How To Install pgAdmin 4 on CentOS 7 / RHEL 7 & Fedora 29 / Fedora 28

<https://docs.bitnami.com/aws/apps/noalyss/administration/configure-pgadmin/>

<https://www.itzgeek.com/how-tos/linux/centos-how-tos/how-to-install-pgadmin-4-on-centos-7-rhel-7-fedora-29-fedora-28.html>

[pgAdmin](https://www.pgadmin.org/) is a free and open source management tool for [PostgreSQL](https://www.itzgeek.com/tag/postgresql) and derivative relation databases such as EDB Advanced Server. It can be installed on multiple OS platforms such as [Linux](https://www.itzgeek.com/category/how-tos/linux), Unix, Mac OS X, and Windows to manager PostgreSQL 9.2 and above.

Deployment Mode

pgAdmin can be run as a web or desktop application.

Server Deployment

In [server deployment](https://www.pgadmin.org/docs/pgadmin4/4.x/server_deployment.html) (web application), it is deployed as a web application behind a web server running as a reverse proxy or using the WSGI interface.

Desktop Deployment

In [desktop deployment](https://www.pgadmin.org/docs/pgadmin4/4.x/desktop_deployment.html) (desktop application), it is deployed to run in desktop mode by utilizing the desktop runtime to host the application. When the runtime is launched from system-tray, it runs the pgAdmin server and launches a web browser to render the user interface.

In this guide, we will install [pgAdmin 4](https://www.itzgeek.com/tag/pgadmin) on [CentOS 7 / RHEL 7](https://www.itzgeek.com/tag/centos-7) & [Fedora 29 / Fedora 28](https://www.itzgeek.com/category/how-tos/linux/fedora-how-tos) as a web application ([server deployment](https://www.pgadmin.org/docs/pgadmin4/4.x/server_deployment.html)).

Prerequisites

Install PostgreSQL Server

This post assumes that you already have PostgreSQL 9.2 and above installed on your system. Otherwise, follow the post: [How To Install PostgreSQL 11 / 10 on CentOS 7 / RHEL 7](https://www.itzgeek.com/how-tos/linux/centos-how-tos/install-postgresql-9-3-on-centos-7.html).

Set up EPEL Repository

We would need to enable the EPEL repository to download dependent packages for pgAdmin.

yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm

Set up PostgreSQL Repository

pgAdmin 4 is available in PostgreSQL repository and your system should have the PostgreSQL repository by now if you have already completed the installation of PostgreSQL. If not, add the PostgreSQL repository using the below command.

### PostgreSQL 11 ###

**# RHEL 7 #**

yum install https://download.postgresql.org/pub/repos/yum/11/redhat/rhel-7-x86\_64/pgdg-redhat11-11-2.noarch.rpm

**# CentoS 7 #**

yum install https://download.postgresql.org/pub/repos/yum/11/redhat/rhel-7-x86\_64/pgdg-centos11-11-2.noarch.rpm

**# Fedora 29 #**

dnf install https://download.postgresql.org/pub/repos/yum/11/fedora/fedora-29-x86\_64/pgdg-fedora11-11-2.noarch.rpm

**# Fedora 28 #**

dnf install https://download.postgresql.org/pub/repos/yum/11/fedora/fedora-28-x86\_64/pgdg-fedora11-11-2.noarch.rpm

### PostgreSQL 10 ##

**# RHEL 7 #**

yum install https://download.postgresql.org/pub/repos/yum/10/redhat/rhel-7-x86\_64/pgdg-redhat10-10-2.noarch.rpm

**# CentoS 7 #**

yum install https://download.postgresql.org/pub/repos/yum/10/redhat/rhel-7-x86\_64/pgdg-centos10-10-2.noarch.rpm

**# Fedora 29 #**

dnf install https://download.postgresql.org/pub/repos/yum/10/fedora/fedora-29-x86\_64/pgdg-fedora10-10-4.noarch.rpm

**# Fedora 28 #**

dnf install https://download.postgresql.org/pub/repos/yum/10/fedora/fedora-28-x86\_64/pgdg-fedora10-10-4.noarch.rpm

Install pgAdmin 4

Once you have PostgreSQL repository configured on your system, run the following command to install pgAdmin 4.

**### RHEL / CentOS ###**

yum -y install pgadmin4

**### Fedora ###**

dnf -y install pgadmin4

Start and enable httpd service.

systemctl start httpd

systemctl enable httpd

Configure pgAdmin 4

We would need to do a few configuration changes prior to accessing the pgAdmin 4.

Copy the pgAdmin 4 sample configuration.

cp /etc/httpd/conf.d/pgadmin4.conf.sample /etc/httpd/conf.d/pgadmin4.conf

Create a pgAdmin log and data directories.

mkdir /var/log/pgadmin4/

mkdir /var/lib/pgadmin4/

Create/Edit config\_local.py file.

vi /usr/lib/python2.7/site-packages/pgadmin4-web/config\_local.py

Add the following settings.

LOG\_FILE = '/var/log/pgadmin4/pgadmin4.log'

SQLITE\_PATH = '/var/lib/pgadmin4/pgadmin4.db'

SESSION\_DB\_PATH = '/var/lib/pgadmin4/sessions'

STORAGE\_DIR = '/var/lib/pgadmin4/storage'

Change permissions of directories so that Apache can write data into it.

chown -R apache:apache /var/lib/pgadmin4/\*

chown -R apache:apache /var/log/pgadmin4/\*

Run the following command to create a user account for the pgAdmin 4 web interface.

python /usr/lib/python2.7/site-packages/pgadmin4-web/setup.py

**Output:**

NOTE: Configuring authentication for SERVER mode.

Enter the email address and password to use for the initial pgAdmin user account:

Email address: admin@itzgeek.local

Password: **xxxxxxxxx**

Retype password: **xxxxxxxxx**

pgAdmin 4 - Application Initialisation

======================================

Restart the Apache web service.

systemctl restart httpd

Firewall

Set up the firewall so that we can access pgAdmin 4 from external machines.

firewall-cmd --permanent --add-service=http

firewall-cmd --reload

SELinux

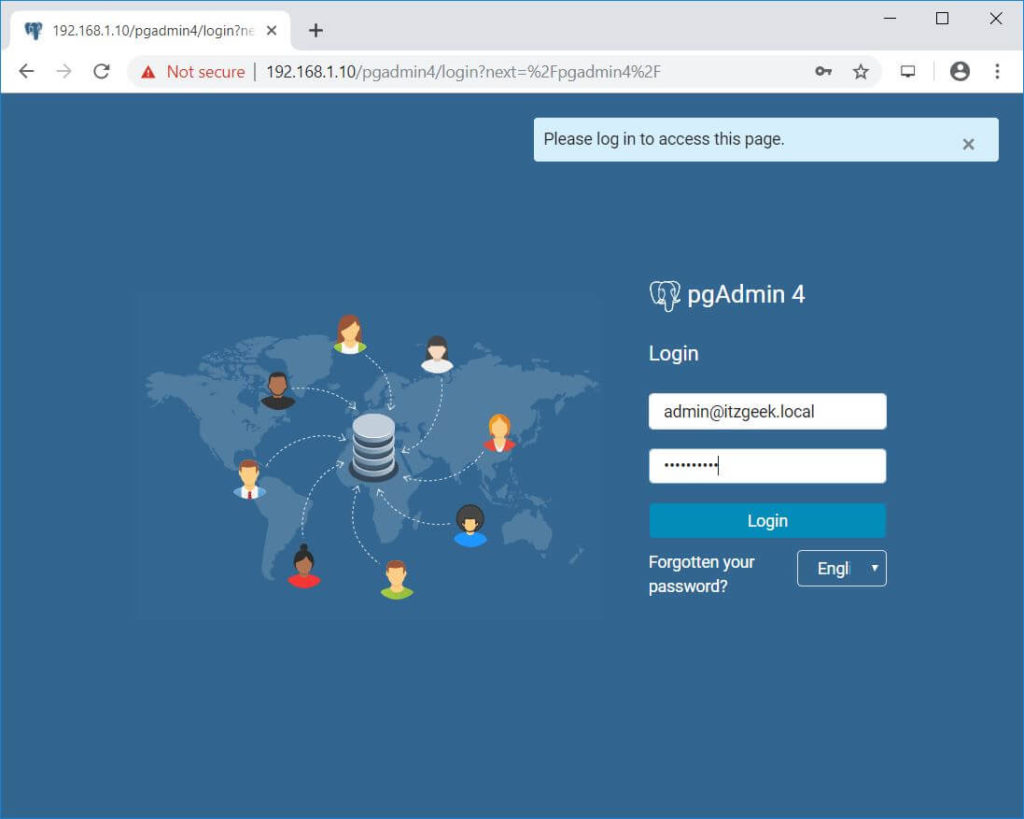
Consider [disabling SELinux permanently](https://www.itzgeek.com/how-tos/linux/centos-how-tos/disable-selinux-on-centos-6-disable-selinux-on-rhel-6.html) for pgAdmin 4 to work properly.

Access pgAdmin 4

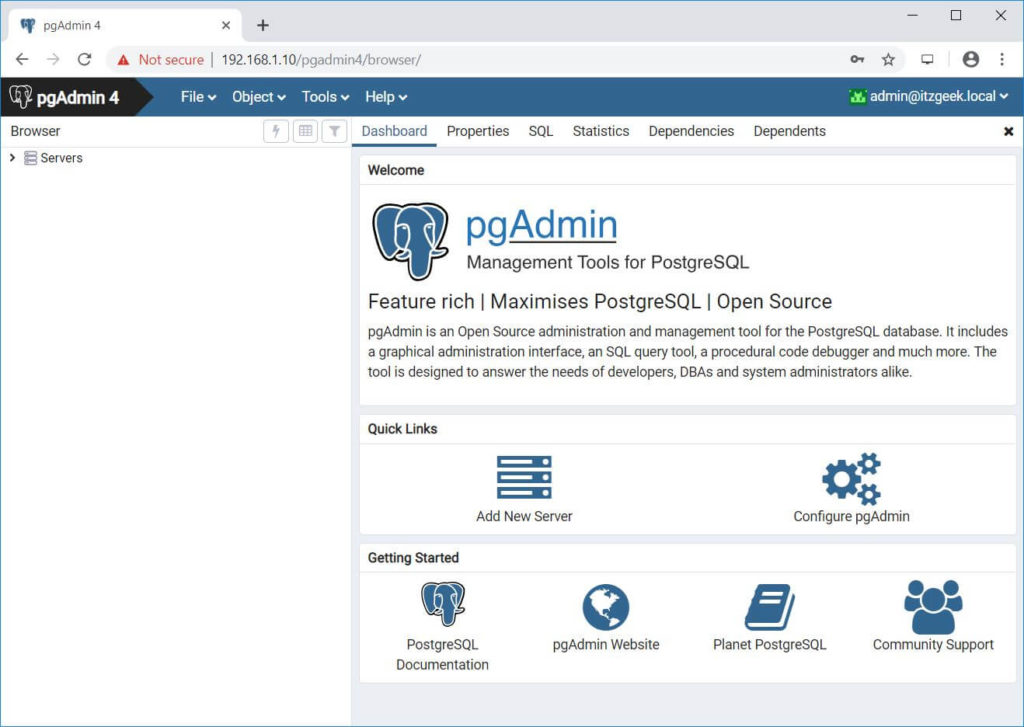
Open a web browser and go to the following URL to access the pgAdmin 4 interface.

http://ip.add.re.ss/pgadmin4

Log in to pgAdmin 4 web interface using the email address and password you have created earlier.

[](https://www.itzgeek.com/wp-content/uploads/2019/03/Install-pgAdmin-4-on-CentOS-7-pgAdmin-Login-Screen.jpg)Install pgAdmin 4 on CentOS 7 – pgAdmin Login Screen

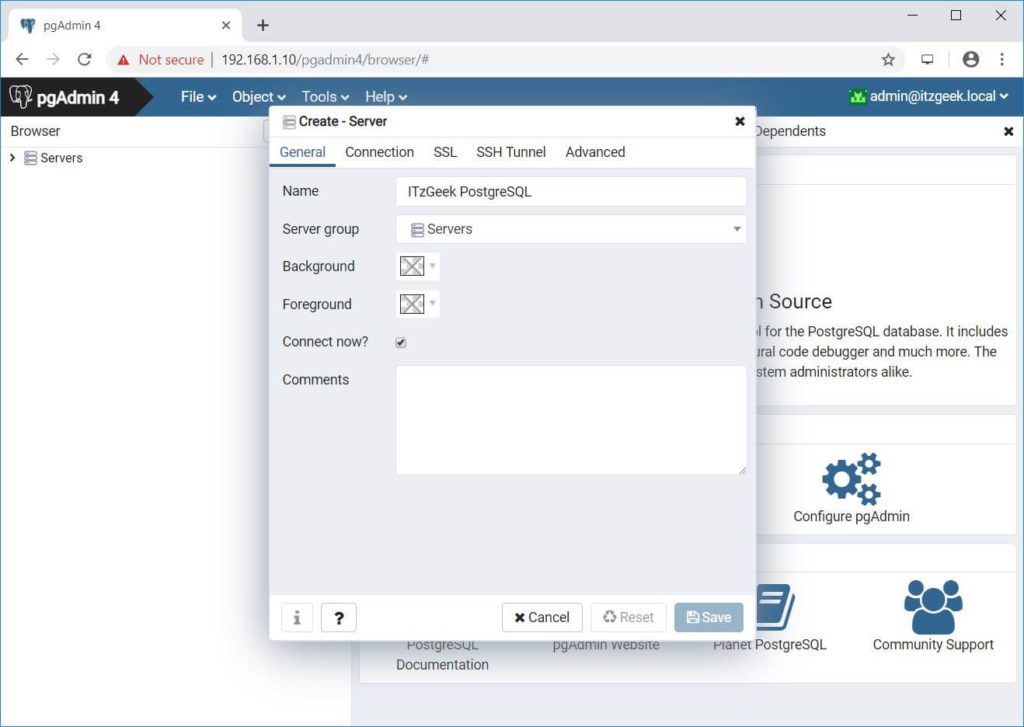
Upon successful login, you should see the pgAdmin 4 interface.

[](https://www.itzgeek.com/wp-content/uploads/2019/03/Install-pgAdmin-4-on-CentOS-7-pgAdmin-Home-Page.jpg)Install pgAdmin 4 on CentOS 7 – pgAdmin Home Page

To manage a PostgreSQL server, you will need to add a new server. Click on Add New Server.

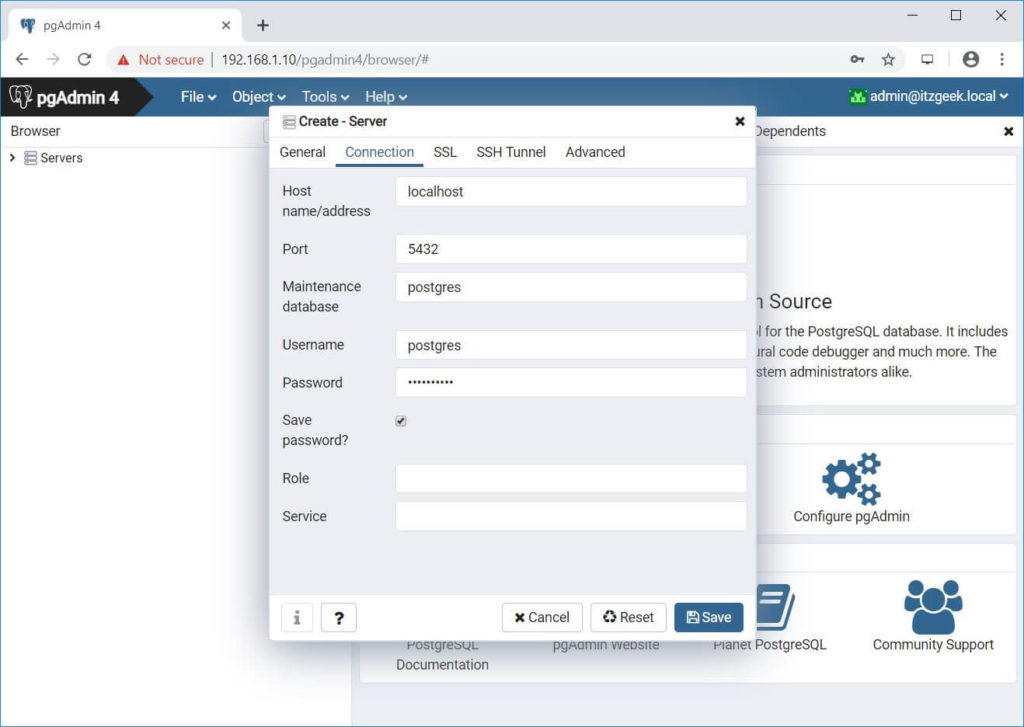
**General Tab:**

**Name:-** Name your PostgreSQL Server

[](https://www.itzgeek.com/wp-content/uploads/2019/03/Install-pgAdmin-4-on-CentOS-7-Add-New-PostgreSQL-Instance.jpg)Install pgAdmin 4 on CentOS 7 – Add New PostgreSQL Instance

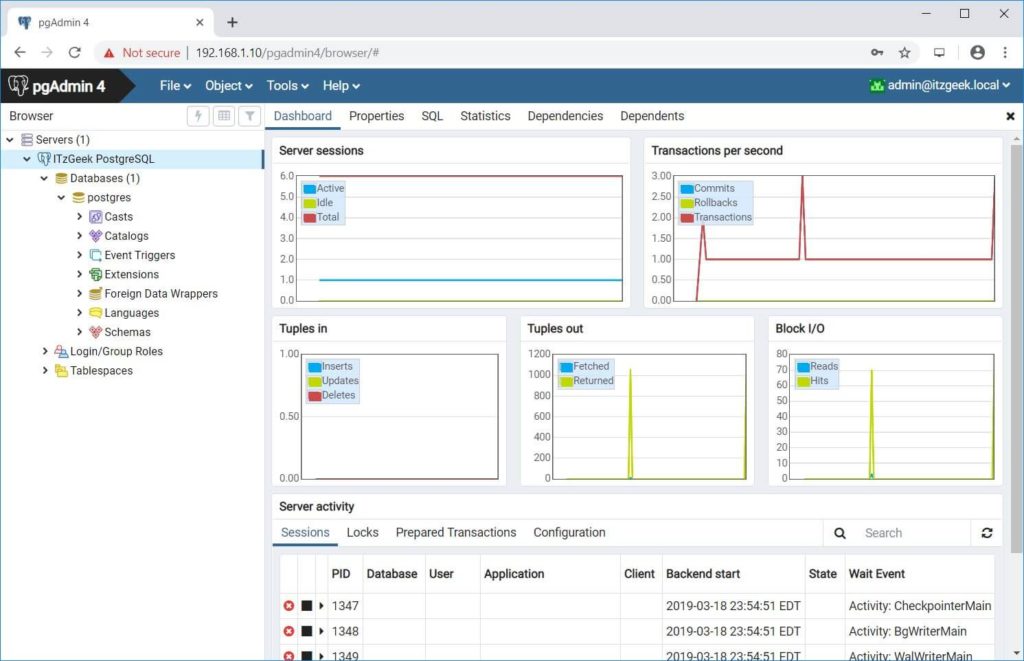
**Connection Tab:**

**Hostname/address:-** Hostname or IP Address of PostgreSQL server  
**Port:-** 5432 (Leave default) – Change it if required  
**Username:-** Username by which you are connecting. In my case, it is postgres.  
**Password:-** Password for the user

[](https://www.itzgeek.com/wp-content/uploads/2019/03/Install-pgAdmin-4-on-CentOS-7-PostgreSQL-Instance-Details.jpg)Install pgAdmin 4 on CentOS 7 – PostgreSQL Instance Details

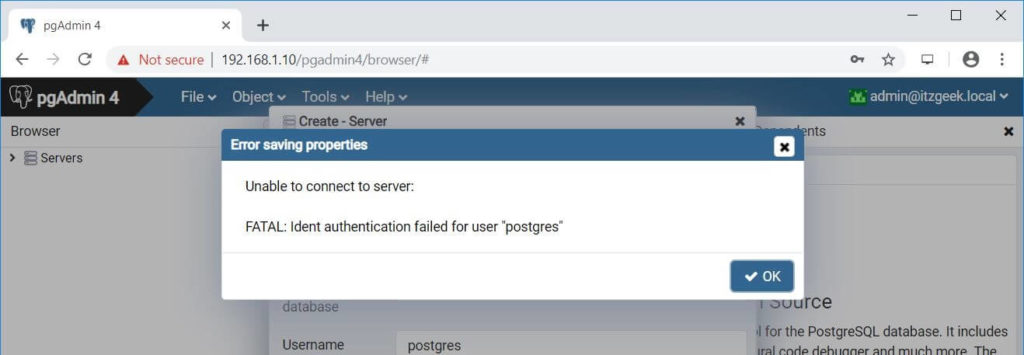
Click Save to save the changes.

If the connection to PostgreSQL server is successful, you should see the following page.

[](https://www.itzgeek.com/wp-content/uploads/2019/03/Install-pgAdmin-4-on-CentOS-7-Database-Statistics.jpg)Install pgAdmin 4 on CentOS 7 – Database Statistics

Troubleshooting

You may get a **Fatal: Ident authentication failed for user** error.

[](https://www.itzgeek.com/wp-content/uploads/2019/03/Install-pgAdmin-4-on-CentOS-7-PostgreSQL-Error.jpg)Install pgAdmin 4 on CentOS 7 – PostgreSQL Error

To fix the issue, edit pg\_hba.conf file.

### PostgreSQL 11 ###

vi /var/lib/pgsql/11/data/pg\_hba.conf

### PostgreSQL 10 ###

vi /var/lib/pgsql/11/data/pg\_hba.conf

Update the below line shown like below.

# TYPE DATABASE USER ADDRESS METHOD

host all all all md5

Restart the PostgreSQL service.

### PostgreSQL 11 ###

systemctl restart postgresql-11

### PostgreSQL 10 ###

systemctl restart postgresql-10

Conclusion

You have successfully installed [pgAdmin 4](https://www.itzgeek.com/tag/pgadmin) on [CentOS 7 / RHEL 7](https://www.itzgeek.com/tag/centos-7) & [Fedora 29 / Fedora 28](https://www.itzgeek.com/category/how-tos/linux/fedora-how-tos) and added one of your PostgreSQL instances to it to manage the database. pgAdmin 4 is similar to phpPgAdmin in terms of managing PostgreSQL databases. You can visit [pgAdmin 4 documentation](https://www.pgadmin.org/docs/pgadmin4/dev/index.html" \t "_blank) for more information.

# Configure PgAdmin 4

NOTE: This section assumes that you have [downloaded and installed pgAdmin 4](https://www.pgadmin.org/download/).

pgAdmin is the most popular and feature-rich platform for administration and development of PostgreSQL databases. Check the [pgAdmin official page](https://www.pgadmin.org/docs/) for more information.

To connect to your remote PostgreSQL database server using pgAdmin 4, follow these steps:

* Make sure that you have your cloud server’s IP address and application credentials ([instructions](https://docs.bitnami.com/aws/faq/get-started/find-credentials/)).
* Open port 5432 in the server firewall ([instructions](https://docs.bitnami.com/aws/faq/administration/use-firewall/)).

IMPORTANT: By default, the database port for the nodes in this solution cannot be accessed over a public IP address. As a result, you will only be able to connect to your database nodes from machines that are running in the same network. For security reasons, we do not recommend making the database port accessible over a public IP address. If you must make it accessible over a public IP address, we recommend restricting access to a trusted list of source IP addresses using firewall rules. For development purposes, you can also use a VPN or SSH tunnel. Refer to the FAQ for information on [accessing restricted ports using an SSH tunnel](https://docs.bitnami.com/aws/faq/get-started/access-ssh-tunnel/) or [opening ports in the server firewall](https://docs.bitnami.com/aws/faq/administration/use-firewall/).

* Connect to your cloud server using PuTTY or another SSH client ([instructions](https://docs.bitnami.com/aws/faq/get-started/connect-ssh/)).
* At the server console, edit the file /opt/bitnami/postgresql/data/pg\_hba.conf and add the following at the end, then save the file:
* host all all all md5
* Edit the file /opt/bitnami/postgresql/data/postgresql.conf and replace this line
* listen\_address='127.0.0.1'

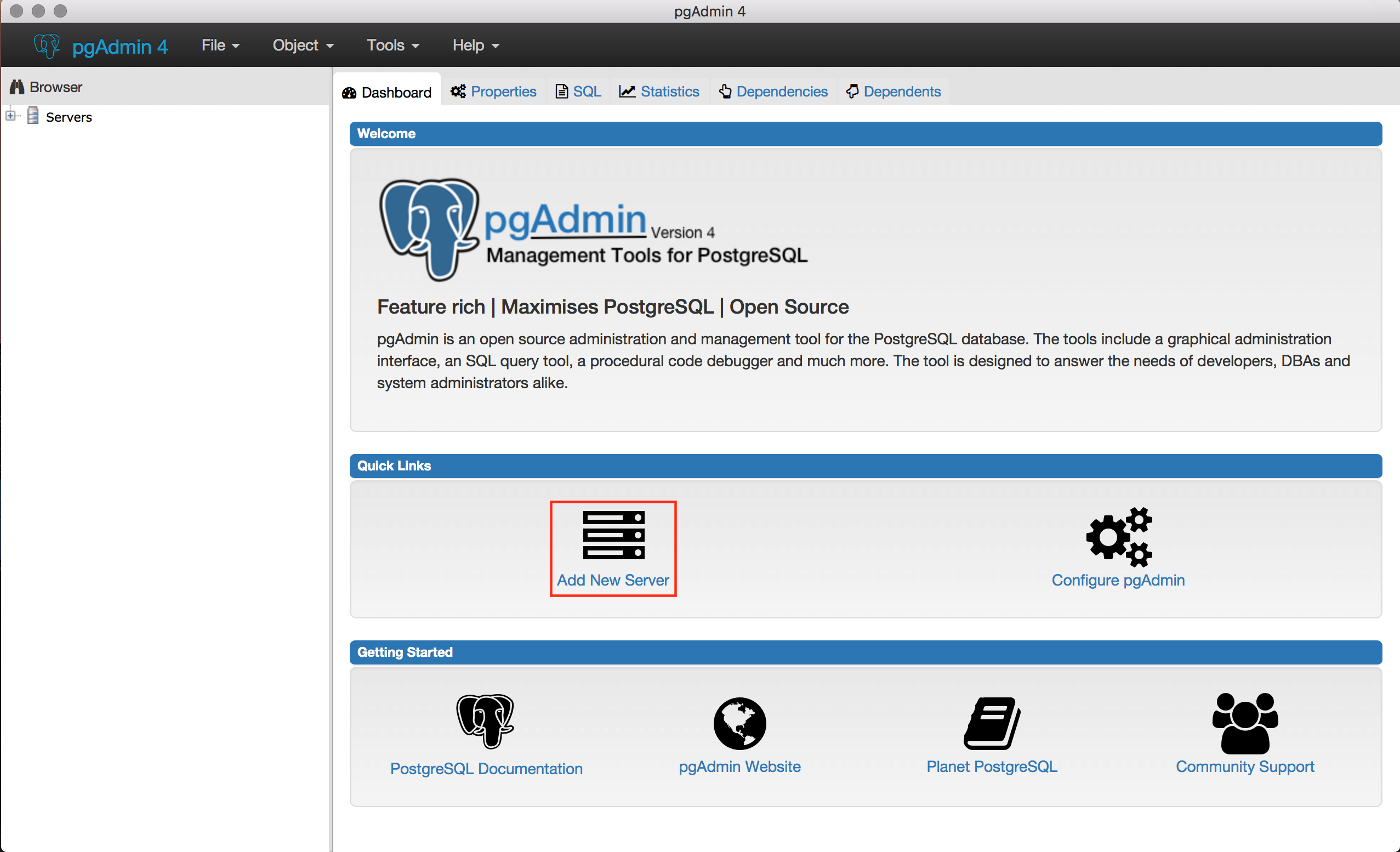
with:

listen\_addresses = '\*'

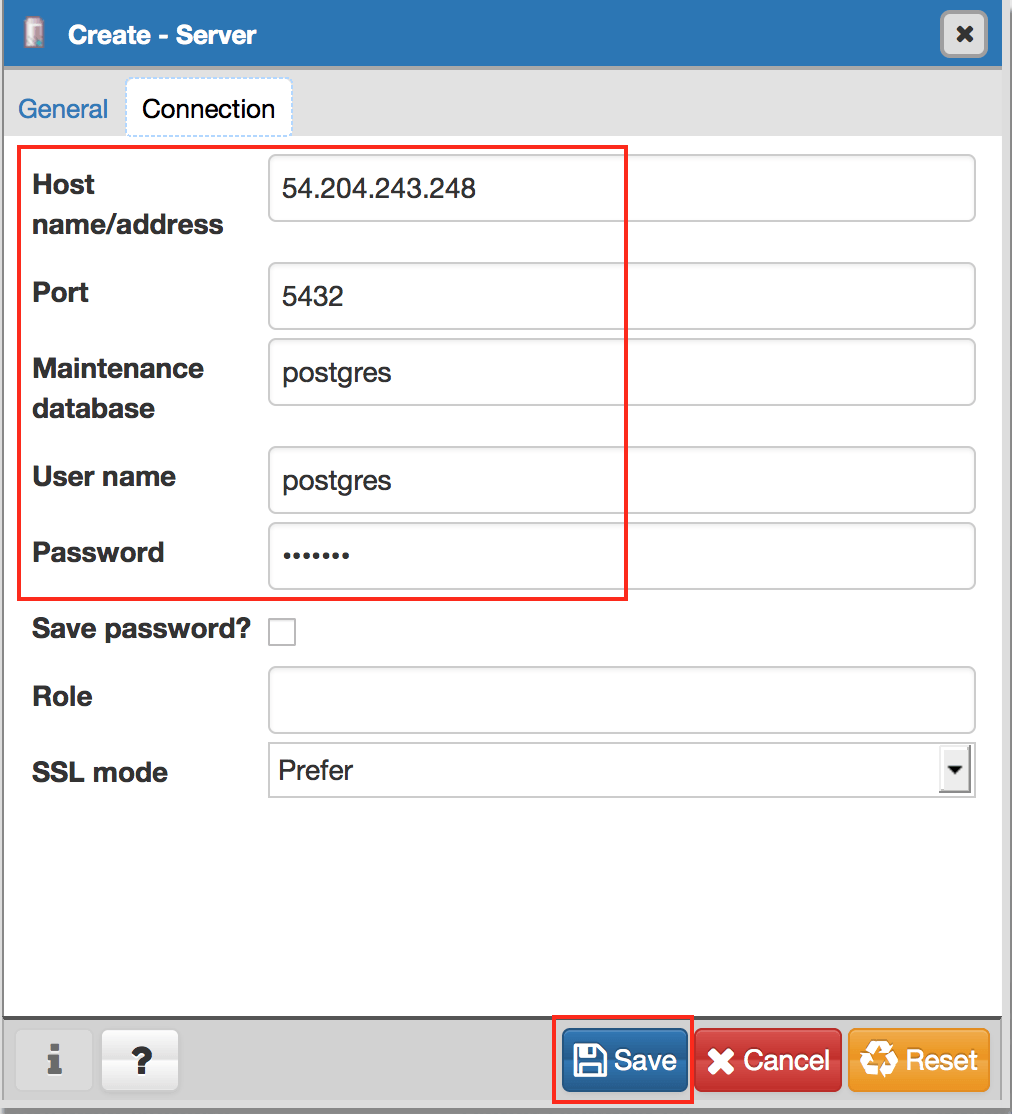
* Save the file.
* Restart the PostgreSQL server:
* sudo /opt/bitnami/ctlscript.sh restart postgresql

Your PostgreSQL server is now configured to accept remote connections, and you can connect to it using pgAdmin 4. Follow these steps:

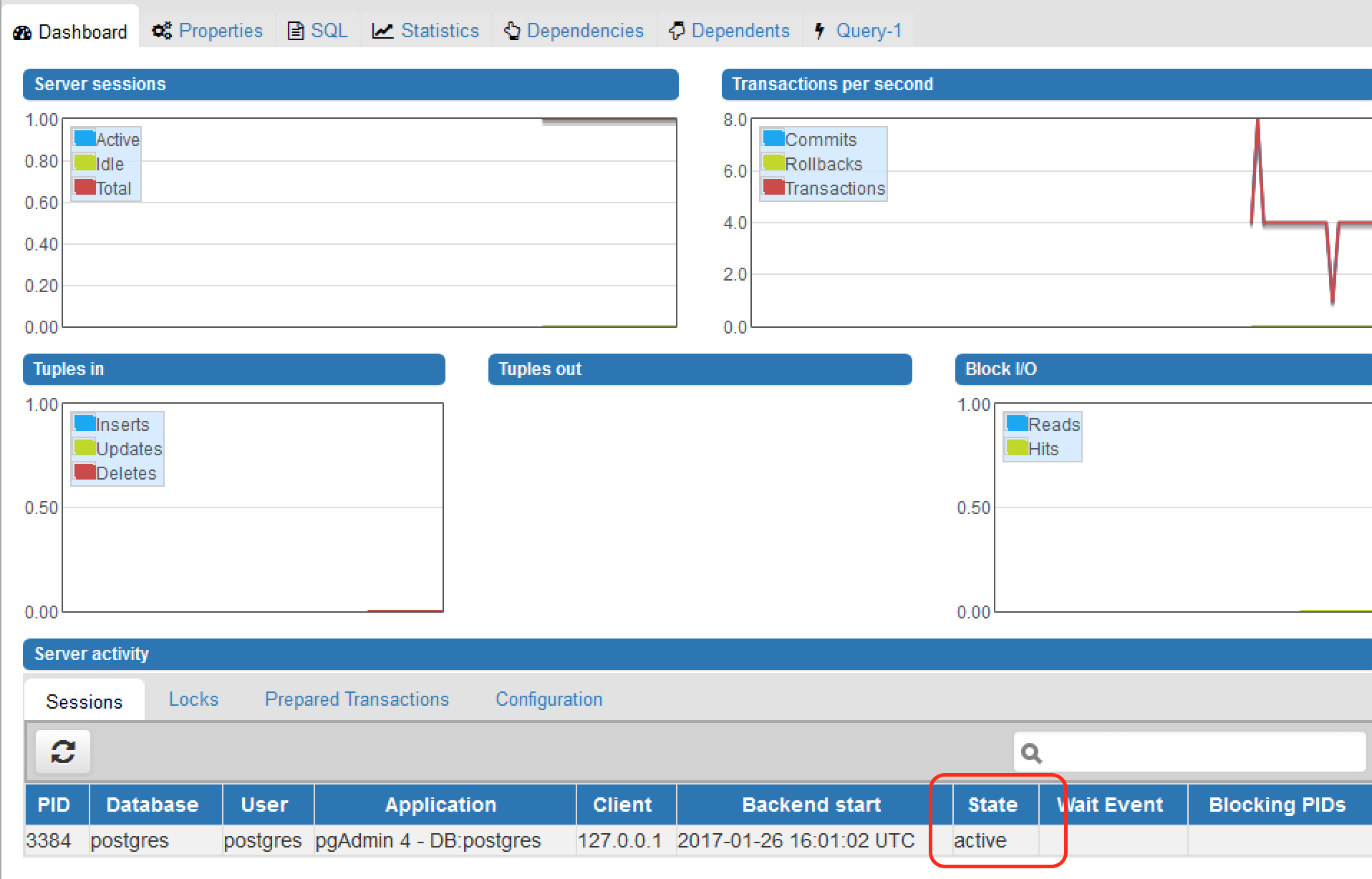
* Launch pgAdmin 4.
* Go to the “Dashboard” tab. In the “Quick Link” section, click “Add New Server” to add a new connection.

[](https://docs.bitnami.com/images/img/components/postgresql/config-pgadmin-1.png)

* Select the “Connection” tab in the “Create-Server” window.
* Then, configure the connection as follows:
* Enter your server’s IP address in the “Hostname/Address” field.
* Specify the “Port” as “5432”.
* Enter the name of the database in the “Database Maintenance” field.
* Enter your username as postgres and password (use the same password you used when previously configuring the server to accept remote connections) for the database.
* Click “Save” to apply the configuration.

[](https://docs.bitnami.com/images/img/components/postgresql/config-pgadmin-2.png)

* Check that the connection between pgAdmin 4 and the PostgreSQL database server is active. Navigate to the “Dashboard” tab and find the state of the server in the “Server activity” section:

[](https://docs.bitnami.com/images/img/components/postgresql/config-pgadmin-3.png)