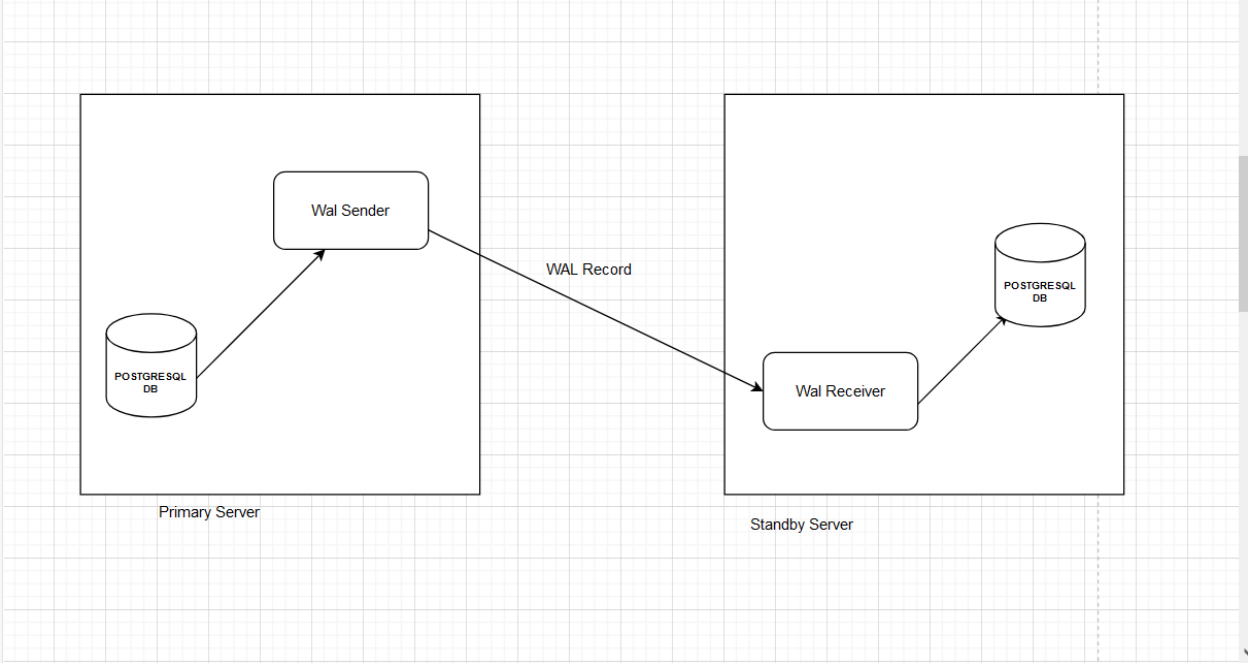

POSTGRESQL Streaming Replication

Primary Server is postgresql write-read server and Standby Server is read-only postgresql server which also terms as DR(Disaster Recovery),where WAL(Write ahead logfile) or Archived logfile will be getting apply from Primary Server

Step-by-Step Guide for PostgreSQL Streaming Replication Setup

1. **Configure the Primary Server (testserver1)**

Edit postgresql.conf

Configure the primary server to allow replication by modifying the following parameters in the PostgreSQL configuration file (postgresql.conf):

listen_addresses = '*'

wal_level = replica

max_wal_senders = 5

wal_keep_segments = 32

hot_standby = on

- listen_addresses = '*': Listen for connections from any IP address.
- wal_level = replica: Set WAL (Write-Ahead Logging) level to replica.
- max_wal_senders = 5: Maximum number of concurrent WAL sender processes.
- wal_keep_segments = 32: Number of WAL segments to retain.
- hot_standby = on: Allow standby queries.

Edit pg_hba.conf

Add a line in pg_hba.conf to allow the replication user to connect from the standby server (testserver2).

host replication replicator 52.187.165.22/32 md5

Replace 52.187.165.22 with the actual IP of the standby server.

Create Replication User

Open the PostgreSQL shell on testserver1 and create a replication user:

CREATE USER replicator WITH REPLICATION PASSWORD 'replicator_password';

Restart PostgreSQL Service

After making changes to the configuration files, restart the PostgreSQL service:

Step 2: Prepare Standby Server (testserver2)

1. **Stop PostgreSQL Service**

Stop the PostgreSQL service on the standby server (testserver2):

Clear Data Directory

Navigate to the PostgreSQL data directory and remove its contents:

C:\Program Files\PostgreSQL\14\data

Step 3: Take a Base Backup from Primary Server

On the standby server (testserver2), use pg_basebackup to take a base backup from the primary server:

1. Open a command prompt and run the following command:

pg_basebackup -h 13.67.60.129 -D "C:\Program Files\PostgreSQL\14\data" -U replicator -P -v -R -X stream -C -S pgstandby1

- -h 13.67.60.129: IP address of the primary server (testserver1).
- -D "C:\Program Files\PostgreSQL\14\data": Destination directory for the backup.
- -U replicator: Replication user.
- -P: Show progress.
- -v: Enable verbose mode.
- -R: Write recovery configuration (creates standby.signal automatically).
- -X stream: Stream WAL during the backup.
- -C -S pgstandby1: Create a replication slot on the primary named pgstandby1.

If the backup is successful, you will see output similar to this:

pg_basebackup: base backup completed

Step 4: Configure Standby Server

1. **Edit postgresql.conf**

On the standby server, update the postgresql.conf file to configure the replication settings:

primary_conninfo = 'user=replicator password=replicator_password host=13.67.60.129 port=5432 sslmode=prefer'

primary_slot_name = 'pgstandby1'

- primary_conninfo: Connection information for the primary server.

- primary_slot_name: Replication slot name created on the primary server.

Restart PostgreSQL Service

Restart the PostgreSQL service on the standby server:

Step 5: Verify Replication

1. **Check Replication Status on Primary Server**

On the primary server (testserver1), you can check the replication status using:

psql -U postgres -c "SELECT * FROM pg_stat_replication;"

You should see an entry for the standby server in the output. Look for the following columns:

- client_addr: IP of the standby server.
- state: Should show streaming.

Example output:

pid | usesysid | usename | application_name | client_addr | client_hostname | client_port | backend_start | backend_xmin | state | sent_lsn | write_lsn | flush_lsn | replay_lsn | write_lag | flush_lag | replay_lag | sync_priority | sync_state | reply_time

------+----------+-----------+-----------------+--------------+----------------+-------------+-------------------------------+-------------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+-------------------------------

7920 | 16410 | replicator | walreceiver | 52.187.165.22 | | 54893 | 2024-10-02 17:28:16.45404+00 | | streaming | 0/5000060 | 0/5000060 | 0/5000060 | 0/5000060 | | | | 0 | async | 2024-10-02 17:28:46.577806+00

Check Replication Status on Standby Server

On the standby server (testserver2), you can check the WAL receiver status:

psql -U postgres -c "SELECT * FROM pg_stat_wal_receiver;"

If replication is set up correctly, you should see output confirming the WAL receiver’s activity.