Item 240

git_comments:

1. **comment:** Call super.toString() so that we have a unique hash/address added to the toString, but also include the file being written to. When comparing the toString() of two different leases, this helps to compare whether or not the two leases are the same object as well as whether or not they point to the same file.
   label: code-design

git_commits:

1. **summary:** NIFI-7856: If a Provenance Event File is ready to be rolled over due to the maximum amount of time having elapsed, avoid rolling over unless there is at least one event written to the event file. Otherwise, we can have multiple RecordWriters / RecordWriterLeases pointing to the same file. This can result in data being overwritten, as well as failing to compress the event file upon rollover. Also added significant DEBUG/TRACE level logging.
   message: NIFI-7856: If a Provenance Event File is ready to be rolled over due to the maximum amount of time having elapsed, avoid rolling over unless there is at least one event written to the event file. Otherwise, we can have multiple RecordWriters / RecordWriterLeases pointing to the same file. This can result in data being overwritten, as well as failing to compress the event file upon rollover. Also added significant DEBUG/TRACE level logging. Signed-off-by: Matthew Burgess <mattyb149@apache.org> This closes #4580
   label: code-design

github_issues:

github_issues_comments:

github_pulls:

1. **title:** NIFI-7856: If a Provenance Event File is ready to be rolled over due …
   body: …to the maximum amount of time having elapsed, avoid rolling over unless there is at least one event written to the event file. Otherwise, we can have multiple RecordWriters / RecordWriterLeases pointing to the same file. This can result in data being overwritten, as well as failing to compress the event file upon rollover. Also added significant DEBUG/TRACE level logging. Thank you for submitting a contribution to Apache NiFi. Please provide a short description of the PR here: #### Description of PR _Enables X functionality; fixes bug NIFI-YYYY._ In order to streamline the review of the contribution we ask you to ensure the following steps have been taken: ### For all changes: - [ ] Is there a JIRA ticket associated with this PR? Is it referenced in the commit message? - [ ] Does your PR title start with **NIFI-XXXX** where XXXX is the JIRA number you are trying to resolve? Pay particular attention to the hyphen "-" character. - [ ] Has your PR been rebased against the latest commit within the target branch (typically `main`)? - [ ] Is your initial contribution a single, squashed commit? _Additional commits in response to PR reviewer feedback should be made on this branch and pushed to allow change tracking. Do not `squash` or use `--force` when pushing to allow for clean monitoring of changes._ ### For code changes: - [ ] Have you ensured that the full suite of tests is executed via `mvn -Pcontrib-check clean install` at the root `nifi` folder? - [ ] Have you written or updated unit tests to verify your changes? - [ ] Have you verified that the full build is successful on JDK 8? - [ ] Have you verified that the full build is successful on JDK 11? - [ ] If adding new dependencies to the code, are these dependencies licensed in a way that is compatible for inclusion under [ASF 2.0] (http://www.apache.org/legal/resolved.html#category-a)? - [ ] If applicable, have you updated the `LICENSE` file, including the main `LICENSE` file under `nifi-assembly`? - [ ] If applicable, have you updated the `NOTICE` file, including the main `NOTICE` file found under `nifi-assembly`? - [ ] If adding new Properties, have you added `.displayName` in addition to .name (programmatic access) for each of the new properties? ### For documentation related changes: - [ ] Have you ensured that format looks appropriate for the output in which it is rendered? ### Note: Please ensure that once the PR is submitted, you check GitHub Actions CI for build issues and submit an update to your PR as soon as possible.

github_pulls_comments:

1. Is this the expected output of the template you attached to the Jira? ``` 2020-10-07 18:33:02,487 WARN [Provenance Repository Maintenance-2-thread-1] o.a.n.p.store.WriteAheadStorePartition Failed to remove Provenance Event file ./provenance_repository/22947.prov; this file should be cleaned up manually 2020-10-07 18:33:02,487 WARN [Provenance Repository Maintenance-2-thread-1] o.a.n.p.store.WriteAheadStorePartition Provenance Event Store Partition[directory=./provenance_repository] Failed to delete oldest event file ./provenance_repository/22947.prov. This file should be cleaned up manually. 2020-10-07 18:33:02,489 INFO [Provenance Repository Maintenance-2-thread-1] o.a.n.p.store.WriteAheadStorePartition Provenance Event Store Partition[directory=./provenance_repository] Deleted ./provenance_repository/22947.prov.gz event file (54.27 MB) due to storage limits 2020-10-07 18:33:05,894 INFO [pool-17-thread-1] o.a.n.wali.SequentialAccessWriteAheadLog Checkpointed Write-Ahead Log with 6806 Records and 0 Swap Files in 22306 milliseconds (Stop-the-world time = 1134 milliseconds), max Transaction ID 23 2020-10-07 18:33:05,894 INFO [pool-17-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully checkpointed FlowFile Repository with 6806 records in 22306 milliseconds 2020-10-07 18:33:08,060 INFO [Timer-Driven Process Thread-8] o.a.n.p.store.WriteAheadStorePartition Successfully rolled over Event Writer for Provenance Event Store Partition[directory=./provenance_repository] after writing 5102 events due to MAX_BYTES_REACHED ```
2. +1 LGTM, reproduced the issue then verified it works with this PR. Thanks for the fix! Merging to main

github_pulls_reviews:

jira_issues:

1. **summary:** Provenance failed to be compressed after nifi upgrade to 1.12
   description: We upgraded our nifi cluster from 1.11.3 to 1.12.0. The nodes come up and everything looks to be functional. I can see 1.12.0 is running. Later on, we discovered that the data provenance is missing. From checking our logs, we see tons of errors compressing the logs. {code} 2020-09-28 03:38:35,205 ERROR [Compress Provenance Logs-1-thread-1] o.a.n.p.s.EventFileCompressor Failed to compress ./provenance_repository/2752821.prov on rollover {code} This didn't happen in 1.11.3. Is this a known issue? We are considering reverting back if there is no solution for this since we can't go prod with no/broken data provenance.

jira_issues_comments:

1. Do we have any pointers how to address/debug this? appreciate it.
2. [~leeyoda] this is not something I've run into. Do you have a stack trace in nifi-app.log?
3. Here is the stack trace of one incident, hopefully it is helpful. Also attached the ls results, it seems that these files are all compressed fine but the logs seem to show that it doesn't exist. A race condition? {code} 2020-09-27 21:37:34,747 INFO [Clustering Tasks Thread-3] o.a.n.c.c.ClusterProtocolHeartbeater Heartbeat created at 2020-09-27 21:37:34,616 and sent to 10.51.8.18:9999 at 2020-09-27 21:37:34,747; send took 131 millis 2020-09-27 21:37:39,660 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Initiating checkpoint of FlowFile Repository 2020-09-27 21:37:39,660 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully checkpointed FlowFile Repository with 15079 records in 0 milliseconds 2020-09-27 21:37:49,109 INFO [pool-61-thread-1] c.a.s.k.clientlibrary.lib.worker.Worker Current stream shard assignments: shardId-000000000000 2020-09-27 21:37:49,110 INFO [pool-61-thread-1] c.a.s.k.clientlibrary.lib.worker.Worker Sleeping ... 2020-09-27 21:37:59,660 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Initiating checkpoint of FlowFile Repository 2020-09-27 21:37:59,660 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully checkpointed FlowFile Repository with 15079 records in 0 milliseconds 2020-09-27 21:38:02,196 INFO [pool-43-thread-1] c.a.s.k.clientlibrary.lib.worker.Worker Current stream shard assignments: shardId-000000000012 2020-09-27 21:38:02,196 INFO [pool-43-thread-1] c.a.s.k.clientlibrary.lib.worker.Worker Sleeping ... 2020-09-27 21:38:19,660 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Initiating checkpoint of FlowFile Repository 2020-09-27 21:38:19,660 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully checkpointed FlowFile Repository with 15079 records in 0 milliseconds 2020-09-27 21:38:20,688 INFO [Timer-Driven Process Thread-6] o.a.nifi.groups.StandardProcessGroup StandardProcessGroup[identifier=9e102d08-0174-1000-ffff-ffffdb703545,name=ContactLookup] is not the most recent version of the flow that is under Version

Control; current version is 3; most recent version is 7 2020-09-27 21:38:20,691 INFO [Timer-Driven Process Thread-6] o.a.nifi.groups.StandardProcessGroup StandardProcessGroup[identifier=4b226950-0174-1000-0000-000064a82b74,name=EcomdashOrderProcessingMain] is not the most recent version of the flow that is under Version Control; current version is 8; most recent version is 10 2020-09-27 21:38:20,694 INFO [Timer-Driven Process Thread-6] o.a.nifi.groups.StandardProcessGroup StandardProcessGroup[identifier=e366c899-0173-1000-0000-000026d80b41,name=ContactLookup] is not the most recent version of the flow that is under Version Control; current version is 5; most recent version is 7 2020-09-27 21:38:20,697 INFO [Timer-Driven Process Thread-6] o.a.nifi.groups.StandardProcessGroup StandardProcessGroup[identifier=a17c8629-0173-1000-0000-0000055a79e8,name=HandleFailedMessages] is not the most recent version of the flow that is under Version Control; current version is 2; most recent version is 3 2020-09-27 21:38:34,799 INFO [Framework Task Thread Thread-3] o.a.n.p.store.WriteAheadStorePartition Successfully rolled over Event Writer for Provenance Event Store Partition[directory=./provenance_repository] due to MAX_TIME_REACHED 2020-09-27 21:38:34,799 ERROR [Compress Provenance Logs-1-thread-2] o.a.n.p.s.EventFileCompressor Failed to compress ./provenance_repository/1693519.prov on rollover java.io.FileNotFoundException: ./provenance_repository/1693519.prov (No such file or directory) at java.io.FileInputStream.open0(Native Method) at java.io.FileInputStream.open(FileInputStream.java:195) at java.io.FileInputStream.<init>(FileInputStream.java:138) at org.apache.nifi.provenance.serialization.EventFileCompressor.compress(EventFileCompressor.java:164) at org.apache.nifi.provenance.serialization.EventFileCompressor.run(EventFileCompressor.java:115) at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) at java.util.concurrent.FutureTask.run(FutureTask.java:266) at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149) at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) at java.lang.Thread.run(Thread.java:748) 2020-09-27 21:38:34,799 WARN [Compress Provenance Logs-1-thread-2] o.a.n.p.s.EventFileCompressor Failed to delete ./provenance_repository/1693519.prov; this file should be cleaned up manually 2020-09-27 21:38:34,887 INFO [Clustering Tasks Thread-3] o.a.n.c.c.ClusterProtocolHeartbeater Heartbeat created at 2020-09-27 21:38:34,748 and sent to 10.51.8.18:9999 at 2020-09-27 21:38:34,887; send took 139 millis 2020-09-27 21:38:39,660 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Initiating checkpoint of FlowFile Repository 2020-09-27 21:38:39,660 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully checkpointed FlowFile Repository with 15079 records in 0 milliseconds 2020-09-27 21:38:54,111 INFO [pool-61-thread-1] c.a.s.k.clientlibrary.lib.worker.Worker Current stream shard assignments: shardId-000000000000 2020-09-27 21:38:54,111 INFO [pool-61-thread-1] c.a.s.k.clientlibrary.lib.worker.Worker Sleeping ... 2020-09-27 21:38:59,661 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Initiating checkpoint of FlowFile Repository 2020-09-27 21:38:59,661 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully checkpointed FlowFile Repository with 15079 records in 0 milliseconds 2020-09-27 21:39:03,202 INFO [pool-43-thread-1] c.a.s.k.clientlibrary.lib.worker.Worker Current stream shard assignments: shardId-000000000012 2020-09-27 21:39:03,202 INFO [pool-43-thread-1] c.a.s.k.clientlibrary.lib.worker.Worker Sleeping ... 2020-09-27 21:39:19,661 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Initiating checkpoint of FlowFile Repository 2020-09-27 21:39:19,661 INFO [pool-15-thread-1] o.a.n.c.r.WriteAheadFlowFileRepository Successfully checkpointed FlowFile Repository with 15079 records in 0 milliseconds 2020-09-27 21:39:20,156 INFO [Write-Ahead Local State Provider Maintenance] org.wali.MinimalLockingWriteAheadLog org.wali.MinimalLockingWriteAheadLog@1fe275d8 checkpointed with 4 Records and 0 Swap Files in 4 milliseconds (Stop-the-world time = 0 milliseconds, Clear Edit Logs time = 0 millis), max Transaction ID 1312 {code}

4. **body:** [~leeyoda] thanks for the updated logs & screenshot from 'ls' command. Does this happen frequently, or just once or twice? If only once or twice does it happen during or shortly after startup? Or after NiFi has been running for a while? I can't think of any changes in 1.12.0 that may have affected this, so wondering if perhaps it's related to restarted moreso than changing to 1.12.0. The interesting thing is that, based on the logs and the screenshot, that file already was compressed. So not sure why it was attempting to compress it again... the good news is that it shouldn't cause any problems, given that it's already compressed. But would definitely prefer to resolve the issue, regardless.
   label: code-design
5. From our logs, it happens every hour (seems that rollover MAX_TIME_REACHED is met, not sure the exact schedule is), see screenshot. It happens consistently after the restart, the cluster has been running for

4+ days. The issue for us is that the data provenance is missing for some processors (never show up after the upgrade so latest record was 25th) and data provenance is displaying either incomplete or delayed records. This can be a huge issue for our prod troubleshooting if we move this to our prod env. Attached one prov file as well. !screenshot-2.png!

6. Thanks. Can you provide what properties you have in nifi.properties for the Provenance Repository. E.g.: {code}# Provenance Repository Properties nifi.provenance.repository.implementation=org.apache.nifi.provenance.WriteAheadProvenanceRepository nifi.provenance.repository.encryption.key.provider.implementation= nifi.provenance.repository.encryption.key.provider.location= nifi.provenance.repository.encryption.key.id= nifi.provenance.repository.encryption.key= # Persistent Provenance Repository Properties nifi.provenance.repository.directory.default=./provenance_repository nifi.provenance.repository.max.storage.time=30 days nifi.provenance.repository.max.storage.size=10 GB nifi.provenance.repository.rollover.time=1 mins nifi.provenance.repository.rollover.size=100 MB nifi.provenance.repository.query.threads=2 nifi.provenance.repository.index.threads=2 nifi.provenance.repository.compress.on.rollover=true nifi.provenance.repository.always.sync=false # Comma-separated list of fields. Fields that are not indexed will not be searchable. Valid fields are: # EventType, FlowFileUUID, Filename, TransitURI, ProcessorID, AlternateIdentifierURI, Relationship, Details nifi.provenance.repository.indexed.fields=EventType, FlowFileUUID, Filename, ProcessorID, Relationship # FlowFile Attributes that should be indexed and made searchable. Some examples to consider are filename, uuid, mime.type nifi.provenance.repository.indexed.attributes= # Large values for the shard size will result in more Java heap usage when searching the Provenance Repository # but should provide better performance nifi.provenance.repository.index.shard.size=500 MB # Indicates the maximum length that a FlowFile attribute can be when retrieving a Provenance Event from # the repository. If the length of any attribute exceeds this value, it will be truncated when the event is retrieved. nifi.provenance.repository.max.attribute.length=65536 nifi.provenance.repository.concurrent.merge.threads=2 {code}
7. sure, this is our setting around provenance {code} # Provenance Repository Properties nifi.provenance.repository.implementation=org.apache.nifi.provenance.WriteAheadProvenanceRepository nifi.provenance.repository.encryption.key.provider.implementation= nifi.provenance.repository.encryption.key.provider.location= nifi.provenance.repository.encryption.key.id= nifi.provenance.repository.encryption.key= # Persistent Provenance Repository Properties nifi.provenance.repository.directory.default=./provenance_repository nifi.provenance.repository.max.storage.time=30 days nifi.provenance.repository.max.storage.size=10 GB nifi.provenance.repository.rollover.time=10 mins nifi.provenance.repository.rollover.size=100 MB nifi.provenance.repository.query.threads=2 nifi.provenance.repository.index.threads=2 nifi.provenance.repository.compress.on.rollover=true nifi.provenance.repository.always.sync=false # Comma-separated list of fields. Fields that are not indexed will not be searchable. Valid fields are: # EventType, FlowFileUUID, Filename, TransitURI, ProcessorID, AlternateIdentifierURI, Relationship, Details nifi.provenance.repository.indexed.fields=EventType, FlowFileUUID, Filename, ProcessorID, Relationship # FlowFile Attributes that should be indexed and made searchable. Some examples to consider are filename, uuid, mime.type nifi.provenance.repository.indexed.attributes= # Large values for the shard size will result in more Java heap usage when searching the Provenance Repository # but should provide better performance nifi.provenance.repository.index.shard.size=500 MB # Indicates the maximum length that a FlowFile attribute can be when retrieving a Provenance Event from # the repository. If the length of any attribute exceeds this value, it will be truncated when the event is retrieved. nifi.provenance.repository.max.attribute.length=65536 nifi.provenance.repository.concurrent.merge.threads=2 # Volatile Provenance Respository Properties nifi.provenance.repository.buffer.size=100000 {code}
8. [~markap14] any chance that you have looked at this issue? Thanks
9. I've tried replicating the issue but so far haven't been able to.
10. Thanks for the reply. Do you mind trying ReplaceText 1.12.0 which for us, doesn't show any data provenance since the upgrade. A few records in the 28th pop up randomly.
11. So far I've been unable to reproduce any issues. I've tried with ReplaceText, though this shouldn't matter at all, since the processor implementation is very much divorced from the provenance repository implementation. [~leeyoda] how do you typically view your Provenance events? By right-clicking on the processor and choosing Provenance Events there, or by going to the global menu in the top-right corner and choosing Provenance? I'm curious, if you use the global menu and then search by processor id if you'll see any different results or not.

12. Hi [~markap14], that's a great question since that's one major reason that we upgraded to 1.12.0 since sometimes the data provenance is missing in the processor view but visible in the global view in 1.11.3 that was fixed in 1.12.0. So to answer your question, we do both and in 1.12.0, the missing and delaying data provenance records are consistent from either view. In global view, we usually get component id and search that way since it is unique. We are speculating that some processes is compressing the files before the scheduled time and those ones didn't make it to the lucene index to be searched. Will try to dig around the logs more to provide information that I can could help you debug further.
13. Another observations that we had from flipping the log levels to debug was that: the provenance files are zipped up *10 mins before* the scheduled run, and then it tried to look for a .prov file then it couldn't find it which resulted the error since it is already compressed.
14. [~leeyoda] I finally managed to replicate the issue! I've put up a PR and will attach a small template to this Jira that contains the flow that I used to recreate the issue.
15. Attached template as NIFI-7856.xml.
16. woohoo! Keep me posted please and appreciate your time and effort looking into this. I assume the fix would be in 1.13.0 or another patch version of 1.12. Let me know.
17. Hi Mark, I see your PR hasn't been merged yet however the status of this ticket is "PATCH AVAILABLE". Does that mean there will be a patch version including the fix?
18. [~leeyoda] Patch Available means that there's a fix/PR available but it hasn't been merged yet.
19. Commit a73cd6a610f2e4a43e82bb26e2e4f983b9bfa1a5 in nifi's branch refs/heads/main from Mark Payne [https://gitbox.apache.org/repos/asf?p=nifi.git;h=a73cd6a] NIFI-7856: If a Provenance Event File is ready to be rolled over due to the maximum amount of time having elapsed, avoid rolling over unless there is at least one event written to the event file. Otherwise, we can have multiple RecordWriters / RecordWriterLeases pointing to the same file. This can result in data being overwritten, as well as failing to compress the event file upon rollover. Also added significant DEBUG/TRACE level logging. Signed-off-by: Matthew Burgess <mattyb149@apache.org> This closes #4580