[Azure Synapse Analytics] Job failed with error message 'DF-EXEC-1 The connection is closed'

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

Issue

When customer use Azure Synapse Analytics connector as Sink. He may find job will fail with following error message.

```
{DF-EXEC-1 The connection is closed.}
```

By checking Kusto with query in DatabrickLoggingTest table:

cluster('adfcus.kusto.windows.net').database('AzureDataFactory').DatabrickLoggingTest | union cluster('adfneu.kusto.windows.net').database('AzureDataFactory').DatabrickLoggingTest | where * contains"e42b4904-870d-4949-ba83-73b84d2e7b68"

You may find the detailed error message:

```
shaded.msdataflow.com.microsoft.sqlserver.jdbc.SQLServerException: The connection is closed.
at shaded.msdataflow.com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDriverError(SQLServerException.ja
at shaded.msdataflow.com.microsoft.sqlserver.jdbc.SQLServerConnection.checkClosed(SQLServerConnection.java:121
at shaded.msdataflow.com.microsoft.sqlserver.jdbc.SQLServerConnection.rollback(SQLServerConnection.java:3508)
at org.apache.spark.sql.execution.datasources.jdbc.JdbcUtils$.savePartition(JdbcUtils.scala:713)
at org.apache.spark.sql.execution.datasources.jdbc.JdbcUtils$$anonfun$saveTable$1.apply(JdbcUtils.scala:839)
at org.apache.spark.sql.execution.datasources.jdbc.JdbcUtils$$anonfun$saveTable$1.apply(JdbcUtils.scala:839)
at org.apache.spark.rdd.RDD$$anonfun$foreachPartition$1$$anonfun$apply$28.apply(RDD.scala:987)
at org.apache.spark.rdd.RDD$$anonfun$foreachPartition$1$$anonfun$apply$28.apply(RDD.scala:987)
at org.apache.spark.SparkContext$$anonfun$runJob$5.apply(SparkContext.scala:2321)
at org.apache.spark.SparkContext$$anonfun$runJob$5.apply(SparkContext.scala:2321)
at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.scala:90)
at org.apache.spark.scheduler.Task.doRunTask(Task.scala:140)
at org.apache.spark.scheduler.Task.run(Task.scala:113)
at org.apache.spark.executor.Executor$TaskRunner$$anonfun$13.apply(Executor.scala:533)
at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1541)
at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:539)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
at java.lang.Thread.run(Thread.java:748)
```

Root Cause

This is highly possible caused by transient network issue.

Please check if customer's data has a large data volume. Since for write to Azure Synapse Analytics, dataflow client need to maintain a connection to write data into Azure Synapse Analytics server. If the data volume is large, the connection need to be maintained for longer time. The longer the connection is needed, the higher possible it meets transient network issue. And finally failed the whole job.

If customer's data doesn't have a large data volume, please involve SQL DW CSS engineer to help analysis customer's Azure Synapse Analytics server health status, see if the CPU/memory is full.

Resolution

Please check if customer open 'enable staging' in Sink. If not, please suggest customer to open it. This is the recommend way and have the best perf and better stability. https://docs.microsoft.com/en-us/azure/data-factory/connector-azure-sql-data-warehouse?tabs=data-factory#sink-transformation

Another way is to suggest customer set more partition number in Optimize tab to reduce the data volume in each partition.

If the data volume is not that large, please reach to SQL DW CSS engineer for further help.

The final approach is to suggest customer split the input data volume, like make several seperate dataflows to load all data.

Additional Information:

- Icm Reference: https://portal.microsofticm.com/imp/v3/incidents/details/263253478/home
- Author: Xiaojin Wang
- Reviewer: Xiaojin Wang; Shawn Xiao