYarnScheduler.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/FairScheduler.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/LeafQueue.java

⌄ ◯ **Jason Darrell Lowe** added a comment - 10/Jan/17 22:06

Attaching the patch for branch-2.8.

⌄ ◯ **Junping Du** added a comment - 10/Jan/17 22:17

Thanks Jason. I reopen the ticket for kicking off the jenkins.

⌄ ◯ **Hadoop QA** added a comment - 11/Jan/17 01:23

❌ **-1 overall**

Vote	Subsystem	Runtime	Comment
0	reexec	0m 20s	Docker mode activated.
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 2 new or modified test files.
+1	mvninstall	7m 30s	branch-2.8 passed
+1	compile	0m 38s	branch-2.8 passed with JDK v1.8.0_111
+1	compile	0m 38s	branch-2.8 passed with JDK v1.7.0_121
+1	checkstyle	0m 24s	branch-2.8 passed
+1	mvnsite	0m 40s	branch-2.8 passed
+1	mvneclipse	0m 18s	branch-2.8 passed
+1	findbugs	1m 20s	branch-2.8 passed
+1	javadoc	0m 26s	branch-2.8 passed with JDK v1.8.0_111
+1	javadoc	0m 27s	branch-2.8 passed with JDK v1.7.0_121
+1	mvninstall	0m 35s	the patch passed
+1	compile	0m 29s	the patch passed with JDK v1.8.0_111
+1	javac	0m 29s	the patch passed
+1	compile	0m 34s	the patch passed with JDK v1.7.0_121
+1	javac	0m 34s	the patch passed

-0	checkstyle	0m 21s	hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager: The patch generated 1 new + 364 unchanged - 2 fixed = 365 total (was 366)
+1	mvnsite	0m 37s	the patch passed
+1	mvneclipse	0m 14s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	findbugs	1m 29s	the patch passed
+1	javadoc	0m 25s	the patch passed with JDK v1.8.0_111
+1	javadoc	0m 23s	the patch passed with JDK v1.7.0_121
-1	unit	77m 4s	hadoop-yarn-server-resourcemanager in the patch failed with JDK v1.7.0_121.
+1	asflicense	0m 20s	The patch does not generate ASF License warnings.
		172m 41s	

Reason	Tests
JDK v1.8.0_111 Failed junit tests	hadoop.yarn.server.resourcemanager.TestAMAuthorization
	hadoop.yarn.server.resourcemanager.TestWorkPreservingRMRestart
	hadoop.yarn.server.resourcemanager.TestClientRMTokens
JDK v1.7.0_121 Failed junit tests	hadoop.yarn.server.resourcemanager.TestAMAuthorization
	hadoop.yarn.server.resourcemanager.TestWorkPreservingRMRestart
	hadoop.yarn.server.resourcemanager.TestClientRMTokens

Subsystem	Report/Notes
Docker	Image:yetus/hadoop:5af2af1
JIRA Issue	YARN-4148
JIRA Patch URL	https://issues.apache.org/jira/secure/attachment/12846682/YARN-4148-branch-2.8.003.patch
Optional Tests	asflicense compile javac javadoc mvninstall mvnsite unit findbugs checkstyle
uname	Linux 051f77f91850 3.13.0-105-generic #152-Ubuntu SMP Fri Dec 2 15:37:11 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/hadoop/patchprocess/precommit/personality/provided.sh
git revision	branch-2.8 / f5e837e
Default Java	1.7.0_121
Multi-JDK versions	/usr/lib/jvm/java-8-oracle:1.8.0_111 /usr/lib/jvm/java-7-openjdk-amd64:1.7.0_121
findbugs	v3.0.0
checkstyle	https://builds.apache.org/job/PreCommit-YARN-Build/14630/artifact/patchprocess/diff-checkstyle-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt
unit	https://builds.apache.org/job/PreCommit-YARN-Build/14630/artifact/patchprocess/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager-jdk1.7.0_121.txt
JDK v1.7.0_121 Test Results	https://builds.apache.org/job/PreCommit-YARN-Build/14630/testReport/
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager U: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager
Console output	https://builds.apache.org/job/PreCommit-YARN-Build/14630/console
Powered by	Apache Yetus 0.5.0-SNAPSHOT http://yetus.apache.org

This message was automatically generated.

⌄ ◯ Junping Du added a comment - 11/Jan/17 01:59

The test failures should be unrelated to the patch. TestAMAuthorization and TestClientRMTokens get tracked in HADOOP-12687 and TestWorkPreservingRMRestart get tracked in ~~YARN-5349~~.
+1 on 2.8 patch. Committing it now.

⌄ ◯ Junping Du added a comment - 11/Jan/17 02:11

I have commit the patch to branch-2.8 and branch-2.8.0. Thanks jlowe for delivering the fix! Also, thanks hex108 for reporting the issue and gsaha for review comments.

⌄ People

Assignee:

◯ Jason Darrell Lowe

Reporter:

◯ Jun Gong

Votes:

0 Vote for this issue

Watchers:

20 Start watching this issue

⌄ Dates

Created:

11/Sep/15 14:04

Updated:

25/Oct/19 20:26

Resolved:

11/Jan/17 02:11