# Investigating GT connectivity issues

Last updated by | Charlene Wang | Oct 12, 2020 at 7:46 PM PDT

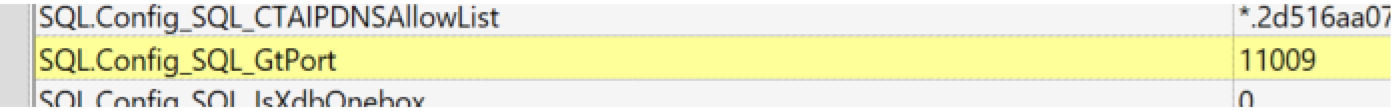
Contents

- Requirements
- Investigation/Analysis
- Mitigations
  - PG's Action
  - Customer's Action
- Root Cause Classification

## Requirements

Global Transactions rely on UCS connections for traffic related to transaction management.

1. In scenarios where GT is executed within single instance (multiple databases) there are no special networking requirements, all communication remains within the instance.
2. In scenarios where multiple instances participate, but those instances are part of the same Subnet, there are no additional requirements on a Subnet level (requirements from customer), however, instances are expected to use custom port for GT traffic from range 11k-12k. This is done via application parameter override on Worker.CL application level, so each instance is expected to have following override:

   | | |
   |---|---|
   | SQL.Config_SQL_CTAIPDNSAllowList | *.2d516aa07 |
   | SQL.Config_SQL_GtPort | 11009 |
   | SQL.Config_SQL_IsXdbOnebox | 0 |

3. In scenarios where 2 or more instances from different Subnets participate in GT following requirements apply:
   1. **Port 5024 needs to be open an all Subnets for all instances participating**. This port is used to initiate UCS connection for GT through Gateway.PDC (per-ring Gateway process for managing all TDS and UCS traffic).
   2. **Port range 11k-12k need to be open on all Subnets for all instances participating**. This range is used to allocate port for a single instance and use this port afterwards the UCS connection is established through Gateway.
4. **Global Transactions using MI public endpoint are not supported.**

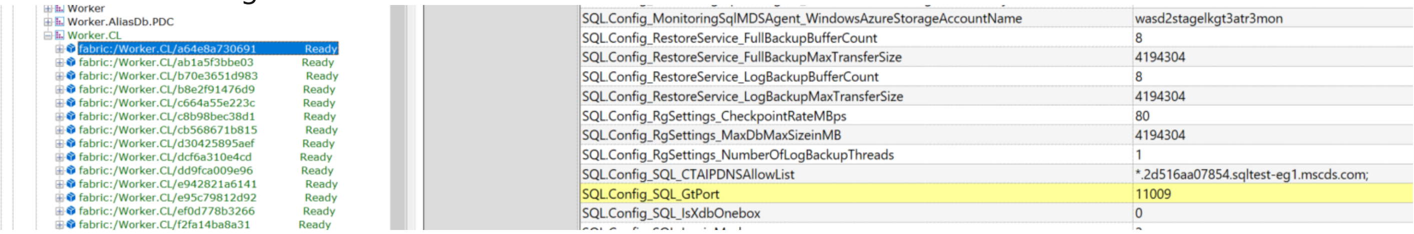
To summarize requirements in this table:

| Requirement\scenario | Cross-DB GT (single instance) | Cross-instance GT (single Subnet) | Cross-instance GT (multiple Subnets) | GT using public MI DNS |
|---|---|---|---|---|
| SQL.Config_SQL_GtPort | No | Yes | Yes | N/A |
| Port 5024 | No | No | Yes | N/A |
| Range 11k-12k | No | No | Yes | N/A |

## Investigation/Analysis

To validate instance connectivity is properly configured for GT in cross-instance scenarios, use steps below:

1. In Service Fabric Explorer look for application parameter override "SQL.Config_SQL_GtPort" and validate it has value from range 11k-12k.



```
Worker
Worker.AliasDb.PDC
Worker.CL
  fabric:/Worker.CL/a64e8a730691    Ready
  fabric:/Worker.CL/ab1a5f3bbe03    Ready
  fabric:/Worker.CL/b70e3651d983    Ready
  fabric:/Worker.CL/b8e2f91476d9    Ready
  fabric:/Worker.CL/c664a55e223c    Ready
  fabric:/Worker.CL/c8b98bec38d1    Ready
  fabric:/Worker.CL/cb568671b815    Ready
  fabric:/Worker.CL/d30425895aef    Ready
  fabric:/Worker.CL/dcf6a310e4cd    Ready
  fabric:/Worker.CL/dd9fca009e96    Ready
  fabric:/Worker.CL/e942821a6141    Ready
  fabric:/Worker.CL/e95c79812d92    Ready
  fabric:/Worker.CL/ef0d778b3266    Ready
  fabric:/Worker.CL/f2fa14ba8a31    Ready
```

| | |
|---|---|
| SQL.Config_MonitoringSqlMDSAgent_WindowsAzureStorageAccountName | wasd2stagelkgt3atr3mon |
| SQL.Config_RestoreService_FullBackupBufferCount | 8 |
| SQL.Config_RestoreService_FullBackupMaxTransferSize | 4194304 |
| SQL.Config_RestoreService_LogBackupBufferCount | 8 |
| SQL.Config_RestoreService_LogBackupMaxTransferSize | 4194304 |
| SQL.Config_RgSettings_CheckpointRateMBps | 80 |
| SQL.Config_RgSettings_MaxDbMaxSizeinMB | 4194304 |
| SQL.Config_RgSettings_NumberOfLogBackupThreads | 1 |
| SQL.Config_SQL_CTAIPDNSAllowList | *.2d516aa07854.sqltest-eg1.mscds.com; |
| SQL.Config_SQL_GtPort | 11009 |
| SQL.Config_SQL_IsXdbOnebox | 0 |

2. For cross-instance scenarios, test GT UCS connectivity on port 5024 using CAS actions below (do this for all combinations of instances participating in a global transaction):

```
Test-TcpConnectionFromNodeAgent -FabricClusterName tr3.lkgtst3-a.worker.sqltest-eg1.mscds.com -NodeName
```

   In this example, first instance is located on tr3.lkgtst3... and node DB8C.0 and it tries to contact second instance by DNS on port 5024.

3. To test connectivity on port from range 11k-12k, first look for exact port value in SQL.Config_SQL_GtPort of the target instance. From cluster and node of the source instance try reaching node of remote instance. For this you'll need DNS of remote instance node. This can be found in CMS: select * from azure_dns_records where dns_name like '%tr2368%'

   Use following CAS to test connectivity: Test-TcpConnectionFromNodeAgent -FabricClusterName tr2365.southeastasia1-a.worker.database.windows.net -NodeName "DB8C.0" -TargetHostName DB8C0.tr2368.southeastasia1-a.worker.database.windows.net -TargetPort 11004 -NumberOfRounds 2

4. Following Kusto table contains more information on UCS connections established from the Managed Instance: MonUcsConnections

## Mitigations

### PG's Action

In case instance port for **GT SQL.Config_SQL_GtPort** was not overrode, it can be configured manually by raising event "BackfillCommunicationLinkPort" or ""BackfillGlobalTransactionsPort" on ManagedServerInstanceStateMachine. SQL processes requires restart to pick up new value, so make sure failover is done afterwards.

Please raise an ICM to product group so they can help restart SQL process. CSS should check with customer for the proper mitigation time since this will cause downtime and failover.

### Customer's Action

For other mitigation related to ports 5024, 11k-12k on a Subnet level, customer must open these port through their NSGs, nothing can be done on service side.

## Root Cause Classification

**How good have you found this content?**

🙂 🙁