

# Monitor short live connections

Last updated by | Hamza Aqel | Sep 14, 2022 at 7:21 AM PDT

Some customers are willing to see if they can monitor their short live connections in their PostgreSQL Flexible servers and set some notifications or alerts to achieve that, you do that through diagnostic settings as in the below example:

Prerequisites:

Make sure to enable or set these before moving forward:

Enable Diagnostic settings:

 Refresh  Feedback

Diagnostic settings are used to configure streaming export of platform logs and metrics for a resource to the destination of your choice. You may create up to five different diagnostic settings to send different logs to independent destinations. [Learn more about diagnostic settings](#)

Diagnostic settings





Name	Storage account	Event hub	Log Analytics workspace	Partner solution	Edit setting
No diagnostic settings defined					


[+ Add diagnostic setting](#)

Click 'Add Diagnostic setting' above to configure the collection of the following data:

- PostgreSQL Server Logs
- AllMetrics

make sure that parameter log\_disconnections is set to ON (default value):

 Save  Discard  Reset all to default  Feedback

This list shows modifiable server parameters. Please click on the info box to get more details about a particular parameter, like the allowed values and data type. [Learn more](#) 

 log\_disconnections

Parameter name	↑↓	VALUE		Parameter type	↑↓	Description
log_disconnections		ON OFF		Read Only		Logs end of a session, including duration.

Run this query under Monitoring – Logs (Log analytics):

AzureDiagnostics

```
| where ResourceProvider == "MICROSOFT.DBFORPOSTGRESQL"
| where Category == "PostgreSQLLogs"
| where TimeGenerated >= ago(2d)
| where Message contains "disconnection: session time"
| extend pgmessage = tostring(split(Message, "disconnection: session time: ")[-1])
| extend myuser = tostring(split(tostring(split(pgmessage, " database=")[-2]), " user=")[-1])
| extend hours = todecimal(substring(pgmessage, 0, 1))
| extend minutes = todecimal(substring(pgmessage, 2, 2))
| extend seconds = todecimal(substring(pgmessage, 5, 2))
| extend milliseconds = todecimal(substring(pgmessage, 7, 4))
```

```

| extend connection_life_seconds = hours*60*60+minutes*60+seconds+milliseconds

| where myuser != 'azuresu'

| extend connection_type = case(connection_life_seconds < 60 , strcat("Short Live Connection"),
connection_life_seconds between (60 .. 1200) , strcat("Normal Live
Connection"),connection_life_seconds >1200, strcat("Long Live Connections"), "")

| summarize max(connection_life_seconds) by TimeGenerated,connection_type

| render timechart

```

- This is based on our recommendation and experience:
- short live .. if connection\_life\_seconds is less than 60s.
  - normal if 60s<connection\_life\_seconds < 1200s
  - long if it is > 1200s

