

Error 40613 State 126

Last updated by | Subbu Kandhaswamy | Mar 7, 2022 at 2:28 PM PST

Contents

- [Issue:](#)
- [Troubleshoot](#)
 - [Using ASC](#)
 - [Using Kusto](#)
- [RCA Template](#)
- [Classification](#)

Issue:

Master database unable to handle login requests to database that is currently not attached to the instance (non-existing database) and requests from user to reach the database get piled up.

This can occur as a result of a quorum loss and DB not attached to the instance resulting in lookup failure.

Note: Logins to non-existent databases cause master to open an expensive XODBC connection to itself, which is also a potential cause for the failure.

Troubleshoot

We will see 40613 with State 126 (LoginSessDb_DbNotFound) emitted when the lookup of the DB fails in these cases.

Using ASC

We deduct the scenarios thru ASC SQL troubleshooter results you can review under SQL Troubleshooter -> Connectivity -> Data Explorer section.

- If you're new to ASC, follow the instructions to generate SQL-troubleshooter results from here -> [How to generate or use SQL troubleshooter results](#)
- Confirm the occurrence of the issue from the results in Data Explorer
- Check if the issue is still occurring (if customer reported as ongoing)

Using Kusto

- To deduct the occurrence of this issue using Kusto, run the following query by passing resource information and impact time frames.

```

MonLogin
| where TIMESTAMP between (datetime(yyyy-mm-dd hh:mm:ss)..datetime(yyyy-mm-dd hh:mm:ss))
| where logical_server_name =~ <ServerName>
| where database_name == <DatabaseName>
| where event == "process_login_finish" and package == "sqlserver"
| where is_success == 0
| summarize count(), min(originalEventTimestamp), max(originalEventTimestamp) by package, error, state, state_

```

Additional scenarios

- When you connect to logical master using 'sqlcmd' it will timeout during the time of the issue.
- We may also notice 18456 with State 126 or State 5 in the telemetry.

Run the below query to confirm the behavior

```

let startTime = ago(1d);
let threshold = 100;
MonLogin
| where originalEventTimestamp > startTime
| where package == "sqlserver"
| where error == 18456 and (state == 126 or state == 5)
| summarize bad_logins = count(), start_bad_logins = min(originalEventTimestamp), end_bad_logins = max(originalEventTimestamp)
| where bad_logins > threshold
| join (
  MonLogin
  | where originalEventTimestamp > startTime
  | where event == "process_login_finish"
  | where error == 26078 or (error == 40613 and state == 84)
  | summarize failed_logins = count(), unavailable_databases = makeset(database_name), start_failures = min(originalEventTimestamp)
  | where failed_logins > threshold
) on logical_server_name
| where db_appname != master_appname
| summarize affected_databases = makeset(unavailable_databases), failed_logins = sum(failed_logins), start_failures = min(start_failures)
| extend mitigation = strcat('Run following command, then kill SQL process of logical master: Set-ServerConfig

```

Mitigation Steps

If the issue persists and customer is impacted, the mitigation step may require failing over (killing) the logical master's, so please escalate to Gateway team by following our [escalation process](#)

RCA Template

Summary of Impact:

Between {StartTime} and {EndTime}, Database {DatabaseName} on Server {ServerName} was unavailable due to logical master database issue.

Cause:

The login requests to the SQL database first reaches the logical master db (except for contained user access) for both SQL Server and Windows authentication to confirm the credentials are valid to connect to user database.

In this case the database you're trying to reach either going thru failover process and not attached to the sql instance OR the database does not exist.

- If the database name is valid, please retry the connection again in few seconds (*retry logic is recommended, if not enabled already*) to resume access to the database.
- This issue can also occur, if the database doesn't exist (recently deleted Or using incorrect database name)

Next Steps:

We will continue to investigate and improve our automation to identify such node issues more quickly. We sincerely apologize for the inconvenience caused.

Classification

Root Cause: Azure SQL DB v2\connectivity\

How good have you found this content?

