To install PostgreSQL 9.4.19 on Ubuntu 22.04, you need to follow a few specific steps, since this version is not included in the default package repositories of Ubuntu 22.04. You'll have to use the PostgreSQL Global Development Group's repository, as PostgreSQL 9.4 is considered an older release.

Below are the steps and commands to install PostgreSQL 9.4.19 on Ubuntu 22.04:

**Step 1: Install Necessary Packages**

First, update your package list and install necessary packages that will allow you to add a new repository.

bash

Copy code

# Update package list

sudo apt update

# Install necessary packages

sudo apt install -y wget gnupg2 lsb-release

**Step 2: Add the PostgreSQL APT Repository**

Add the PostgreSQL Global Development Group's repository, which contains older versions of PostgreSQL, including 9.4.

1. **Download the PostgreSQL GPG key**:

bash

Copy code

wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -

1. **Add the PostgreSQL APT repository**:

bash

Copy code

sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb\_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'

**Step 3: Update Package List Again**

Update the package list to include packages from the newly added PostgreSQL repository.

bash

Copy code

sudo apt update

**Step 4: Install PostgreSQL 9.4**

Install PostgreSQL 9.4 by specifying the version:

bash

Copy code

sudo apt install -y postgresql-9.4

**Step 5: Verify PostgreSQL Installation**

Verify that PostgreSQL 9.4 is installed and running correctly:

1. **Check PostgreSQL version**:

bash

Copy code

psql --version

You should see output similar to:

scss

Copy code

psql (PostgreSQL) 9.4.19

1. **Check PostgreSQL service status**:

bash

Copy code

sudo systemctl status postgresql

Ensure that the service is running. If it's not running, you can start it with:

bash

Copy code

sudo systemctl start postgresql

**Step 6: Configure PostgreSQL**

Once PostgreSQL 9.4 is installed, you might want to configure it to suit your needs. Here's how you can configure the default settings and create a new user and database:

1. **Switch to the PostgreSQL user**:

bash

Copy code

sudo -i -u postgres

1. **Create a new PostgreSQL user**:

bash

Copy code

createuser --interactive

You will be prompted to enter a username and decide whether the user should be a superuser.

1. **Create a new PostgreSQL database**:

bash

Copy code

createdb mydatabase

Replace mydatabase with the desired name of your database.

1. **Access the PostgreSQL prompt**:

bash

Copy code

psql

* + **List databases**:

sql

Copy code

\l

* + **Create a table**:

sql

Copy code

CREATE TABLE employees (

id SERIAL PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

email VARCHAR(100),

hire\_date DATE

);

* + **Insert data into the table**:

sql

Copy code

INSERT INTO employees (first\_name, last\_name, email, hire\_date)

VALUES

('John', 'Doe', 'john.doe@example.com', '2024-07-01'),

('Jane', 'Smith', 'jane.smith@example.com', '2024-06-15'),

('Alice', 'Johnson', 'alice.johnson@example.com', '2024-05-20');

* + **Query the table**:

sql

Copy code

SELECT \* FROM employees;

* + **Exit the PostgreSQL prompt**:

sql

Copy code

\q

1. **Exit the PostgreSQL user session**:

bash

Copy code

exit

**Step 7: Secure PostgreSQL**

It's important to secure your PostgreSQL installation:

1. **Set a password for the PostgreSQL user**:

bash

Copy code

sudo -i -u postgres psql

In the PostgreSQL prompt, set a password:

sql

Copy code

\password postgres

Enter and confirm your new password, then exit the prompt with \q.

1. **Configure PostgreSQL to accept remote connections** (if needed):

Edit the pg\_hba.conf file:

bash

Copy code

sudo nano /etc/postgresql/9.4/main/pg\_hba.conf

Add or modify a line to allow remote connections:

css

Copy code

host all all 0.0.0.0/0 md5

Edit the postgresql.conf file:

bash

Copy code

sudo nano /etc/postgresql/9.4/main/postgresql.conf

Uncomment and set the listen\_addresses directive:

plaintext

Copy code

listen\_addresses = '\*'

**Restart PostgreSQL** to apply changes:

bash

Copy code

sudo systemctl restart postgresql

**Summary**

By following these steps, you can install PostgreSQL 9.4.19 on Ubuntu 22.04, set up a database and a table, and ensure your installation is secure and ready for use. If you have any more questions or need further assistance, feel free to ask!