

# Manual Database Install

## Contents:

- *Pre-requisites*
- *Manual DB Install for PostgreSQL*
  - *Launch PostgreSQL*
  - *Trifacta database*
  - *Jobs database*
  - *Scheduling database*
  - *Time-based Trigger database*
  - *Configuration Service database*
  - *Artifact Storage Service database*
  - *Job Metadata Service database*
- *Manual DB Install for MySQL*
  - *Launch MySQL*
  - *Trifacta database*
  - *Jobs database*
  - *Scheduling database*
  - *Time-based Trigger database*
  - *Configuration Service database*
  - *Artifact Storage Service database*
  - *Job Metadata Service database*

---

As needed, you can perform manual creation of users and individual databases used by the Trifacta® platform.

**Tip:** In most installation scenarios, manual database installs are not needed.

## Pre-requisites

1. The database server must be installed and available:
  - a. See *Install Databases for PostgreSQL*.
  - b. See *Install Databases for MySQL*.
2. The databases must be initialized.

**NOTE:** If you are using non-standard database names, users, passwords, port numbers, that information must be applied during installation and afterwards in the Trifacta platform configuration.

For more information, see *Configure the Databases*.

## Manual DB Install for PostgreSQL

Use the following steps to manually create the users and databases required by the platform in PostgreSQL.

### Launch PostgreSQL

#### Pre-requisites for PostgreSQL:

- The installing user must have write permissions to the directory from which the commands are executed.
- The installing user must have sudo privileges.

1. Switch to the Postgres user. Launch psql.

```
sudo su - postgres
```

**NOTE:** Unless the port number for postgres has been modified, it should be listening at the default value: 5432.

2. Launch psql using the following command, applying the admin password when prompted:

```
psql --host=${POSTGRES_HOST_NAME} --port=${POSTGRES_HOST_PORT} --username=${POSTGRES_ADMIN_USERNAME} --password=${POSTGRES_ADMIN_PASSWORD}
```

where:

Parameter	Description
host	Public DNS value for database server instance
port	Port number value for database server instance
username	Admin username for database server instance
password	Password for admin username

3. Execute the following commands using the postgres user.

**NOTE:** The values in platform configuration must match the values that you use below. Below are the default values.

### Trifacta database

```
CREATE ROLE trifacta LOGIN ENCRYPTED PASSWORD '<pwd_trifacta>';  
CREATE DATABASE trifacta WITH OWNER trifacta;
```

### Jobs database

```
CREATE ROLE trifactaactiviti WITH LOGIN ENCRYPTED PASSWORD '<pwd_trifactaactiviti>';  
CREATE DATABASE "trifacta-activiti" WITH OWNER trifactaactiviti;
```

### Scheduling database

```
CREATE ROLE trifactaschedulingservice LOGIN ENCRYPTED PASSWORD '<pwd_trifactaschedulingservice>';  
CREATE DATABASE trifactaschedulingservice WITH OWNER trifactaschedulingservice;
```

For more information on scheduling, see *Configure Automator*.

### Time-based Trigger database

```
CREATE ROLE trifactatimebasedtriggerservice LOGIN ENCRYPTED PASSWORD '<pwd_trifactatimebasedtriggerservice>';  
CREATE DATABASE trifactatimebasedtriggerservice WITH OWNER trifactatimebasedtriggerservice;
```

For more information on scheduling, see *Configure Automator*.

## Configuration Service database

```
CREATE ROLE trifactaconfigurationservice LOGIN ENCRYPTED PASSWORD '<pwd_trifactaconfigurationservice>';
CREATE DATABASE trifactaconfigurationservice WITH OWNER trifactaconfigurationservice;
```

## Artifact Storage Service database

```
CREATE ROLE trifactaartifactstorageservice LOGIN ENCRYPTED PASSWORD '<pwd_trifactaartifactstorageservice>';
CREATE DATABASE trifactaartifactstorageservice WITH OWNER trifactaartifactstorageservice;
```

## Job Metadata Service database

```
CREATE ROLE trifactajobmetadataservice LOGIN ENCRYPTED PASSWORD '<pwd_trifactajobmetadataservice>';
CREATE DATABASE trifactajobmetadataservice WITH OWNER trifactajobmetadataservice;
```

## Manual DB Install for MySQL

Use the following steps to manually initialize the databases required by the platform in MySQL.

### Launch MySQL

#### Pre-requisites for MySQL:

- The installing user must have write permissions to the directory from which the commands are executed.
- The installing user must have sudo privileges.

1. Login to MySQL as the root user. When prompted, enter the password for `root` user:

```
mysql -u root -p
```

2. Execute the following commands using the `root` user.

**NOTE:** The values in platform configuration must match the values that you use below. Below are the default values.

### Trifacta database

```
Create USER 'trifacta' IDENTIFIED BY '<pwd_trifacta>';
CREATE DATABASE trifacta;
Grant all on trifacta.* to trifacta;
```

### Jobs database

```
Create USER 'trifactaactiviti' IDENTIFIED BY '<pwd_trifactaactiviti>';
CREATE DATABASE trifactaactiviti;
Grant all on trifactaactiviti.* to trifactaactiviti;
```

### Scheduling database

```
Create USER 'trifactaschedulingservice' IDENTIFIED BY '<pwd_trifactaschedulingservice>';
CREATE DATABASE trifactaschedulingservice;
Grant all on trifactaschedulingservice.* to trifactaschedulingservice;
```

For more information on scheduling, see *Configure Automator*.

### **Time-based Trigger database**

```
Create USER 'trifactatimebasedtriggerservice' IDENTIFIED BY '<pwd_trifactatimebasedtriggerservice>';  
CREATE DATABASE trifactatimebasedtriggerservice;  
Grant all on trifactatimebasedtriggerservice.* to trifactatimebasedtriggerservice;
```

For more information on scheduling, see *Configure Automator*.

### **Configuration Service database**

```
Create USER 'trifactaconfigurationservice' IDENTIFIED BY '<pwd_trifactaconfigurationservice>';  
CREATE DATABASE trifactaconfigurationservice;  
Grant all on trifactaconfigurationservice.* to trifactaconfigurationservice;
```

### **Artifact Storage Service database**

```
Create USER 'trifactaartifactstorageservice' IDENTIFIED BY '<pwd_trifactaartifactstorageservice>';  
CREATE DATABASE trifactaartifactstorageservice;  
Grant all on trifactaartifactstorageservice.* to trifactaartifactstorageservice;
```

### **Job Metadata Service database**

```
Create USER 'trifactajobmetadataservice' IDENTIFIED BY '<pwd_trifactajobmetadataservice>';  
CREATE DATABASE trifactajobmetadataservice;  
Grant all on trifactajobmetadataservice.* to trifactajobmetadataservice;
```