|  |  |  |
| --- | --- | --- |
| ICANlogo.jpg | **GeoSpatial Implementation:**  Postgresql **only** installation on CentOS 7 & 8 | 320-10 |

# **How To Install PostgreSQL 12 (only) on CentOS 7 / CentOS 8**

( <https://computingforgeeks.com/how-to-install-postgresql-12-on-centos-7/> )

By Josphat Mutai June 15, 2020

This guide will walk you through the steps used to install PostgreSQL 12 on CentOS 7 / CentOS 8 Linux server?. PostgreSQL is an object-relational database management system based on POSTGRES 4.2. PostgreSQL 12 is available for Production use by Developers and Database Administrators.

PostgreSQL project provides a repository of packages of all supported versions for the most common distributions. Among the distributions supported are all Red Hat family of which includes CentOS, Fedora, Scientific Linux, Oracle Linux and Red Hat Enterprise Linux.

For Ubuntu users, check: [Install PostgreSQL 12 on Ubuntu](https://computingforgeeks.com/install-postgresql-12-on-ubuntu/)

Use the steps below to install PostgreSQL 12 on CentOS 8 / CentOS 7.

**Step 1: Add PostgreSQL Yum Repository to CentOS 7 / CentOS 8**

The PostgreSQL Yum Repository will integrate with your normal systems and patch management, and provide automatic updates for all supported versions of PostgreSQL throughout the support lifetime of PostgreSQL.

It can be added to CentOS system by running the command below:

**CentOS 8:**

sudo yum -y install <https://download.postgresql.org/pub/repos/yum/reporpms/EL-8-x86_64/pgdg-redhat-repo-latest.noarch.rpm>

**CentOS 7:**

sudo yum -y install <https://download.postgresql.org/pub/repos/yum/reporpms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm>

You can get more information on installed package by running the command:

$ rpm -qi pgdg-redhat-repo

Name : pgdg-redhat-repo

Version : 42.0

Release : 4

Architecture: noarch

Install Date: Thu 19 Sep 2019 06:34:53 PM UTC

Group : System Environment/Base

Size : 6915

License : PostgreSQL

Signature : DSA/SHA1, Wed 17 Apr 2019 04:12:42 AM UTC, Key ID 1f16d2e1442df0f8

Source RPM : pgdg-redhat-repo-42.0-4.src.rpm

Build Date : Wed 17 Apr 2019 04:12:41 AM UTC

Build Host : koji-centos7-x86-64-pgbuild

Relocations : (not relocatable)

Vendor : PostgreSQL Global Development Group

URL : <https://yum.postgresql.org>

Summary : PostgreSQL PGDG RPMs- Yum Repository Configuration for Red Hat / CentOS / Scientific Linux

Description :

This package contains yum configuration for Red Hat Enterprise Linux, CentOS

and Scientific Linux. and also the GPG key for PGDG RPMs.

**Step 2: Install PostgreSQL 12 on CentOS 8 / CentOS 7**

With the YUM repository added, we can install PostgreSQL 12 on CentOS 7/8 with the following command.

**PostgreSQL 12 on CentOS 8**

Disable the built-in PostgreSQL module:

sudo dnf -qy module disable postgresql

Then Install both client and server packages:

sudo dnf -y install postgresql12 postgresql12-server

**PostgreSQL 12 on CentOS 7**

Install PostgreSQL client and server packages:

sudo yum -y install epel-release yum-utils

sudo yum-config-manager --enable pgdg12

sudo yum install postgresql12-server postgresql12

***Sample installation output:***

Dependencies Resolved

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

Package Arch Version Repository Size

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

Installing:

postgresql12 x86\_64 12beta4-1PGDG.rhel7 pgdg12-testing 1.8 M

postgresql12-server x86\_64 12beta4-1PGDG.rhel7 pgdg12-testing 5.4 M

Installing for dependencies:

libicu x86\_64 50.2-3.el7 base 6.9 M

postgresql12-libs x86\_64 12beta4-1PGDG.rhel7 pgdg12-testing 383 k

python3 x86\_64 3.6.8-10.el7 base 69 k

python3-libs x86\_64 3.6.8-10.el7 base 7.0 M

python3-pip noarch 9.0.3-5.el7 base 1.8 M

python3-setuptools noarch 39.2.0-10.el7 base 629 k

Transaction Summary

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

Install 2 Packages (+6 Dependent packages)

Total download size: 24 M

Installed size: 104 M

Downloading packages:

(1/8): libicu-50.2-3.el7.x86\_64.rpm | 6.9 MB 00:00:00

warning: /var/cache/yum/x86\_64/7/pgdg12-testing/packages/postgresql12-libs-12beta4-1PGDG.rhel7.x86\_64.rpm: Header V4 DSA/SHA1 Signature, key ID 442df0f8: NOKEY

Public key for postgresql12-libs-12beta4-1PGDG.rhel7.x86\_64.rpm is not installed

(2/8): postgresql12-libs-12beta4-1PGDG.rhel7.x86\_64.rpm | 383 kB 00:00:00

(3/8): python3-3.6.8-10.el7.x86\_64.rpm | 69 kB 00:00:00

(4/8): python3-setuptools-39.2.0-10.el7.noarch.rpm | 629 kB 00:00:00

(5/8): postgresql12-12beta4-1PGDG.rhel7.x86\_64.rpm | 1.8 MB 00:00:00

(6/8): python3-libs-3.6.8-10.el7.x86\_64.rpm | 7.0 MB 00:00:00

(7/8): postgresql12-server-12beta4-1PGDG.rhel7.x86\_64.rpm | 5.4 MB 00:00:00

(8/8): python3-pip-9.0.3-5.el7.noarch.rpm | 1.8 MB 00:00:00

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

Total 15 MB/s | 24 MB 00:00:01

Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-PGDG

Importing GPG key 0x442DF0F8:

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

Fingerprint: 68c9 e2b9 1a37 d136 fe74 d176 1f16 d2e1 442d f0f8

Package : pgdg-redhat-repo-42.0-4.noarch (installed)

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

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

Installing : libicu-50.2-3.el7.x86\_64 1/8

Installing : postgresql12-libs-12beta4-1PGDG.rhel7.x86\_64 2/8

Installing : python3-libs-3.6.8-10.el7.x86\_64 3/8

Installing : python3-setuptools-39.2.0-10.el7.noarch 4/8

Installing : python3-3.6.8-10.el7.x86\_64 5/8

Installing : python3-pip-9.0.3-5.el7.noarch 6/8

Installing : postgresql12-12beta4-1PGDG.rhel7.x86\_64 7/8

Installing : postgresql12-server-12beta4-1PGDG.rhel7.x86\_64 8/8

Verifying : postgresql12-libs-12beta4-1PGDG.rhel7.x86\_64 1/8

Verifying : python3-pip-9.0.3-5.el7.noarch 2/8

Verifying : libicu-50.2-3.el7.x86\_64 3/8

Verifying : python3-libs-3.6.8-10.el7.x86\_64 4/8

Verifying : postgresql12-12beta4-1PGDG.rhel7.x86\_64 5/8

Verifying : postgresql12-server-12beta4-1PGDG.rhel7.x86\_64 6/8

Verifying : python3-setuptools-39.2.0-10.el7.noarch 7/8

Verifying : python3-3.6.8-10.el7.x86\_64 8/8

Installed:

postgresql12.x86\_64 0:12beta4-1PGDG.rhel7 postgresql12-server.x86\_64 0:12beta4-1PGDG.rhel7

Dependency Installed:

libicu.x86\_64 0:50.2-3.el7 postgresql12-libs.x86\_64 0:12beta4-1PGDG.rhel7 python3.x86\_64 0:3.6.8-10.el7 python3-libs.x86\_64 0:3.6.8-10.el7

python3-pip.noarch 0:9.0.3-5.el7 python3-setuptools.noarch 0:39.2.0-10.el7

**Step 3: Initialize and start database service**

***Tac insert:***

If data directory is different from the default in ***/usr/lib/system/system/postgresql-12.service***, need to add an override file (drop-in file):

* Create drop-in directory ***postgresql-12.service.d*** at ***/etc/systemd/system***
* Insert a file ***anyName.conf*** containing

[Service]

Environment=PGDATA={data location}

* Issue ***systemctl daemon-reload*** to reload changed files

After installation, database initialization is required before service can be started.

***sudo /usr/pgsql-12/bin/postgresql-12-setup initdb –D {path to data folder}***  (if data folder different from default)

The database main configuration ifile is written to: */var/lib/pgsql/12/data/postgresql.conf (unless moved)*

Start and enable the database server service.

***sudo systemctl enable --now postgresql-12***

Confirm that the service is started without any errors.

$ **systemctl status postgresql-12**

● postgresql-12.service - PostgreSQL 12 database server

Loaded: loaded (/usr/lib/systemd/system/postgresql-12.service; enabled; vendor preset: disabled)

Active: active (running) since Thu 2019-09-19 18:50:10 UTC; 39s ago

Docs: <https://www.postgresql.org/docs/12/static/>

Process: 10647 ExecStartPre=/usr/pgsql-12/bin/postgresql-12-check-db-dir ${PGDATA} (code=exited, status=0/SUCCESS)

Main PID: 10652 (postmaster)

CGroup: /system.slice/postgresql-12.service

├─10652 /usr/pgsql-12/bin/postmaster -D /var/lib/pgsql/12/data/

├─10654 postgres: logger

├─10656 postgres: checkpointer

├─10657 postgres: background writer

├─10658 postgres: walwriter

├─10659 postgres: autovacuum launcher

├─10660 postgres: stats collector

└─10661 postgres: logical replication launcher

Sep 19 18:50:10 cent7.novalocal systemd[1]: Starting PostgreSQL 12 database server...

Sep 19 18:50:10 cent7.novalocal postmaster[10652]: 2019-09-19 18:50:10.207 UTC [10652] LOG: starting PostgreSQL 12beta4 on x86\_64-pc-lin... 64-bit

Sep 19 18:50:10 cent7.novalocal postmaster[10652]: 2019-09-19 18:50:10.209 UTC [10652] LOG: listening on IPv6 address "::1", port 5432

Sep 19 18:50:10 cent7.novalocal postmaster[10652]: 2019-09-19 18:50:10.209 UTC [10652] LOG: listening on IPv4 address "127.0.0.1", port 5432

Sep 19 18:50:10 cent7.novalocal postmaster[10652]: 2019-09-19 18:50:10.214 UTC [10652] LOG: listening on Unix socket "/var/run/postgresq...L.5432"

Sep 19 18:50:10 cent7.novalocal postmaster[10652]: 2019-09-19 18:50:10.229 UTC [10652] LOG: listening on Unix socket "/tmp/.s.PGSQL.5432"

Sep 19 18:50:10 cent7.novalocal postmaster[10652]: 2019-09-19 18:50:10.254 UTC [10652] LOG: redirecting log output to logging collector process

Sep 19 18:50:10 cent7.novalocal postmaster[10652]: 2019-09-19 18:50:10.254 UTC [10652] HINT: Future log output will appear in directory "log".

Sep 19 18:50:10 cent7.novalocal systemd[1]: Started PostgreSQL 12 database server.

Hint: Some lines were ellipsized, use -l to show in full.

If you have a running Firewall service and remote clients should connect to your database server, allow PostgreSQL service.

sudo firewall-cmd --add-service=postgresql --permanent

sudo firewall-cmd --reload

**Step 4: Set PostgreSQL admin user’s password**

Set PostgreSQL admin user

$ sudo su - postgres

~]$ psql -c "alter user postgres with password 'StrongPassword'"

ALTER ROLE

**Step 5: Enable remote access (Optional)**

Edit the file /var/lib/pgsql/12/data/postgresql.conf and set Listen address to your server IP address or “**\***” for all interfaces.

listen\_addresses = '192.168.10.10'

Also set PostgreSQL to accept remote connections

$ sudo vim /var/lib/pgsql/12/data/pg\_hba.conf

# Accept from anywhere

host all all 0.0.0.0/0 md5

# Accept from truqsudo /varsted subnet

host all all 192.168.18.0/24 md5

Restart database service after committing the change.

sudo systemctl restart postgresql-12

Connecting to remote database:

$ psql -U <dbuser> -h <serverip> -p 5432 <dbname>

**Step 6: Install pgAdmin 4 Web interface**

pgAdmin is the leading Open Source feature-rich PostgreSQL administration and development platform that runs on Linux, Unix, Mac OS X, and Windows. Here is the link for the installation of pgAdmin4 on CentOS.q

[How To Install pgAdmin 4 on CentOS 8 Linux](https://computingforgeeks.com/how-to-install-pgadmin-4-on-centos-linux/)

[Install pgAdmin4 on CentOS 7](https://computingforgeeks.com/how-to-install-pgadmin-on-centos-fedora/)

PostgreSQL 12 has been installed on CentOS 7 / CentOS 8 Linux system. Spare some minutes to go through [PostgreSQL 12 documentation](https://www.postgresql.org/docs/12/index.html).