

# Migration from PostgreSQL database to MS SQL Server



## Document Overview

M365 Manager Plus comes bundled with a PostgreSQL database to store the product's data from its reporting, monitoring, management, and auditing modules. The solution also supports MS SQL Server and enables you to migrate product data from the built-in PostgreSQL to an MS SQL database. This guide will walk you through the database migration process.

**Supported versions of MS SQL Server:** 2008 R2, 2012, 2014, 2016, 2017

## Prerequisites for the migration process

Make sure that the following conditions are satisfied in the MS SQL Server to which you want to migrate the data:

1. The SQL Server browser must be up and running.
2. For SQL Server network configuration, TCP/IP protocol must be enabled.
3. All the client protocols must be enabled.
4. MS SQL Server access is delegated to a user with **sysadmin** and **db\_owner** permissions at the server and database levels respectively.

Refer to [Appendix A](#) for configuring prerequisites 1, 2, and 3. Refer to [Appendix B](#) for configuring prerequisite 4.

## Migrating from PostgreSQL to MS SQL

Migrating data from M365 Manager Plus' default PostgreSQL database to MS SQL consists of the following two steps:

1. [Backing up the M365 Manager Plus database](#)
2. [Migrating PostgreSQL data to MS SQL](#)

### Step 1: Backing up the M365 Manager Plus database

If you're migrating the database of a fresh installation, then you can skip this step.

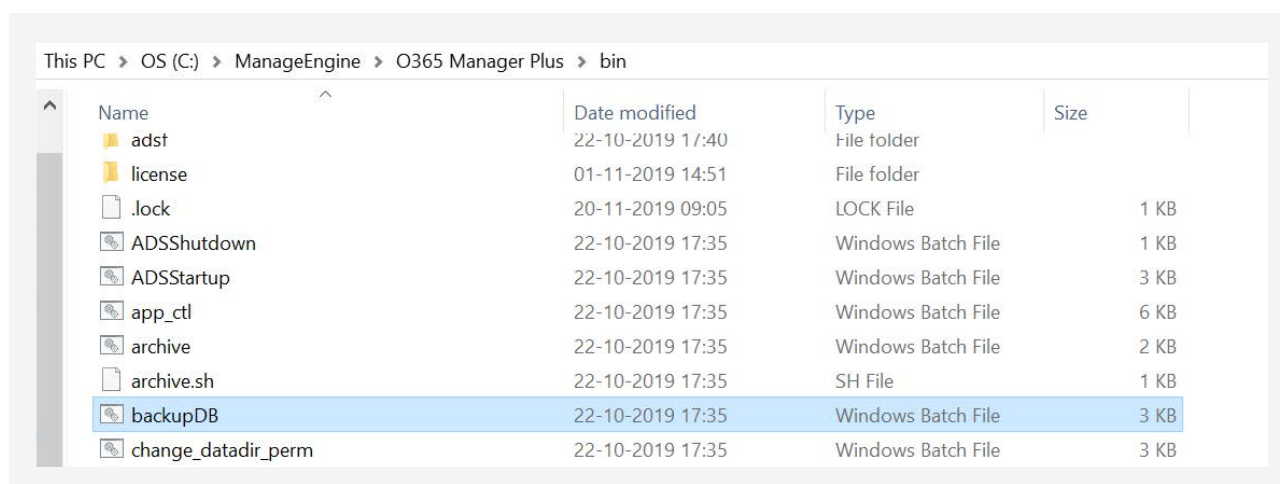
1. **Stop M365 Manager Plus** (it's essential to stop M365 Manager Plus before performing the backup).
  - a. If M365 Manager Plus is running in console mode, click the **Windows** icon. Search for **M365 Manager Plus**, and choose **Stop M365 Manager Plus**.

- b. If M365 Manager Plus is running as a service, click the **Windows** icon. Search for **Services**. In the window that appears, choose **ManageEngine M365 Manager Plus**, and select the **Stop the service** option on the left side.

2. Navigate to **<M365 Manager Plus installation directory>\bin**.

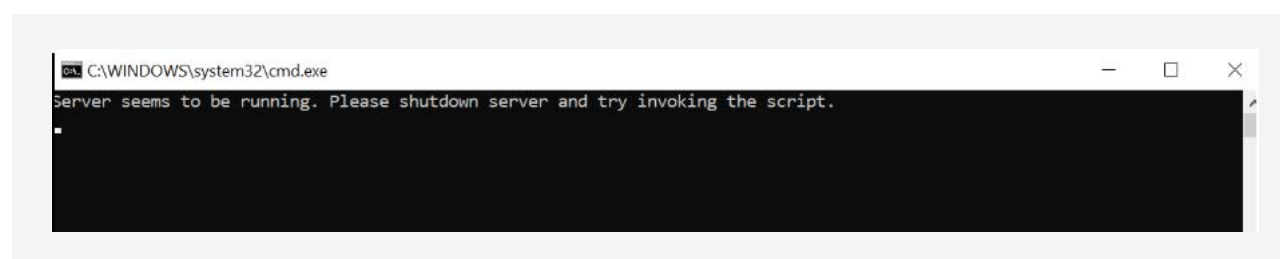
**Note:** By default, M365 Manager Plus gets installed in: C:\Program Files\ManageEngine\M365 Manager Plus

3. Run the **backupDB** (Windows batch) file as an administrator. Do not terminate until the process is finished.



4. Data in the default database of M365 Manager Plus will be backed up and stored under **<M365 Manager Plus installation directory>\backup\OfflineBackup\_<Backup Time>**.

**Note:** If the below error shows up while running the **backupDB** file, make sure that the product has been stopped before proceeding.



## Step 2: Migrating PostgreSQL Data to MS SQL

**Note:** Make sure that M365 Manager Plus is stopped before the migration process.

1. If MS SQL Server is installed in a **remote computer**, install the necessary command line utilities and native Client, and then proceed to the next step.

**Note:** Download the corresponding command line utilities and native clients from the links below. On clicking the **Download** button in the page displayed, you will see a listing of SQL Native Client or command line utilities. The command line utilities have the term **SQLCMD** in them, and the native client file can be found under the name **sqlncli**. Install the corresponding SQL Native Client or command line utilities as per the MS SQL Server version and CPU type of the machine in which M365 Manager Plus is installed.

SQL Server version	Command line utilities	Native client
2008 R2	<a href="#">Download</a>	<a href="#">Download</a>
2012	<a href="#">Download</a>	<a href="#">Download</a>
2014	<a href="#">Download</a>	<a href="#">Download</a>
2016	<a href="#">Download</a>	Not Needed
2017	<a href="#">Download</a>	Not Needed

2. Copy the following files to <M365 Manager Plus installation directory>\bin folder.

i. **bcp.exe**- <MSSQL installation directory>\Tools\Binn\bcp.exe

ii. **bcp.rll**- <MSSQL installation directory>\Tools\Binn\Resources\1033\bcp.rll

3. If your firewall is enabled in the Microsoft SQL Server machine, the TCP and UDP ports must be opened.

4. If you are migrating to a Microsoft SQL server that has **Force encryption** enabled, follow the steps mentioned below.

a. Convert your certificate to CER format.

i. Open **IIS Manager**.

ii. In the middle pane, click **Server Certificates**.

iii. Open the certificate you want to use, and click the **Details** tab.

iv. Click **Copy to file**.

v. Click **Next** in the *Certificate Export Wizard* that appears.

vi. On the *Export Private Key* screen, select **No, do not export the private key**, and click **Next**.

vii. On the *Export File Format* screen, select either **DER encoded binary X.509 (.CER)** or **Base-64 encoded X.509 (.CER)**, and click **Next**.

viii. Enter a **name** for the file, click **Next**, and then **Finish**.

or

b. Export SSL certificate from SQL Configuration Manager

i. Open **Microsoft Management Console**.

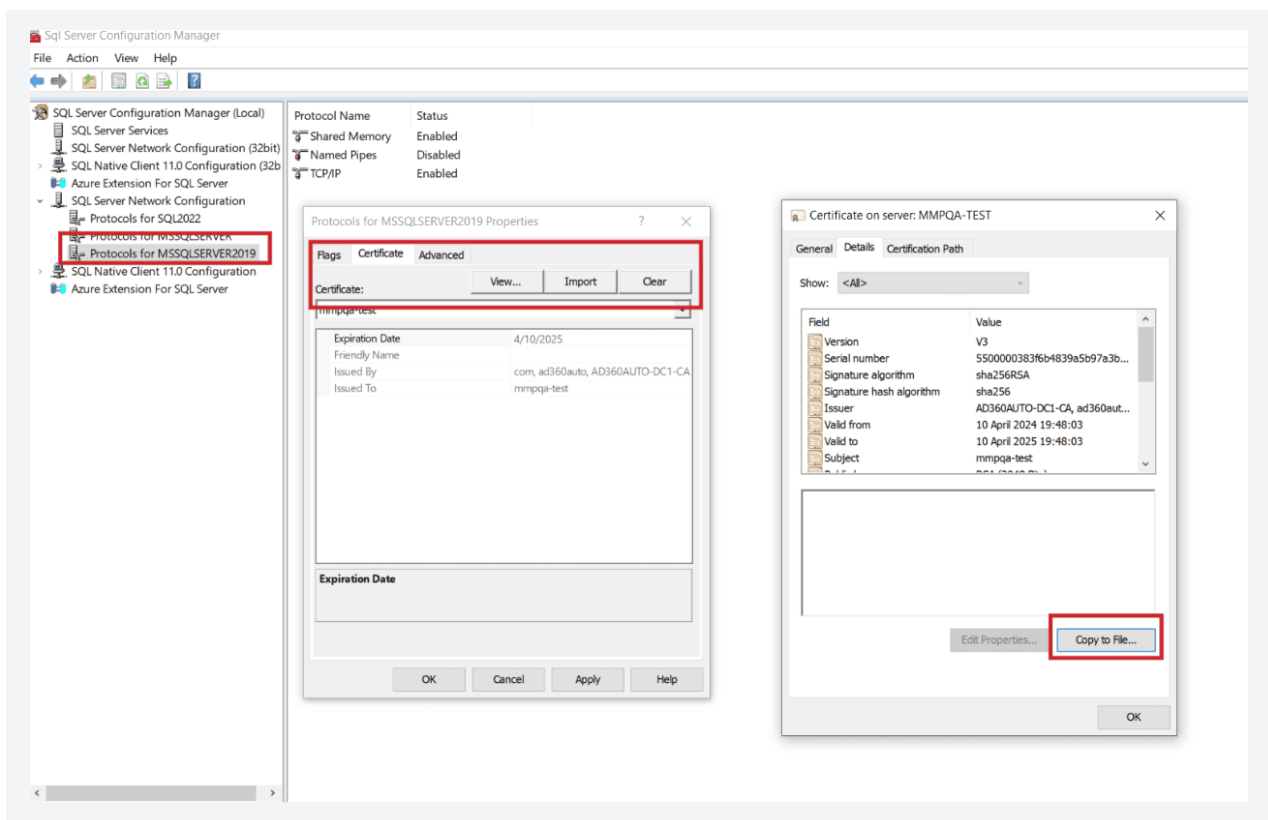
ii. Navigate to **SQL Server Configuration Manager > SQL Server Network Configuration**.

iii. Select the protocols for the server that you wish to export the SSL certificate from.

iv. Right click on the selected option and click **Properties**.

v. Click on the **Certificate** tab, select the certificate that you wish to export, and click **View**.

vi. Click **Copy to File** in the new window that opens and save the certificate in a folder of your choice.



5. Open **Command Prompt** and navigate to **<Installation directory>\jre\bin**. Use the command below to associate the certificate with the Java KeyStore.

```
keytool -import -v -trustcacerts -alias myserver -file pathofthecert\certname.cer -keystore  
"..\lib\security\cacerts" -keypass changeit -storepass changeit -noprompt
```

where **pathofthecert** is the location where the certificate has been stored and **certname** is the certificate name.

6. The certificate will be added to your Java KeyStore.

7. Navigate to **<M365 Manager Plus directory>\bin**.

8. Run the **ChangeDB** batch file as an administrator. In the *Database Setup Wizard* that appears, select **Microsoft SQL Server** as the **Server Type**.

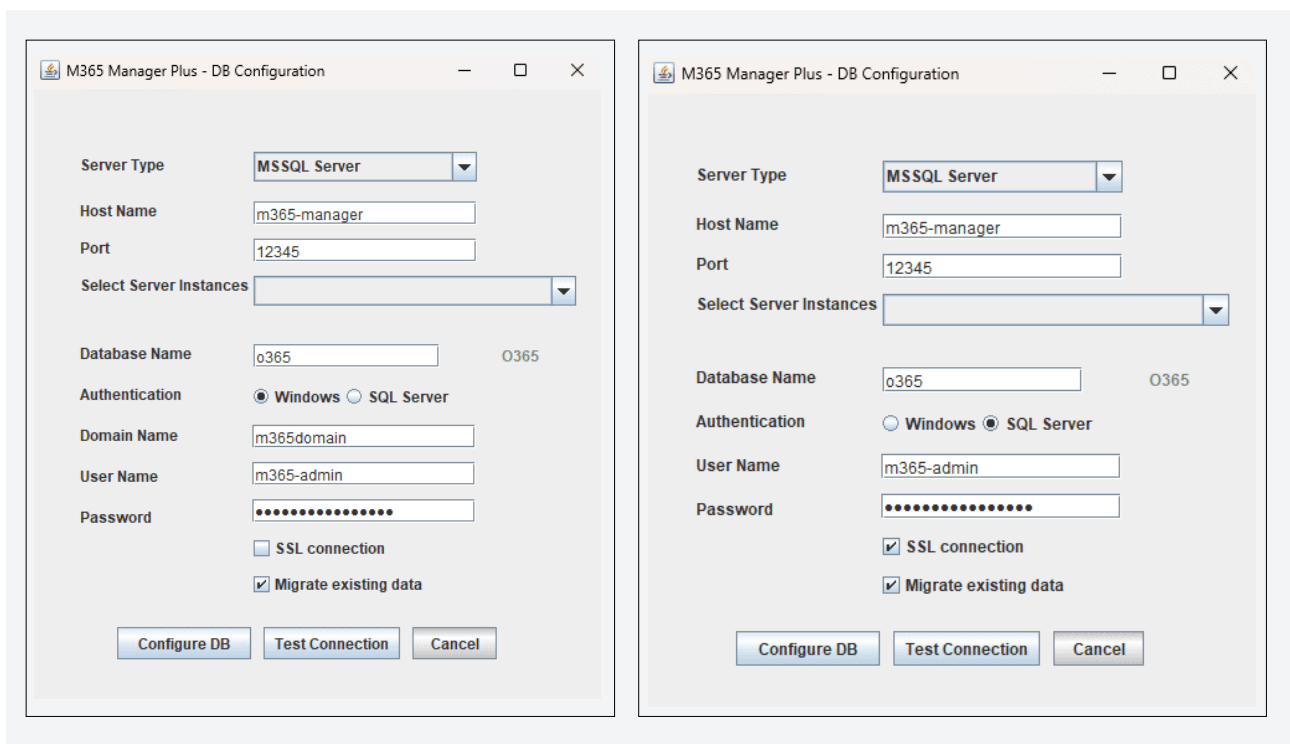
9. Enter the **Host Name** and **Port** number of Microsoft SQL Server.

10. Select the **SQL Server Instance** from the drop-down.

11. Select the **Authentication** type.

a. If you choose **Windows Authentication**, provide the **Domain Name**, **User Name**, and **Password** of the user account that has access to the server.

b. If you choose **SQL Server Authentication**, provide the **User Name** and **Password** of the user who has access to the MS SQL Server instance.



10. If the Microsoft SQL server you wish to migrate to has **Force encryption** on, check the **SSL connection** box.
11. If you want to migrate all the data stored in the default database and also keep a copy of it in the newly-migrated database, check the box next to **Migrate existing data**. If you do not want to migrate the data, you can leave it unchecked.

**Note:** By choosing either option, data will remain in the default PostgreSQL database even after the migration. However, the Microsoft SQL database will be active and used to store all the product data.

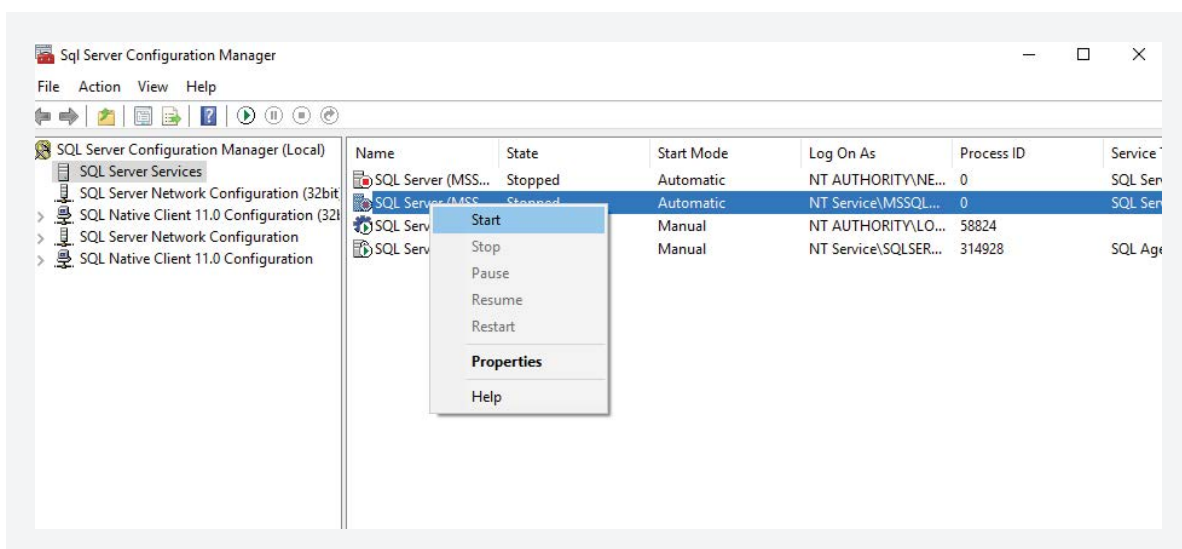
12. Click on **Test Connection**. If the test fails, then recheck your credentials and try again.
13. Click **Save** to finish the Microsoft SQL Server configuration.
14. Start the **M365 Manager Plus Server/Service** to work with Microsoft SQL Server as the database.

## Appendix A

### Configuring MS SQL Server

If you already have a functional MS SQL Server, then this step is not required. Follow the steps below to configure a new MS SQL Server installation.

1. Run **compmgmt.msc** in Command Prompt to open the SQL Server Configuration Manager.
2. Go to SQL Server Services, and ensure that SQL Server Browser is running.



3. In the left pane, click **SQL Server Network Configuration**, double-click **Protocols for SQLEXPRESS**.

**Note:** SQLEXPRESS is the instance name provided while configuring MS SQL Server. This name will be used for reference.

4. In the pop-up that appears, go to the **Protocol** tab, and enable **TCP/IP**. Restart the **SQL Server SQLEXPRESS Service**.
5. Go back to **SQL Server Configuration Manager**. In the left pane:
  - Navigate to **SQL Server Network Configuration > Protocols for SQLEXPRESS**, and enable all the **protocols**.
  - Navigate to **SQL Native Client Configuration > Client Protocols**, and enable all the **protocols**.

## Appendix B

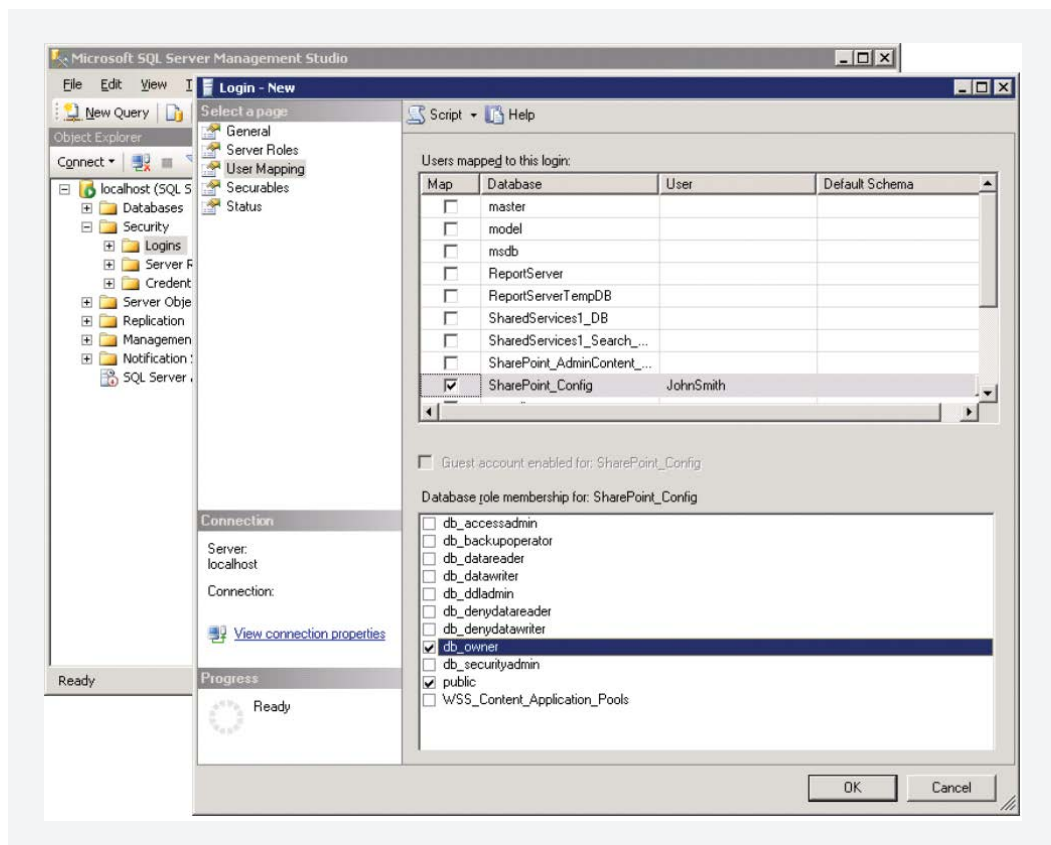
### Delegating Microsoft SQL Server access to users

To complete the migration process, you need a **database admin** account for the MS SQL connection. If M365 Manager Plus is already running as a Windows service, you can use its **service account** as the MS SQL database admin account. Check if the account has the required privileges. If there is no previously configured service account in MS SQL Server, follow the steps given below to create a new user account, and assign required permissions to it.

1. Log in to **SQL Server Management Studio**.
2. In the left pane, navigate to **Machine Name > SQLEXPRESS > Security > Logins**.
3. Right-click on **Logins**, and select **New Login**.
4. Provide a **Login Name**, and choose whether to use **Windows Authentication** or **SQL Server Authentication**.
  - If you choose **Windows Authentication**, enter the **Windows NT name** of the user to whom access must be granted.
  - If you choose **SQL Server Authentication**, you will be prompted to create a new **Username** and **Password**.
5. The new user must have the **sysadmin** role in the **server level** and **db\_owner** role in the **database level**. Follow these steps to provide the sysadmin and db\_owner role permissions to the new user:



- Navigate to **Machine Name > SQLEXPRESS > Security > Logins**, and right-click the **user > Properties**.
- Go to **Server Roles**, check the **sysadmin** box, and click **OK**.
- Go to **User Mapping** in the left pane. In the *Users mapped to this login* list, check the **database** and in the **Database role membership** for list, select **db\_owner**, and click **OK**.



**Note:** For details about user roles, refer the documents in the following links:

- For Server-Level Roles: <http://msdn.microsoft.com/en-us/library/ms188659.aspx>
- For Database-Level Roles: <http://msdn.microsoft.com/en-us/library/ms189121.aspx>

In general, the configured account requires any one of the three sets of privileges listed in the table below to complete the migration process successfully.

	Required database role	Required permissions
Set 1	db_owner	Not required
Set 2	db_datareader, db_datawriter, db_ddladmin, db_backupoperator.	Not required
Set 3	db_ddladmin	ALTER ANY TABLE, ALTER ANY AGGREGATE, ALTER ANY DEFAULT, ALTER ANY FUNCTION, ALTER ANY PROCEDURE, ALTER ANY QUEUE, ALTER ANY RULE, ALTER ANY SYNONYM, ALTER ANY TYPE, ALTER ANY VIEW, ALTER ANY XML SCHEMA COLLECTION, ALTER ANY REFERENCES, CONTROL ON CERTIFICATE::[ZOHO_CERT] TO [user], CONTROL ON SYMMETRIC KEY::[##MS_DatabaseMasterKey##] TO [user], CONTROL ON SYMMETRIC KEY::[ZOHO_SYMM_KEY] TO [user]

## Important:

Please note that you must have the **db\_owner** permission while migrating PostgreSQL to Microsoft SQL for the first time. After a successful migration, you can revoke the **db\_owner** permission for the account, and provide the set 2 or set 3 permissions.

## Our Products

AD360 | Log360 | ADManager Plus | ADAudit Plus | ADSelfService Plus  
Exchange Reporter Plus | RecoveryManager Plus

## ManageEngine M365 Manager Plus

M365 Manager Plus is an extensive Microsoft 365 tool used for reporting, managing, monitoring, auditing, and creating alerts for critical incidents. With its user-friendly interface, you can easily manage Exchange Online, Azure Active Directory, Skype for Business, OneDrive for Business, Microsoft Teams, and other Microsoft 365 services from a single console.

\$ Get Quote

↓ Download