Table of contents

TimescaleDB Installation and Configuration Manual (on Debian-Based Systems)

This manual provides detailed instructions for installing and configuring TimescaleDB on Linux, specifically for Debian-based systems like Ubuntu. TimescaleDB is an open-source time-series SQL database built on PostgreSQL, designed to provide fast analytics, scalability, and ease of use for time-series data.

Complete documentation :

https://docs.timescale.com/self-hosted/latest/install/installation-linux/#installing-self-hosted-timescaledb-on-debian-based-systems

Installation

1. Prepare Your System

First, ensure your system is prepared for the installation by adding the PostgreSQL third-party repository to access the latest PostgreSQL packages. Execute the following commands as the root user or use `sudo`:

```
$ cd /var/lib/postgresql
$ apt install gnupg postgresql-common apt-transport-https lsb-release
wget
```

2. Run the PostgreSQL Repository Setup Script

To include the latest PostgreSQL versions in your package list, run the PostgreSQL repository setup script. Replace `jammy` with the codename of your Ubuntu version if different:

```
$ sudo /usr/share/postgresql-common/pgdg/apt.postgresql.org.sh jammy
```

3. Add the TimescaleDB Third Party Repository

Incorporate the TimescaleDB repository into your system's package sources:

```
$ echo "deb https://packagecloud.io/timescale/timescaledb/ubuntu/ jammy
main" | sudo tee /etc/apt/sources.list.d/timescaledb.list
```

4. Install the TimescaleDB GPG Key

Secure your installation by adding the TimescaleDB GPG key:

```
$ wget --quiet -O - https://packagecloud.io/timescale/timescaledb/gpgkey
| sudo gpg --dearmor -o /etc/apt/trusted.gpg.d/timescaledb.gpg
```

5. Update Local Repository List

Refresh your local package index to recognize the newly added repositories:

```
$ apt update
```

6. Install TimescaleDB

Install TimescaleDB along with support for PostgreSQL 16:

```
$ apt install timescaledb-2-postgresql-16
```

Configuration

After successfully installing TimescaleDB, proceed to configure your database.

Configure Database

Utilize the `timescaledb-tune` utility to automatically adjust your PostgreSQL configuration settings for optimal performance with TimescaleDB, and answer questions (*yes* by default)

```
$ sudo timescaledb-tune
```

Setup

Final steps to set up and verify your TimescaleDB installation.

Restart PostgreSQL Service

Apply the changes made by `timescaledb-tune` by restarting the PostgreSQL service:

```
$ systemctl restart postgresql
```

Connect to PostgreSQL

Use the PostgreSQL command-line tool, `psql`, to connect to your PostgreSQL server:

```
$ psql -U postgres -h localhost -p 5433
```

Create a New Database

Once connected at the *psql* prompt, create a new database named `tsdb` (or your chosen name) for your time-series data:

```
postgres=# CREATE DATABASE tsdb;
```

Connect to Your New Database

Switch to your newly created database within the `psql` session:

```
postgres=# \c tsdb;
```

Enable TimescaleDB Extension

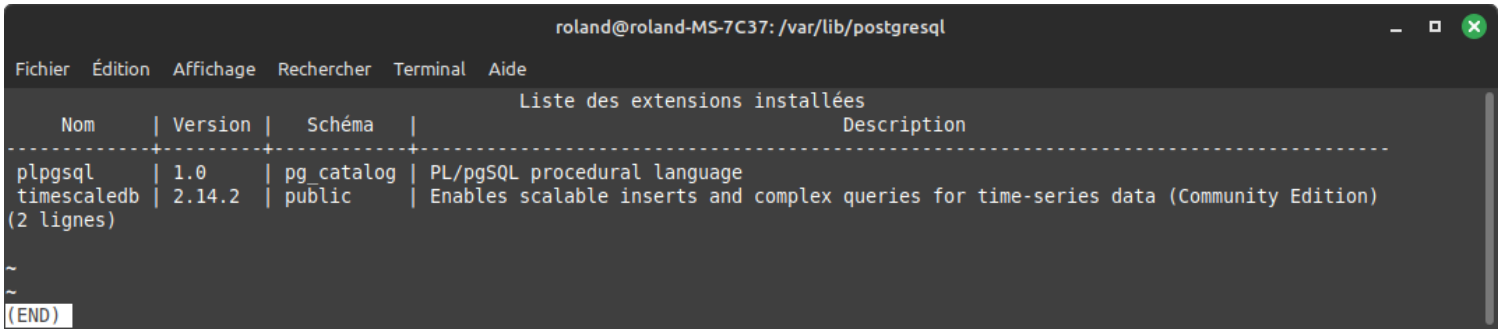
Activate the TimescaleDB extension to enable its features on your database:

```
tsdb=# CREATE EXTENSION IF NOT EXISTS timescaledb;
```

Verify Installation

Confirm the TimescaleDB extension is installed and active by listing all installed extensions:

```
tsdb=# \dx
tsdb=# \q
```



Conclusion

You have successfully installed and configured TimescaleDB on your Debian-based system. This manual walked you through adding the necessary repositories, installing TimescaleDB, configuring your database with `timescaledb-tune`, and enabling the TimescaleDB extension. It's now ready for you to begin creating, managing, and querying time-series data within a robust PostgreSQL environment.