Connection timeouts

Last updated by | Vitor Tomaz | Dec 14, 2021 at 11:26 AM PST

Contents

Public Doc Reference

Troubleshooting connectivity issues and other errors with Microsoft Azure SQL Database

https://docs.microsoft.com/en-us/azure/sql-database/troubleshoot-connectivity-issues-microsoft-azure-sql-database

Troubleshooting

Step 1: Understand their network environment

Ask the scoping questions:

Can I know where you try to connect from to the SQL MI? Are you trying to connect from VM in the same VNET as MI or different VNET? Could you please provide the source machine VNET name and machine name?

Or are you trying to connect from on-premise network? If so, do you use express route, site to site or point to site?

Can I know when the issue started to happen? Was it connecting fine before?

Check which scenario that customer uses https://docs.microsoft.com/en-us/azure/sql-database/sql-database-managed-instance-connect-app

Step 2: Use ASC

We can get MI subnet info from ASC under SQL MI.

Once we get info from step 1, we can check the corresponding configurations of source network in ASC under resource explorer too.

Basic info to check if they matches to what documentation requires:

https://docs.microsoft.com/en-us/azure/sql-database/sql-database-managed-instance-connectivity-architecture# network-requirements

Route table: you can add entries to the route table to route traffic that has on-premises private IP ranges as a destination through the virtual network gateway or virtual network appliance (NVA).

NSG: Can remove NSG to confirm if it is NSG problem if possible.

DNS: when using custom DNS, make sure it resolves correctly(we can use dnslookup and see if customer can see the same thing as us). And add the azure recursive IP.

Service endpoint: Cannot use service endpoint for MI

a. VNET peering

Note it does not work in different regions as global peering is not supported in MI. VM connection works but not MI because MI uses the basic load balancer which does not support global peering.

Peering can be checked in ASC. https://docs.microsoft.com/en-us/azure/sql-database/sql-database-managed-instance-connect-app# connect-an-application-inside-a-different-vnet

b. VNET-to-VNET VPN gateway

(Azure portal, PowerShell, Azure CLI)

c. Site to site

Check the local network gateway in azure includes all the on-premise IP ranges.- Can be found in ASC too!

In the MI route table, check if we have route to on-premise IP range to VPN gateway.

When the connections goes to on-premise, ensure that on-premise firewall does not block SQL MI connections(port 1433 and 11000-11999)

d. Point to site

Point to site is connecting from public IP so the route table does not have to route to vpn gateway. Route to internet works fine too.

When the connections goes to on-premise, ensure that on-premise firewall does not block SQL MI connections(port 1433 and 11000-11999)

e. Express route

You can find the VNET list which have connection to Express routes in ASC:

Go to Microsoft.Network/expressRouteCircuits

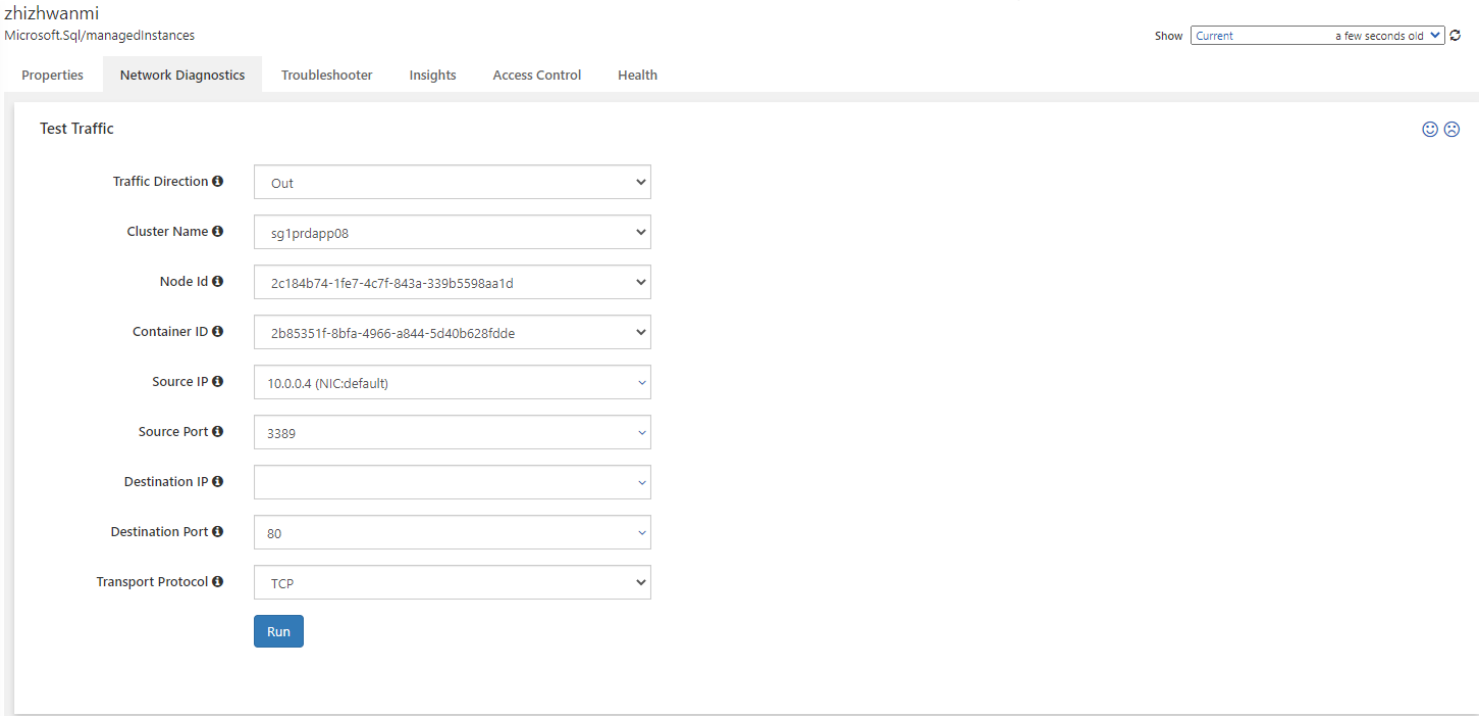
Under properties, go to Dump Circuite Info File uri, look for tunnel name

Or you can find tunnel name under VRF

When the connections goes to on-premise, ensure that on-premise firewall does not block SQL MI connections(port 1433 and 11000-11999)

Network Diagnostics in ASC

You can open ASC -> Resource Explorer(MI)-> Network Diagnostics -> Test Network to perform the network test. If there is network issues like route table, NSG, the test result will show up.



Step 3: Connectivity test using tools

Use [Azure SQL Connectivity Checker]

How good have you found this content?