

PostgreSQL 17

Incremental backup

Ubuntu under Hyper-V

using pg_basebackup and WAL archive replay

<https://www.linkedin.com/in/mariyanclement/>

1. PostgreSQL 17 Installation

```
sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg main" >  
/etc/apt/sources.list.d/pgdg.list'
```

```
wget -qO - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo tee  
/etc/apt/trusted.gpg.d/postgresql.asc
```

```
sudo apt update
```

```
sudo apt install -y postgresql-17
```

Verify installation:

```
psql --version # PostgreSQL 17.x
```

2. Post-Installation Setup (New Ubuntu VM)

Ensure PostgreSQL executables are accessible:

```
nano ~/.bashrc
```

Scroll to the bottom and add:

```
export PATH=$PATH:/usr/lib/postgresql/17/bin
```

Save and exit (Ctrl+O, Enter, then Ctrl+X in nano).

Apply changes immediately:

```
source ~/.bashrc
```

3. Configuration Directory Setup

PostgreSQL config files are in:

`/etc/postgresql/17/main/`

Do not modify config files in `/var/lib/postgresql/17/main` (data dir).

Create symbolic links instead of copying to avoid future maintenance issues:

```
sudo ln -s /etc/postgresql/17/main/postgresql.conf /var/lib/postgresql/17/main/postgresql.conf
```

```
sudo ln -s /etc/postgresql/17/main/pg_hba.conf /var/lib/postgresql/17/main/pg_hba.conf
```

```
sudo ln -s /etc/postgresql/17/main/pg_ident.conf /var/lib/postgresql/17/main/pg_ident.conf
```

4. Setup WAL Archiving

Edit the config:

```
sudo nano /etc/postgresql/17/main/postgresql.conf
```

Uncomment and update:

```
wal_level = replica
```

```
archive_mode = on
```

```
archive_command = 'test ! -f /var/lib/postgresql/17/archive/%f && cp %p /var/lib/postgresql/17/archive/%f'
```

Also # comment below line

```
#include_dir = 'conf.d'
```

Then:

```
sudo mkdir -p /var/lib/postgresql/17/archive
```

```
sudo chown postgres:postgres /var/lib/postgresql/17/archive
```

```
sudo chmod 700 /var/lib/postgresql/17/archive
```

Restart PostgreSQL:

```
/usr/lib/postgresql/17/bin/pg_ctl -D /var/lib/postgresql/17/main restart
```

5. Create Sample Demo Data

As postgres user:

```
psql
```

```
CREATE DATABASE demodb;
```

```
\c demodb
```

```
CREATE TABLE evidence (  
    id SERIAL PRIMARY KEY,  
    action TEXT,  
    created_at TIMESTAMP DEFAULT now()  
);
```

```
INSERT INTO evidence (action) VALUES ('Base data - before backup');
```

6. Perform Base Backup

Create backup destination:

```
sudo mkdir /var/backups/pg_base  
sudo chown postgres:postgres /var/backups/pg_base
```

Then run:

```
sudo -u postgres pg_basebackup -D /var/backups/pg_base -F tar -X fetch -P
```

Unpack the backup:

```
sudo mkdir /var/backups/pgdata_restored  
cd /var/backups/pgdata_restored  
sudo tar -xf /var/backups/pg_base/base.tar
```

Create symlinks to config: (assuming current dir /var/backups/pgdata_restored)

```
sudo ln -s /etc/postgresql/17/main/postgresql.conf postgresql.conf  
sudo ln -s /etc/postgresql/17/main/pg_hba.conf pg_hba.conf  
sudo ln -s /etc/postgresql/17/main/pg_ident.conf pg_ident.conf
```

7. Insert More Data After Backup

Back in the original cluster:

```
INSERT INTO evidence (action) VALUES ('Post-backup WAL entry 1');
```

```
INSERT INTO evidence (action) VALUES ('Post-backup WAL entry 2');
```

Verify data:

```
SELECT * FROM evidence ORDER BY id;
```


8. Restore and Replay WAL

Stop PostgreSQL:

```
/usr/lib/postgresql/17/bin/pg_ctl -D /var/lib/postgresql/17/main stop
```

Rename original:

```
sudo mv /var/lib/postgresql/17/main /var/lib/postgresql/17/main.bak
```

Move restored data:

```
sudo mv /var/backups/pgdata_restored /var/lib/postgresql/17/main
```

```
sudo chown -R postgres:postgres /var/lib/postgresql/17/main
```

```
sudo chmod 700 /var/lib/postgresql/17/main
```

Create recovery.signal:

```
sudo -u postgres touch /var/lib/postgresql/17/main/recovery.signal
```

Edit postgresql.conf (linked already) to add:

```
sudo nano /etc/postgresql/17/main/postgresql.conf
```

```
restore_command = 'cp /var/lib/postgresql/17/archive/%f %p'
```

Start PostgreSQL:

```
/usr/lib/postgresql/17/bin/pg_ctl -D /var/lib/postgresql/17/main start
```

9. Verify WAL Recovery

```
psql -d demodb
```

```
SELECT * FROM evidence ORDER BY id;
```

You should see:

- Base data
- WAL replayed entries

1. /etc/postgresql/17/main/

Purpose: Contains configuration files for the PostgreSQL 17 main cluster.

Common Files:

postgresql.conf – main server configuration.

pg_hba.conf – client authentication rules.

pg_ident.conf – user mapping.

start.conf – cluster startup control.

This is the primary configuration directory, often symlinked or referenced during upgrades and automation.

2. /var/lib/postgresql/17/main/

Purpose: This is the PostgreSQL data directory where all database files are stored.

Includes:

Data for system and user databases.

WAL files, base/, pg_wal/, pg_stat/, etc.

Consider this the heart of your PostgreSQL instance – backing up this is essential!

3. /usr/lib/postgresql/17/bin/

Purpose: Contains PostgreSQL server binaries.

Examples:

postgres – the actual server binary.

initdb, pg_ctl, pg_basebackup, pg_dump, psql, etc.

These are versioned binaries – useful if you manage multiple PostgreSQL versions side-by-side.

4. /usr/lib/postgresql/17/lib/

Purpose: Contains shared libraries (.so files) used by PostgreSQL 17 server.

Examples:

Internal plugins like adminpack, pg_stat_statements, or custom C extensions.

5. /lib/postgresql/17/lib/

Purpose: Similar to /usr/lib/postgresql/17/lib/, it may contain additional or system-level shared libraries for PostgreSQL.

Note: On some systems, this may be a symlink or remain unused depending on packaging.

6. /lib/postgresql/17/bin/

Unusual/Optional: Normally not used by default packaging.

Might appear:

Due to manual builds or path misconfigurations.

If a third-party package placed binaries here by mistake.

7. /var/log/postgresql/

Default log file directory (if enabled).

Controlled via postgresql.conf → log_directory.

8. /etc/init.d/postgresql or /lib/systemd/system/postgresql.service

Startup scripts or systemd unit files to manage PostgreSQL as a service.

9. /usr/share/postgresql/17/

Contains default configuration templates and extension SQL files (e.g., for CREATE EXTENSION).