Error 40613, State 22

Last updated by | Tayyab Mohd Sayyed | Apr 4, 2022 at 4:39 PM PDT

Contents

Issue

Database unavailable for customer occurs during tcp connection open to xdbhost on the DB node. This occurs before socket duplication and mostly due to networking related issue or XDBHost issue and not SQL Server related.

Troubleshooting

Using Azure Support Center

ASC Insight

We detect this issue in Azure Support center and generate insight against the impacted resource, with required details. Ensure the issue timeframe is within the impact timeframe and related to customer downtime before sharing the insights with customer.

ASC Resource Explorer

If you're new to Resource explorer,

- Click on Resource explorer from ASC landing/main page.
- Check the impacted Subscription, Server, Database from the left section.
- FYI - By design resource explorer auto runs diagnostics for current time (right section).

To run (New) diagnostics Or choose previously run (Existing) diagnostics, choose the [Diagnostics] tab and click [Create Report] Or choose available diagnostics (eye icon) to view as shown below.

Then choose

- Connectivity
- Database explorer
- Navigate to the failures based on the impact, in this case **40613 state 22** to review and confirm the issue.

Proxy timeout issue due to TCP timeout - 40613 State -22 /Lookup state = 10060.

Proceed with Troubleshooting ONLY if ASC failed to assist.

Using Kusto

Depending on the lookup_code for this state, we will be able to narrow down the cause for this occurrence.

- Lookup code:1460 - Login spikes causing the login fail (Lookup code = 1460) and NO backend related problems.
- Lookup Code:10060 - Issue related to XDbHost (Slow / Stuck) and XdbHost process is hung. This means, client is being throttled due to slow connections to backend.

Refer Gateway Proxy debugging for analyzing the issue based on the lookup state.

```
MonLogin
| where TIMESTAMP >= ago(5h)
| where logical_server_name =~ "laazmxsqlsqp001" and database_name =~ "OceanProdMX" and event == "process_logi
| summarize count() by error, state, result, instance_name, instance_port,driver_name, driver_version,bin(orig
```

error	state	result	instance_name	instance_port	d
0	0	e_crContinueSameState	d61a3998c6b8.tr30.southcentralus1-a.worker.database.windows.net ↗	11102	N JI C
0	0	e_crContinueSameState	d61a3998c6b8.tr30.southcentralus1-a.worker.database.windows.net ↗	11102	N JI C
40613	22	_crDisconnect	d61a3998c6b8.tr30.southcentralus1-a.worker.database.windows.net ↗	11102	N JI C

e_crContinueSameState = Proxy e_crContinue = Redirect e_crDisconnect = Disconnected

From above, We can see high number of connections that are going to proxy and they are using JDBC 4.0 which is consider as down level client. https://blogs.msdn.microsoft.com/sqlcat/2016/09/08/connect-to-azure-sql-database-v12-via-redirection/ ↗

The driver in these connection failure is JDBC 4.0, This driver doesn't supported for redirection as indicated in the article above

We can quickly check if that client connections are inside Azure or outside using CSS Toolkit web ↗

IP address 104.214.119.0 came out as allow listed from Networks.xml deployed world wide and also it is inside Azure. By default redirection is enabled for all clients with in Azure, since this customer using down level client they are going through proxy route and we do see logins were disconnected because it was in the queue for too long. Since customer has retry logic in their application, it was transparent. We also noticed that customer application is averaging around 400-500 logins/minute and some times more. By design, when there are more login disconnections, Resource health check displays as database as unavailable and portal blade also reads same information and that is reason customer seeing DB unavailable in portal but actual availability just fine.

RCA Template (Proxy Throttling Scenario)

Summary of the Impact : We identified that connections utilizing the Proxy method of connecting to database <**DatabaseName**> on server <**ServerName**> were being throttled. This occurs as a result of slow connections to the backend depending on connection policy and traffic.

Cause : By default, connections originating outside of the Azure network boundary will use the proxy method which will be a shared common endpoint for connecting to the database(s) in that region. If performance regressions are seen or heavy traffic from a client are seen, throttling can occur. When using the proxy method, only the 1433 outbound port from the clients network needs to be allowed. Clients within Azure use the redirect method for authentication. While establishing a connection, this method provides a redirect token to the client, which includes the specific endpoint of the SQL database. This instance will be listening on a dynamic port within the range of 11000-11999 and 14000-14999 which will require the network where the client resides to allow those outbound ports in addition to the standard 1433 port for SQL.

Mitigation: If this is causing issues for your application workload you can force all connections to your database to use the redirect method to connect, avoiding the potential of this scenario occurring.

Classification

Root Cause: Azure SQL DB v2\Connectivity\Login \ Bydesign - User Error)

How good have you found this content?

😊 🙁