The semaphore timeout period has expired

Last updated by | Vitor Pombeiro | Mar 11, 2022 at 9:04 AM PST

Contents

Error Message

> TITLE: Connect to Server
>
> Cannot connect to testr.database.windows.net.
>
> ADDITIONAL INFORMATION:
> A connection was successfully established with the server, but then an error occurred during the *pre-login handshake*. (provider: TCP Provider, error: 0 - The **semaphore timeout** period has expired.) (Microsoft SQL Server, Error: 121) For help, click: http://go.microsoft.com/fwlink?ProdName=Microsoft%20SQL%20Server&EvtSrc=MSSQLServer&EvtID=121&LinkId=20476

Analysis

This is client side issue. "**semaphore timeout**" is a 100% Windows kernel error that can occur for a very wide variety of reasons, typically due to a network card or driver-related issue. This appears as a SQL error because Windows passes this to the SQL process, so it is often mistaken to be a SQL error, when it's a client OS-level error.

This is not because of database memory resource pressure. At pre-login handshake stage, the request has not reached the database yet.

Mitigation

This is a client side issue, not an Azure Gateway-side or SQL-side issue. Check customer firewall settings. Most likely they have a firewall allowing TCP connections but preventing data to flow to destination.

Azure Service Endpoints and Subnets

For troubleshooting this error when it involves Service Endpoints and Subnets -- it is helpful to check the following:

- Is the Azure VM (or other Azure Resource that is trying to access SQL DB) on a different subnet than the SQL Database?

- Go to *Azure Portal* > *Virtual Networks* > *Select 'Service Endpoints'* > *Select the 'Microsoft.SQL' endpoint*
- Make sure that any other Subnets that will use the SQL Database are listed in here, if they are not listed click '+Add' to add the other subnets/services.
- NEXT:
  - Go to *Azure Portal* > *SQL DB* > *Firewall Configuration* > *Scroll down to Service Endpoint RULES*
    - Here you need to add the Rule for any SubNET that uses SQL DB
      - (Even the SubNET that the SQL DB itself is in!)
    - Click create Rule and add the Subnets that use SQL DB

On-premise client machines

If this error occurs for non-Azure, on-premise client machines, the issue might be caused by a local corporate network proxy/firewall or even a client machine firewall (e.g. Windows firewall). The customer needs to involve their network team and investigate locally with network traces and reviewing package flow.

Another possible cause is the TCP configuration of the Windows client machine. The customer may try to change one or all of the following settings on their client:

- Configure TCP Keep Alive settings: follow steps in Handling dropped connections using TCP Keepalive
- Disable TCP Chimney Offload: `netsh int ip set chimney DISABLED`
- Disable RSS: `netsh int tcp set global rss=disabled`
- Disable auto-tuning: `netsh int tcp set global autotuninglevel=disabled`

  (execute `netsh` commands from a command prompt window)

How good have you found this content?