VPS March 19, 2019 4min Read Edward S.

How to Install PostgreSQL on CentOS 7

PostgreSQL is a powerful and open source relational database platform. It's scalable across multiple platforms and is a widely used and well-loved tool. In this tutorial, we'll show you how to install PostgreSQL on CentOS 7!

How Does PostgreSQL work?

PostgreSQL uses SQL for relational queries and JSON for non-relational queries. One advantage of using PostgreSQL is its immense community support! Its open source nature means that a lot of developers contribute to this utility's growth.

PostgreSQL is scalable, reliable and accompanied by advanced optimization features. In most cases, people assume advanced optimization and data types are supported by only commercial databases like **SQL Server** and Oracle. Contrary to this belief, PostgreSQL provides all of this and many more advanced features, definitely making it a worthwhile addition to your **VPS**.

PostgreSQL is extremely simple to start using and master.

Here we will demonstrate how to install PostgreSQL on CentOS 7.

You can install PostgreSQL on CentOS 7 in using one of two methods:

1. Install PostgreSQL from existing CentOS repositories
2. Install from the PostgreSQL repository

Let us check each of this method in more detail:

First Method – Install PostgreSQL on CentOS 7 using the CentOS repositories

The CentOS 7 repository contains PostgreSQL. Note that it may not have the latest version of PostgreSQL. At the time of writing the repository hosts PostgreSQL version 9.2.15.

1. Access Your Server

Remember, before starting to install PostgreSQL on CentOS 7, we need to access our VPS server with SSH. Check out our **PuTTY tutorial** if you're having trouble.

following command:

```
sudo yum install postgresql-server postgresql-contrib
```

This might take some time to complete.

3. Initialize the Database

Once the installation is done, you can initialize the database using the below command:

```
sudo postgresql-setup initdb
```

4. Start the Database

After initializing the database, you can start the database using:

```
sudo systemctl start postgresql
```

5. (Optional) Enable PostgreSQL

This completes our database installation and initialization. If required you can configure PostgreSQL to start on every system reboot automatically.

```
sudo systemctl enable postgresql
```

Second Method – Install PostgreSQL on CentOS 7 using the PostgreSQL repository

With this method, you can download any latest version of PostgreSQL from the official website. You can select the PostgreSQL version from the PostgreSQL yum repositories. Currently, this shows PostgreSQL version 9.6 as the latest version.

1. Access Your Server

As with the previous method, first we have to access our server using SSH.

2. Download PostgreSQL Using Wget

Now we will download PostgreSQL version 9.6.3. Similarly, you can download any version. This can be done using the wget command.

```
wget https://download.postgresql.org/pub/repos/yum/9.6/redhat/rhel-7-
x86_64/pgdg-centos96-9.6-3.noarch.rpm
```

This also requires RPM or **RedHat Package manager** along with EPEL (Extra Packages Enterprise Linux) repositories. This is required for additional PostgreSQL dependencies.

3. Install PosgreSQL on CentOS 7 with the Downloaded Package

This can be installed using the below command:

4. Update Yum

Update yum so that your changes get reflected. Use the below command for this:

```
sudo yum update
```

5. Complete the PostgreSQL Install Process for CentOS 7

This completes our prerequisites for PostgreSQL installation. Next, we can install this using the below command:

```
sudo yum install postgresql96-server postgresql96-contrib
```

This completes our PostgreSQL installation.

6. Initialize the Database

Next, you can initialize the database using:

```
sudo /usr/pgsql-9.6/bin/postgresql96-setup initdb
```

7. Restart PostgreSQL

You can restart PostgreSQL using:

```
sudo systemctl start postgresql-9.6
```

8. (Optional) Enable PostgreSQL Launch on Reboot

In case you want PostgreSQL to start at system reboot automatically then you can optionally use the below command:

```
sudo systemctl enable postgresql-9.6
```

PostgreSQL Basic Setup

In Linux by default, a user named **postgres** is created once PostgreSQL is installed. You can change the user's password with the following command:

```
sudo passwd postgres
```

You will be prompted to enter the new password twice.

Next, you can switch to the PostgreSQL prompt and change the password for the PostgreSQL postgres user using:

```
su - postgres
```

If you receive an error, you can set a valid shell on the user with the following command:

```
su - postgres
```

To change the password, use the below command where you add your new password instead of the **NewPassword**:

```
psql -d template1 -c "ALTER USER postgres WITH PASSWORD 'NewPassword';"
```

You can switch to the PostgreSQL client shell using:

```
psql postgres
```

Here you can check the list of available commands by typing **\h**. You can use **\h** followed by the command for which you need more information. To exit the environment you can type **\q**.

The createdb command lets you create new databases. Suppose we want to create a new database named **testDB** using the **postgres** Linux user. The command we would use would look like this:

```
createdb testDB
```

You can create a new role using the createuser command. Below is an example where we are creating a role named samplerole using the postgres Linux user.

```
createuser samplerole –pwprompt
```

Here you will be prompted to set a password for the user.

Optionally you can assign the ownership of our newly created database to a specific postgres user or role. This can be done with a command like this one:

```
createdb testDB -O samplerole
```

In the above command, replace samplerole with the role you want to use.

You can connect to this new database using the command bellow:

```
psql testDB
```

In case you want to use a specific user or role to log in, use the command as shown below:

```
psql testDB -U samplerole
```

This will prompt you to enter the password.

You can use **\l** or **\list** commands to show all the databases. To know the current database you're using, you can use **\c**. In case you want more information about connections such as the socket, port, etc., then you can use cannot be retrieved.

To delete a database, you can use:

```
dropdb testDB
```

PostgreSQL similar to other databases allows:

- Table creation
- Table deletion
- Table Updates
- Column Addition
- Drop column
- Query a table
- Alter commands
- Grant Privileges

The syntax for all of these commands is similar to most database commands. You can list all the tables by using the **\dt** command. To list all roles, you use the **\du** command. To learn more, we encourage you to read the **official documentation**!

Conclusion

With this, we have shown you PostgreSQL installation in CentOS 7 and also walked you through the basic PostgreSQL setup. There are several things that can still be learned in PostgreSQL. This being a starting point, from here you can start exploring more about PostgreSQL commands.

The Author

Edward S. / **@edvardasstabinskas**

Edward is Hostinger's copywriter. He's an expert communicator with years of experience in IT as a writer, marketer, and Linux enthusiast. IT is a core pillar of his life, personal and professional. Edward's goal is to encourage millions to achieve an impactful online presence. He also really loves dogs, guitars, and everything related to space.

Featured tutorials

How to Make a Website – The All-in-One Guide

Read More →

21 Aug • WORDPRESS

How to Start a Blog – The Complete Guide

Read More →

12 Jul • DOMAIN

How to Buy a Domain Name – A Simple Guide

Read More →

Related tutorials

02 Dec • VPS

How to Make a Minecraft Server – The Complete Tutorial

29 Nov • VPS

Iptables Tutorial – Securing Ubuntu VPS with Linux Firewall

21 Nov • SSH VPS

How to Install and Use Nano Text Editor: A Beginner's Tutorial

Read More →

Leave a reply

Comment*

Enter your comment

Name*

Enter your name

Email*

Enter your email

□ By using this form you agree with the storage and handling of your data by this website. * (You need to accept this checkbox.)

This site uses Akismet to reduce spam. **Learn how your comment data is processed**.

HOSTING

Web Hosting

VPS Hosting

Cloud Hosting

WordPress Hosting

Email Hosting

CMS Hosting

Ecommerce Hosting

Free Website Hosting

Website Builder

Cheap Web Hosting

Buy Hosting

Windows VPS

Minecraft Server Hosting

Web Design Services

DOMAINS

Domain Checker

Domain Transfer

Free Domain

XYZ Domain

Cheap SSL Certificate

Buy Domain Name

Whois Checker

Domain Name Search

99 Cent Domains

Free SSL Certificate

INFORMATION

Server Status

Affiliate Program

Payment Methods

Wall of Fame

Compare

Reviews

Student Discount

Pricing

YouTube Creators

HELP

Report Abuse

Knowledge Base

Tutorials

Blog

COMPANY

About Hostinger

Contact Us

Our Technology

Career

LEGAL

Privacy Policy

Terms of Service

TUTORIALS