

# Microsoft SQL Server Integration

The Microsoft SQL Server integration package allows you to search, observe, and visualize the SQL Server audit logs, as well as performance and transaction log metrics.

---

## Requirements

Microsoft SQL Server is installed and has connectivity with the CyTech Log Collector.

*Note.* For more information regarding Microsoft SQL Server Installation, click the link (<https://learn.microsoft.com/en-us/sql/database-engine/install-windows/install-sql-server?view=sql-server-ver16>) for more info.

---

## Microsoft SQL Server permissions

Before you can start sending data to the Log Collector, make sure you have the necessary Microsoft SQL Server permissions.

If you browse Microsoft Developer Network (MSDN) for the following tables, you will find a "Permissions" section that defines the permission needed for each table.

- `transaction_log`:
  - `sys.databases`
  - `sys.dm_db_log_space_usage`
  - `sys.dm_db_log_stats (DB_ID)` (Available on SQL Server (MSSQL) 2016 (13.x) SP 2 and later)
- `performance`:
  - `sys.dm_os_performance_counters`

Please make sure the user has the permissions to system as well as user-defined databases. For the particular user used in the integration, the following requirements are met:

### User setup options:

- Grant specific permissions as mentioned in the MSDN pages above.
- Alternatively, use `sysadmin` role (includes all required permissions): This can be configured via SQL Server Management Studio (SSMS) in `Server Roles`. Read more about joining a role in the SQL Server documentation.

### User Mappings (using SQL Server Management Studio (SSMS)):

- Open SSMS and connect to your server.

- Navigate to "Object Explorer" > "Security" > "Logins".
  - Right-click the user and select "Properties".
  - In the "User Mapping" tab, select the appropriate database and grant the required permissions.
- 

## Setup

Below you'll find more specific details on setting up the Microsoft SQL Server integration.

### Named Instance

Microsoft SQL Server has a feature that allows running multiple databases on the same host (or clustered hosts) with separate settings. Establish a named instance connection by using the instance name along with the hostname (e.g. `host/instance_name` or `host:named_instance_port`) to collect metrics. Details of the host configuration are provided below.

### Host Configuration

As part of the input configuration, you need to provide the user name, password and host details. The host configuration supports both named instances or default (no-name) instances, using the syntax below.

*Note: This integration supports collecting metrics from a single host. For multi-host metrics, each host can be run as a new integration.*

#### Connecting to Default Instance (host):

- `host` (e.g. `localhost` (Instance name is not needed when connecting to default instance))

Note. IP Address of the SQL Server will be needed for the integration

- `host:port` (e.g. `localhost:1433`)

Note. Default port is 1433

#### Connecting to Named Instance (host):

- `host/instance_name` (e.g. `localhost/namedinstance_01`)
- `host:named_instance_port` (e.g. `localhost:60873`)

*If you need further assistance, kindly contact our support at [support@cytechint.com](mailto:support@cytechint.com) for prompt assistance and guidance.*

---