Error 40532 State 4

Last updated by | Holger Linke | Mar 24, 2023 at 2:43 AM PDT

Contents

- Issue
- Investigation
- Analysis
  - Scenario 1
  - Scenario 2
- Mitigation
- Public Doc Reference
- Root Cause Classification

Error 40532 State 4 and new gateways

Issue

The customer reported connectivity issues without any change on their side. The error message was similar to the following:

> ADO NET Source has failed to acquire the connection {D24D668F-DF92-4FED-A10B-7C66C77D6736} Cannot open server 'XXXXXXXX' requested by the login. The login failed.'.

The error might occur consistently or intermittently.

Investigation

> **Note:** The symptoms are similar to the issue described in Wiki article "Error 40532 State 4 on master database". The main difference appears to be that the error 40532 state 4 is reported for either the `master` database or the user database - but this is still uncertain for now due to the lack of comparable case scenarios. Make sure to check both articles for narrowing down the cause of your customer's issue.

Check for failed logins on MonLogin. The customer error message comes up as Error 40532 State 4:

```
let srv = "servername";
let db = "databasename";
MonLogin
| where TIMESTAMP >= ago(4d)
| filter logical_server_name =~ srv
| where database_name =~ "master" or database_name =~ db
| extend logical_server_name = tolower(logical_server_name), database_name = tolower(database_name)
| where event == "process_login_finish"
| where is_success == 0
| summarize
    count(),
    min_TIMESTAMP = format_datetime(min(TIMESTAMP), "yyyy-MM-dd HH:mm:ss"),
    max_TIMESTAMP = format_datetime(max(TIMESTAMP), "yyyy-MM-dd HH:mm:ss")
    by database_name , error, ['state'], peer_address , is_user_error, is_vnet_address, application_name, alia
| order by database_name, error asc, ['state'] asc


Sample output:
database_name  error  state  peer_address  is_user_error  is_vnet_address  application_name   ClusterName
-------------- ------ ------ ------------- -------------- ---------------- ------------------ ----------------
databasename   40532  4      13.93.114.x   True           False            Microsoft JDBC...  cr2.westeurope1-
```

The `ClusterName` column indicates that CR2 (Control Ring 2) is used as gateway for the request.
You can check the current gateway by sending a `ping` or `nslookup` from a command line, like this:

```
ping servername.database.windows.net
Pinging cr4.westeurope1-a.control.database.windows.net [104.40.168.105] with 32 bytes of data:



nslookup servername.database.windows.net

Server:  dsldevice.lan
Address:  fe80::c6ea:1dff:fef0:37e0

Non-authoritative answer:
Name:    cr7.westeurope1-a.control.database.windows.net
Address:  52.236.184.163
Aliases:  servername.database.windows.net
          dataslice9.westeurope.database.windows.net
          dataslice9westeurope.trafficmanager.net
```

The results show different control rings: CR2, CR4, and CR7. Depending on the region, there might be less or more than that.
Get the IP addresses for the control rings that you have identified so far:

```
ping cr2.westeurope1-a.control.database.windows.net
Pinging cr2.westeurope1-a.control.database.windows.net [40.68.37.158] with 32 bytes of data:

ping cr4.westeurope1-a.control.database.windows.net
Pinging cr4.westeurope1-a.control.database.windows.net [104.40.168.105] with 32 bytes of data:

ping cr7.westeurope1-a.control.database.windows.net
Pinging cr7.westeurope1-a.control.database.windows.net [52.236.184.163] with 32 bytes of data:
```

Let's check the documented Gateway IP addresses to see how these ping results align with the expected gateway addresses:

Region name	Gateway IP addresses	Gateway IP address subnets
West Europe	40.68.37.158, 104.40.168.105, 52.236.184.163	104.40.169.32/29, 13.69.112.168/29, 52.236.184.32/29

These should be exactly the IP addresses that were returned from the ping and nslookup - in this example they are.

Analysis

Scenario 1

In the example shown above, the control ring names all resolved to valid gateway IP addresses. This means that the cause of the issue is very likely on the customer's network. For example: the customer's application is hosted in their corporate on-premise environment and their internet proxy server allows only individual IP addresses. The error would occur if the on-premise proxy blocked the IP address for CR2. If CR4 and/or CR7 are allowed, they might see intermittent behaviour depending on what gateway node is used; if CR4 and CR7 are also blocked, the error occurs consistently.

Scenario 2

In another case, the control ring shown in MonLogin was no longer available because of a previous Azure SQL Database traffic migration to newer Gateways. The customer had either allowed only the static gateway IP address, or their DNS cache was still pointing to the decommissioned control ring.

Mitigation

This issue requires further investigation from the customer's network side. They possibly have blocked general traffic to Azure and only allowed specific IP addresses. The strong recommendation is to allow traffic to the Gateway IP address subnets as explained in Azure SQL Database and Azure Synapse Analytics connectivity architecture.

In the case of a recent change in the gateway IP address, the custommer should clear their DNS cache, or reduce TTL (Time to Live) of the DNS record.

Public Doc Reference

Azure SQL Database traffic migration to newer Gateways

Root Cause Classification

Cases resolved by this TSG should be coded to the following root cause:
Azure SQL DB v2\Connectivity\Network-Client\Other

How good have you found this content?