https://repost.aws/knowledge-center/dms-sequence-errors-cdc-oracle

**Why did my AWS DMS CDC task using Oracle as a source fail with a "Sequence doesn't exist" message?**

7 minute read

0

*I want to migrate data from my on-premise or Amazon Relational Database Service (Amazon RDS) for Oracle database using AWS Database Migration Service (AWS DMS). The AWS DMS change data capture (CDC) task runs as expected, but then fails with an error similar to this: "Oracle CDC maximum retry counter exceeded" (archivelog sequence does not exist)" How can I troubleshoot and resolve this error?*

Short description

When you use an Oracle database as a source for your migration task, AWS DMS gets the data from the table during the full-load phase. During the CDC phase, AWS DMS reads from the archived redo logs. Then, AWS DMS captures the redo logs from the source Oracle database, and applies only the committed changes to the target database.

You might see a sequence log error that looks like this:

**"03980512: 2022-05-23T12:33:11 [SOURCE\_CAPTURE ]E: Archived Redo log with the sequence 232488 does not exist, thread 1 [1022318] (oradcdc\_thread.c:624"**

To troubleshoot this error, follow these steps:

1.    Run this query on the source Oracle database to check if the archive log sequence is present on the source. For example, this query checks for redo log sequence 232488:

select name, dest\_id, thread#,

sequence#, archived, applied, deleted, status, first\_time, next\_time,

completion\_time from v$archived\_log where sequence# =232488;

2.    Run this command on the location of the archive log files on the source server. This command checks if the archive log is physically present:

ls -l \* 232488\*

3.    Check the archivelog destination (log\_archive\_dest) on the source Oracle database.

The examples in this article investigate this error with two different root causes:

* AWS DMS is looking for the redo log in the wrong DEST\_ID
* DEL is YES and the archivelog sequence doesn't exist on source Oracle

Resolution

Example 1 - AWS DMS is looking for the redo log in the wrong DEST\_ID

The error shows that the archived redo log sequence 232488 is missing on the source Oracle DB. This might happen if the log is deleted from the source Oracle database. This causes the task to fail.

01788702: 2022-06-07T17:10:31:206453 [SOURCE\_CAPTURE ]D: Going to

prepare the statement 'select supplemental\_log\_data\_min, DATABASE\_ROLE,

SUPPLEMENTAL\_LOG\_DATA\_PK, SUPPLEMENTAL\_LOG\_DATA\_ALL from v$database'

(oracle\_endpoint\_conn.c:114)

01788702: 2022-06-07T17:10:31:209648 [SOURCE\_CAPTURE ]I: Database role is 'PHYSICAL STANDBY' (oracle\_endpoint\_conn.c:139)

1.    Run this query on the source database to see if archive log sequence 232488 exists in the source:

select name, dest\_id, thread#, sequence#, archived, applied, deleted,

status, first\_time, next\_time, completion\_time from v$archived\_log

where sequence# in (232488);

Output:

NAME DEST\_ID THREAD# SEQUENCE# ARC APPLIED DEL S FIRST\_TIM NEXT\_TIME COMPLETIO

--------------------------------------------- ---------- ---------- ---------- --- --------- --- - --------- --------- ---------

/orafra/prdsvbo/arc/1\_232488\_950180179.arc 2 1 232488 YES YES NO A 07-JUN-22 07-JUN-22 07-JUN-22

2.    **Check the DEL column. If this column lists NO, then the archivedlog sequence exists in the source database. If it lists YES, then the archiveredlog doesn't exist, and might have been purged from the source because of the retention period**.

In this example, the output indicates that archivelog sequence exists in the source. But, the AWS DMS task still fails with an error that says the sequence doesn't exist. For troubleshooting when DEL is YES, see Example 2.

3.    Next, check the DEST\_ID column. By default, AWS DMS captures the redo logs on DEST\_ID 1. In this example, you see this snippet in the logs:

01788702: 2022-06-07T17:10:31:658376 [SOURCE\_CAPTURE ]I: Used Oracle archived Redo log destination id is '1' (oracdc\_merger.c:639)

01788702: 2022-06-07T17:10:31:658420 [SOURCE\_CAPTURE ]I: Oracle instance uses more than one archived Redo log destination id. Please configure the correct destination id, if Redo logs of '1' destination cannot be accessed (oracdc\_merger.c:642)

So, the AWS DMS task fails because it is looking for the redo log in DEST\_ID 1, but the redo log file is present in DEST\_ID 2.

4.    To mitigate this error, **use this extra connection attribute (ECA)** in the source endpoint:

additionalArchivedLogDestId=2

For more information about additionalArchivedLogDestID, see [Source data types for Oracle](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Source.Oracle.html#CHAP_Source.Oracle.ConnectionAttrib).

5.    After you configure the ECA, resume the AWS DMS task. In this example, the logs show that AWS DMS can now capture the available redo log sequence 232488 from DEST\_ID 2.

01898667: 2022-06-08T05:45:08:535588 [SOURCE\_CAPTURE ]D: Going to retrieve archived REDO log with sequence 232488, thread 1 (oradcdc\_thread.c:510)

01898667: 2022-06-08T05:45:08:535607 [SOURCE\_CAPTURE ]T: Use a prepared statement to access v$archived\_log, thread 1 (oradcdc\_thread.c:587)

01898667: 2022-06-08T05:45:08:598396 [SOURCE\_CAPTURE ]D: Going to open Redo Log with original name '/orafra/prdsvbo/arc/1\_232488\_950180179.arc', thread id '1' (oradcdc\_redo.c:492)

01898667: 2022-06-08T05:45:08:599614 [SOURCE\_CAPTURE ]D: archived Redo log '/orafra/prdsvbo/arc/1\_232488\_950180179.arc' with sequence 232488 is opened, thread id '1' (oradcdc\_redo.c:747)

Example 2 - DEL is YES and the archivelog sequence doesn't exist on source Oracle

As previously detailed in step 2, if DEL is YES, then it's possible that the archivelog sequence was purged from the source Oracle database.

**Note**: Oracle instances delete the archive log files to minimize the space that the archive logs take up.

**Steps for On-premises or Amazon Elastic Compute Cloud (Amazon EC2)**

If you scheduled the Recovery Manager (RMAN) to delete noprompt archivelog until time SYSDATE-1, then this schedule deletes all archive log files older than one day. To increase this, modify this command to SYSDATE-2, or turn off the schedule.

Follow these steps to find the current retention period, and then increase it.

1.    Connect to RMAN.

2.    Show all:

CONFIGURE ARCHIVELOG DELETION POLICY TO NONE; # default

**Note**: The default Archivelog deletion policy value is NONE.

3.    Change the deletion policy value to a value that's sufficient to retain the archivelogs on the source on-premises database:

CONFIGURE ARCHIVELOG DELETION POLICY BACKED UP integer TIMES TO DEVICE TYPE

CONFIGURE ARCHIVELOG DELETION POLICY TO BACKED UP 2 TIMES TO SBT;

**Steps for Amazon RDS**

Amazon RDS deletes the archive log files every five minutes, and keeps a copy in an Amazon Simple Storage Solution (Amazon S3) bucket. You can use this copy to do a point-in-time restore.

1.    Increase the retention of the archive log files using the steps in [Performing common log-related tasks for Oracle DB instances](https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.Oracle.CommonDBATasks.Log.html).

2.    Verify the retention period that's defined for the archive log files:

exec RDSADMIN.RDSADMIN\_UTIL.SHOW\_CONFIGURATION;

3.    Check if the archive log file that you fetched in the previous query is physically available on the source database server.

SQL>select name, archived, deleted, status, sequence# from v$archived\_log where sequence# = 232488;

**Note**: In Amazon RDS, the deleted column shows NO, even though it was purged. This information comes from the control file.

4.    To check if the redo log exists in Amazon RDS, run this query:

select \* from table(RDSADMIN.RDS\_FILE\_UTIL.LISTDIR('ARCHIVELOG\_DIR')) where filename='redolog-5924417-1-1013939085.arc' order by mtime desc;

**Note**: AWS DMS doesn't control the purging of the archive redo logs. Archive redo logs are purged by the source on-premises or RDS for Oracle database. Instead, AWS DMS confirms that it can't find the log when it tries to process the next LSN.

5.    If its available, restore the missing archive log of sequence 232488 in the source destination, and then resume the task. All the redo logs following 232488 must be present on the source before you can successfully resume the AWS DMS task.

If you can't restore the missing archive log sequence, try restarting the task from the full-load phase, and then start the migration over.

Related information

[Extra connection attributes when using Oracle as a source for AWS DMS](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Source.Oracle.html#CHAP_Source.Oracle.ConnectionAttrib)

[Using an Oracle database as a source for AWS DMS](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Source.Oracle.html)

[Performing common log-related tasks for Oracle DB instances](https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.Oracle.CommonDBATasks.Log.html)