

Connection Failures due to Gateway Migration, Upgrade

Last updated by | Charlene Wang | May 10, 2021 at 8:19 PM PDT

Contents

- [Issue](#)
- [Troubleshooting:](#)
- [Mitigation](#)

Issue

Customer unable to connect to database as a result of gateway hardware migration in the region either intermittently or consistently. If firewall rules on-premises that depend on IP address of a specific Gateway Or using a custom DNS server that resolves to a specific Gateway, customer may experience this error message when a connection is attempted.

"A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not accessible. Verify that the instance name is correct and that SQL Server is configured to allow remote connections."

Troubleshooting:

If you are working on an incident where the customer's logins were succeeding before the migration date and have not been working post migration, please follow the steps below to ensure that the reason is due to IP address of the new CR to which the server got migrated is not present in the customer's allow-list firewall rules.

1. Do an nslookup for the customer's server to see where it is landing

```
a. In this example, I am using the server testserver
nslookup testserver.database.windows.net
Server: UnKnown
Address: 2001:4898::1050:1050
```

Non-authoritative answer:

```
Name: cr3.japaneast1-a.control.database.windows.net
Address: 40.79.184.8
Aliases: testserver.database.windows.net
         dataslice5.japaneast.database.windows.net
         dataslice5japaneast.trafficmanager.net
```

For JapanEast, the servers were migrated to dataslice5.

You can check the table - Data slices for Migration - to see what is the new dataslice for the region

2. Use the following MonLogin query to see if the connections to the old CR were successful or not.

```
Execute: [Web] [Desktop] [Web (Lens)] [Desktop (SAW)] https://sqlazureja2.kustomfa.windows.net/sqlazur
let svr = '<replace with server name from ICM>';
let incidentReportedTime = datetime('<replace with incident reported time>');
MonLogin
| where originalEventTimestamp between ((incidentReportedTime - 2d) .. (incidentReportedTime))
| where logical_server_name =~ svr
| where event == 'process_login_finish'
|   where package =~ 'xdbgateway' and AppName =~ 'Gateway'
|   where is_success == 1
| summarize count(), min(originalEventTimestamp), max(originalEventTimestamp) by ClusterName
```

****Example.****

```
Execute: [Web] [Desktop] [Web (Lens)] [Desktop (SAW)] https://sqlazureja2.kustomfa.windows.net/sqlazur
Execute: [Web] [Desktop] [Web (Lens)] [Desktop (SAW)] https://sqlazureja2.kustomfa.windows.net/sqlazur
let svr = 'testserver';
MonLogin
| where logical_server_name =~ svr
| where event == 'process_login_finish'
| where package =~ 'xdbgateway' and AppName =~ 'Gateway'
| where is_success == 1
| summarize count(), min(originalEventTimestamp), max(originalEventTimestamp) by ClusterName
```

****Result****

ClusterName	count_	min_originalEventTimestamp	max_originalEventTimestamp
cr2.japaneast1-a.control.database.windows.net	1	2020-09-08 18:30:53.8702356	2020-09-08 18:

3. Run the same MonLogin query to see if there is any telemetry for the past 1hr. If you see any incoming tele

```
Execute: [Web] [Desktop] [Web (Lens)] [Desktop (SAW)] https://sqlazureja2.kustomfa.windows.net/sqlazur
let svr = '<replace with server name from ICM>';
MonLogin
| where originalEventTimestamp > ago(1h) // you might have to change the duration based on the Custome
| where logical_server_name =~ svr
| where event == 'process_login_finish'
| where package =~ 'xdbgateway' and AppName =~ 'Gateway'
| summarize count(), min(originalEventTimestamp), max(originalEventTimestamp) by ClusterName
```

4. Run sqlcmd to see if the database is up and healthy and is able to accept connections

```
PS E:\Azure_SQL_DB\Connectivity_Infrastructure\Migration> sqlcmd -S testserver.database.windows.net -d
Sqlcmd: Error: Microsoft SQL Server Native Client 11.0 : Cannot open server 'testserver' requested by
```

On running sqlcmd, if you see either this firewall error or login failed for user 'test', it implies

5. Check HPDB is see if the server was recently migrated or not.

```
a. Connect to healthproperties server (or vzkjq52y5r.database.windows.net), database Gen4MigrationDB
b. Then you can query the MigrationTable to check if migration completed.
```

```
SELECT *
FROM [dbo].[MigrationTable]
where LogicalServerName in ('<server name from incident>')
```

6. If you see that the error 40532 -> it implies that the customer is connecting to the incorrect CR. So they might have to update their custom setup to use the new CR.



Mitigation

If you are seeing extremely high refresh rates, you might have to update the change tracking retention value for the SQL Alias Databases. Instructions can be found here: [How to modify the change tracking for Sql Alias Database](#)

1. Ask the customer to whitelist the IP addresses present here for the region they are concerned with - https://supportability.visualstudio.com/AzureSQLDB/_wiki/wikis/AzureSQLDB.wiki/399388/Connection-Failures-due-to-Gateway-Migration-Upgrade

****Customer ready content**** We recommend that you allow outbound traffic to IP addresses for all the A

2. If the customer says that they need extra time to whitelist the IP address and they would like to be revert
 - a. To do so, you will have to run the following commands :
Go to the "Gateway Slices.xts" view in XTS
 - b. In the "Target Gateway Slices" tab, you will need to select the dns_name and gateway_slice_id to wh

Once you have correct dns_name and gateway_slice_id, the "Move Server" tab will be populated w
You will have to select the correct environment and azure cluster before you run the commands

Example :

```
Select-SqlAzureEnvironment Prod
```

```
Select-SqlAzureCluster wasd-prod-japaneast1-a-cr1
```

- c. You will have to execute the command corresponding to the Start Migration action

Example :

```
Start-MigrateServerToSlice -ServerName testserver -TargetGatewaySliceId 4059489b-ff5c-4e41-93f
```

- d. Wait for about 10 mins for the DNS propagation to complete. You can additionally perform the follow

```
- Use the following query in CMS to see if the server is in WaitingBeforeContinueMigration sta
SET QUOTED_IDENTIFIER OFF;
SELECT distinct sb.server_name as serverName, sb.state as state, cr.proxy_override_val
FROM dbo.logical_servers as ls
INNER JOIN dbo.server_slice_bindings as sb
ON ls.name = sb.server_name
INNER JOIN dbo.sql_alias_cache_records as cr on ls.name = cr.server_name
INNER JOIN dbo.gateway_slices as gs
ON sb.gateway_slice_id = gs.gateway_slice_id
WHERE gs.gateway_slice_id IN ("<target_gateway_slice_id>") and ls.name = '<server_name
order by state desc
```

Example:

```
SET QUOTED_IDENTIFIER OFF;
SELECT distinct sb.server_name as serverName, sb.state as state, cr.proxy_override_val
FROM dbo.logical_servers as ls
INNER JOIN dbo.server_slice_bindings as sb
ON ls.name = sb.server_name
INNER JOIN dbo.sql_alias_cache_records as cr on ls.name = cr.server_name
INNER JOIN dbo.gateway_slices as gs
ON sb.gateway_slice_id = gs.gateway_slice_id
WHERE gs.gateway_slice_id IN ("585c8a95-fdff-4391-ba7d-ee4e2ce470c2") and ls.name = 't
order by state desc
```

You should that the server's state is : WaitingBeforeContinueMigration

- e. Then execute the command corresponding to the Complete Migration action

Example

```
Complete-MigrateServerToSlice -ServerName testserver
```

- f. You can perform the following steps for verification

- Query CMS to see if the server state is now "Ready"

```
SET QUOTED_IDENTIFIER OFF;
SELECT distinct sb.server_name as serverName, sb.state as state, cr.proxy_override_val
FROM dbo.logical_servers as ls
INNER JOIN dbo.server_slice_bindings as sb
ON ls.name = sb.server_name
INNER JOIN dbo.sql_alias_cache_records as cr on ls.name = cr.server_name
INNER JOIN dbo.gateway_slices as gs
ON sb.gateway_slice_id = gs.gateway_slice_id
WHERE gs.gateway_slice_id IN ("<target_gateway_slice_id>") and ls.name = '<server_name
order by state desc
```

Example:

```
SET QUOTED_IDENTIFIER OFF;  
SELECT distinct sb.server_name as serverName, sb.state as state, cr.proxy_override_val  
FROM dbo.logical_servers as ls  
INNER JOIN dbo.server_slice_bindings as sb  
ON ls.name = sb.server_name  
INNER JOIN dbo.sql_alias_cache_records as cr on ls.name = cr.server_name  
INNER JOIN dbo.gateway_slices as gs  
ON sb.gateway_slice_id = gs.gateway_slice_id  
WHERE gs.gateway_slice_id IN ("585c8a95-fdff-4391-ba7d-ee4e2ce470c2") and ls.name = 't'  
order by state desc
```

You should that the server's state is : Ready

- Do an nslookup on the server to see if the server resolve to the intended dns_name

g. Additionally, you can query MonLogin for the servers that you wish to migrate to see the traffic mo

Tags for Searchability: migration, servermigration, start-migrateservertoslice, complete-migratetoserverslice, WaitingBeforeContinueMigration, logins, failing, login failed

How good have you found this content?

