

OOM with MEMORYCLERK_SQLQUERYPLAN

Last updated by | Vitor Tomaz | Feb 24, 2023 at 3:27 AM PST

Contents

- [Issue](#)
- [Investigation / Analysis](#)
 - [ASC](#)
 - [OOM on master database](#)
 - [OOM on user databases](#)
 - [Kusto](#)
- [Mitigation](#)
 - [OOM on master database](#)
 - [OOM on user databases](#)
- [Public Doc Reference](#)
- [Internal Doc Reference](#)

Out-of-Memory errors related to MEMORYCLERK_SQLQUERYPLAN

Issue

The customer is seeing that for several of their databases health alert are raised as: "Unavailable : Resource health event (Unplanned)". The issue is possibly reported several times either shortly one after the other within several minutes, or spread over a longer period.

The full event is shown as:

At 14:03, Tuesday, 18 January 2022 UTC, the Azure monitoring system received the following information regarding your resource servername/databasename:

We're sorry your SQL database is unavailable at this time. Currently, Azure shows the downtime for your SQL database resource at a one-minute granularity. The actual downtime is likely less than a minute – average is 2s. Reason for unavailability, if found, will typically be published here 45 minutes after the outage time. Please check back then for more information.

The applications might also see logins failing with error 40613 and a state of either 84 (LogicalMasterConnectionFailed) or 85 (CloudExtensionAutheticateFailed).

Investigation / Analysis

Use the following steps to check the telemetry for further symptoms and details.

ASC

1. From ASC -> SQL Troubleshooter -> Downtime Reasons tab, you can see the frequent unavailability and if they are related to out-of-memory (OOM) issues.
2. On the same tab, check for which database the issue is reported. It might be either for the master database or the user databases.

If you see availability errors 40613 state 84 without the out-of-memory (OOM) detail, then look into article [Error 40613, State 84](#).

OOM on master database

3. Go to Performance -> Memory tab, you can find which is the top memory clerk when OOM happens. In this scenario, the top memory consumer is "MEMORYCLERK_SQLQUERYPLAN".
4. Go to Performance -> Overview tab, you see the top wait is "XE_FILE_TARGET_TVF" and "XIO_IOSTATS_BLOBLIST_RWLOCK".
5. Check with customer if they are using table-value function [sys.fn_xe_file_target_read_file](#) to read external files near the time you are facing this issue, to read extended events or audit files from a blob storage for example.

OOM on user databases

3. Go to Performance -> Memory tab, you can find which is the top memory clerk when OOM happens. In this scenario, the top memory consumer is "MEMORYCLERK_SQLQUERYPLAN".
4. Then check with the customer on the following questions:
 - Are the applications using [cursors](#) in their queries?
 - Is the customer running [DBCC CHECKTABLE \(Transact-SQL\)](#) or [DBCC CHECKDB \(Transact-SQL\)](#) around when they see the OOM occur?
 - Have they noticed that the OOM happen during specific queries or workloads that they are running? If so, do you have the query hash or information about those specific queries?

Both cursors and DBCC CHECKTABLE/CHECKDB use MEMORYCLERK_SQLQUERYPLAN, which is the main memory consumer causing the OOM.

Kusto

Run the following Kusto query to confirm the 40613 errors and their state.

```
let ServerName = "servername";
MonLogin
| where logical_server_name == ServerName
  and originalEventTimestamp >= datetime(2022-01-18 09:00)
  and error == 40613
  and event == "process_login_finish"
  and (is_success == false or is_normal_logout == 0 or error > 0)
| summarize min(originalEventTimestamp), max(originalEventTimestamp), count() by database_name, error, state,
| project min_originalEventTimestamp, max_originalEventTimestamp, database_name, error, state, state_desc, cou
| order by min_originalEventTimestamp asc
```

Sample output:

min_originalEventTimestamp	max_originalEventTimestamp	database_name	error	state	state
2022.01.18 09:59:17	2022.01.18 09:59:17	database1	40613	84	Logi
2022.01.18 09:59:18	2022.01.18 10:01:43	database2	40613	84	Logi
2022.01.18 09:59:21	2022.01.18 10:01:25	database3	40613	84	Logi
2022.01.18 09:59:22	2022.01.18 10:01:25	database4	40613	84	Logi
2022.01.18 09:59:22	2022.01.18 10:01:25	database5	40613	84	Logi
2022-01-18 09:59:59	2022.01.18 10:01:25	database6	40613	84	Logi
2022-01-18 10:00:05	2022.01.18 10:01:25	database7	40613	84	Logi
2022.01.18 10:01:05	2022.01.18 10:01:43	database3	40613	85	Clo
2022.01.18 10:01:05	2022.01.18 10:02:00	database5	40613	85	Clo
2022.01.18 10:01:05	2022.01.18 10:02:00	database2	40613	85	Clo
2022.01.18 10:01:05	2022.01.18 10:04:49	master	40613	14	Insta
2022.01.18 10:01:06	2022.01.18 10:01:06	database4	40613	114	Retr

If you only see availability errors 40613 state 84 without corresponding state 85 entries, then also look into article [Error 40613, State 84](#).

Mitigation

OOM on master database

- Use alternative ways to read extended event files, like PowerShell cmdlet [Read-SQLXEvent](#) 🔗.

OOM on user databases

- Tune the queries that are using cursors. Replace the cursors with set-based operations as much as possible.
- Avoid using [DBCC CHECKTABLE \(Transact-SQL\)](#) 🔗 or [DBCC CHECKDB \(Transact-SQL\)](#) 🔗 at times with high workloads. Run the DBCC commands outside of normal business hours, if possible.

Public Doc Reference

- [sys.dm_os_memory_clerks \(Transact-SQL\)](#) 🔗

Internal Doc Reference

- [lcM 283871988](#) 

How good have you found this content?

