Difference in TumblingWindow & Scheduled trigger

Last updated by | Jackie Huang | Jan 4, 2022 at 12:24 AM PST

Contents

Introduction

This TSG would help us understand the difference in reporting the final status of the Tumbling Window Trigger Failure and the Scheduled Trigger Failure. Before proceeding further with this TSG, make sure to refer the following article for a brief comparison between Tumbling Window Trigger and Scheduled Trigger:

Trigger type comparison: https://docs.microsoft.com/en-us/azure/data-factory/concepts-pipeline-execution-triggers#trigger-type-comparison

The tumbling window trigger run waits for the triggered pipeline run to finish. Its run state reflects the state of the triggered pipeline run. For example, if a triggered pipeline run is cancelled, the corresponding tumbling window trigger run is marked cancelled. This is different from the "fire and forget" behavior of the schedule trigger, which is marked successful as long as a pipeline run started.

Also, Schedule Trigger is a stateless trigger. It does not wait for the pipeline to complete before moving to succeeded state. If the user sets concurrency=1 in the pipeline, the multiple pipeline runs will be queued, but the schedule trigger run will complete as soon as the trigger run trigger the pipeline. Tumbling Window Trigger is a stateful trigger. If concurrency=1 is set of the tumbling Window Trigger, the next window will only be executed when the first one is done.

Scenario

Customer had configured email alerts to report any pipeline failures. The alerts were setup based on the **trigger failure**, if a trigger fails it should send out emails to the admins regarding the failure. As per customer, it was working fine in the past, however, recently they observed there are no alerts generated even though the pipelines were failing.

Resolution

There's a difference how ScheduleTrigger and TumblingWindowTrigger will report a success or failure.

The status of the **TumblingWindow** trigger will be based on the final status of the pipeline execution. If the pipeline succeeds, TW trigger will have status as succeeded. Similarly, if the pipeline final status is failed, TW

trigger will have status as failed.

However, incase of **ScheduledTrigger**, as soon as the trigger is executed successfully, the status of the scheduled trigger will be reported as success, irrespective of the final status of the pipeline whether the pipeline execution succeeds or fails.

When we checked with the customer, he confirmed that he has changed the triggers from TumblingWindow to Scheduled for his pipelines. Hence, he was not getting an alert, even though the status of the pipelines were failing, the scheduled trigger execution status was success, and since, the scheduled trigger status was success, the alerts were not getting fired.

Investigation

You can query if the pipeline was triggered by a particular trigger (TumblingWindow or Scheduled) or another pipeline using the below Kusto function:

```
GetChainedPipelineFromCurrentToPrevious(@'XXXXXXX')  //provide runId as parameter
```

You may also use the following Kusto query to check the status of the Trigger execution:

```
cluster('adfcus').database('AzureDataFactory').TriggerRuns | union cluster('adfneu').database('AzureDataFactor
| where TIMESTAMP >= ago(7d)
| where dataFactoryName =~ "<dataFactoryName>"
| where triggerId == "<triggerId>"
| project PreciseTimeStamp, dataFactoryName, triggerName, triggerId, triggerType, triggerEvent, status, trigge
| sort by PreciseTimeStamp desc
```

Conclusion

Customer should create two different alerts - 1 based on the pipeline failure and other based on the trigger failure. So if the trigger fails, customer will get an alert, and if the trigger succeeds and pipeline fails, he will get an alert too.

Additional Information:

- **Icm References**: Icm link
- **Author**: vimals
- **Reviewer**: vimals
- **Keywords**:

How good have you found this content?