Could not open a connection (slash) Connection attempt failed

Last updated by | Holger Linke | Aug 19, 2022 at 1:04 AM PDT

Contents

Issue

The Managed Instance has been setup and configured correctly. The admin is able to connect from SSMS to MI and has configured the Transactional Replication environment successfully. Network connectivity, NSGs, routing, firewll, DNS etc. are all configured as expected.

But when the replication agent jobs start the agents, they are unable to connect and failing with connection or login timeout errors.

Investigation/Analysis

The connection or login timeout issue can be caused by a Loopback Connection problem, which can occur when the MI's Worker.CL application runs on the same node as the gateway.

Mitigation

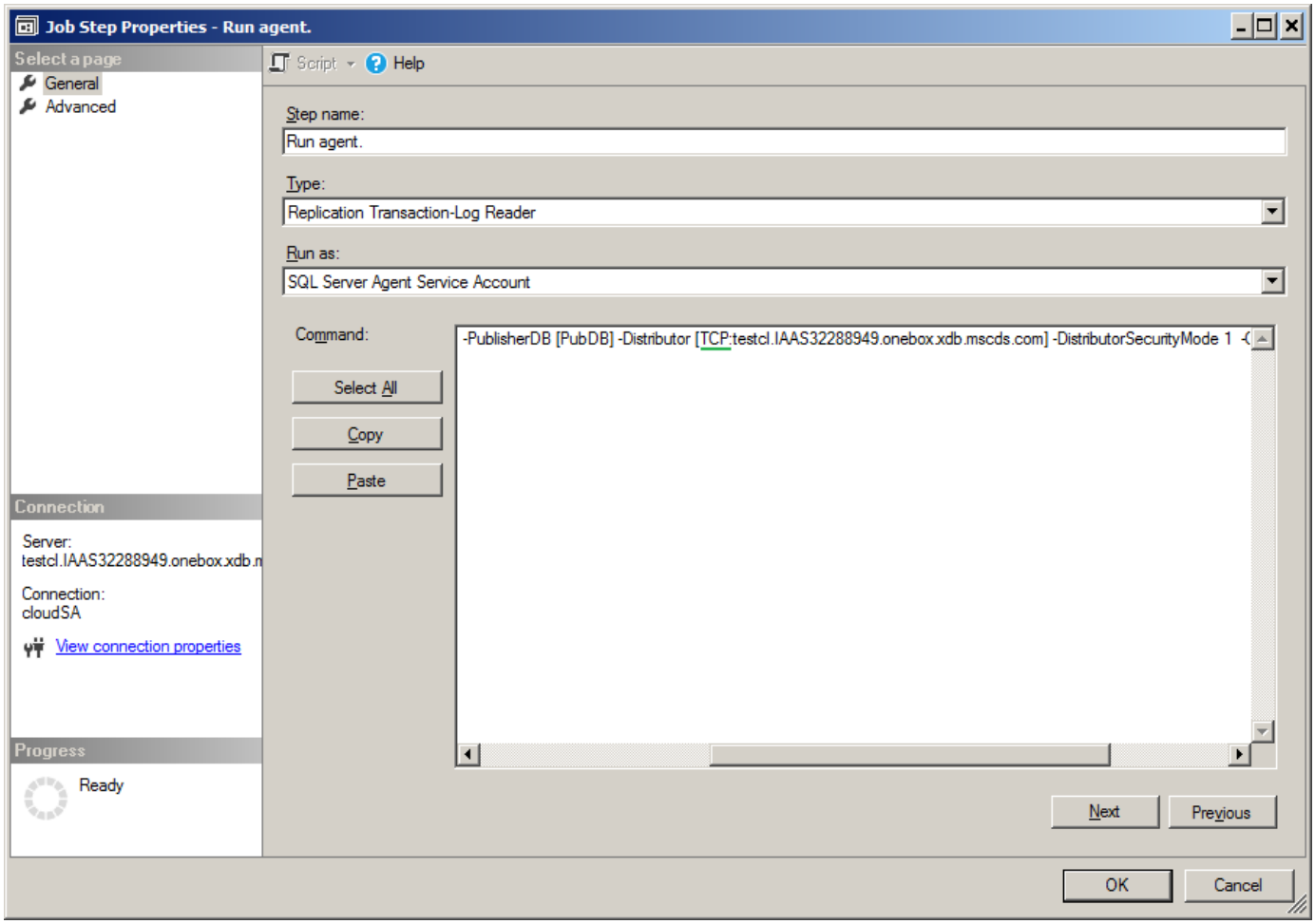
There are two possible mitigation options available - the first can be attempted by the admin user, the second requires an IcM.

Mitigation 1: Configure longer timeouts and specify TCP protocol

In order to mitigate the issue on user side, add `-LoginTimeout 150` to the end of the replication agents' command line. To do so, execute the following SQL statement on the Distributor:

```
-- Increase the default login timeouts for all replication jobs
update msdb..sysjobsteps
set command = command + N' -LoginTimeout 150'
where subsystem in ('Distribution','LogReader','Snapshot')
and command not like '%-LoginTimeout %'
```

In a second step, enforce Distributor connections to use the TCP protocol by adding the 'tcp:' prefix to the Distribution server name as shown below:



Or execute the following SQL statement on the Distributor:

```
-- Explicitly set the tcp protocol for the connections:
update msdb..sysjobsteps
set command = replace(command,' -Distributor [',' -Distributor [tcp:')
where subsystem in ('Distribution','LogReader','Snapshot')
and command not like '% -Distributor \[tcp:%' escape '\'
```

Mitigation 2: Open IcM to fix the issue at the tenant ring level

The steps for the PG would be:

- expand tenant ring,
- run GW application on dedicated nodes where Worker.CL should not run
- then failover GW to run aside from Worker.CL application.

See TSGs for details:

- TSGCL0087 Managed Server And Gateway On Same Node
- SOPCL00119: Set minimum number of nodes for PDC
- CRGW00032 - Kill Gateway

e.g. Locate MI and TR details in XTS, identify id Worker.CL application runs on the same node, allocated for one of GW applications and if so, run following script in DS console, spawned from XTS, to failover conflicting GW

application:

```
Select-SqlAzureEnvironment Prod
Select-SqlAzureCluster wasd-prod-indiasouth1-a-tr40
Get-FabricNode -NodeName DB.1 -NodeClusterName tr40.indiasouth1-a.worker.database.windows.net  | Kill-Process
```

How good have you found this content?