Disable CDC fails with error 22831

Last updated by | Akio Hose | Mar 28, 2022 at 12:23 AM PDT

Contents

Issue

The customer wanted to disable Change Data Capture (CDC) using sys.sp_cdc_disable_db but fails with error 22831.

```
Msg 22831, Level 16, State 1, Procedure sys.sp_cdc_disable_db_internal, Line 293 [Batch Start Line 42]
Could not update the metadata that indicates database cdctest is not enabled for Change Data Capture. The fail
```

Possible root cause

Error 9002 can be caused by several reasons as outlined below. However, in this case (IcM 281109301), the issue was caused by the capture job not running (which for some reason the Change Data Capture job didn't exist).

Common reasons for a full transaction log

- Log not being truncated
- Disk volume is full
- Log size is set to a fixed maximum value or autogrow is disabled
- Replication or availability group synchronization that is unable to complete

DML operations performed on the a CDC enabled table will be marked as "REPLICATE" in the transaction log to indicate the record is part of the CDC transaction. The main role of the CDC capture job is to scan the transaction log records marked for replication and insert records to the cdc.<capture_instance>_CT table. Following the scanning process, log records are marked as distributed which will allow truncation of those records during a transaction log backup. If the capture job isn't running, records in the transaction log marked for replication are retained and these records (and possibly other log records after the "REPLICATE" records) will eventually fill up the transaction log.

Mitigation

There are two options to mitigate the issue. The first method is straightforward and mostly recommended. If for some reason altering the transaction log size is not possible, the customer can follow the second method.

- Method 1: Expand the transaction log size
- Method 2: Force marking the log records as "distributed"

Method 1

This method is straightforward and mostly recommend. See [Increase the Size of a Database] for details.

Method 2

We can use the [sp_repldone] stored procedure to mark the log records having "REPLICATE" as distributed. After marking those records as distributed, wait for the transaction log backup to run (Usually runs every 5 to 10 minutes - [Backup frequency]), and this will truncate the log records and make some room within the transaction log to enable to run sys.sp_cdc_disable_db successfully.

```
-- Command to mark all REPLICATE log records as distributed.
EXEC sp_repldone @xactid = NULL, @xact_seqno = NULL, @numtrans = 0, @time = 0, @reset = 1
```

If Change Data Capture and transactional replication coexists, the subscription should be reinitialize using a new snapshot. This is because the sp_repldone issued in the above example will mark other log records (that are part of transactional publication) as distributed as well, and may cause some record "gaps" between the publication database and the subscription database. See [Reinitialize a Subscription] on steps to reinitialize a subscription.

Other useful commands

Here are couple of useful commands to verify some items

- Find out current transaction log usage

  You can use [DBCC SQLPERF (Transact-SQL)] to find out current log space usage.

```
DBCC SQLPERF(LOGSPACE)
```

- Find out when the transaction log backup started

```
sp_readerrorlog 0, 1, 'backup log started'
```

- Find out current records pending to be distributed

  The sp_repltrans stored procedure will return list of transactions that are marked for replication and are pending for distribution. This can be used to check if there pending records for replication in the transaction log. Make sure that the CDC capture job is NOT running (or Log Reader Agent job in case of coexistence of CDC and Transactional Replication) when running these stored procedure otherwise the execution might fail due to existing connection.

The sp_replflush procedure is called to ensure to release connections from the current session to the transaction log file. Note that this command might fail if another session (Such as the capture job running in the background) is running for scanning. Make s

```
exec sp_replflush
exec sp_repltrans
exec sp_replflush
```

How good have you found this content?

🙂 🙁