Hadoop HDFS / HDFS-9289

Make DataStreamer#block thread safe and verify genStamp in commitBlock

Details

Type:	Bug	Status:	CLOSED
Priority:	Critical	Resolution:	Fixed
Affects Version/s:	2.7.1	Fix Version/s:	2.8.0, 2.7.2, 2.6.3, 3.0.0-alpha1
Component/s:	None		
Labels:	None		
Target Version/s:	2.7.3		
Hadoop Flags:	Reviewed		

Description

we have seen a case of corrupt block which is caused by file complete after a pipelineUpdate, but the file complete with the old block genStamp. This caused the replicas of two datanodes in updated pipeline to be viewed as corrupte. Propose to check genstamp when commit block

Attachments

HDFS-9289.1.patch	3 kB	22/Oct/15 19:15
HDFS-9289.2.patch	10 kB	23/Oct/15 16:27
HDFS-9289.3.patch	11 kB	27/Oct/15 20:46
HDFS-9289.4.patch	2 kB	28/Oct/15 21:47
HDFS-9289.5.patch	12 kB	29/Oct/15 22:24
HDFS-9289.6.patch	10 kB	29/Oct/15 22:40
HDFS-9289.7.patch	12 kB	03/Nov/15 16:31
HDFS-9289.branch-2.7.patch	12 kB	04/Nov/15 17:30
HDFS-9289.branch-2.patch	11 kB	03/Nov/15 22:18
HDFS-9289-branch-2.6.patch	12 kB	21/Nov/15 07:55

Issue Links

is related to

HDFS-9298 remove replica and not add replica with wrong genStamp	RESOLVED

relates to

HDFS-8498 Blocks can be committed with wrong size	RESOLVED

Activity

Elliott Neil Clark added a comment - 22/Oct/15 21:05

We just had this something very similar happen on a prod cluster. Then the datanode holding the only complete block was shut off for repair.

```
15/10/22 06:29:32 INFO hdfs.StateChange: BLOCK* allocateBlock: /TESTCLUSTER-
HBASE/WALs/hbase4544.test.com,16020,1444266312515/hbase4544.test.com%2C16020%2C1444266312515.default.1445520572440. BP-
1735829752-10.210.49.21-1437433901380 blk_1190230043_116735085{blockUCState=UNDER_CONSTRUCTION, primaryNodeIndex=-1,
replicas=[ReplicaUnderConstruction[[DISK]DS-8d0a91de-8a69-4f39-816e-de3a0fa8a3aa:NORMAL:10.210.81.33:50010|RBW],
ReplicaUnderConstruction[[DISK]DS-52d9a122-a46a-4129-ab3d-d9041de109f8:NORMAL:10.210.31.48:50010|RBW],
ReplicaUnderConstruction[[DISK]DS-c734b72e-27de-4dd4-a46c-7ae59f6ef792:NORMAL:10.210.31.38:50010|RBW]]}
15/10/22 06:32:48 INFO namenode.FSNamesystem: updatePipeline(block=BP-1735829752-10.210.49.21-
1437433901380:blk_1190230043_116735085, newGenerationStamp=116737586, newLength=201675125, newNodes=
[10.210.81.33:50010, 10.210.81.45:50010, 10.210.64.29:50010], clientName=DFSClient_NONMAPREDUCE_1976436475_1)
15/10/22 06:32:48 INFO namenode.FSNamesystem: updatePipeline(BP-1735829752-10.210.49.21-
1437433901380:blk_1190230043_116735085) successfully to BP-1735829752-10.210.49.21-
1437433901380:blk_1190230043_116737586
15/10/22 06:32:50 INFO BlockStateChange: BLOCK* addStoredBlock: blockMap updated: 10.210.64.29:50010 is added to
blk_1190230043_116737586{blockUCState=UNDER_CONSTRUCTION, primaryNodeIndex=-1, replicas=
[ReplicaUnderConstruction[[DISK]DS-8d0a91de-8a69-4f39-816e-de3a0fa8a3aa:NORMAL:10.210.81.33:50010|RBW],
ReplicaUnderConstruction[[DISK]DS-d5f7fff9-005d-4804-a223-b6e6624d3af2:NORMAL:10.210.81.45:50010|RBW],
ReplicaUnderConstruction[[DISK]DS-0620aef7-b6b2-4a23-950c-09373f68a815:NORMAL:10.210.64.29:50010|FINALIZED]]} size
201681322
15/10/22 06:32:50 INFO BlockStateChange: BLOCK* addStoredBlock: blockMap updated: 10.210.81.45:50010 is added to
blk_1190230043_116737586{blockUCState=UNDER_CONSTRUCTION, primaryNodeIndex=-1, replicas=
[ReplicaUnderConstruction[[DISK]DS-8d0a91de-8a69-4f39-816e-de3a0fa8a3aa:NORMAL:10.210.81.33:50010|RBW],
ReplicaUnderConstruction[[DISK]DS-0620aef7-b6b2-4a23-950c-09373f68a815:NORMAL:10.210.64.29:50010|FINALIZED],
```

The patch will help but the issue will still be there. Is there some way to keep the genstamps from getting out of sync?

Elliott Neil Clark added a comment - 22/Oct/15 21:28

Also can we add the expected and encountered genstamps to the exception message

Hadoop QA added a comment - 22/Oct/15 21:34

☒ **-1 overall**

Vote	Subsystem	Runtime	Comment
-1	pre-patch	18m 50s	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	8m 59s	There were no new javac warning messages.
+1	javadoc	11m 27s	There were no new javadoc warning messages.
+1	release audit	0m 32s	The applied patch does not increase the total number of release audit warnings.
+1	checkstyle	0m 46s	There were no new checkstyle issues.
+1	whitespace	0m 0s	The patch has no lines that end in whitespace.
+1	install	2m 9s	mvn install still works.
+1	eclipse:eclipse	0m 36s	The patch built with eclipse:eclipse.
-1	findbugs	0m 30s	Post-patch findbugs hadoop-hdfs-project/hadoop-hdfs compilation is broken.
+1	findbugs	0m 30s	The patch does not introduce any new Findbugs (version) warnings.
+1	native	0m 13s	Pre-build of native portion
-1	hdfs tests	0m 25s	Tests failed in hadoop-hdfs.
		44m 31s	

Reason	Tests
Failed build	hadoop-hdfs

Subsystem	Report/Notes
Patch URL	http://issues.apache.org/jira/secure/attachment/12768113/HDFS-9289.1.patch

Optional Tests	javadoc javac unit findbugs checkstyle
git revision	trunk / 0fce5f9
hadoop-hdfs test log	https://builds.apache.org/job/PreCommit-HDFS-Build/13136/artifact/patchprocess/testrun_hadoop-hdfs.txt
Test Results	https://builds.apache.org/job/PreCommit-HDFS-Build/13136/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-HDFS-Build/13136/console

This message was automatically generated.

Chang Li added a comment - 22/Oct/15 21:55

Hi eclark, I think the case you gave is not the same and the corrupt block doesn't seem to be caused by gen stamp reverse mismatch. So your initial pipeline has node 33, 48, 38. Then after pipeline update it has node 33, 45, 29. Then node 38 is marked corrupt due to gen stamp mismatch, which is what it should be. Then node 29(with correct gen stamp) were told to replicate to some other node, and then client report block node 29 as corrupt. This case of corruption doesn't seem like to be caused by gen stamp mismatch on namenode side but a report from client ("because client machine reported it")

Chang Li added a comment - 22/Oct/15 21:56

will update patch with info of expected and encountered gen stamp and unit test soon.

Elliott Neil Clark added a comment - 23/Oct/15 00:50

```
15/10/22 09:37:36 INFO BlockStateChange: BLOCK NameSystem.addToCorruptReplicasMap: blk_1190230043 added as corrupt on
10.210.31.38:50010 by hbase4678.test.com/10.210.31.38 because reported RBW replica with genstamp 116735085 does not
match COMPLETE block's genstamp in block map 116737586
```

Block lengths on "corrupt" replicas is the same as on the non-corrupt. The only difference is the genstamp.

Chang Li added a comment - 23/Oct/15 16:27

.2 patch include test. also include info of encountered genStamp and expected genStamp in exception

Chang Li added a comment - 23/Oct/15 16:29

eclark, block on 10.210.31.38 should be marked as corrupt because it's from old pipeline right?

Elliott Neil Clark added a comment - 23/Oct/15 18:44

It had all of the data and the same md5sums when I checked. So the only thing different was genstamps. Not really sure about why that happened. But I didn't mean to side track this jira.

Test looks nice.

Zhe Zhang added a comment - 24/Oct/15 00:42

lichangleo Thanks for reporting the issue.

> but the file complete with the old block genStamp.

How did that happen? So the client somehow had an old GS? IIUC the `updatePipeline` protocol is below (using `client_GS`, `DN_GS`, and `NN_GS` to denote the 3 copies of GS):

1. Client asks for new GS from NN through `updateBlockForPipeline`. After this, `client_GS` is new, both `DN_GS` and `NN_GS` are old
2. Client calls `createBlockOutputStream` to update DN's GS. After this, both `client_GS` and `DN_GS` are new, `NN_GS` is old
3. Client calls `updatePipeline`. After this, all 3 GSes should be new

Maybe step 3 failed, and then client tried to complete the file? It'd be ideal if you could extend the unit test to reproduce the error without the fix (or paste the error log). Thanks!

Chang Li added a comment - 24/Oct/15 15:38

Hi zhz, here is the log,

```
INFO hdfs.StateChange: BLOCK* allocateBlock:
/projects/FETLDEV/Benzene/benzene_stg_transient/primer/201510201900/_temporary/1/_temporary/attempt_1444859775697_31140_
m-01028. BP-1052427332-98.138.108.146-1350583571998 blk_3773617405_1106111498065{blockUCState=UNDER_CONSTRUCTION,
primaryNodeIndex=-1, replicas=[ReplicaUnderConstruction[[DISK]DS-0a28b82a-e3fb-4e42-b925-
e76ebd98afb4:NORMAL:10.216.32.61:1004|RBW], ReplicaUnderConstruction[[DISK]DS-236c19ee-0a39-4e53-9520-
c32941ca1828:NORMAL:10.216.70.49:1004|RBW], ReplicaUnderConstruction[[DISK]DS-fc7c2dab-9309-46be-b5c0-
52be8e698591:NORMAL:10.216.70.43:1004|RBW]]}
2015-10-20 19:49:20,392 [IPC Server handler 63 on 8020] INFO namenode.FSNamesystem: updatePipeline(block=BP-1052427332-
98.138.108.146-1350583571998:blk_3773617405_1106111498065, newGenerationStamp=1106111511603, newLength=107761275,
newNodes=[10.216.70.49:1004, 10.216.70.43:1004],
clientName=DFSClient_attempt_1444859775697_31140_m_001028_0_1424303982_1)
2015-10-20 19:49:20,392 [IPC Server handler 63 on 8020] INFO namenode.FSNamesystem: updatePipeline(BP-1052427332-
98.138.108.146-1350583571998:blk_3773617405_1106111498065) successfully to BP-1052427332-98.138.108.146-
1350583571998:blk_3773617405_1106111511603
2015-10-20 19:49:20,400 [IPC Server handler 96 on 8020] INFO hdfs.StateChange: DIR* completeFile:
/projects/FETLDEV/Benzene/benzene_stg_transient/primer/201510201900/_temporary/1/_temporary/attempt_1444859775697_31140_
m-01028 is closed by DFSClient_attempt_1444859775697_31140_m_001028_0_1424303982_1
```

You can see the file complete after a pipeline update. The block changed its genStamp from blk_3773617405_1106111498065 to blk_3773617405_1106111511603. But then the two nodes in the updated pipeline are marked as corrupted. When I do fsck, it shows

```

hdfs fsck /projects/FETLDEV/Benzene/benzene_stg_transient/primer/201510201900/part-m-01028
Connecting to namenode via http://uraniumtan-nn1.tan.ygrid.yahoo.com:50070
FSCK started by hdfs (auth:KERBEROS_SSL) from /98.138.131.190 for path
/projects/FETLDEV/Benzene/benzene_stg_transient/primer/201510201900/part-m-01028 at Wed Oct 21 15:04:56 UTC 2015
.
/projects/FETLDEV/Benzene/benzene_stg_transient/primer/201510201900/part-m-01028: CORRUPT blockpool BP-1052427332-
98.138.108.146-1350583571998 block blk_3773617405

/projects/FETLDEV/Benzene/benzene_stg_transient/primer/201510201900/part-m-01028:  Replica placement policy is violated
for BP-1052427332-98.138.108.146-1350583571998:blk_3773617405_1106111498065. Block should be additionally replicated on
1 more rack(s).
```

it shows the blk with old gen stamp blk_3773617405_1106111498065.

Chang Li added a comment - 26/Oct/15 16:21

have met another case in our cluster

```
2015-10-23 04:38:08,544 [IPC Server handler 11 on 8020] INFO hdfs.StateChange: BLOCK* allocateBlock:
/projects/wcc/wcc1/data/2015/10/22/05/Content-9892.temp.
gz.temp._COPYING_. BP-1161836467-98.137.240.59-1438814573258 blk_1427767166_354062734{blockUCState=UNDER_CONSTRUCTION,
primaryNodeIndex=-1, replicas=[Replica
UnderConstruction[[DISK]DS-a04f60ed-6700-4e93-8a52-555301e07d3b:NORMAL:10.213.43.41:1004|RBW],
ReplicaUnderConstruction[[DISK]DS-7e0de56b-17ba-4164-8b19-67a9
f9f84c2c:NORMAL:10.213.46.123:1004|RBW], ReplicaUnderConstruction[[DISK]DS-14a850d1-deb9-496b-b5ed-
bb57010a8b56:NORMAL:10.213.46.96:1004|RBW]]}
2015-10-23 04:39:35,588 [IPC Server handler 5 on 8020] INFO namenode.FSNamesystem: updatePipeline(block=BP-1161836467-
98.137.240.59-1438814573258:blk_1427767
166_354062734, newGenerationStamp=354080525, newLength=24505255, newNodes=[10.213.46.123:1004, 10.213.46.96:1004],
clientName=DFSClient_NONMAPREDUCE_12621589
81_1)
2015-10-23 04:39:35,588 [IPC Server handler 5 on 8020] INFO namenode.FSNamesystem: updatePipeline(BP-1161836467-
98.137.240.59-1438814573258:blk_1427767166_35
4062734) successfully to BP-1161836467-98.137.240.59-1438814573258:blk_1427767166_354080525
2015-10-23 04:39:35,595 [IPC Server handler 50 on 8020] INFO hdfs.StateChange: DIR* completeFile:
/projects/wcc/wcc1/data/2015/10/22/05/Content-9892.temp.gz.temp._COPYING_ is closed by
DFSClient_NONMAPREDUCE_1262158981_1
```

This is also a complete file before a pipelineupdate.
jsp page shows three nodes which currently holds the replica of blk_1427767166. One of the node is 10.213.43.41, which is the first node in old pipeline and dropped out in the updated pipeline and the replica currently in that node has old gen stamp. The other two nodes are later replicated after the first node in old pipeline sent in its block report. The two nodes in the updated pipeline were marked as corrupted until the node 10.213.43.41 sent in its block report.

Chang Li added a comment - 26/Oct/15 16:25

zhz, before we figure out the root cause of this strange case, should let this jira to be a temporary fix?

Zhe Zhang added a comment - 26/Oct/15 16:50

lichangleo Thanks for sharing the logs! I'll look at the patch and logs and post a review today.

Jing Zhao added a comment - 26/Oct/15 18:01

Hi lichangleo, what is the current conf setting of the replace-datanode-on-failure policy in your cluster? From the log looks like you disabled it?

Chang Li added a comment - 26/Oct/15 18:12

Hi jingzhao, we are currently taking the default one, and the default is true.

Zhe Zhang added a comment - 26/Oct/15 23:35

lichangleo I think the below log shows that the client does have new GS `1106111511603` because the parameter `newBlock` is passed in from the client. So IIUC even if we check GS when completing file, as the patch does, it won't stop the client from completing / closing the file. Or could you describe how you think the patch can avoid this error? Thanks..

```
2015-10-20 19:49:20,392 [IPC Server handler 63 on 8020] INFO namenode.FSNamesystem: updatePipeline(BP-1052427332-
98.138.108.146-1350583571998:blk_3773617405_1106111498065) successfully to BP-1052427332-98.138.108.146-
1350583571998:blk_3773617405_1106111511603
```

Chang Li added a comment - 26/Oct/15 23:55

zhz, you are the right, the client had the new genstamp, but the problem I am trying to point out is that the client after updatepipeline with the new gen stamp it later completed file with the old gen stamp. So my patch is trying to prevent client from complete file with old genstamp after it updated pipeline with new genstamp

Walter Su added a comment - 27/Oct/15 06:53

The patch hides a potential bigger bug. We should find it out and address it.
Hi, lichangleo. I'll very appreciate if you could enable debug level of `NameNode.blockStateChangeLog` and attach more logs? Or instructions about how to reproduce it.

Chang Li added a comment - 27/Oct/15 14:28

Hi, walter.k.su, I don't know in which cluster this strange case will happen again and I can't enable the debug message of NameNode.blockStateChangeLog across all clusters. I will look into the root cause of how this strange problem happen.

Zhe Zhang added a comment - 27/Oct/15 17:04

> the client after updatepipeline with the new gen stamp it later completed file with the old gen stamp

This looks very strange. But why do you think this happened? Did you see logs that the file was completed with an old GS?

Chang Li added a comment - 27/Oct/15 17:40

zhz, I don't have the log show the file was completed with an old GS. But by look up the block from jsp page right now, I can see that the block blk_3773617405 currently has replica on host **657n26.*.com, ***656n04.*.com, and ***656n38.**.com,
by going to those datanode, I see the replica on those datanodes have replica with old genstamp.

```
bash-4.1$ hostname
***657n26.***.com
bash-4.1$ ls -l /grid/2/hadoop/var/hdfs/data/current/BP-1052427332-98.138.108.146-
1350583571998/current/finalized/subdir236/subdir212/blk_3773617405*
-rw-r--r-- 1 hdfs users 107761275 Oct 23 18:00 /grid/2/hadoop/var/hdfs/data/current/BP-1052427332-98.138.108.146-
1350583571998/current/finalized/subdir236/subdir212/blk_3773617405
-rw-r--r-- 1 hdfs users    841895 Oct 23 18:00 /grid/2/hadoop/var/hdfs/data/current/BP-1052427332-98.138.108.146-
1350583571998/current/finalized/subdir236/subdir212/blk_3773617405_1106111498065.meta

bash-4.1$ hostname
***656n04.***.com
bash-4.1$ ls -l /grid/1/hadoop/var/hdfs/data/current/BP-1052427332-98.138.108.146-
1350583571998/current/finalized/subdir236/subdir212/blk_3773617405*
-rw-r--r-- 1 hdfs users 107761275 Oct 21 19:14 /grid/1/hadoop/var/hdfs/data/current/BP-1052427332-98.138.108.146-
1350583571998/current/finalized/subdir236/subdir212/blk_3773617405
-rw-r--r-- 1 hdfs users    841895 Oct 21 19:14 /grid/1/hadoop/var/hdfs/data/current/BP-1052427332-98.138.108.146-
```

```
1350583571998/current/finalized/subdir236/subdir212/blk_3773617405_1106111498065.meta

bash-4.1$ hostname
***656n38.***.com
bash-4.1$ ls -l /grid/3/hadoop/var/hdfs/data/current/BP-1052427332-98.138.108.146-
1350583571998/current/finalized/subdir236/subdir212/blk_3773617405*
```

Zhe Zhang added a comment - 27/Oct/15 18:36

The fact that all 3 DNs have old GS doesn't mean the client also has an old GS. Is the above log from the same cluster as previous logs ?

In these cases, is there any replica with the correct (new) GS? If so it doesn't look a bug. If all replicas of a block have old GS, then it's more suspicious.

Chang Li added a comment - 27/Oct/15 20:46

.3 patch set block in DataStreamer to volatile

Chang Li added a comment - 27/Oct/15 20:58

zhz, yes, the above log is from the same cluster as the first log I post.

The two replicas in two datanodes from updated pipeline had new GS but they were marked as corrupt because the block commit with old genstamp.
The complete story happened in that cluster is: there were initially 3 datanodes in pipeline d1, d2, d3. Then pipelineupdate happen with only d2 and d3 with new GS. Then file complete with old GS and d2 and d3 were marked corrupt. Then after 1 day, full block report from d1 came in, and NN found out d1 has the the right block with "correct" old GS but d1 is under replicated, so NN told d1 to replicate its replica with old GS to the other two nodes, d4, d5. So the all 3DNs I showed above were d1, d4, and d5 having old GS.
I think there probabaly exist some cache coherence issue since

```
protected ExtendedBlock block;
```

lack volatile. That could also explain why this issue didn't happen frequently.

Zhe Zhang added a comment - 27/Oct/15 22:08

> I think there probabaly exist some cache coherence issue

This sounds possible. Maybe the `DFSOutputStream` thread uses a stale copy of `block` in `completeFile`, after `block` is updated by the `DataStreamer` thread.

> Then pipelineupdate happen with only d2 and d3 with new GS. Then file complete with old GS and d2 and d3 were marked corrupt.

Do you have any log showing that "replica marked as corrupt because its GS is newer than the block GS on NN"?

Regardless, making `DataStreamer#block` volatile is a good change. Ideally we should add a test to emulate the cache coherency problem but it doesn't look easy.

Hadoop QA added a comment - 28/Oct/15 04:00

☒ -1 overall

Vote	Subsystem	Runtime	Comment
-1	pre-patch	30m 31s	Pre-patch trunk has 1 extant Findbugs (version 3.0.0) warnings.
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	tests included	0m 0s	The patch appears to include 2 new or modified test files.
+1	javac	11m 6s	There were no new javac warning messages.

+1	javadoc	16m 4s	There were no new javadoc warning messages.
+1	release audit	0m 39s	The applied patch does not increase the total number of release audit warnings.
-1	checkstyle	3m 30s	The applied patch generated 1 new checkstyle issues (total was 161, now 161).
+1	whitespace	0m 1s	The patch has no lines that end in whitespace.
+1	install	2m 33s	mvn install still works.
+1	eclipse:eclipse	1m 5s	The patch built with eclipse:eclipse.
+1	findbugs	8m 5s	The patch does not introduce any new Findbugs (version 3.0.0) warnings.
+1	native	5m 22s	Pre-build of native portion
-1	hdfs tests	78m 6s	Tests failed in hadoop-hdfs.
+1	hdfs tests	0m 59s	Tests passed in hadoop-hdfs-client.
		159m 31s	

Reason	Tests
Failed unit tests	hadoop.hdfs.server.datanode.TestDataNodeHotSwapVolumes
	hadoop.hdfs.server.datanode.TestDirectoryScanner
	hadoop.hdfs.TestRecoverStripedFile
	hadoop.hdfs.server.namenode.ha.TestEditLogTailer
	hadoop.hdfs.server.blockmanagement.TestPendingInvalidateBlock
	hadoop.hdfs.TestEncryptionZones

Subsystem	Report/Notes
Patch URL	http://issues.apache.org/jira/secure/attachment/12769100/HDFS-9289.3.patch
Optional Tests	javadoc javac unit findbugs checkstyle
git revision	trunk / 68ce93c
Pre-patch Findbugs warnings	https://builds.apache.org/job/PreCommit-HDFS-Build/13238/artifact/patchprocess/trunkFindbugsWarningshadoop-hdfs.html
checkstyle	https://builds.apache.org/job/PreCommit-HDFS-Build/13238/artifact/patchprocess/diffcheckstylehadoop-hdfs.txt
hadoop-hdfs test log	https://builds.apache.org/job/PreCommit-HDFS-Build/13238/artifact/patchprocess/testrun_hadoop-hdfs.txt
hadoop-hdfs-client test log	https://builds.apache.org/job/PreCommit-HDFS-Build/13238/artifact/patchprocess/testrun_hadoop-hdfs-client.txt
Test Results	https://builds.apache.org/job/PreCommit-HDFS-Build/13238/testReport/
Java	1.7.0_55
uname	Linux asf907.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-HDFS-Build/13238/console

This message was automatically generated.

Chang Li added a comment - 28/Oct/15 15:45

zhz, no we don't have this log because we didn't enable the blockStateChangeLog. How do you propose we should proceed with this jira?

Jing Zhao added a comment - 28/Oct/15 17:55

Making DataStreamer#block volatile is a good change, the GS validation on the NN side also looks good to me. Maybe we do not need a new type of `InvalidGenStampException` though. And to log the detailed information of the block with mismatching GS on the NN side will also be useful.

Chang Li added a comment - 28/Oct/15 21:47

Thanks jingzhao for proposal! I have updated .4 patch which log mismatched GS block info on NN and change DataStreamer#block to volatile

Zhe Zhang added a comment - 28/Oct/15 22:20

Thanks Jing for sharing the thoughts.

I think the GS validation in `BlockManager#commitBlock` is a little tricky. What if the `updatePipeline` RPC call has successfully finished NN side changes but failed in sending response to client? Should we allow the client to commit the block?

GS is used to determine whether a replica is stale, but the client doesn't have a replica. Among the 3 attributes of a block (ID, size, GS), client should always have the same ID as NN, and should always have fresher size than NN. So maybe the right thing to do is to discard the client reported GS in `commitBlock`, but I'm not so sure.

How about we use this JIRA to commit the `volatile` change (which should fix the reported issue) and dedicate a follow-on JIRA to the `commitBlock` GS validation change?

Jing Zhao added a comment - 28/Oct/15 22:40

> What if the updatePipeline RPC call has successfully finished NN side changes but failed in sending response to client? Should we allow the client to commit the block?

The RPC call will fail on the client side and the client will not use the old GS to commit the block.

> How about we use this JIRA to commit the volatile change (which should fix the reported issue) and dedicate a follow-on JIRA to the commitBlock GS validation change?

I agree with this proposal. Let's only log a warning msg on the NN side but not throw exception in this jira.

Daryn Sharp added a comment - 29/Oct/15 14:41

I worked with Chang on this issue and can't think of a scenario in which it's legitimate for the client to misreport the genstamp - whether the pipeline was updated or not.

Consider a more extreme case: The client wrote more data after the pipeline recovered and misreports the older genstamp. That's silent data corruption!

I'd like to see an exception here rather than later.

Chang Li added a comment - 29/Oct/15 20:37

Thanks jingzhao, zhz and daryn for reivew and valuable discussions!
Some additional info about several cases of mismatched GS we encountered is that they all happened after pipelineupdate for datanode close recovery, so no mismatched size of commit happen only mismatched GS.
Could we reach a consensus of whether we should log warn of mismatched GS block info or throw exception?

Zhe Zhang added a comment - 29/Oct/15 20:57

> That's silent data corruption!

daryn I agree it's a silent data corruption in the current logic because we update the NN's copy of the GS with the reported GS from the client:

```
// BlockInfo#commitBlock
this.set(getBlockId(), block.getNumBytes(), block.getGenerationStamp());
```

Throwing an exception (and therefore denying the commitBlock) turns this into an explicit failure, which is better. But it's still a data loss because the data written by the client after `updatePipeline` becomes invisible.

So I think at least for this particular bug (lacking `volatile`), the right thing to do is to avoid changing NN's copy of GS when committing block (so we should avoid changing blockID as well). The only thing we should commit is `numBytes`. Of course we should still print a `WARN` or `ERROR` when GSes mismatch. As a safer first step we should at least avoid decrementing NN's copy of block GS.

In general, if a client misreports GS, does it indicate a likelihood of misreported `numBytes` – and therefore we should deny the `commitBlock`? It's hard to say; the `volatile` bug here is only for GS. But since we have already ensured the NN's copy of block `numBytes` never decrements, the harm of a misreported `numBytes` is not severe.

Jing Zhao added a comment - 29/Oct/15 21:13

> In general, if a client misreports GS, does it indicate a likelihood of misreported numBytes – and therefore we should deny the commitBlock?

Currently NN only depends on the reported length from the client to determine the block length (not considering lease recovery scenario). So the only check we can do about the length is the existing one: `assert block.getNumBytes() <= commitBlock.getNumBytes()`.

> But it's still a data loss because the data written by the client after updatePipeline becomes invisible.

Throwing an exception here may not necessarily mean that the data written after updatePipeline will get lost. In most cases the data can still get recovered during the lease recovery, considering the replicas have already get persisted in DataNodes before client sends out the commit/complete request to NN (since the client has received the last response from the pipeline at that time).

So throwing exception here should be the correct behavior and may not be that risky.

Zhe Zhang added a comment - 29/Oct/15 21:26

Thanks Jing for the explanation. I agree it's reasonable to throw an exception in `commitBlock` and rely on lease recovery to bring the block back to full strength in this case.

Zhe Zhang added a comment - 29/Oct/15 21:29

A small ask for the next rev:

```
// BlockInfo#commitBlock
- this.set(getBlockId(), block.getNumBytes(), block.getGenerationStamp());
+ this.setNumBytes(block.getNumBytes());
```

We also need to add a test in the 04 patch. Otherwise LGTM.

Chang Li added a comment - 29/Oct/15 22:24

Thanks jingzhao and zhz for further review!
Uploaded .5 patch which not only log detailed error message but also throw exception. Also commit does not modify NN's copy of the block GS as Zhang suggested. Test is included.

Chang Li added a comment - 29/Oct/15 22:40

decide to not add new exception, update .6 patch

Hadoop QA added a comment - 30/Oct/15 03:05

☒ **-1 overall**

Vote	Subsystem	Runtime	Comment
0	reexec	0m 14s	docker + precommit patch detected.
+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	3m 8s	trunk passed
+1	compile	1m 4s	trunk passed with JDK v1.8.0_66
+1	compile	1m 4s	trunk passed with JDK v1.7.0_79

+1	checkstyle	0m 22s	trunk passed
+1	mvneclipse	0m 29s	trunk passed
-1	findbugs	2m 3s	hadoop-hdfs-project/hadoop-hdfs in trunk cannot run convertXmlToText from findbugs
+1	javadoc	1m 39s	trunk passed with JDK v1.8.0_66
+1	javadoc	2m 18s	trunk passed with JDK v1.7.0_79
+1	mvninstall	1m 12s	the patch passed
+1	compile	1m 2s	the patch passed with JDK v1.8.0_66
+1	javac	1m 2s	the patch passed
+1	compile	1m 1s	the patch passed with JDK v1.7.0_79
+1	javac	1m 1s	the patch passed
-1	checkstyle	0m 21s	Patch generated 1 new checkstyle issues in hadoop-hdfs-project (total was 247, now 247).
+1	mvneclipse	0m 29s	the patch passed
+1	whitespace	0m 0s	Patch has no whitespace issues.
+1	findbugs	4m 14s	the patch passed
+1	javadoc	1m 30s	the patch passed with JDK v1.8.0_66
+1	javadoc	2m 19s	the patch passed with JDK v1.7.0_79
-1	unit	69m 1s	hadoop-hdfs in the patch failed with JDK v1.8.0_66.
+1	unit	0m 55s	hadoop-hdfs-client in the patch passed with JDK v1.8.0_66.
+1	unit	67m 54s	hadoop-hdfs in the patch passed with JDK v1.7.0_79.
+1	unit	0m 57s	hadoop-hdfs-client in the patch passed with JDK v1.7.0_79.
-1	asflicense	0m 20s	Patch generated 56 ASF License warnings.
		168m 34s	

Reason	Tests
JDK v1.8.0_66 Failed junit tests	hadoop.hdfs.server.datanode.TestBlockScanner
	hadoop.hdfs.server.namenode.TestFSImage
	hadoop.hdfs.server.namenode.ha.TestDFSUpgradeWithHA

Subsystem	Report/Notes
Docker	Client=1.7.0 Server=1.7.0 Image:test-patch-base-hadoop-date2015-10-30
JIRA Patch URL	https://issues.apache.org/jira/secure/attachment/12769654/HDFS-9289.6.patch
JIRA Issue	~~HDFS-9289~~
Optional Tests	asflicense javac javadoc mvninstall unit findbugs checkstyle compile
uname	Linux 480999ad7ea6 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	/home/jenkins/jenkins-slave/workspace/PreCommit-HDFS-Build/patchprocess/apache-yetus-e77b1ce/precommit/personality/hadoop.sh
git revision	trunk / e5b1733
Default Java	1.7.0_79
Multi-JDK versions	/usr/lib/jvm/java-8-oracle:1.8.0_66 /usr/lib/jvm/java-7-openjdk-amd64:1.7.0_79
findbugs	v3.0.0
findbugs	https://builds.apache.org/job/PreCommit-HDFS-Build/13286/artifact/patchprocess/branch-findbugs-hadoop-hdfs-project_hadoop-hdfs.html

checkstyle	https://builds.apache.org/job/PreCommit-HDFS-Build/13286/artifact/patchprocess/diff-checkstyle-hadoop-hdfs-project.txt
unit	https://builds.apache.org/job/PreCommit-HDFS-Build/13286/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.8.0_66.txt
unit test logs	https://builds.apache.org/job/PreCommit-HDFS-Build/13286/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.8.0_66.txt
JDK v1.7.0_79 Test Results	https://builds.apache.org/job/PreCommit-HDFS-Build/13286/testReport/
asflicense	https://builds.apache.org/job/PreCommit-HDFS-Build/13286/artifact/patchprocess/patch-asflicense-problems.txt
modules	C: hadoop-hdfs-project/hadoop-hdfs hadoop-hdfs-project/hadoop-hdfs-client U: hadoop-hdfs-project
Max memory used	224MB
Powered by	Apache Yetus http://yetus.apache.org
Console output	https://builds.apache.org/job/PreCommit-HDFS-Build/13286/console

This message was automatically generated.

Zhe Zhang added a comment - 30/Oct/15 06:12

Thanks Chang for the update. The main code change looks good. I'll take a closer look at the test tomorrow. I just updated the JIRA title. Please see if it looks OK.

Chang Li added a comment - 30/Oct/15 14:21

Thanks zhz, the jira title looks good to me. Looking forward to your further review.

Daryn Sharp added a comment - 02/Nov/15 15:48 - edited

I'd rather see the InvalidGenStampException instead of a generic IOE. Else it's hard for client code to intelligently deal with exceptions and for tests to verify that the correct/expected IOE was thrown.

Zhe Zhang added a comment - 02/Nov/15 23:49

The test looks reasonable. I can see 2 minor issues:

1. `DFSTestUtil#addBlockToFile` is very similar as `addStripedBlockToFile`. At the minimum we should change the comment (not adding a "block group"). Ideally we should consolidate the two methods with a switch for striped or not.
2. The `TestCommitBlock` class only has 1 test and should be named more specifically, something like `TestCommitBlockWithInvalidGenStamp`

If we create a new `InvalidGenStampException` class, I guess we should apply it to all GS-related NN exceptions? E.g. `BlockManager#checkReplicaCorrupt`. How about we do it under a separate JIRA?

Chang Li added a comment - 03/Nov/15 16:31

Thanks daryn and zhz for review and comment!
I have talked to Daryn and he agrees to relegate the new exception work to a new jira.
the updated .7 address zhz's suggestion of consolidate two methods of `addBlockToFile` and `addStripedBlockToFile` and rename `TestCommitBlock` class to `TestCommitBlockWithInvalidGenStamp`

Zhe Zhang added a comment - 03/Nov/15 17:25

Thanks Chang. +1 on the latest patch pending Jenkins.

I'll file a follow-on JIRA to cleanup `addBlockToFile` a bit. Maybe we can consolidate `numStripes` and `len`.

Hadoop QA added a comment - 03/Nov/15 19:37

☒ **-1 overall**

Vote	Subsystem	Runtime	Comment
0	reexec	0m 8s	docker + precommit patch detected.
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 3 new or modified test files.
+1	mvninstall	3m 22s	trunk passed
+1	compile	1m 7s	trunk passed with JDK v1.8.0_66
+1	compile	1m 1s	trunk passed with JDK v1.7.0_79
+1	checkstyle	0m 22s	trunk passed
+1	mvneclipse	0m 28s	trunk passed
-1	findbugs	2m 5s	hadoop-hdfs-project/hadoop-hdfs in trunk cannot run convertXmlToText from findbugs
+1	javadoc	1m 39s	trunk passed with JDK v1.8.0_66
+1	javadoc	2m 19s	trunk passed with JDK v1.7.0_79
+1	mvninstall	1m 14s	the patch passed
+1	compile	1m 4s	the patch passed with JDK v1.8.0_66
+1	javac	1m 4s	the patch passed
+1	compile	1m 13s	the patch passed with JDK v1.7.0_79
+1	javac	1m 13s	the patch passed
-1	checkstyle	0m 24s	Patch generated 1 new checkstyle issues in hadoop-hdfs-project (total was 247, now 247).
+1	mvneclipse	0m 33s	the patch passed
+1	whitespace	0m 0s	Patch has no whitespace issues.
+1	findbugs	4m 14s	the patch passed
+1	javadoc	1m 33s	the patch passed with JDK v1.8.0_66
+1	javadoc	2m 22s	the patch passed with JDK v1.7.0_79
-1	unit	73m 29s	hadoop-hdfs in the patch failed with JDK v1.8.0_66.
+1	unit	1m 2s	hadoop-hdfs-client in the patch passed with JDK v1.8.0_66.
-1	unit	72m 38s	hadoop-hdfs in the patch failed with JDK v1.7.0_79.
+1	unit	1m 1s	hadoop-hdfs-client in the patch passed with JDK v1.7.0_79.
-1	asflicense	0m 22s	Patch generated 56 ASF License warnings.
		178m 46s	

Reason	Tests
JDK v1.8.0_66 Failed junit tests	hadoop.hdfs.server.namenode.ha.TestFailureToReadEdits
	hadoop.hdfs.server.blockmanagement.TestNodeCount
JDK v1.7.0_79 Failed junit tests	hadoop.hdfs.server.blockmanagement.TestPendingInvalidateBlock
	hadoop.hdfs.server.namenode.ha.TestDNFencing
	hadoop.hdfs.TestRollingUpgrade
	hadoop.hdfs.TestFileCreationDelete
	hadoop.hdfs.server.blockmanagement.TestUnderReplicatedBlocks

Subsystem	Report/Notes
Docker	Client=1.7.0 Server=1.7.0 Image:test-patch-base-hadoop-date2015-11-03
JIRA Patch URL	https://issues.apache.org/jira/secure/attachment/12770342/HDFS-9289.7.patch
JIRA Issue	~~HDFS-9289~~
Optional Tests	asflicense javac javadoc mvninstall unit findbugs checkstyle compile

uname	Linux e3e78233b1ef 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	/home/jenkins/jenkins-slave/workspace/PreCommit-HDFS-Build/patchprocess/apache-yetus-1a9afee/precommit/personality/hadoop.sh
git revision	trunk / 957f031
Default Java	1.7.0_79
Multi-JDK versions	/usr/lib/jvm/java-8-oracle:1.8.0_66 /usr/lib/jvm/java-7-openjdk-amd64:1.7.0_79
findbugs	v3.0.0
findbugs	https://builds.apache.org/job/PreCommit-HDFS-Build/13360/artifact/patchprocess/branch-findbugs-hadoop-hdfs-project_hadoop-hdfs.html
checkstyle	https://builds.apache.org/job/PreCommit-HDFS-Build/13360/artifact/patchprocess/diff-checkstyle-hadoop-hdfs-project.txt
unit	https://builds.apache.org/job/PreCommit-HDFS-Build/13360/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.8.0_66.txt
unit	https://builds.apache.org/job/PreCommit-HDFS-Build/13360/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.7.0_79.txt
unit test logs	https://builds.apache.org/job/PreCommit-HDFS-Build/13360/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.8.0_66.txt https://builds.apache.org/job/PreCommit-HDFS-Build/13360/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.7.0_79.txt
JDK v1.7.0_79 Test Results	https://builds.apache.org/job/PreCommit-HDFS-Build/13360/testReport/
asflicense	https://builds.apache.org/job/PreCommit-HDFS-Build/13360/artifact/patchprocess/patch-asflicense-problems.txt
modules	C: hadoop-hdfs-project/hadoop-hdfs hadoop-hdfs-project/hadoop-hdfs-client U: hadoop-hdfs-project
Max memory used	226MB
Powered by	Apache Yetus http://yetus.apache.org
Console output	https://builds.apache.org/job/PreCommit-HDFS-Build/13360/console

This message was automatically generated.

Hadoop QA added a comment - 03/Nov/15 19:56

☒ **-1 overall**

Vote	Subsystem	Runtime	Comment
0	reexec	0m 5s	docker + precommit patch detected.
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 3 new or modified test files.
+1	mvninstall	3m 13s	trunk passed
+1	compile	0m 55s	trunk passed with JDK v1.8.0_60
+1	compile	0m 53s	trunk passed with JDK v1.7.0_79
+1	checkstyle	0m 19s	trunk passed
+1	mvneclipse	0m 27s	trunk passed
-1	findbugs	1m 51s	hadoop-hdfs-project/hadoop-hdfs in trunk cannot run convertXmlToText from findbugs
+1	javadoc	1m 25s	trunk passed with JDK v1.8.0_60
+1	javadoc	2m 12s	trunk passed with JDK v1.7.0_79
+1	mvninstall	1m 6s	the patch passed

+1	compile	0m 54s	the patch passed with JDK v1.8.0_60
+1	javac	0m 54s	the patch passed
+1	compile	0m 54s	the patch passed with JDK v1.7.0_79
+1	javac	0m 54s	the patch passed
-1	checkstyle	0m 19s	Patch generated 1 new checkstyle issues in hadoop-hdfs-project (total was 247, now 247).
+1	mvneclipse	0m 26s	the patch passed
+1	whitespace	0m 0s	Patch has no whitespace issues.
+1	findbugs	3m 53s	the patch passed
+1	javadoc	1m 28s	the patch passed with JDK v1.8.0_60
+1	javadoc	2m 11s	the patch passed with JDK v1.7.0_79
-1	unit	61m 48s	hadoop-hdfs in the patch failed with JDK v1.8.0_60.
+1	unit	0m 48s	hadoop-hdfs-client in the patch passed with JDK v1.8.0_60.
-1	unit	64m 57s	hadoop-hdfs in the patch failed with JDK v1.7.0_79.
+1	unit	0m 55s	hadoop-hdfs-client in the patch passed with JDK v1.7.0_79.
-1	asflicense	0m 19s	Patch generated 58 ASF License warnings.
		155m 57s	

Reason	Tests
JDK v1.8.0_60 Failed junit tests	hadoop.hdfs.server.blockmanagement.TestNodeCount
	hadoop.hdfs.server.balancer.TestBalancerWithMultipleNameNodes
JDK v1.7.0_79 Failed junit tests	hadoop.fs.shell.TestHdfsTextCommand
	hadoop.hdfs.TestRollingUpgrade

Subsystem	Report/Notes
Docker	Client=1.7.1 Server=1.7.1 Image:test-patch-base-hadoop-date2015-11-03
JIRA Patch URL	https://issues.apache.org/jira/secure/attachment/12770342/HDFS-9289.7.patch
JIRA Issue	~~HDFS-9289~~
Optional Tests	asflicense javac javadoc mvninstall unit findbugs checkstyle compile
uname	Linux 2117bffd150f 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	/home/jenkins/jenkins-slave/workspace/PreCommit-HDFS-Build/patchprocess/apache-yetus-1a9afee/precommit/personality/hadoop.sh
git revision	trunk / 957f031
Default Java	1.7.0_79
Multi-JDK versions	/usr/lib/jvm/java-8-oracle:1.8.0_60 /usr/lib/jvm/java-7-openjdk-amd64:1.7.0_79
findbugs	v3.0.0
findbugs	https://builds.apache.org/job/PreCommit-HDFS-Build/13362/artifact/patchprocess/branch-findbugs-hadoop-hdfs-project_hadoop-hdfs.html
checkstyle	https://builds.apache.org/job/PreCommit-HDFS-Build/13362/artifact/patchprocess/diff-checkstyle-hadoop-hdfs-project.txt
unit	https://builds.apache.org/job/PreCommit-HDFS-Build/13362/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.8.0_60.txt

unit	https://builds.apache.org/job/PreCommit-HDFS-Build/13362/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.7.0_79.txt
unit test logs	https://builds.apache.org/job/PreCommit-HDFS-Build/13362/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.8.0_60.txt https://builds.apache.org/job/PreCommit-HDFS-Build/13362/artifact/patchprocess/patch-unit-hadoop-hdfs-project_hadoop-hdfs-jdk1.7.0_79.txt
JDK v1.7.0_79 Test Results	https://builds.apache.org/job/PreCommit-HDFS-Build/13362/testReport/
asflicense	https://builds.apache.org/job/PreCommit-HDFS-Build/13362/artifact/patchprocess/patch-asflicense-problems.txt
modules	C: hadoop-hdfs-project/hadoop-hdfs hadoop-hdfs-project/hadoop-hdfs-client U: hadoop-hdfs-project
Max memory used	225MB
Powered by	Apache Yetus http://yetus.apache.org
Console output	https://builds.apache.org/job/PreCommit-HDFS-Build/13362/console

This message was automatically generated.

Chang Li added a comment - 03/Nov/15 20:48

Thanks zhz for further review! I have test those above failed unit tests above on my own machine with my patch on, and those test passed. They are not related to the changes of my patch.

Zhe Zhang added a comment - 03/Nov/15 21:40

Thanks Change. I just committed the patch to trunk. Do you mind creating a patch for branch-2? The main conflict is on the test.

Zhe Zhang added a comment - 03/Nov/15 21:40

Fixing the JIRA for 3.0 for now. Will reopen and update fix version once branch-2 patch is done.

Chang Li added a comment - 03/Nov/15 22:18

Thanks zhz for commit!
have created and uploaded a branch-2 patch

Zhe Zhang added a comment - 03/Nov/15 22:24

Thanks Chang. I just committed to branch-2 as well. Changed fix version to 2.8.0.

Hudson added a comment - 03/Nov/15 22:30

FAILURE: Integrated in Hadoop-trunk-Commit #8750 (See https://builds.apache.org/job/Hadoop-trunk-Commit/8750/)
~~HDFS-9289~~. Make DataStreamer#block thread safe and verify genStamp in (zhz: rev dac0463a4e20dfb3a802355919fc22b8e017a4e1)

- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockManager.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/DFSTestUtil.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestCommitBlockWithInvalidGenStamp.java
- hadoop-hdfs-project/hadoop-hdfs/CHANGES.txt
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockInfo.java
- hadoop-hdfs-project/hadoop-hdfs-client/src/main/java/org/apache/hadoop/hdfs/DataStreamer.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestQuotaWithStripedBlocks.java

Hudson added a comment - 04/Nov/15 00:32

FAILURE: Integrated in Hadoop-Mapreduce-trunk #2564 (See https://builds.apache.org/job/Hadoop-Mapreduce-trunk/2564/)
~~HDFS-9289~~. Make DataStreamer#block thread safe and verify genStamp in (zhz: rev dac0463a4e20dfb3a802355919fc22b8e017a4e1)

- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestQuotaWithStripedBlocks.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/DFSTestUtil.java
- hadoop-hdfs-project/hadoop-hdfs/CHANGES.txt

- hadoop-hdfs-project/hadoop-hdfs-client/src/main/java/org/apache/hadoop/hdfs/DataStreamer.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestCommitBlockWithInvalidGenStamp.java
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockInfo.java
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockManager.java

Hudson added a comment - 04/Nov/15 00:48

SUCCESS: Integrated in Hadoop-Yarn-trunk #1357 (See https://builds.apache.org/job/Hadoop-Yarn-trunk/1357/)
~~HDFS-9289~~. Make DataStreamer#block thread safe and verify genStamp in (zhz: rev dac0463a4e20dfb3a802355919fc22b8e017a4e1)

- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockInfo.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestCommitBlockWithInvalidGenStamp.java
- hadoop-hdfs-project/hadoop-hdfs/CHANGES.txt
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockManager.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/DFSTestUtil.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestQuotaWithStripedBlocks.java
- hadoop-hdfs-project/hadoop-hdfs-client/src/main/java/org/apache/hadoop/hdfs/DataStreamer.java

Hudson added a comment - 04/Nov/15 01:01

SUCCESS: Integrated in Hadoop-Yarn-trunk-Java8 #634 (See https://builds.apache.org/job/Hadoop-Yarn-trunk-Java8/634/)
~~HDFS-9289~~. Make DataStreamer#block thread safe and verify genStamp in (zhz: rev dac0463a4e20dfb3a802355919fc22b8e017a4e1)

- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestQuotaWithStripedBlocks.java
- hadoop-hdfs-project/hadoop-hdfs-client/src/main/java/org/apache/hadoop/hdfs/DataStreamer.java
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockManager.java
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockInfo.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/DFSTestUtil.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestCommitBlockWithInvalidGenStamp.java
- hadoop-hdfs-project/hadoop-hdfs/CHANGES.txt

Hudson added a comment - 04/Nov/15 03:11

FAILURE: Integrated in Hadoop-Mapreduce-trunk-Java8 #623 (See https://builds.apache.org/job/Hadoop-Mapreduce-trunk-Java8/623/)
~~HDFS-9289~~. Make DataStreamer#block thread safe and verify genStamp in (zhz: rev dac0463a4e20dfb3a802355919fc22b8e017a4e1)

- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/DFSTestUtil.java
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockInfo.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestCommitBlockWithInvalidGenStamp.java
- hadoop-hdfs-project/hadoop-hdfs/CHANGES.txt
- hadoop-hdfs-project/hadoop-hdfs-client/src/main/java/org/apache/hadoop/hdfs/DataStreamer.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestQuotaWithStripedBlocks.java
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockManager.java

Hudson added a comment - 04/Nov/15 05:00

FAILURE: Integrated in Hadoop-Hdfs-trunk-Java8 #568 (See https://builds.apache.org/job/Hadoop-Hdfs-trunk-Java8/568/)
~~HDFS-9289~~. Make DataStreamer#block thread safe and verify genStamp in (zhz: rev dac0463a4e20dfb3a802355919fc22b8e017a4e1)

- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockInfo.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestCommitBlockWithInvalidGenStamp.java
- hadoop-hdfs-project/hadoop-hdfs/CHANGES.txt
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/DFSTestUtil.java
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockManager.java
- hadoop-hdfs-project/hadoop-hdfs-client/src/main/java/org/apache/hadoop/hdfs/DataStreamer.java

- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestQuotaWithStripedBlocks.java

---

Hudson added a comment - 04/Nov/15 05:17

FAILURE: Integrated in Hadoop-Hdfs-trunk #2505 (See https://builds.apache.org/job/Hadoop-Hdfs-trunk/2505/)
~~HDFS-9289~~. Make DataStreamer#block thread safe and verify genStamp in (zhz: rev dac0463a4e20dfb3a802355919fc22b8e017a4e1)

- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestQuotaWithStripedBlocks.java
- hadoop-hdfs-project/hadoop-hdfs/CHANGES.txt
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/server/namenode/TestCommitBlockWithInvalidGenStamp.java
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockManager.java
- hadoop-hdfs-project/hadoop-hdfs-client/src/main/java/org/apache/hadoop/hdfs/DataStreamer.java
- hadoop-hdfs-project/hadoop-hdfs/src/test/java/org/apache/hadoop/hdfs/DFSTestUtil.java
- hadoop-hdfs-project/hadoop-hdfs/src/main/java/org/apache/hadoop/hdfs/server/blockmanagement/BlockInfo.java

---

Rushabh Shah added a comment - 04/Nov/15 16:25

zhz: Given that we are going to release 2.7.2 shortly , do we need to cherry pick it to 2.7.2 ?

Kihwal Lee added a comment - 04/Nov/15 16:58

2.7.2 is already cut. I will target it to 2.7.3 and cherry-pick to branch-2.7.

Kihwal Lee added a comment - 04/Nov/15 17:05

> I will target it to 2.7.3 and cherry-pick to branch-2.7.

We can't cherry-pick since the files/components have been split and moved around.
lichangleo, can you post 2.7 version of the patch?

Chang Li added a comment - 04/Nov/15 17:30

Hi kihwal, have uploaded branch 2.7 patch

Zhe Zhang added a comment - 04/Nov/15 17:35

Thanks for the discussions above, 2.7.3 sounds a good target version for this change.

Kihwal Lee added a comment - 04/Nov/15 18:12

Looks like a correct port of the patch. The new test succeeds.
Committed to branch-2.7.

Sangjin Lee added a comment - 20/Nov/15 23:46

Does this issue exist in 2.6.x? Should this be backported to branch-2.6?

Zhe Zhang added a comment - 21/Nov/15 00:57

I think this is a good addition to branch-2.6.

Zhe Zhang added a comment - 21/Nov/15 00:58

sjlee0 Yes this is a valid bug even for 2.6.x.

Sangjin Lee added a comment - 21/Nov/15 01:02

Thanks zhz! Could you please cherry-pick the commit to branch-2.6?

Zhe Zhang added a comment - 21/Nov/15 07:55

Thanks sjlee0. I'm attaching a branch-2.6 patch. Am I updating CHANGES.TXT correctly? If looks good to you I'll push to branch-2.6.

Sangjin Lee added a comment - 23/Nov/15 22:24

Hi zhz, the only differences I see between the 2.7 commit and the 2.6 patch are a couple of type changes (e.g. BlockInfoContiguous), right? If so, I think it might be good to follow a cherry-pick route, rather than creating a separate patch and commit.

If you're not familiar with it already, you can do

```
git cherry-pick -x <commit id of the commit on branch-2.7>
```

Naturally CHANGES.txt will come up as a conflict, and you can just make it like what's proposed in the proposed patch. You'll also want to touch up the other files just like what you did in the patch. You then stage those files that were further modified. Finally you can finalize the cherry-pick via

```
git cherry-pick --continue
```

If you're not comfortable with that, I'd be OK with a new separate commit too this time. Let me know if you have a question either way.

Zhe Zhang added a comment - 24/Nov/15 17:46

Thanks Sangjin. Yes the conflicts were mostly trivial. I was just not sure if I did the right update to CHANGES.TXT. I just committed the change to branch-2.6.

Vinod Kumar Vavilapalli added a comment - 13/Jan/16 22:41

Pulled this into 2.7.2 to keep the release up-to-date with 2.6.3. Changing fix-versions to reflect the same.

People

Assignee:

Chang Li

Reporter:

Chang Li

Votes:

0 Vote for this issue

Watchers:

18 Start watching this issue

Dates

Created:

22/Oct/15 19:13

Updated:

06/Jan/17 00:53

Resolved:

03/Nov/15 21:40