

Error 40532 State 150

Last updated by | Holger Linke | Feb 10, 2023 at 4:56 AM PST

Contents

- [Issue](#)
- [Investigation](#)
- [Analysis](#)
 - [Scenario 1](#)
 - [Scenario 2](#)
- [Mitigation](#)
 - [Scenario 1](#)
 - [Scenario 2](#)
- [Public Doc Reference](#)
- [Root Cause Classification](#)

Issue

The customer is trying to connect to Azure SQL Database and it is failing with the following symptoms:

Customer-reported error message:

Error 40532

Cannot open server 'servername' requested by the login. The login failed.
ClientConnectionId:068845b9-086c-48c2-a44e-1d5ec014bac9

Error 40914

Cannot open server 'servername' requested by the login. Client is not allowed to access the server.

MonLogin reports the error as:

Error 40532 State 150

Investigation

The customer has very likely made a mistake with configuring the firewall rules of their Azure SQL Database server for accessing it from a virtual network. It's important to understand the the configuration is per subnet inside of a virtual network and not for the whole virtual network.

Check the `MonLogin` Kusto telemetry to get more details about the symptoms:

```

let srv = "servername";
let startTime = datetime(2023-02-09 04:00:00Z);
let endTime = datetime(2023-02-09 16:00:00Z);
let timeRange = ago(1d);
MonLogin
| where TIMESTAMP >= startTime
| where TIMESTAMP <= endTime
//| where TIMESTAMP >= timeRange
| where event == "process_login_finish"
| where error == '40532'
| where logical_server_name =~ srv or LogicalServerName =~ srv
| project PreciseTimeStamp, SubscriptionId, logical_server_name, database_name, package, event, error, state,

```

Compare this output with the actual configuration as it is shown in the CMS tables ("the source of truth"):

```

select rule_name, vnet_name, subnet_name, vnet_traffic_tag, subnet_traffic_tag
from vnet_firewall_rules
where logical_server_name = 'servername'

```

Sample output MonLogin (abbreviated):

SubscriptionId	database_name	package	error	state	peer_address	is_vnet_address	v
50eac240-453d-4c1b-8f46-480d9b41f717	datasenname	sqlserver	40532	150	0.0.0.x	True	3
50eac240-453d-4c1b-8f46-480d9b41f717	datasenname	sqlserver	40532	150	0.0.0.x	True	3
50eac240-453d-4c1b-8f46-480d9b41f717	datasenname	sqlserver	40532	150	0.0.0.x	True	3
50eac240-453d-4c1b-8f46-480d9b41f717	datasenname	sqlserver	40532	150	0.0.0.x	True	3

Sample output vnet_firewall_rules:

rule_name	vnet_name	subnet_name	vnet_traffic_tag	subnet_traffic_tag	vnet_subscription_id
vnetrulename1	vnetname1	subnetname1	309766167	3	50eac240-453d-4c1b-8f46-480d9b41f717
vnetrulename2	vnetname2	subnetname2	309017938	1	70981115-5fc9-4faa-a503-18e1541c0

The following columns are relevant:

- Confirm that the `state` of error 40532 is 150. If it is not 150, then this TSG doesn't apply; look for a TSG that covers the state you are seeing.
- If `vnet_gre_key == vnet_traffic_tag` then the customer configured this VNet to access the database
- If `vnet_gre_key == vnet_traffic_tag` but `vnet_subnet_id != subnet_traffic_tag` then the VNet firewall rule was set for one subnet, but the customer is trying to access the database from another subnet on this VNet.
- If `vnet_gre_key != vnet_traffic_tag` then the customer configured a different VNet for the database, or no VNet at all.

In the example output above, we can see that `vnet_gre_key == vnet_traffic_tag == 309766167` and `vnet_subnet_id == 5 != subnet_traffic_tag == 3`, so the customer has indeed misconfigured their environment and is trying to access the database from the wrong subnet.

Analysis

Scenario 1

The error 40532 state 150 occurs due to a missing virtual network rule for the originating subnet. If the subnets have SQL Endpoints enabled, the traffic will take an internal backbone route, and the Azure SQL Database recognizes this traffic as originating from a subnet in the customer's VNet.

If the firewall doesn't have a VNet rule created, the connection will be blocked. It is essentially the same as a traditional firewall rule blocking an IP address that is not allowed.

These errors will manifest as error 40914 or error 40532, state 150 in MonLogin on the `sqlserver` package.

Scenario 2

The error 40532 state 150 can also occur when connecting two PaaS servers that are hosted in different subscriptions, for example, if the connection is attempted between Azure SQL Databases residing on different subscriptions.

The root cause is the same as on scenario 1: the lack of a virtual network rule on the target server firewall.

Mitigation

Scenario 1

The issue might have started by turning on VNet service endpoints for Microsoft.Sql in the subnet. This enables the endpoints for Azure SQL Database, Azure Synapse Analytics, Azure Database for PostgreSQL server, Azure Database for MySQL server and Azure Database for MariaDB.

The customer possibly tried to achieve the following:

- Aiming to connect to SQL Database using service endpoints, Microsoft.Sql was enabled in the subnet but the VNet rule for the originating subnet in the 'Firewalls and virtual networks' settings on the server was not added.
- Aiming to connect to other database service, like Azure Database for MySQL as an example, Azure SQL Database was also impacted. If the customer intended to enable it only for MySQL, they possibly haven't considered that connections to SQL Database from this subnet might fail if VNet firewall rules are not set for SQL Database.





Possible mitigation steps:

- Add the VNet rule for the originating subnet in the 'Firewalls and virtual networks'.
- Consider removing the Microsoft.Sql service endpoint from the subnet, but this will impact all the services mentioned above.

Scenario 2

To mitigate the issue, you need to add a firewall rule for the VNet/Subnet of the source server to the target server.

Public Doc Reference

- [Use virtual network service endpoints and rules for servers in Azure SQL Database](#) 
- [Details about virtual network rules](#) 
- [Add a virtual network firewall rule to your server](#) 
- [Troubleshoot errors 40914 and 40615](#) 

Root Cause Classification

Cases resolved by this TSG should be coded to the following root cause:

Root Cause: Azure SQL v3\Connectivity\Network (Azure)

How good have you found this content?

