The master/slave database replication is a process of copying (syncing) data from a database on one server (the master) to a database on another server (the slaves). The main benefit of this process is to distribute databases to multiple machines, so when the master server has a problem, there is a backup machine with same data available for handling requests without interruption.

PostgreSQL provides several ways to replicate a database. It can be used for backup purposes and to provide a high availability database server. Here we will see how to install and configure PostgreSQL replication by using hot standby mode. Hot standby mode is easy to configure.

Hot standby mode requires 2 database servers, we will use Ubuntu as operating system on both servers.

1. **Master Server** - accepts connections from the client with read and write permissions.
2. **Slave Server** - the standby server runs copy of the data from the master server with read-only permission.

**Prerequisites**

* 2 Ubuntu servers - 1 for master and 1 for slave.
* Root privileges on the servers.

**Step 1 - Install PostgreSQL on Master and Slave Server**

* update the Ubuntu repository

**sudo apt-get update**

* install PostgreSQL with all its dependencies

**sudo apt-get install postgresql postgresql-client postgresql-contrib**

* Give a new password for postgres user

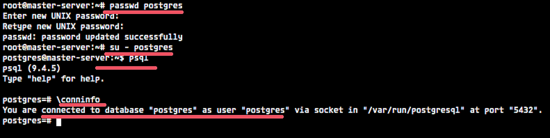
**passwd postgres**

* Testing the PostgreSQL

**su - postgres**

**psql**

**\conninfo**



## Step 2 - Configure Master-server

We will configure the **'master server'** with IP address **<IP-M>**. We will create a new user/role with special permission to perform the replication, then we edit the PostgreSQL configuration file to enable the hot standby replication mode.

* Switch to the PostgreSQL user

**su - postgres**

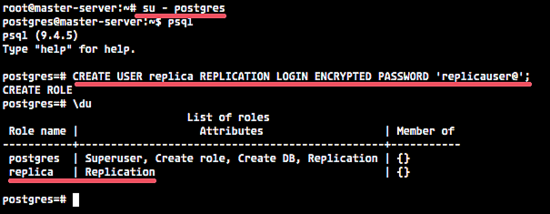
* Access the Postgres shell with the psql command and type in this PostgreSQL query to create the new user/role:

**psql**

**CREATE USER replica REPLICATION LOGIN ENCRYPTED PASSWORD 'replicauser@';**

* Check new replica user with PostgreSQL

**\du**



* Go to the PostgreSQL directory '/etc/postgresql/9.5/main' to edit the configuration file

**vi /etc/postgresql/9.5/main/postgresql.conf**

* + update

listen\_addresses = '0.0.0.0'

wal\_level = hot\_standby

archive\_mode = on

* + add

**archive\_command = 'cp -i %p /var/lib/postgresql/9.5/main/archive/%f'**

* + update

**max\_wal\_senders = 3**

**wal\_keep\_segments = 8**

* Create a new directory inside of the 'main' directory for the archive configuration - run the command below as postgres user:

**mkdir -p /var/lib/9.5/main/archive/**

* Edit pg\_hba.conf file to allow the replication connection:

**vi /etc/postgresql/9.5/main/pg\_hba.conf**

* In the end of the line, add a new configuration for user 'replica' to make the connection.

**host replication replica <IP-S>/24 md5**

## Step 3 - Slave-server Configuration

* Configure the slave server like the master server. Use su to become the postgres user and go to the PostgreSQL configuration directory & edit the postgresql.conf.

**su - postgres**

**vi /etc/postgresql/9.5/main/postgresql.conf**

* + Update

**listen\_addresses = '0.0.0.0'**

**wal\_level = hot\_standby**

**max\_wal\_senders = 3**

**wal\_keep\_segments = 8**

**hot\_standby = on**

## Step 4 - Synchronize Data from Master server to Slave server

* Stop PostgreSQL on the slave server:

**systemctl stop postgresql**

* Login to the postgres user and rename the 'main' directory to 'main\_original' as a backup.

**su - postgres**

**mv 9.5/main 9.5/main\_original**

* Run the command below to copy data from the master server to slave server:

**pg\_basebackup -h <IP-M> -D /var/lib/postgresql/9.5/main -U replica -v –P**



* Go to the new 'main' directory and create the new recovery file 'recovery.conf'

**vi /var/lib/postgresql/9.4/main/recovery.conf**

* Add below line in the recovery.conf file:

**standby\_mode = 'on'**

**primary\_conninfo = 'host=<IP-M> port=5432 user=replica password=replicauser@'**

**restore\_command = 'cp //var/lib/postgresql/9.5/main/archive/%f %p'**

**trigger\_file = '/tmp/postgresql.trigger.5432'**

* Back to the root user with exit and start PostgreSQL with systemctl command:

**exit**

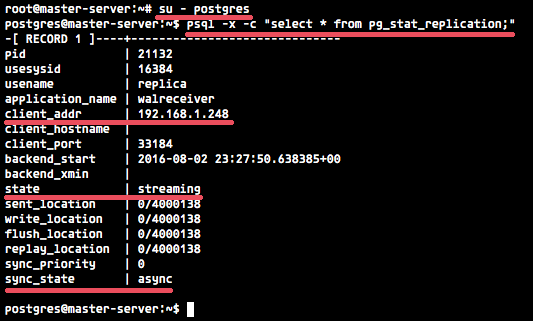
**systemctl start postgresql**

## Step 5 – Testing

* Go to the master server and log into the postgres user, then run the command below to see the replication info.

**su - postgres**

**psql -x -c "select \* from pg\_stat\_replication;"**

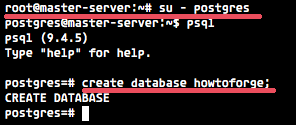


* create a new database from the master server and then check that the database exist on the slave server.

**su - postgres**

**psql**

**create database howtoforge;**



* Login to the slave server and check that the 'howtoforge' database has been mirrored to the slave server automatically.

**su - postgres**

**psql**

**\list**

