A severe error occurred on the current command

Last updated by | Holger Linke | Mar 1, 2023 at 4:38 AM PST

Contents

Issue

The user/client would see some form of the following stack trace. However the key identifier for this kind of issue is bold text in below example.

> {"TimeUtc":"2018-07-03 20:39:59:3981 +0" ,"LogLevel": "Error", "LogData":{ Exception: "SqlException",
> ExceptionMessage: **"A severe error occurred on the current command. The results, if any, should be discarded. Operation cancelled by user."**,
> Message: null, CustomerId: "5263", ApiMethod: "GetRelatedItems", 5263/GetRelatedItems/24/c53cdffa-3605-420c-a8a7-4381c665d910?ProductID=905038&IsSimilar=no", Parameters: [("ProductID": "905038"), ("IsSimilar": "no"), ("RsaServerCookie": "")],
> StackTrace: "System.Data.SqlClient.SqlException (0x80131904): **A severe error occurred on the current command. The results, if any, should be discarded. Operation cancelled by user.**
> at System.Data.SqlClient.SqlConnection.OnError(SqlException exception, Boolean breakConnection, Action`1 wrapCloseInAction)
> at System.Data.SqlClient.TdsParser.ThrowExceptionAndWarning(TdsParserStateObject stateObj, Boolean callerHasConnectionLock, Boolean asyncClose)
> at System.Data.SqlClient.TdsParser.TryRun(RunBehavior runBehavior, SqlCommand cmdHandler, SqlDataReader dataStream, BulkCopySimpleResultSet bulkCopyHandler, TdsParserStateObject stateObj, Boolean& dataReady) at System.Data.SqlClient.SqlDataReader.TryConsumeMetaData()"

This is a SQL Client exception which could be thrown out of severa reasons, e.g. incomplete or bad request from both engine or client side. The error message that is exposed to the user or application log usually contains additional text, which determines more precisely what had happened.

Investigation / Analysis

1. Create an ASC troubleshooter report. Check the page "Downtime Reasons -> SQL Dumps" for any dumps that match the time of the issue. Also check for any other downtime reasons and excessive resource consumption at that time.

2. Check the `MonSQLSystemHealth` SQL errorlog in Kusto for any error message that is pointing to an exception, crash, service restart, or SQL dump:

```
let srv = "servername";
let startTime = datetime(2023-02-16 04:00:00Z);
let endTime = datetime(2023-02-16 05:00:00Z);
let timeRange = ago(1d);
MonSQLSystemHealth
| where TIMESTAMP >= startTime
| where TIMESTAMP <= endTime
//| where  TIMESTAMP >= timeRange
| where LogicalServerName =~ srv
//| where AppName =~ "appname"
//| where error_id > 0
//| where message contains "text to search for"
| project originalEventTimestamp, NodeName, LogicalServerName, AppName, error_id, message
| order by originalEventTimestamp asc
| limit 10000
```

Mitigation

If there are dumps for the time when the issue occurred, then the database probably was unavailable due to failovers. Open an IcM for getting the dump downloaded and analyzed for RCA.

If there aren't any dumps at the time, then most likely the application is cancelling the request before it completes. Identify with the customer what SQL driver the application is using (SQL Native Client, ODBC, JDBC, OLE DB, ...). Then open a collab with the team that is supporting the SQL driver and analyze it further from the client side.

How good have you found this content?