

Table Of Contents

Table Of Contents 1

Step # 1: Allow remote IP address to access PostgreSQL 2

Step # 2: Allow communication over TCP/IP 2

Step # 3: Restart PostgreSQL server 2

Step # 4: Test your setup 2

See also: 3

[Home](#) > [Faq](#) > [Debian / Ubuntu](#)

PostgreSQL psql: could not connect to server: Connection refused

Posted by [Vivek Gite](#) <vivek@nixcraft.com>

Q. When I try to connect remote PostgreSQL, I am receiving an error which reads as follows:

*psql: could not connect to server: Connection refused
Is the server running on host host.domain.com and accepting
TCP/IP connections on port 5432?*

How do I fix this problem? I am using CentOS 4.x version.

A. First make sure PostgreSQL server has been started to remote server.

```
# /etc/init.d/postgresql start
```

If it is running and you get above error, you need to add enable TCP/IP support. By default, the PostgreSQL server only allows connections to the database from the local machine or localhost. This is a security feature.

Step # 1: Allow remote IP address to access PostgreSQL

You need to open file called `/var/lib/pgsql/data/pg_hba.conf`. Login as postgres user using su command:

```
$ su - postgres  
$ vi /var/lib/pgsql/data/pg_hba.conf
```

Now append following line. Let us say you would like to give access to 192.168.1.0/24 network:

```
host all all 192.168.1.0 255.255.255.0 trust
```

Please replace 192.168.1.0 and 255.255.255.0 to reflect the actual network IP address range of the clients system in your own network.

Save close the file.

Step # 2: Allow communication over TCP/IP

You need to open PostgreSQL configuration file `/var/lib/pgsql/data/postgresql.conf`

```
$ vi /var/lib/pgsql/data/postgresql.conf
```

Now bind and open TCP/IP port by setting `tcpip_socket` to true:

```
tcpip_socket = true
```

Save and close the file.

Step # 3: Restart PostgreSQL server

Restart the PostgreSQL server with the following command

```
# /etc/init.d/postgresql restart
```

This will open default port 5432.

Step # 4: Test your setup

Use psql command from client system as follows:

```
psql -h PostgreSQL-IP-ADDRESS -U USERNAME -d DATABASENAME
```

[1]

Connect to remote server by IP address 192.168.1.5 and login using vivek user to connect to sales database, use:

```
$ psql -h 192.168.1.5 -U vivek -d sales
```

Where,

- **-h 192.168.1.5** : Specifies the host name of the machine or IP address (192.168.1.5) on which the server is running.
- **-U vivek** : Connect to the database as the vivek username instead of the default. You must have [account and permission](#) ^[2] to connect as vivek user.
- **-d sales** : Specifies the name of the database (sales) to connect to.

See also:

=> [How do I add a user to PostgreSQL database server?](#) ^[2]

4000+ howtos and counting! Want to read more Linux / UNIX howtos, tips and tricks? Subscribe to our [daily email](#) newsletter or [weekly newsletter](#) to make sure you don't miss a single tip/tricks. Alternatively, subscribe via [RSS/XML](#) feed.

Article printed from Frequently Asked Questions About Linux / UNIX: <http://www.cyberciti.biz/faq/>

URL to article: <http://www.cyberciti.biz/faq/postgresql-remote-access-or-connection/>

URLs in this post:

[1] Image: <http://www.cyberciti.biz/faq/faq/category/postgresql/>

[2] account and permission: <http://www.cyberciti.biz/faq/faq/howto-add-postgresql-user-account/>