Update Linux server packages by using **dnf** command.

[root@postgres-01 ~]# dnf update -y

Last metadata expiration check: 0:00:58 ago on Sun 22 Nov 2020 08:26:21 PM PKT.

Dependencies resolved.

Nothing to do.

Complete!

Our Linux operating system is already up-to-date.

Check the Linux operating system and Kernel version.

[root@postgres-01 ~]# uname -r

4.18.0-193.28.1.el8\_2.x86\_64

[root@postgres-01 ~]# cat /etc/redhat-release

CentOS Linux release 8.2.2004 (Core)

**Installing Postgres Official Yum Repository:**

Although Postgres database server is available in Linux yum repositories, but to install latest version, we have to add PostgreSQL official yum repository in our Linux server.

[root@postgres-01 ~]# dnf install -y https://download.postgresql.org/pub/repos/yum/reporpms/EL-8-x86\_64/pgdg-redhat-repo-latest.noarch.rpm

Last metadata expiration check: 0:02:50 ago on Sun 22 Nov 2020 08:26:21 PM PKT.

pgdg-redhat-repo-latest.noarch.rpm 6.4 kB/s | 11 kB 00:01

Dependencies resolved.

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

Package Architecture Version Repository Size

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

Installing:

pgdg-redhat-repo noarch 42.0-14 @commandline 11 k

Transaction Summary

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

Install 1 Package

Total size: 11 k

Installed size: 11 k

Downloading Packages:

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing : 1/1

Installing : pgdg-redhat-repo-42.0-14.noarch 1/1

Verifying : pgdg-redhat-repo-42.0-14.noarch 1/1

Installed:

pgdg-redhat-repo-42.0-14.noarch

Complete!

Disable the PostgreSQL module in standard yum repository by executing following command.

[root@postgres-01 ~]# dnf -qy module disable postgresql

**Installing Postgres Database Server on CentOS 8:**

Install Postgres database server 13 (latest at the time of this writing) on your Linux server by using **dnf** command.

[root@postgres-01 ~]# dnf install -y postgresql13-server

Last metadata expiration check: 0:00:30 ago on Sun 22 Nov 2020 08:30:03 PM PKT.

Dependencies resolved.

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

Package Arch Version Repository Size

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

Installing:

postgresql13-server x86\_64 13.1-1PGDG.rhel8 pgdg13 5.6 M

Installing dependencies:

postgresql13 x86\_64 13.1-1PGDG.rhel8 pgdg13 1.4 M

postgresql13-libs x86\_64 13.1-1PGDG.rhel8 pgdg13 410 k

Transaction Summary

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

Install 3 Packages

Total download size: 7.4 M

Installed size: 31 M

Downloading Packages:

(1/3): postgresql13-libs-13.1-1PGDG.rhel8.x86\_6 82 kB/s | 410 kB 00:05

(2/3): postgresql13-13.1-1PGDG.rhel8.x86\_64.rpm 134 kB/s | 1.4 MB 00:10

(3/3): postgresql13-server-13.1-1PGDG.rhel8.x86 276 kB/s | 5.6 MB 00:20

--------------------------------------------------------------------------------

Total 365 kB/s | 7.4 MB 00:20

warning: /var/cache/dnf/pgdg13-e81daebfc8b779ec/packages/postgresql13-13.1-1PGDG.rhel8.x86\_64.rpm: Header V4 DSA/SHA1 Signature, key ID 442df0f8: NOKEY

PostgreSQL 13 for RHEL/CentOS 8 - x86\_64 1.6 MB/s | 1.7 kB 00:00

Importing GPG key 0x442DF0F8:

Userid : "PostgreSQL RPM Building Project <pgsqlrpms-hackers@pgfoundry.org>"

Fingerprint: 68C9 E2B9 1A37 D136 FE74 D176 1F16 D2E1 442D F0F8

From : /etc/pki/rpm-gpg/RPM-GPG-KEY-PGDG

Key imported successfully

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing : 1/1

Installing : postgresql13-libs-13.1-1PGDG.rhel8.x86\_64 1/3

Running scriptlet: postgresql13-libs-13.1-1PGDG.rhel8.x86\_64 1/3

Installing : postgresql13-13.1-1PGDG.rhel8.x86\_64 2/3

Running scriptlet: postgresql13-13.1-1PGDG.rhel8.x86\_64 2/3

Running scriptlet: postgresql13-server-13.1-1PGDG.rhel8.x86\_64 3/3

Installing : postgresql13-server-13.1-1PGDG.rhel8.x86\_64 3/3

Running scriptlet: postgresql13-server-13.1-1PGDG.rhel8.x86\_64 3/3

Verifying : postgresql13-13.1-1PGDG.rhel8.x86\_64 1/3

Verifying : postgresql13-libs-13.1-1PGDG.rhel8.x86\_64 2/3

Verifying : postgresql13-server-13.1-1PGDG.rhel8.x86\_64 3/3

Installed:

postgresql13-13.1-1PGDG.rhel8.x86\_64

postgresql13-libs-13.1-1PGDG.rhel8.x86\_64

postgresql13-server-13.1-1PGDG.rhel8.x86\_64

Complete!

You need to execute following command once to initialize the Postgres database.

[root@postgres-01 ~]# /usr/pgsql-13/bin/postgresql-13-setup initdb

Initializing database ... OK

Enable and start Postgres database service.

[root@postgres-01 ~]# systemctl enable --now postgresql-13

Created symlink /etc/systemd/system/multi-user.target.wants/postgresql-13.service â /usr/lib/systemd/system/postgresql-13.service.

Check the status of Postgres database service.

[root@postgres-01 ~]# systemctl status postgresql-13.service

â postgresql-13.service - PostgreSQL 13 database server

Loaded: loaded (/usr/lib/systemd/system/postgresql-13.service; enabled; vend>

Active: active (running) since Sun 2020-11-22 20:32:04 PKT; 24s ago

Docs: https://www.postgresql.org/docs/13/static/

Process: 1598 ExecStartPre=/usr/pgsql-13/bin/postgresql-13-check-db-dir ${PGD>

Main PID: 1603 (postmaster)

Tasks: 8 (limit: 5879)

Memory: 17.5M

CGroup: /system.slice/postgresql-13.service

ââ1603 /usr/pgsql-13/bin/postmaster -D /var/lib/pgsql/13/data/

ââ1605 postgres: logger

ââ1607 postgres: checkpointer

ââ1608 postgres: background writer

ââ1609 postgres: walwriter

ââ1610 postgres: autovacuum launcher

ââ1611 postgres: stats collector

ââ1612 postgres: logical replication launcher

Nov 22 20:32:04 postgres-01.centlinux.com systemd[1]: Starting PostgreSQL 13 da>

Nov 22 20:32:04 postgres-01.centlinux.com postmaster[1603]: 2020-11-22 20:32:04>

Nov 22 20:32:04 postgres-01.centlinux.com postmaster[1603]: 2020-11-22 20:32:04>

Nov 22 20:32:04 postgres-01.centlinux.com systemd[1]: Started PostgreSQL 13 dat>

Check the version of installed PostgreSQL server.

[root@postgres-01 ~]# psql -V

psql (PostgreSQL) 13.1

Switch to **postgres** user and connect to **psql** shell to set admin user password.

[root@postgres-01 ~]# su - postgres

[postgres@postgres-01 ~]$ psql

psql (13.1)

Type "help" for help.

postgres=# ALTER USER postgres WITH PASSWORD '123';

ALTER ROLE

postgres=# \q

[postgres@postgres-01 ~]$ exit

logout

**Configure Postgres Database Service for Network Access:**

PostgreSQL default service port is **5432/tcp**. Execute the following command to verify that PostgreSQL service is listening on this port.

[root@postgres-01 ~]# ss -tulpn | grep 5432

tcp LISTEN 0 128 127.0.0.1:5432 0.0.0.0:\* users:(("postmaster",pid=1603,fd=7))

tcp LISTEN 0 128 [::1]:5432 [::]:\* users:(("postmaster",pid=1603,fd=6))

As you may notice that the PostgreSQL service is initially runs on localhost interface only. However, to make our Postgres database available for network clients, you have to configure PostgreSQL to run on all (or on some specific) interfaces.

For this purpose, you have to edit PostgreSQL configuration file in **vim** text editor.

[root@postgres-01 ~]# vi /var/lib/pgsql/13/data/postgresql.conf

Search for following directive therein.

# listen\_addresses = 'localhost'

And replace it with the following directive.

listen\_addresses = '\*'

Your Postgres database service is now configured to listen on all network interfaces.

Allow network clients to access PostgreSQL service in **pg\_hba.conf** file.

[root@postgres-01 ~]# echo "host all all 192.168.116.0/24 md5" >> /var/lib/pgsql/13/data/pg\_hba.conf

Restart Postgres database service to apply changes.

[root@postgres-01 ~]# systemctl restart postgresql-13.service

Again check the network services by using **ss** command.

[root@postgres-01 ~]# ss -tulpn | grep 5432

tcp LISTEN 0 128 0.0.0.0:5432 0.0.0.0:\* users:(("postmaster",pid=1781,fd=6))

tcp LISTEN 0 128 [::]:5432 [::]:\* users:(("postmaster",pid=1781,fd=7))

Postgres Service is now running on all network interfaces.

**Configure Linux Firewall for Postgres Database Service:**

In CentOS 8, we already have a firewall service for PostgreSQL, therefore we are allowing it by using firewall-cmd command. However, you can [create a custom firewall service](https://www.centlinux.com/2019/01/3-ways-create-custom-firewalld-service-centos-7.html) of your own, if your PostgreSQL server is configured to listen on a non-default port.

[root@postgres-01 ~]# firewall-cmd --permanent --add-service=postgresql

success

[root@postgres-01 ~]# firewall-cmd --reload

success

Postgres database server has been installed on Linux server.

**Installing pgAdmin Yum Repository on CentOS 8:**

**pgAdmin** is a popular web interface for database administration of PostgreSQL databases.

Although pgAdmin is also provided within PostgreSQL official yum repositories. But it doesn’t work on our CentOS 8 server.

Therefore, we are installing the latest stable version of **pgAdmin** from pgAdmin official yum repository.

For this purpose, first you need to remove the PostgreSQL yum repositories from your Linux server.

[root@postgres-01 ~]# dnf remove -y pgdg-redhat-repo

Dependencies resolved.

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

Package Architecture Version Repository Size

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

Removing:

pgdg-redhat-repo noarch 42.0-14 @@commandline 11 k

Transaction Summary

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

Remove 1 Package

Freed space: 11 k

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing : 1/1

Erasing : pgdg-redhat-repo-42.0-14.noarch 1/1

Verifying : pgdg-redhat-repo-42.0-14.noarch 1/1

Removed:

pgdg-redhat-repo-42.0-14.noarch

Complete!

Now, add the pgAdmin official yum repository in your Linux operating system.

[root@postgres-01 ~]# dnf install -y https://ftp.postgresql.org/pub/pgadmin/pgadmin4/yum/pgadmin4-redhat-repo-1-1.noarch.rpm

Last metadata expiration check: 1 day, 1:30:15 ago on Sun 22 Nov 2020 08:26:21 PM PKT.

pgadmin4-redhat-repo-1-1.noarch.rpm 1.2 kB/s | 6.6 kB 00:05

Dependencies resolved.

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

Package Architecture Version Repository Size

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

Installing:

pgadmin4-redhat-repo noarch 1-1 @commandline 6.6 k

Transaction Summary

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

Install 1 Package

Total size: 6.6 k

Installed size: 4.0 k

Downloading Packages:

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing : 1/1

Installing : pgadmin4-redhat-repo-1-1.noarch 1/1

Verifying : pgadmin4-redhat-repo-1-1.noarch 1/1

Installed:

pgadmin4-redhat-repo-1-1.noarch

Complete!

You are also required to install **EPEL (Extra Packages for Enterprise Linux)** yum repository, because pgAdmin requires some software packages that are not available in standard yum repositories.

Use dnf command and install EPEL yum repository.

[root@postgres-01 ~]# dnf install -y epel-release

Last metadata expiration check: 0:01:00 ago on Mon 23 Nov 2020 09:57:16 PM PKT.

Dependencies resolved.

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

Package Architecture Version Repository Size

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

Installing:

epel-release noarch 8-8.el8 extras 23 k

Transaction Summary

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

Install 1 Package

Total download size: 23 k

Installed size: 32 k

Downloading Packages:

epel-release-8-8.el8.noarch.rpm 31 kB/s | 23 kB 00:00

--------------------------------------------------------------------------------

Total 13 kB/s | 23 kB 00:01

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing : 1/1

Installing : epel-release-8-8.el8.noarch 1/1

Running scriptlet: epel-release-8-8.el8.noarch 1/1

Verifying : epel-release-8-8.el8.noarch 1/1

Installed:

epel-release-8-8.el8.noarch

Complete!

Build cache for newly installed yum repositories.

[root@postgres-01 ~]# dnf makecache

CentOS-8 - AppStream 2.1 kB/s | 4.3 kB 00:02

CentOS-8 - Base 3.7 kB/s | 3.9 kB 00:01

CentOS-8 - Extras 768 B/s | 1.5 kB 00:02

Extra Packages for Enterprise Linux Modular 8 - 24 kB/s | 98 kB 00:04

Extra Packages for Enterprise Linux 8 - x86\_64 92 kB/s | 8.3 MB 01:32

pgadmin4 1.1 kB/s | 2.9 kB 00:02

Metadata cache created.

**Installing pgAdmin Web UI on CentOS 8:**

All repositories has been added. You can now install pgAdmin by using **dnf** command.

[root@postgres-01 ~]# dnf install -y pgadmin4

...

Installed:

apr-1.6.3-9.el8.x86\_64

apr-util-1.6.1-6.el8.x86\_64

apr-util-bdb-1.6.1-6.el8.x86\_64

apr-util-openssl-1.6.1-6.el8.x86\_64

avahi-libs-0.7-19.el8.x86\_64

centos-logos-httpd-80.5-2.el8.noarch

cups-libs-1:2.2.6-33.el8.x86\_64

dejavu-fonts-common-2.35-6.el8.noarch

dejavu-sans-fonts-2.35-6.el8.noarch

fontconfig-2.13.1-3.el8.x86\_64

fontpackages-filesystem-1.44-22.el8.noarch

glx-utils-8.4.0-4.20181118git1830dcb.el8.x86\_64

graphite2-1.3.10-10.el8.x86\_64

harfbuzz-1.7.5-3.el8.x86\_64

httpd-2.4.37-21.module\_el8.2.0+494+1df74eae.x86\_64

httpd-filesystem-2.4.37-21.module\_el8.2.0+494+1df74eae.noarch

httpd-tools-2.4.37-21.module\_el8.2.0+494+1df74eae.x86\_64

libICE-1.0.9-15.el8.x86\_64

libSM-1.2.3-1.el8.x86\_64

libX11-1.6.8-3.el8.x86\_64

libX11-common-1.6.8-3.el8.noarch

libX11-xcb-1.6.8-3.el8.x86\_64

libXau-1.0.8-13.el8.x86\_64

libXdamage-1.1.4-14.el8.x86\_64

libXext-1.3.3-9.el8.x86\_64

libXfixes-5.0.3-7.el8.x86\_64

libXrender-0.9.10-7.el8.x86\_64

libXxf86vm-1.1.4-9.el8.x86\_64

libglvnd-1:1.2.0-6.el8.x86\_64

libglvnd-egl-1:1.2.0-6.el8.x86\_64

libglvnd-glx-1:1.2.0-6.el8.x86\_64

libjpeg-turbo-1.5.3-10.el8.x86\_64

libwayland-client-1.17.0-1.el8.x86\_64

libwayland-server-1.17.0-1.el8.x86\_64

libxcb-1.13.1-1.el8.x86\_64

libxshmfence-1.3-2.el8.x86\_64

mailcap-2.1.48-3.el8.noarch

mesa-libEGL-19.3.4-2.el8.x86\_64

mesa-libGL-19.3.4-2.el8.x86\_64

mesa-libgbm-19.3.4-2.el8.x86\_64

mesa-libglapi-19.3.4-2.el8.x86\_64

mod\_http2-1.11.3-3.module\_el8.2.0+486+c01050f0.1.x86\_64

pcre2-utf16-10.32-1.el8.x86\_64

pgadmin4-4.28-1.el8.noarch

pgadmin4-desktop-4.28-1.el8.x86\_64

pgadmin4-server-4.28-1.el8.x86\_64

pgadmin4-web-4.28-1.el8.noarch

python3-mod\_wsgi-4.6.4-4.el8.x86\_64

python3-pip-9.0.3-16.el8.noarch

python3-setuptools-39.2.0-5.el8.noarch

python36-3.6.8-2.module\_el8.1.0+245+c39af44f.x86\_64

qt5-qtbase-5.12.5-4.el8.x86\_64

qt5-qtbase-common-5.12.5-4.el8.noarch

qt5-qtbase-gui-5.12.5-4.el8.x86\_64

xcb-util-0.4.0-10.el8.x86\_64

xcb-util-image-0.4.0-9.el8.x86\_64

xcb-util-keysyms-0.4.0-7.el8.x86\_64

xcb-util-renderutil-0.3.9-10.el8.x86\_64

xcb-util-wm-0.4.1-12.el8.x86\_64

Complete!

To configure **SELinux** policies, the pgAdmin setup script requires semanage command, which is provided in **policycoreutils-python-utils** packages. Therefore, you should install it before executing pgAdmin setup script.

[root@postgres-01 ~]# dnf install -y policycoreutils-python-utils

Last metadata expiration check: 0:12:43 ago on Mon 23 Nov 2020 10:12:00 PM PKT.

Dependencies resolved.

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

Package Arch Version Repo Size

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

Installing:

policycoreutils-python-utils

noarch 2.9-9.el8 BaseOS 251 k

Installing dependencies:

checkpolicy x86\_64 2.9-1.el8 BaseOS 348 k

python3-audit x86\_64 3.0-0.17.20191104git1c2f876.el8 BaseOS 86 k

python3-libsemanage x86\_64 2.9-2.el8 BaseOS 127 k

python3-policycoreutils noarch 2.9-9.el8 BaseOS 2.2 M

python3-setools x86\_64 4.2.2-2.el8 BaseOS 601 k

Transaction Summary

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

Install 6 Packages

Total download size: 3.6 M

Installed size: 11 M

Downloading Packages:

(1/6): python3-audit-3.0-0.17.20191104git1c2f87 27 kB/s | 86 kB 00:03

(2/6): policycoreutils-python-utils-2.9-9.el8.n 44 kB/s | 251 kB 00:05

(3/6): python3-libsemanage-2.9-2.el8.x86\_64.rpm 46 kB/s | 127 kB 00:02

(4/6): checkpolicy-2.9-1.el8.x86\_64.rpm 40 kB/s | 348 kB 00:08

(5/6): python3-setools-4.2.2-2.el8.x86\_64.rpm 85 kB/s | 601 kB 00:07

(6/6): python3-policycoreutils-2.9-9.el8.noarch 125 kB/s | 2.2 MB 00:18

--------------------------------------------------------------------------------

Total 147 kB/s | 3.6 MB 00:25

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing : 1/1

Installing : python3-setools-4.2.2-2.el8.x86\_64 1/6

Installing : python3-libsemanage-2.9-2.el8.x86\_64 2/6

Installing : python3-audit-3.0-0.17.20191104git1c2f876.el8.x86\_64 3/6

Installing : checkpolicy-2.9-1.el8.x86\_64 4/6

Installing : python3-policycoreutils-2.9-9.el8.noarch 5/6

Installing : policycoreutils-python-utils-2.9-9.el8.noarch 6/6

Running scriptlet: policycoreutils-python-utils-2.9-9.el8.noarch 6/6

Verifying : checkpolicy-2.9-1.el8.x86\_64 1/6

Verifying : policycoreutils-python-utils-2.9-9.el8.noarch 2/6

Verifying : python3-audit-3.0-0.17.20191104git1c2f876.el8.x86\_64 3/6

Verifying : python3-libsemanage-2.9-2.el8.x86\_64 4/6

Verifying : python3-policycoreutils-2.9-9.el8.noarch 5/6

Verifying : python3-setools-4.2.2-2.el8.x86\_64 6/6

Installed:

checkpolicy-2.9-1.el8.x86\_64

policycoreutils-python-utils-2.9-9.el8.noarch

python3-audit-3.0-0.17.20191104git1c2f876.el8.x86\_64

python3-libsemanage-2.9-2.el8.x86\_64

python3-policycoreutils-2.9-9.el8.noarch

python3-setools-4.2.2-2.el8.x86\_64

Complete!

The pgAdmin software comes with a well written configuration script to configure pgAdmin web service. Execute it to create a admin user, configure SELinux policies and Apache web server to deploy pgAdmin web service.

[root@postgres-01 ~]# /usr/pgadmin4/bin/setup-web.sh

Setting up pgAdmin 4 in web mode on a Redhat platform...

Creating configuration database...

NOTE: Configuring authentication for SERVER mode.

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

Email address: ahmer@postgres-01.centlinux.com

Password:

Retype password:

pgAdmin 4 - Application Initialisation

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

Creating storage and log directories...

Configuring SELinux...

The Apache web server is not running. We can enable and start the web server for you to finish pgAdmin 4 installation. Continue (y/n)? y

Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service â /usr/lib/systemd/system/httpd.service.

Apache successfully enabled.

Apache successfully started.

You can now start using pgAdmin 4 in web mode at http://127.0.0.1/pgadmin4

The above script installs a configuration file **(/etc/httpd/conf.d/pgadmin4.conf**) to deploy pgAdmin on Apache web server.

No customization is required in this file. You are only required to enable and start httpd service.

[root@postgres-01 ~]# systemctl enable --now httpd.service

Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service â /usr/lib/systemd/system/httpd.service.

Configure Linux firewall to allow incoming traffic to Apache web server.

[root@postgres-01 ~]# firewall-cmd --permanent --add-service=http

success

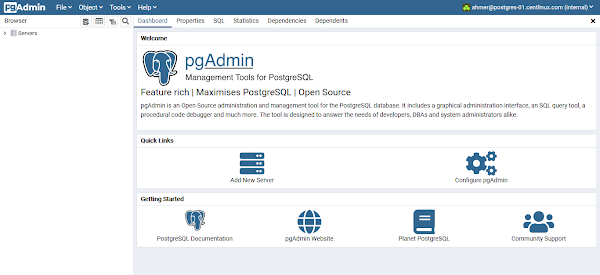
[root@postgres-01 ~]# firewall-cmd --reload

success

Open URL [**http://postgres-01.centlinux.com/pgadmin4/**](http://postgres-01.centlinux.com/pgadmin4/) in a web browser.

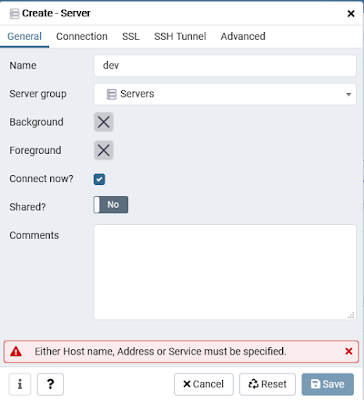
[](https://1.bp.blogspot.com/-IkoqOOw6MqQ/X70DLjUdvGI/AAAAAAAAJRc/mOhaIfWTXg8msbvnrwOCX3sPPxQNA006QCLcBGAsYHQ/s0/01-install-postgresql-pgadmin-on-centos-8-login.png)

Login to pgAdmin as admin user that we have created by **setup-web.sh** script.

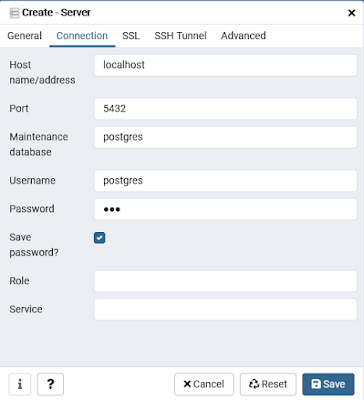
[](https://1.bp.blogspot.com/-1byZucpXBjs/X70DLxuKWMI/AAAAAAAAJRg/pRPIdROeDCoB-fuL72qN66L-1C21w4lBACLcBGAsYHQ/s0/02-install-postgresql-pgadmin-on-centos-8-dashboard.png)

After successful login, you may reach at the pgAdmin dashboard.

To add your PostgreSQL database server in pgAdmin inventory, click on **"Add New Server".**

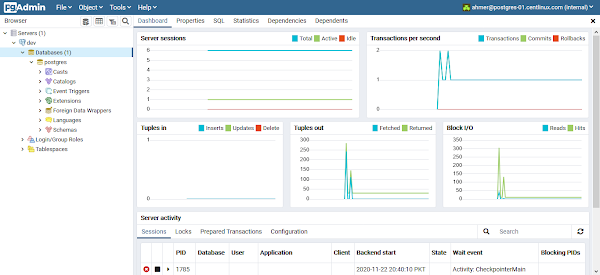
[](https://1.bp.blogspot.com/-DmMFZAEAmWQ/X70DLtcGDII/AAAAAAAAJRY/T1E480ue1901YLELCZOP83kHI4N3lD-HgCLcBGAsYHQ/s0/03-install-postgresql-pgadmin-on-centos-8-add-new-server-general.png)

Provide the Server name and click on **“Connection”** tab.

[](https://1.bp.blogspot.com/-O-zjl2Dz51U/X70DMIeB0yI/AAAAAAAAJRk/yDA-Abl2UO4dyrkNd5GVrolZAB0UGTr2QCLcBGAsYHQ/s0/04-install-postgresql-pgadmin-on-centos-8-add-new-server-connection.png)

Provide the database connection information in this dialog box as we did in the above screenshots.

Click on **"Save"**.

[](https://1.bp.blogspot.com/-bbw3l9ztE-8/X70DMWRPONI/AAAAAAAAJRs/EAoF78y6s6wFUnQm7YixFyydo0LnxOlcgCLcBGAsYHQ/s0/05-install-postgresql-pgadmin-on-centos-8-dashboard.png)

Our Postgres database server has been added in pgAdmin. You can see a tree of you database server in the left side panel.