# ERROR: Too many connections or hit maximum connection limit

Last updated by | Ulisses Alves | Mar 2, 2023 at 4:44 PM PST



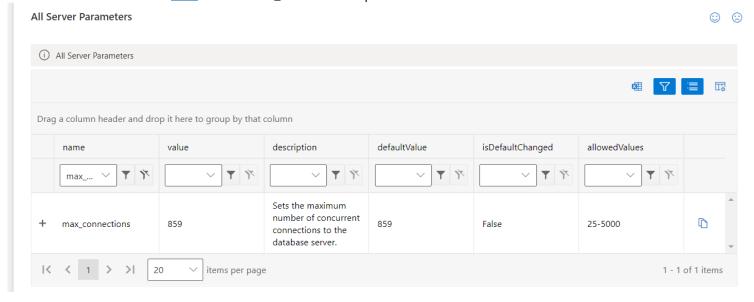
#### Issue

Customer can hit or reach maximum connections limit or too many connections error.

Maximum connections limit is determine by max\_connections parameter, which is determine by server's <a href="SKU">SKU</a> <a href="SKU">

## **Analysis**

Please check customer's <u>SKU</u> ☑ and max\_connections parameter in ASC.



#### **Kusto**

To get errors related to too many connections error, please check monpglogs:

```
MonPgLogs
 where LogicalServerName == 'SERVER_NAME'
 where TIMESTAMP >= datetime(START_TIME) and TIMESTAMP <= datetime(END_TIME)
| where message_id contains "too many"
| project TIMESTAMP, message_id
```

You will get message similar to below:

,	
TIMESTAMP	message_id
2022-10-24 15:11:00.0000000	sorry, too many clients already
2022-10-24 15:11:00.0000000	sorry, too many clients already
2022-10-24 15:11:00.0000000	sorry, too many clients already
2022-10-24 15:11:10.0000000	sorry, too many clients already
2022-10-24 15:11:20.0000000	sorry, too many clients already
2022-10-24 15:11:40.0000000	sorry, too many clients already
2022-10-24 15:11:50.0000000	sorry, too many clients already

You can get all connections to server from kusto using:

```
MonOBPgSqlSessionsStatus
 where LogicalServerName == 'SERVER_NAME'
 where TIMESTAMP >= datetime(START_TIME) and TIMESTAMP <= datetime(END_TIME)
 extend totalnumberofconnections= toint(Idle_cnt) + toint(Active_cnt)
 project TIMESTAMP, LogicalServerName, totalnumberofconnections, Idle_cnt, Active_cnt, Admin_active_cnt, Admi
```

Output will contain below columns:

```
Active_cnt: include both Admin and user count
Idle cnt: include both Admin and user count.
Total number of connections = Active_cnt+Idle_cnt
```

To get more details about connections, please use below table, which will show you, database and application name:

```
MonOBPgSqlSessions
 where LogicalServerName == 'SERVER NAME'
 where TIMESTAMP >= datetime(START TIME) and TIMESTAMP <= datetime(END TIME)
project TIMESTAMP, Database name, Application name, Session duration, State, Query start
```

From above table, you cannot get IP address, if customer wants to know which IP is causing this, he needs to enable <u>logs</u> 🗷

## customer side logs

If customer enabled <u>logs</u> \( \text{logs} \) he can use this query in logs blade to find number of connections per IP:

```
AzureDiagnostics
 where Resource == toupper('SERVER NAME')
 project TimeGenerated, Message
 where Message contains "connection received: host=" and Message !contains "host=127.0" and Message !contains
 extend IPADDRESS=substring(Message,indexof(Message,"host="), indexof(Message,'port=')-indexof(Message,"host=
 project TimeGenerated, Message, IPADDRESS
 summarize count() by IPADDRESS, bin(TimeGenerated,1m)
```

and this query to get too many connection error

```
AzureDiagnostics
  where Resource == toupper('merupg12')
  where LogicalServerName == 'SERVER NAME'
  where TIMESTAMP >= datetime(START TIME) and TIMESTAMP <= datetime(END TIME)
  order by TimeGenerated asc
  project TimeGenerated, Message
  where Message contains "too many"
```

### Limit of concurrent connections into the database (Single Server / Flexible Server)

If cx is receiving the error "Too many connections" but the server is not exceeding the max number of connections, this can be because cx set a limit on the number of concurrent conections for a userid in the database. For example if they set that the user can have only 1 connection, they log into the database in PgAdmin and try to use the same userid connecting into the same database in PSQL they will get 'too many connections' error. To check it, cx will need to execute the following query in the database that they are having the issue:

```
SELECT rolname, rolconnlimit FROM pg roles;
```

The column rolconnlimit will show how many concurrent connections are allowed for the rolname to connect to this database (-1 is unlimited).

In case you need to adjust it, you can use the following command in the database:

```
alter user <userid> with connection limit -1;
```

# Solving issue

Issue can be resolved using one of the following options:

- 1. Use connection pooler or build-in pgbouncer \(\mathbb{Z}\) that is supported in flexile server.
- 2. Increase SKU to handle more connections.