Item 124

git_comments:

git_commits:

1. **summary:** [SPARK-29494][SQL] Fix for ArrayOutofBoundsException while converting string to timestamp
   message: [SPARK-29494][SQL] Fix for ArrayOutofBoundsException while converting string to timestamp ### What changes were proposed in this pull request? * Adding an additional check in `stringToTimestamp` to handle cases where the input has trailing ':' * Added a test to make sure this works. ### Why are the changes needed? In a couple of scenarios while converting from String to Timestamp `DateTimeUtils.stringToTimestamp` throws an array out of bounds exception if there is trailing ':'. The behavior of this method requires it to return `None` in case the format of the string is incorrect. ### Does this PR introduce any user-facing change? No ### How was this patch tested? Added a test in the `DateTimeTestUtils` suite to test if my fix works. Closes #26143 from rahulsmahadev/SPARK-29494. Lead-authored-by: Rahul Mahadev <rahul.mahadev@databricks.com> Co-authored-by: Rahul Shivu Mahadev <51690557+rahulsmahadev@users.noreply.github.com> Signed-off-by: Sean Owen <sean.owen@databricks.com> (cherry picked from commit 4cfce3e5d03b0badb4e9685499be2ab0fca5747a) Signed-off-by: Sean Owen <sean.owen@databricks.com>

github_issues:

github_issues_comments:

github_pulls:

1. **title:** [SPARK-29494][SQL] Fix for ArrayOutofBoundsException while converting string to timestamp
   body: <!-- Thanks for sending a pull request! Here are some tips for you: 1. If this is your first time, please read our contributor guidelines: https://spark.apache.org/contributing.html 2. Ensure you have added or run the appropriate tests for your PR: https://spark.apache.org/developer-tools.html 3. If the PR is unfinished, add '[WIP]' in your PR title, e.g., '[WIP][SPARK-XXXX] Your PR title ...'. 4. Be sure to keep the PR description updated to reflect all changes. 5. Please write your PR title to summarize what this PR proposes. 6. If possible, provide a concise example to reproduce the issue for a faster review. --> ### What changes were proposed in this pull request? * Adding an additional check in `stringToTimestamp` to handle cases where the input has trailing ':' * Added a test to make sure this works. <!-- Please clarify what changes you are proposing. The purpose of this section is to outline the changes and how this PR fixes the issue. If possible, please consider writing useful notes for better and faster reviews in your PR. See the examples below. 1. If you refactor some codes with changing classes, showing the class hierarchy will help reviewers. 2. If you fix some SQL features, you can provide some references of other DBMSes. 3. If there is design documentation, please add the link. 4. If there is a discussion in the mailing list, please add the link. --> ### Why are the changes needed? In a couple of scenarios while converting from String to Timestamp `DateTimeUtils.stringToTimestamp` throws an array out of bounds exception if there is trailing ':'. The behavior of this method requires it to return `None` in case the format of the string is incorrect. <!-- Please clarify why the changes are needed. For instance, 1. If you propose a new API, clarify the use case for a new API. 2. If you fix a bug, you can clarify why it is a bug. --> ### Does this PR introduce any user-facing change? No <!-- If yes, please clarify the previous behavior and the change this PR proposes - provide the console output, description and/or an example to show the behavior difference if possible. If no, write 'No'. --> ### How was this patch tested? Added a test in the `DateTimeTestUtils` suite to test if my fix works. <!-- If tests were added, say they were added here. Please make sure to add some test cases that check the changes thoroughly including negative and positive cases if possible. If it was tested in a way different from regular unit tests, please clarify how you tested step by step, ideally copy and paste-able, so that other reviewers can test and check, and descendants can verify in the future. If tests were not added, please describe why they were not added and/or why it was difficult to add. -->

github_pulls_comments:

1. cc @zsxwing
2. ok to test
3. Actually, trailing spaces shouldn't cause any errors because `stringToTimestamp` trims its input https://github.com/apache/spark/blob/d11cbf2e367317e6a2bedfcd71819ddf485c39c5/sql/catalyst/src/main/scala/org/apache/spark/sql/catalyst/util/DateTimeUtils.sca
4. **[Test build #112189 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/112189/testReport)** for PR 26143 at commit [`7f764d0`](https://github.com/apache/spark/commit/7f764d043ecdabbcb73035630fe06af67ef086a9). * This patch **fails Scala style tests**. * This patch merges cleanly. * This patch adds no public classes.
5. **[Test build #112185 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/112185/testReport)** for PR 26143 at commit [`17edc79`](https://github.com/apache/spark/commit/17edc79d220ba29fe768cc3a5ed5930bcecbded9). * This patch passes all tests. * This patch merges cleanly. * This patch adds no public classes.
6. **[Test build #112186 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/112186/testReport)** for PR 26143 at commit [`421138f`](https://github.com/apache/spark/commit/421138fb54e85d630f94e6223f4ae088a264d1c9). * This patch passes all tests. * This patch merges cleanly. * This patch adds no public classes.
7. **[Test build #112191 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/112191/testReport)** for PR 26143 at commit [`17edc79`](https://github.com/apache/spark/commit/17edc79d220ba29fe768cc3a5ed5930bcecbded9). * This patch passes all tests. * This patch merges cleanly. * This patch adds no public classes.
8. **[Test build #112291 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/112291/testReport)** for PR 26143 at commit [`5ef53b7`](https://github.com/apache/spark/commit/5ef53b7ca639c0bff5db28b9d96ba810974b3207). * This patch passes all tests. * This patch merges cleanly. * This patch adds no public classes.
9. Merged to master/2.4
10. @rahulsmahadev this broke 2.4. I just reverted it from branch-2.4. Could you submit a new PR against branch-2.4? Thanks!
11. Shoot, thanks for catching that @zsxwing . Looks like some test utility code isn't present in 2.4: ``` [error] /home/jenkins/workspace/spark-branch-2.4-test-maven-hadoop-2.7/sql/catalyst/src/test/scala/org/apache/spark/sql/catalyst/util/DateTimeUtilsSuite.scala:586: not found: value defaultZoneId [error] val time = DateTimeUtils.stringToTimestamp(s, defaultZoneId) [error] ^ [error] one error found [error] Compile failed at Oct 18, 2019 2:55:40 PM [15.246s] ```

github_pulls_reviews:

1. You described 2 cases in the PR discription `':'` and `' '`. This test checks the first one only. Could you add at least one more check for `' '`.
2. `stringToTimestamp` returns None becase of invalid time zone `ZZ` at the end. If you replace the string `"2019-10-31T10:59:23ZZ "` by `"2019-10-31T10:59:23Z "`, the functions shouldn't return None
3. You are right this works accurately for trailing spaces, I first enounctered this issue on an older release branch and it did not have the `.trim` . But the fix should work for trailing spaces as well if backported.
4. ```suggestion if (i < segments.length && (b == ':' || b == ' ')) { ```

jira_issues:

jira_issues_comments: