Msg 5061: ALTER DATABASE failed because a lock could not be placed on database 'X'

Last updated by | Radhika Shah | May 30, 2022 at 4:55 PM PDT

Contents

- Issue
- Investigation/Analysis
- Classification

Issue

Today, we worked on a new service request where our customer tried to modify a parameter of a new database, just a couple minutes after that they have created it.

During aprox. 5-10 minutes they were not able to modify this parameter after the database was created and our customer asked why.

Every time that they tried to change the value of this parameter they got the following error message:

Msg 5061, Level 16, State 1, Line 27

ALTER DATABASE failed because a lock could not be placed on database 'databasename'. Try again later.

Msg 5069, Level 16, State 1, Line 27

ALTER DATABASE statement failed.

Investigation/Analysis

Please, remember that the creation of the database is an asynchronous process and even when SQL SERVER Management Studio returned the message that the database has been created, other process may take place in the background for example, enabling TDE, running a backup, etc.. for this new database. If any of this parameter needs to have an exclusive lock during this period of time you could see this error message.

Also, we need to know that just only a few parameters are enabled at the moment of the execution of CREATE DATABASE as TSQL or using SQL Server Management Studio - https://docs.microsoft.com/en-us/sql/t-sql/statements/create-database-transact-sql?view=azuresqldb-m...

How we found it:

During the execution of ALTER DATABASE and before having the error message that is not possible to obtain the exclusive lock, we found that some operations are doing in the background blocking any exclusive locking using this article: https://techcommunity.microsoft.com/t5/azure-database-support-blog/lesson-learned-22-how-to-identify...

If you want to automate this process using a script, for example, to change the containment parameter:

We suggested to use the following script as an example:

1. First, create a store procedure to change the parameter of the database.

```
CREATE OR ALTER PROCEDURE ChangeStatus
AS
BEGIN
BEGIN TRY
        ALTER DATABASE <databasename> set containment =partial
END TRY
BEGIN CATCH
    PRINT @@ERROR
END CATCH
END
```

2. Second, create a script to retry the operation until this operation has been finished

```
DECLARE @VALUE AS INT = -1
DECLARE @nTimes as INT = 0
WHILE(@nTimes <=10)
BEGIN
SELECT @Value = (SELECT Containment FROM sys.databases WHERE name='databasename')
   IF @VALUE <> 0 BREAK;
   SET @nTimes = @ntimes +1
   IF @Value = 0
BEGIN
   PRINT 'Doing # '
   EXEC ChangeStatus
   WAITFOR DELAY '00:02:00'
END
END
```

Also available at https://techcommunity.microsoft.com/t5/azure-database-support-blog/lesson-learned-118-having-alter-database-failed-because-a-lock/ba-p/1105276

Classification

Root cause tree:

How good have you found this content?