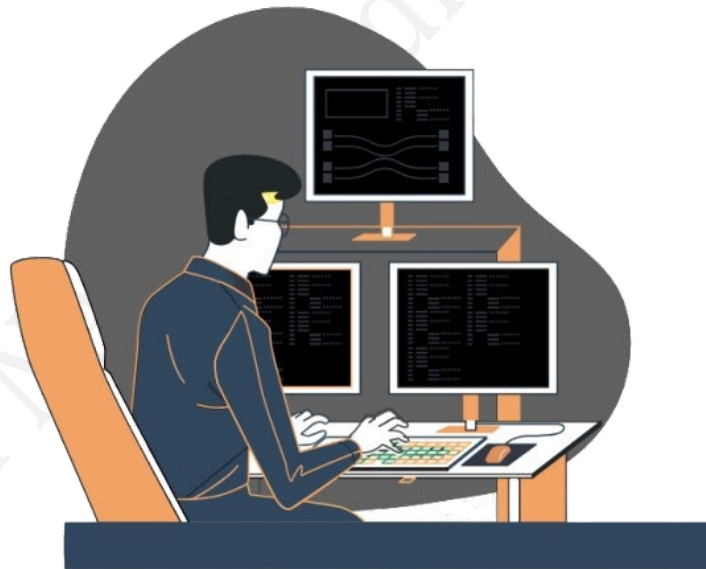
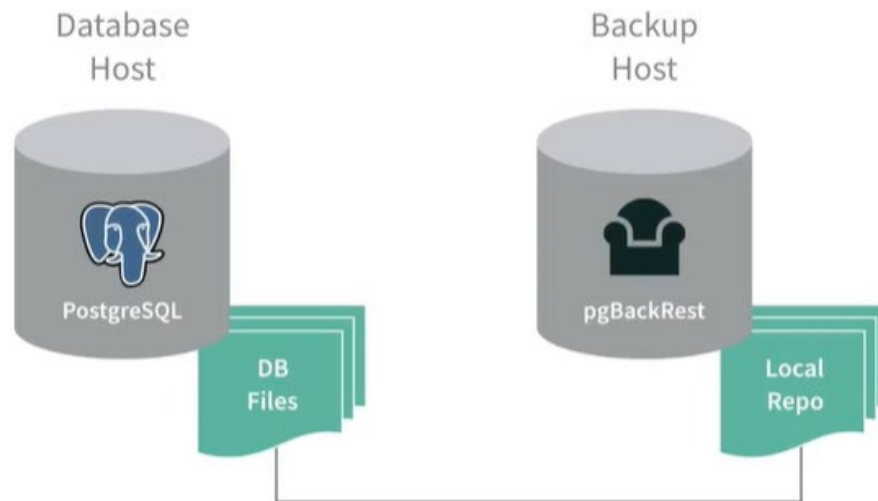


# PostgreSQL Backup and Restore Using pgBackRest



## Standard Operating Procedure (SOP) for PostgreSQL Backup & Restore using pgBackRest.



### 1. Introduction

pgBackRest is a robust backup and restore solution for PostgreSQL that supports full, differential, and incremental backups, point-in-time recovery (PITR), multi-repository storage, and advanced retention policies. This documentation outlines the complete setup of pgBackRest in both single and remote host environments, covering installation, configuration, backup scheduling, restoration, and PITR scenarios using both timestamps and LSNs.

### 2. Overview

#### 2.1 What is pgBackRest & Why this is use instead of other tools ?

**pgBackRest** is a reliable, efficient and secure backup and restore solution for PostgreSQL databases. It is widely used in production environments to ensure data integrity, support point-in-time recovery (PITR) and manage large-scale PostgreSQL deployments with ease. It is an open-source tool designed to address limitations of traditional PostgreSQL backup methods (like `pg_basebackup`, `pg_dump` or `pg_restore` etc.)

## 2.2 Architecture Overview

- **pgBackRest Command-Line Tool:** Central CLI used to run backups, restores, WAL archiving, etc.
- **Repository Host (Backup Server):** Stores backup files, WAL archives and manages retention policies and metadata.
- **Database Host:** PostgreSQL database nodes to be backed up.
- **Stanza:** A named configuration context that links a PostgreSQL instance to a backup repository. Supports multiple stanzas for multiple clusters.

## 2.3 Backup Types:

Full – Captures entire database

Differential – Based on last full backup

Incremental – Based on last backup of any type

## 2.4 WAL Archiving

pgBackRest captures and archives Write-Ahead Log (WAL) files to support PITR (Point-in-Time Recovery).

## 3. Pre-requisite

### 3.1 Install PostgreSQL 14

*yum update -y*

*yum install -y postgresql14-server*

```
[root@postgresql-test ~]# dnf install -y postgresql14-server
Last metadata expiration check: 0:00:08 ago on Tue 13 May 2025 12:10:50 PM +06.
Dependencies resolved.
=====
Package                               Architecture      Version           Repository        Size
=====
Installing:
postgresql14-server                  x86_64            14.18-1PGDG.rhel8 pgdg14            5.7 M
Installing dependencies:
libicu                               x86_64            60.3-2.el8_1     baseos            8.8 M
postgresql14                         x86_64            14.18-1PGDG.rhel8 pgdg14            1.6 M
postgresql14-libs                   x86_64            14.18-1PGDG.rhel8 pgdg14            283 k
Transaction Summary
-----
Install 4 Packages

Total download size: 16 M
Installed size: 64 M
Downloading Packages:
(1/4): postgresql14-libs-14.18-1PGDG.rhel8.x86_64.rpm      621 kB/s | 283 kB  00:00
(2/4): libicu-60.3-2.el8_1.x86_64.rpm                     7.6 MB/s | 8.8 MB  00:01
(3/4): postgresql14-14.18-1PGDG.rhel8.x86_64.rpm          1.0 MB/s | 1.6 MB  00:01
(4/4): postgresql14-server-14.18-1PGDG.rhel8.x86_64.rpm    1.9 MB/s | 5.7 MB  00:02
-----
Total                                                     3.7 MB/s | 16 MB  00:04
PostgreSQL 14 for RHEL / Rocky / AlmaLinux 8 - x86_64
Importing GPG key 0x08B40D20:
 Userid : "PostgreSQL RPM Repository <pgsql-pkg-yum@lists.postgresql.org>"
 Fingerprint: D4BF 0BAE 67A0 B4C7 A10B CC02 40BC A2B4 08B4 0D20
 From : /etc/pki/rpm-gpg/PGDG-RPM-GPG-KEY-RHEL
Key imported successfully
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction:
Preparing :                                                     1/1
Installing : postgresql14-libs-14.18-1PGDG.rhel8.x86_64      1/4
Running scriptlet: postgresql14-libs-14.18-1PGDG.rhel8.x86_64 1/4
Installing : libicu-60.3-2.el8_1.x86_64                      2/4
Running scriptlet: libicu-60.3-2.el8_1.x86_64                  2/4
Installing : postgresql14-14.18-1PGDG.rhel8.x86_64           3/4
Running scriptlet: postgresql14-14.18-1PGDG.rhel8.x86_64     3/4
Running scriptlet: postgresql14-server-14.18-1PGDG.rhel8.x86_64 4/4
Installing : postgresql14-server-14.18-1PGDG.rhel8.x86_64   4/4
Running scriptlet: postgresql14-server-14.18-1PGDG.rhel8.x86_64 4/4
Verifying : libicu-60.3-2.el8_1.x86_64                       1/4
Verifying : postgresql14-14.18-1PGDG.rhel8.x86_64           2/4
Verifying : postgresql14-libs-14.18-1PGDG.rhel8.x86_64      3/4
Verifying : postgresql14-server-14.18-1PGDG.rhel8.x86_64   4/4

Installed:
libicu-60.3-2.el8_1.x86_64      postgresql14-14.18-1PGDG.rhel8.x86_64  postgresql14-libs-14.18-1PGDG.rhel8.x86_64
postgresql14-server-14.18-1PGDG.rhel8.x86_64
```

### 3.2 Check the status & enable, start of postgresql14 after installing

*systemctl enable postgresql-14*  
*systemctl start postgresql-14*  
*systemctl status postgresql-14*

```
[root@pgBackRest_Implement ~]# systemctl enable postgresql-14
Created symlink /etc/systemd/system/multi-user.target.wants/postgresql-14.service - /usr/lib/systemd/system/postgresql-14.service.
[root@pgBackRest_Implement ~]# systemctl start postgresql-14
[root@pgBackRest_Implement ~]# systemctl status postgresql-14
● postgresql-14.service - PostgreSQL 14 database server
   Loaded: loaded (/usr/lib/systemd/system/postgresql-14.service; enabled; vendor preset: disabled)
   Active: active (running) since Wed 2025-05-21 14:05:16 +06; 5s ago
     Docs: https://www.postgresql.org/docs/14/static/
   Process: 1919 ExecStartPre=/usr/pgsql-14/bin/postgresql-14-check-db-dir ${PGDATA} (code=exited, status=0/SUCCESS)
    Main PID: 1924 (postmaster)
      Tasks: 8 (limit: 23197)
     Memory: 16.8M
    CGroup: /system.slice/postgresql-14.service
            └─1924 /usr/pgsql-14/bin/postmaster -D /var/lib/pgsql/14/data/
              └─1926 postgres: logger
                └─1928 postgres: checkpoint writer
                  └─1929 postgres: background writer
                    └─1930 postgres: walwriter
                      └─1931 postgres: autovacuum launcher
                        └─1932 postgres: stats collector
                          └─1933 postgres: logical replication launcher

May 21 14:05:16 pgBackRest_Implement systemd[1]: Starting PostgreSQL 14 database server...
May 21 14:05:16 pgBackRest_Implement postmaster[1924]: 2025-05-21 14:05:16.706 +06 [1924] LOG:  redirecting log output to logging collector process
May 21 14:05:16 pgBackRest_Implement postmaster[1924]: 2025-05-21 14:05:16.706 +06 [1924] HINT:  Future log output will appear in directory "log".
May 21 14:05:16 pgBackRest_Implement systemd[1]: Started PostgreSQL 14 database server.
```

### 3.3 Dependency to install pgBackRest

*yum install epel-release -y*

```
[root@pgBackRest_Implement ~]# yum install epel-release -y
Last metadata expiration check: 0:15:26 ago on Wed 21 May 2025 01:38:22 PM +06.
Dependencies resolved.

Package Architecture Version Repository Size
Installing:
epel-release noarch 8-21.el8 extras 24 k

Transaction Summary
-----
Install 1 Package

Total download size: 24 k
Installed size: 34 k
Downloading Packages:
epel-release-8-21.el8.noarch.rpm 258 kB/s | 24 kB 00:00
-----
Total 22 kB/s | 24 kB 00:01
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      : 
  Installing     : epel-release-8-21.el8.noarch 1/1
  Running scriptlet: epel-release-8-21.el8.noarch 1/1
Many EPEL packages require the CodeReady Builder (CRB) repository.
It is recommended that you run /usr/bin/crb enable to enable the CRB repository.

  Verifying      : epel-release-8-21.el8.noarch 1/1

Installed:
  epel-release-8-21.el8.noarch

Complete!
[root@pgBackRest_Implement ~]#
```

### yum install vim wget -y

```
[root@pgBackRest Implement ~]# yum install vim wget -y
Extra Packages for Enterprise Linux 8 - x86_64
Last metadata expiration check: 0:00:09 ago on Wed 21 May 2025 01:55:35 PM +06.
Dependencies resolved.

=====
Package                                Architecture      Version            Repository          Size
=====
Installing:
vim-enhanced                           x86_64             2:8.0.1763-19.el8_6.4  appstream           1.4 M
wget                                    x86_64             1.19.5-12.el8_10      appstream           733 k
Installing dependencies:
gpm-libs                               x86_64             1.20.7-17.el8         appstream           38 k
libmetalink                             x86_64             0.1.3-7.el8           baseos              32 k
vim-common                              x86_64             2:8.0.1763-19.el8_6.4  appstream           6.3 M
vim-filesystem                          noarch             2:8.0.1763-19.el8_6.4  appstream           49 k
=====
Transaction Summary
=====
Install 6 Packages

Total download size: 8.5 M
Installed size: 33 M
Downloading Packages:
(1/6): gpm-libs-1.20.7-17.el8.x86_64.rpm 485 kB/s | 38 kB 00:00
(2/6): libmetalink-0.1.3-7.el8.x86_64.rpm 392 kB/s | 32 kB 00:00
(3/6): vim-filesystem-8.0.1763-19.el8_6.4.noarch.rpm 668 kB/s | 49 kB 00:00
(4/6): wget-1.19.5-12.el8_10.x86_64.rpm 3.4 MB/s | 733 kB 00:00
(5/6): vim-enhanced-8.0.1763-19.el8_6.4.x86_64.rpm 4.2 MB/s | 1.4 MB 00:00
(6/6): vim-common-8.0.1763-19.el8_6.4.x86_64.rpm 7.4 MB/s | 6.3 MB 00:00
-----
Total                                     3.0 MB/s | 8.5 MB 00:02
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing : vim-filesystem-2:8.0.1763-19.el8_6.4.noarch 1/1
  Installing : vim-common-2:8.0.1763-19.el8_6.4.x86_64 1/6
  Installing : gpm-libs-1.20.7-17.el8.x86_64 2/6
  Installing : gpm-libs-1.20.7-17.el8.x86_64 3/6
  Running scriptlet: gpm-libs-1.20.7-17.el8.x86_64 3/6
  Installing : libmetalink-0.1.3-7.el8.x86_64 4/6
  Installing : wget-1.19.5-12.el8_10.x86_64 5/6
  Running scriptlet: wget-1.19.5-12.el8_10.x86_64 5/6
```

## 3.4 Install pgBackRest

### yum install pgbackrest -y

```
[root@postgres-test ~]# yum install pgbackrest -y
Last metadata expiration check: 0:00:26 ago on Tue 13 May 2025 12:46:55 PM +06.
Dependencies resolved.

=====
Package                                Architecture      Version            Repository          Size
=====
Installing:
pgbackrest                             x86_64            2.55.1-1PGDG.rhel8  pgdg-common         468 k
=====
Transaction Summary
=====
Install 1 Package

Total download size: 468 k
Installed size: 1.1 M
Downloading Packages:
pgbackrest-2.55.1-1PGDG.rhel8.x86_64.rpm 1.0 MB/s | 468 kB 00:00
-----
Total                                     1.0 MB/s | 468 kB 00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing : pgbackrest-2.55.1-1PGDG.rhel8.x86_64 1/1
  Running scriptlet: pgbackrest-2.55.1-1PGDG.rhel8.x86_64 1/1
  Installing : pgbackrest-2.55.1-1PGDG.rhel8.x86_64 1/1
  Running scriptlet: pgbackrest-2.55.1-1PGDG.rhel8.x86_64 1/1
  Verifying : pgbackrest-2.55.1-1PGDG.rhel8.x86_64 1/1
Installed:
pgbackrest-2.55.1-1PGDG.rhel8.x86_64
Complete!
```

## 3.5 Configuration the pgBackRest (Database Server)


```
mkdir -p /data/backups
chown postgres:postgres /data/backups
chown postgres:postgres /var/log/pgbackrest
mkdir /etc/pgbackrest
mv /etc/pgbackrest.conf /etc/pgbackrest/
```

```
chown postgres:postgres /etc/pgbackrest/pgbackrest.conf
chmod 640 /etc/pgbackrest/pgbackrest.conf
su - postgres
cd /etc/pgbackrest/
ls -lrth
vim pgbackrest.conf
```

```
[root@pgBackRest_Implement ~]# mkdir -p /data/backups
[root@pgBackRest_Implement ~]# chown postgres:postgres /data/backups
[root@pgBackRest_Implement ~]# chown postgres:postgres /var/log/pgbackrest
[root@pgBackRest_Implement ~]# mkdir /etc/pgbackrest
[root@pgBackRest_Implement ~]# mv /etc/pgbackrest.conf /etc/pgbackrest/
[root@pgBackRest_Implement ~]# chown postgres:postgres /etc/pgbackrest/pgbackrest.conf
[root@pgBackRest_Implement ~]# chmod 640 /etc/pgbackrest/pgbackrest.conf
[root@pgBackRest_Implement ~]# su - postgres
Last login: Wed May 21 14:11:14 +06 2025 on pts/0
[postgres@pgBackRest_Implement ~]$ cd /etc/pgbackrest/
[postgres@pgBackRest_Implement pgbackrest]$ ls -lrth
total 4.0K
-rw-r----- 1 postgres postgres 82 May  6 00:30 pgbackrest.conf
[postgres@pgBackRest_Implement pgbackrest]$ vim pgbackrest.conf
```

Insert the configuration text

```
[global]
repo1-path=/data/backups
repo1-host-user=postgres
log-level-file=detail
log-level-console=detail
start-fast=y
[demo]
pg1-path=/var/lib/pgsql/14/data/
pg1-host-user=postgres
```

 root@pgBackRest\_Implement:/var/lib/pgsql/14/data

```
[global]
repo1-path=/data/backups
repo1-host-user=postgres
log-level-file=info
log-level-console=info
start-fast=y

[demo]
pg1-port=5432
pg1-path=/var/lib/pgsql/14/data/
pg1-host-user=postgres
```

### 3.6 Configuration the pgBackRest (Backup Server) [If Backup server will be separated, then this config will be needed otherwise not]

Install all required same as Database server and Insert the below conf text in pgbackrest.conf file. Here is the path: /etc/pgbackrest

```
[global]
repo1-path=/data/backups
log-level-file=detail
start-fast=y

[demo]
pg1-host=192.168.56.102
pg1-path=/var/lib/pgsql/14/data/
pg1-host-user=postgres
```

## 4. Stanza Concept

### 4.1 What is Stanza and why is this need in pgBackRest ?

Stanza is a metadata which is required by pbackrest to perform backups and restores on the database of the database server. So, Once everything has been said, like creation of the configuration files and directories, we need to have the password less SSH and all that configuration has been completed. Then we need to proceed with configuring the stanza the metadata of which is required for pgBackRest to take the backups.

### 4.2 PostgreSQL Config change before create stanza

Change the below config parameter in the postgresql.conf file. Here is the location:

```
vim /var/lib/pgsql/14/data/postgresql.conf
archive_mode = on
archive_command = 'pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo archive-push %p'
```

```
# - Archiving -
archive_mode = on          # enables archiving; off, on, or always
                           # (change requires restart)
archive_command = 'pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo archive-push %p'
```

### 4.3 Restart the PostgreSQL to impact the config change

*systemctl restart postgresql-14*

### 4.4 Stanza Create

*su - postgres*

*pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail stanza-create*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail stanza-create
2025-05-21 15:52:13.050 P00 INFO: stanza-create command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=4218-07c96e9a --log-level-console=detail --log-level-file=info --pg1-path=/var/lib/pgsql/14/data/ --pg1-port=5432 --repo1-path=/data/backups --stanza=demo
2025-05-21 15:52:13.066 P00 INFO: stanza-create for stanza 'demo' on repo1
2025-05-21 15:52:13.088 P00 INFO: stanza-create command end: completed successfully (39ms)
```

### 4.5 Stanza creation check

*pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail check*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail check
2025-05-21 16:05:06.959 P00 INFO: check command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=4432-7ade2fa9 --log-level-console=detail --log-level-file=info --pg1-path=/var/lib/pgsql/14/data/ --pg1-port=5432 --repo1-path=/data/backups --stanza=demo
2025-05-21 16:05:06.973 P00 INFO: check repo1 configuration (primary)
2025-05-21 16:05:07.003 P00 INFO: check repo1 archive for WAL (primary)
2025-05-21 16:05:07.606 P00 INFO: WAL segment 00000001000000000000000000000001 successfully archived to '/data/backups/archive/demo/14-1/0000000100000000/000000010000000000000001-2c55c0f0d25b375c4c6f95d3083781332d33e9e4.gz' on repo1
2025-05-21 16:05:07.606 P00 INFO: check command end: completed successfully (648ms)
```

### 4.6 If stanza config same server (DB & Backup Same Server)

For Here is the location: /etc/pgbackrest/pgbackrest.conf

*[global]*

*repo1-path=/data/backups*

*repo1-host-user=postgres*

*log-level-file=detail*

*start-fast=y*

*[demo]*

*pg1-port=5432*

*pg1-path=/var/lib/pgsql/14/data/*

*pg1-host-user=postgres*

For example, if we are taking full backup on Sunday and taking daily incremental backups, we need to



## 5. Take Full, Incremental, Differential Backup Concept

### 5.1 Full backup by using pgBackRest

For example, if we are taking full backup on Sunday and taking daily incremental backups, we need to perform PITR on Usingsrday on some timeline of Wednesday, then we need to restore backups of Sunday, Monday, Tuesday and Wednesday; time taken by incremental backups is way less than full backup and depends on how many transactions have taken place after the last backup.

```
pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail backup --type=full
```

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info backup --type=full
2025-05-21 16:41:09.495 P00 INFO: backup command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=5064-e9321dc4 --log-level-console=info --log-level-f
ile=info --pg1-path=/var/lib/pgsql/14/data/ --pg1-port=5432 --repol-path=/data/backups --stanza=demo --start-fast --type=full
2025-05-21 16:41:09.496 P00 WARN: option 'repol-retention-full' is not set for 'repol-retention-full-type=count', the repository may run out of space
HINT: to retain full backups indefinitely (without warning), set option 'repol-retention-full' to the maximum.
2025-05-21 16:41:09.513 P00 INFO: execute non-exclusive backup start: backup begins after the requested immediate checkpoint completes
2025-05-21 16:41:09.679 P00 INFO: backup start archive = 000000010000000000000005, lsn = 0/5000028
2025-05-21 16:41:09.679 P00 INFO: check archive for prior segments 000000010000000000000004
2025-05-21 16:41:10.011 P00 WARN: resumable backup 20250521-163940F of same type exists -- invalid files will be removed then the backup will resume
2025-05-21 16:41:12.042 P00 INFO: execute non-exclusive backup stop and wait for all WAL segments to archive
2025-05-21 16:41:12.079 P00 INFO: backup stop archive = 000000010000000000000005, lsn = 0/5000100
2025-05-21 16:41:12.100 P00 INFO: check archive for segment(s) 000000010000000000000005:000000010000000000000005
2025-05-21 16:41:12.447 P00 INFO: new backup label = 20250521-163940F
2025-05-21 16:41:12.505 P00 INFO: full backup size = 26.1MB, file total = 951
2025-05-21 16:41:12.505 P00 INFO: backup command end: completed successfully (3011ms)
2025-05-21 16:41:12.505 P00 INFO: expire command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=5064-e9321dc4 --log-level-console=info --log-level-f
ile=info --repol-path=/data/backups --stanza=demo
2025-05-21 16:41:12.507 P00 INFO: option 'repol-retention-archive' is not set - archive logs will not be expired
2025-05-21 16:41:12.507 P00 INFO: expire command end: completed successfully (2ms)
```

### 5.2 Scheduling Full Backup using pgBackRest

This cron job runs a full PostgreSQL backup using pgBackRest every day at 9:10 AM for the demo stanza, logging output in detail.

```
crontab -e
10 9 * * * /usr/bin/pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --
stanza=demo --log-level-console=detail backup --type=full
```

### 5.3 Check the pgBackRest log

```
cd /etc/pgbackrest/pgbackrest.log
cat demo-backup.log | grep 2025-05-13 09:10:00 | head
```

## 5.4 To get backup information

*pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail info*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail info
stanza: demo
  status: ok
  cipher: none

db (current)
  wal archive min/max (14): 000000010000000000000001/000000010000000000000005

  full backup: 20250521-163940F
    timestamp start/stop: 2025-05-21 16:41:09+06 / 2025-05-21 16:41:12+06
    wal start/stop: 000000010000000000000005 / 000000010000000000000005
    database size: 26.1MB, database backup size: 26.1MB
    repol: backup set size: 3.3MB, backup size: 3.3MB
```

*pgbackrest info*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest info
stanza: demo
  status: ok
  cipher: none

db (current)
  wal archive min/max (14): 000000010000000000000001/000000010000000000000005

  full backup: 20250521-163940F
    timestamp start/stop: 2025-05-21 16:41:09+06 / 2025-05-21 16:41:12+06
    wal start/stop: 000000010000000000000005 / 000000010000000000000005
    database size: 26.1MB, database backup size: 26.1MB
    repol: backup set size: 3.3MB, backup size: 3.3MB
```

## 5.5 Incremental backup using pgBackRest

*pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail backup --type=incr*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info backup --type=incr
2025-05-21 16:52:32.989 P00 INFO: backup command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=5242-3572ad09 --log-level-console=info --log-level-file=info --pg1-path=/var/lib/pgsql/14/data/ --pg1-port=5432 --repol-path=/data/backups --stanza=demo --start-fast --type=incr
2025-05-21 16:52:32.989 P00 WARN: option 'repol-retention-full' is not set for 'repol-retention-full-type=count', the repository may run out of space
HINT: to retain full backups indefinitely (without warning), set option 'repol-retention-full' to the maximum.
2025-05-21 16:52:33.008 P00 INFO: last backup label = 20250521-163940F_20250521-165222I, version = 2.55.1
2025-05-21 16:52:33.009 P00 INFO: execute non-exclusive backup start: Backup begins after the requested immediate checkpoint completes
2025-05-21 16:52:33.046 P00 INFO: backup start archive = 000000010000000000000008, lsn = 0/8000028
2025-05-21 16:52:33.046 P00 INFO: check archive for segment 000000010000000000000008
2025-05-21 16:52:34.152 P00 INFO: execute non-exclusive backup stop and wait for all WAL segments to archive
2025-05-21 16:52:34.166 P00 INFO: backup stop archive = 000000010000000000000008, lsn = 0/8000050
2025-05-21 16:52:34.212 P00 INFO: check archive for segment(s) 000000010000000000000008:000000010000000000000008
2025-05-21 16:52:34.544 P00 INFO: new backup label = 20250521-163940F_20250521-165233I
2025-05-21 16:52:34.602 P00 INFO: incr backup size = 16.6KB, file total = 951
2025-05-21 16:52:34.602 P00 INFO: backup command end: completed successfully (1613ms)
2025-05-21 16:52:34.602 P00 INFO: expire command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=5242-3572ad09 --log-level-console=info --log-level-file=info --repol-path=/data/backups --stanza=demo
2025-05-21 16:52:34.604 P00 INFO: option 'repol-retention-archive' is not set - archive logs will not be expired
2025-05-21 16:52:34.604 P00 INFO: expire command end: completed successfully (2ms)
```

## 5.6 Differential backup using pgBackRest

*pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail backup --type=diff*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info backup --type=diff
2025-05-21 16:53:17.456 P00 INFO: backup command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=5266-d5fa8bca --log-level-console=info --log-level-file=info --pg1-path=/var/lib/pgsql/14/data/ --pg1-port=5432 --repol-path=/data/backups --stanza=demo --start-fast --type=diff
2025-05-21 16:53:17.457 P00 WARN: option 'repol-retention-full' is not set for 'repol-retention-full-type=count', the repository may run out of space
HINT: to retain full backups indefinitely (without warning), set option 'repol-retention-full' to the maximum.
2025-05-21 16:53:17.496 P00 INFO: last backup label = 20250521-163940F, version = 2.55.1
2025-05-21 16:53:17.496 P00 INFO: execute non-exclusive backup start: backup begins after the requested immediate checkpoint completes
2025-05-21 16:53:17.748 P00 INFO: backup start archive = 00000001000000000000000B, lsn = 0/B000028
2025-05-21 16:53:17.748 P00 INFO: check archive for prior segment 00000001000000000000000A
2025-05-21 16:53:18.136 P00 INFO: execute non-exclusive backup stop and wait for all WAL segments to archive
2025-05-21 16:53:18.180 P00 INFO: backup stop archive = 00000001000000000000000B, lsn = 0/B000100
2025-05-21 16:53:18.218 P00 INFO: check archive for segment(s) 00000001000000000000000B:00000001000000000000000B
2025-05-21 16:53:18.360 P00 INFO: new backup label = 20250521-163940F_20250521-165317D
2025-05-21 16:53:18.416 P00 INFO: diff backup size = 18KB, file total = 951
2025-05-21 16:53:18.416 P00 INFO: backup command end: completed successfully (961ms)
2025-05-21 16:53:18.416 P00 INFO: expire command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=5266-d5fa8bca --log-level-console=info --log-level-file=info --repol-path=/data/backups --stanza=demo
2025-05-21 16:53:18.417 P00 INFO: option 'repol-retention-archive' is not set - archive logs will not be expired
2025-05-21 16:53:18.417 P00 INFO: expire command end: completed successfully (1ms)
```

## 5.7 Taken Full, Incremental, Differential backup are shown

```
[postgres@pgBackRest_Implement ~]$ pgbackrest info
stanza: demo
status: ok
cipher: none

db (current)
wal archive min/max (14): 00000001000000000000000000000000/0000000100000000000000000000000D

full backup: 20250521-163940F
timestamp start/stop: 2025-05-21 16:41:09+06 / 2025-05-21 16:41:12+06
wal start/stop: 00000001000000000000000000000005 / 00000001000000000000000000000005
database size: 26.1MB, database backup size: 26.1MB
repol: backup set size: 3.3MB, backup size: 3.3MB

diff backup: 20250521-163940F 20250521-165317D
timestamp start/stop: 2025-05-21 16:53:17+06 / 2025-05-21 16:53:18+06
wal start/stop: 0000000100000000000000000000000B / 0000000100000000000000000000000B
database size: 26.1MB, database backup size: 18KB
repol: backup set size: 3.3MB, backup size: 1.9KB
backup reference total: 1 full

incr backup: 20250521-163940F 20250521-165456I
timestamp start/stop: 2025-05-21 16:54:56+06 / 2025-05-21 16:54:58+06
wal start/stop: 0000000100000000000000000000000D / 0000000100000000000000000000000D
database size: 26.1MB, database backup size: 19.5KB
repol: backup set size: 3.3MB, backup size: 2KB
backup reference total: 1 full, 1 diff
```

## 6. Restore Database using pgBackRest

### 6.1 Restore Database by pgBackRest. [If data file is corrupted. Can't login postgresSQL]

Before restoration, the database service should be stopped.

*systemctl stop postgresql-14*

Lets see, how to happens. Recover will start from last backup which was taken recently.

*su - postgres*

*pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info restore*

```
[root@pgBackRest_Implement 14]# su - postgres
Last login: Wed May 21 17:19:46 +06 2025 on pts/0
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info restore
2025-05-21 17:20:48.690 P00 INFO: restore command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=5822-40126d8f --log-level-console=info --log-level=info --pgl-path=/var/lib/pgsql/14/data/ --repol-path=/data/backups --stanza=demo
2025-05-21 17:20:48.698 P00 INFO: repol: restore backup set 20250521-163940F_20250521-165456I, recovery will start at 2025-05-21 16:54:56
2025-05-21 17:20:49.823 P00 INFO: write updated /var/lib/pgsql/14/data/postgresql.auto.conf
2025-05-21 17:20:49.829 P00 INFO: restore global/pg control (performed last to ensure aborted restores cannot be started)
2025-05-21 17:20:49.830 P00 INFO: restore size = 26.1MB, file total = 951
2025-05-21 17:20:49.831 P00 INFO: restore command end: completed successfully (1152ms)
```

If you are facing the error. "Unable to restore the path  
/var/lib/pgsql/14/data/ because it contains files".

Create empty data directory or rename existing data directory:

```
cd /var/lib/pgsql/14/
mv data/ data_1
```

```
[root@pgBackRest_Implement 14]# su - postgres
Last login: Wed May 21 17:20:43 +06 2025 on pts/0
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info restore
2025-05-21 17:23:13.183 P00 INFO: restore command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=5930-bfd4d0dc --log-level-console=info --log-level-file=info --pg1-path=/var/lib/pgsql/14/data/ --repol-path=/data/backups --stanza=demo
2025-05-21 17:23:13.189 P00 INFO: repol: restore backup set 20250521-163940F_20250521-165456I, recovery will start at 2025-05-21 16:54:56
2025-05-21 17:23:13.190 P00 ERROR: [040]: unable to restore to path '/var/lib/pgsql/14/data' because it contains files
HINT: try using --delta if this is what you intended.
2025-05-21 17:23:13.190 P00 INFO: restore command end: aborted with exception [040]
[postgres@pgBackRest_Implement ~]$ cd /var/lib/pgsql/14/
[postgres@pgBackRest_Implement 14]$ ls
backups data data_1 initdb.log
[postgres@pgBackRest_Implement 14]$ mv data/ data_2
[postgres@pgBackRest_Implement 14]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info restore
2025-05-21 17:24:02.760 P00 INFO: restore command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=5954-ladd7228 --log-level-console=info --log-level-file=info --pg1-path=/var/lib/pgsql/14/data/ --repol-path=/data/backups --stanza=demo
2025-05-21 17:24:02.777 P00 INFO: repol: restore backup set 20250521-163940F_20250521-165456I, recovery will start at 2025-05-21 16:54:56
2025-05-21 17:24:04.109 P00 INFO: write updated /var/lib/pgsql/14/data/postgresql.auto.conf
2025-05-21 17:24:04.115 P00 INFO: restore global/pg_control (performed last to ensure aborted restores cannot be started)
2025-05-21 17:24:04.117 P00 INFO: restore size = 26.1MB, file total = 951
2025-05-21 17:24:04.117 P00 INFO: restore command end: completed successfully (1359ms)
[postgres@pgBackRest_Implement 14]$ exit
logout
[postgres@pgBackRest_Implement 14]$ systemctl start postgresql-14
After Crash ta DB, Restore done and DB up successfully
```

Then execute the restore command using pgbackrest. Hopefully, the command will be executed successfully.

## 6.2 Start the Postgres service

```
systemctl start postgresql-14
```

## 6.3 Verify and compare before and after situation take the backup.

Need to store previous situation/data to compare the backup restoration when backup whether correct or not.

## 6.4 Restoring Delta Changes

First full backup is required. Then some changes are occurred then take a incremental backup then below steps are followed to restore the delta changes. Delta changes are those changes where in the pgbackrest determines which file have been generated after the last restore was done and which files should be available on the database cluster. So, considering our scenario on another server whatever backup has been generated after last restore was done. Those backup files or incremental files or differential files changes should get restored using the delta params.

```
systemctl stop postgresql-14
```

```
su - postgres
```

```
pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info restore --delta
```

```
systemctl start postgresql-14
```

## 6.5 Restoring Selected Database

Differential backup taken then below the command perform

```
systemctl stop postgresql-14
```

```
su - postgres
```

```
cd /var/lib/pgsql/14/
```

```
mv /data /data_3
```

```
pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info restore --db-include=test2
```

```
systemctl start postgresql-14
```

```
[root@pgBackRest_Implement ~]# systemctl stop postgresql-14
[root@pgBackRest_Implement ~]# su - postgres
Last login: Thu May 22 09:52:54 +06 2025 on pts/0
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info restore --db-include=student
2025-05-22 09:53:37.489 P00 INFO: restore command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --db-include=student --delta --exec-id=1659-9e7c
vel-console=info --log-level=info --pgl-path=/var/lib/pgsql/14/data/ --repol-path=/data/backups --stanza=demo
2025-05-22 09:53:37.502 P00 INFO: repol: restore backup set 20250521-175835F 20250521-180010I, recovery will start at 2025-05-21 18:00:10
2025-05-22 09:53:37.502 P00 INFO: nothing to filter - all user databases have been selected
2025-05-22 09:53:37.503 P00 INFO: remove invalid files/links/paths from '/var/lib/pgsql/14/data'
2025-05-22 09:53:38.249 P00 INFO: write updated /var/lib/pgsql/14/data/postgresql.auto.conf
2025-05-22 09:53:38.255 P00 INFO: restore global/pg_control (performed last to ensure aborted restores cannot be started)
2025-05-22 09:53:38.257 P00 INFO: restore size = 34.7MB, file total = 1247
2025-05-22 09:53:38.257 P00 INFO: restore command end: completed successfully (770ms)
```

## 6.6 Point-in-Recovery using pgBackRest

Firstly login the postgres. And executed the query.

```
select now(); -- Store it
```

Then delete some data by executing the DELETE operation, which will be recover next time. Then take Incremental backup and PITR time will be before incremental backup are taken or select now(); query output time. Then execute below command to reach the previous setup.

```
systemctl stop postgresql-14
```

```
su - postgres
```

```
pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=detail --delta --type=time "--target=2025-05-13 10:37:09" --target-action=promote restore
```

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info --delta --type=time "--target=2025-05-22 10:22:17" --target-action=promote restore
2025-05-22 10:29:44.051 P00 INFO: restore command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --delta --exec-id=1923-6a80b5cb --log-level-console=info --log-level=info --pgl-path=/var/lib/pgsql/14/data/ --repol-path=/data/backups --stanza=demo --target="2025-05-22 10:22:17" --target-action=promote --type=time
2025-05-22 10:29:44.060 P00 INFO: repol: restore backup set 20250521-175835F 20250521-180010I, recovery will start at 2025-05-21 18:00:10
2025-05-22 10:29:44.061 P00 INFO: remove invalid files/links/paths from '/var/lib/pgsql/14/data'
2025-05-22 10:29:44.469 P00 INFO: write updated /var/lib/pgsql/14/data/postgresql.auto.conf
2025-05-22 10:29:44.475 P00 INFO: restore global/pg_control (performed last to ensure aborted restores cannot be started)
2025-05-22 10:29:44.477 P00 INFO: restore size = 34.7MB, file total = 1247
2025-05-22 10:29:44.477 P00 INFO: restore command end: completed successfully (427ms)
```

To verify the database again login postgresql & use the database then check the data.

```
[root@pgBackRest_Implement ~]# systemctl start postgresql-14
[root@pgBackRest_Implement ~]# su - postgres
Last login: Thu May 22 10:29:38 +06 2025 on pts/0
[postgres@pgBackRest_Implement ~]$ psql
psql (14.18)
Type "help" for help.

postgres=# \c student
You are now connected to database "student" as user "postgres".
student=# select * from personal_information;
 id | full_name
----+-----
  1 | Nahid
  2 | Opu
(2 rows)
```

To check the log for PITR.

When PITR running, showing in the log at the last stage "write updated /var/lib/pgsql/14/data/postgresql.auto.conf". To show the details log

`cat /var/lib/pgsql/14/data/postgresql.auto.conf`

`systemctl start postgresql-14`

```
[postgres@pgBackRest Implement ~]$ cat /var/lib/pgsql/14/data/postgresql.auto.conf
# Do not edit this file manually!
# It will be overwritten by the ALTER SYSTEM command.

# Recovery settings generated by pgBackRest restore on 2025-05-21 17:24:04
restore_command = 'pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo archive-get %f "%p"'

# Recovery settings generated by pgBackRest restore on 2025-05-22 10:29:44
restore_command = 'pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo archive-get %f "%p"'
recovery_target_time = '2025-05-22 10:22:17'
recovery_target_action = 'promote'
```

Then Check the data that I have deleted, that will be recover.

## 6.7 Point-in-Recovery using LSN number

To collect the LSN number, execute the query.

`select pg_current_wal_lsn();` -- Store it

Then delete some data by executing the DELETE operation or anything else, which will be recover next time. Then take differential backup.

```
student=# select pg_current_wal_lsn();
 pg_current_wal_lsn
-----
0/17000768
(1 row)

student=# select * from personal_information;
 id | full_name
-----+-----
  1 | Nahid
  2 | Opu
  3 | Karim
  4 | Lablu
  5 | Jamil
(5 rows)

student=# delete from personal_information where id=4;
DELETE 1
student=# select * from personal_information;
 id | full_name
-----+-----
  1 | Nahid
  2 | Opu
  3 | Karim
  5 | Jamil
(4 rows)
```

`su - postgres`

`pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info backup --type=diff`



Then PITR using LSN number

```
systemctl stop postgresql-14
```

```
su - postgres
```

```
pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info --delta --type=lsn "--target=0/950009B8" --target-action=promote restore
```

```
[root@pgBackRest_Implement ~]# systemctl stop postgresql-14
[root@pgBackRest_Implement ~]# su - postgres
Last login: Thu May 22 10:30:20 +06 2025 on pts/0
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info --delta --type=lsn "--target=0/17000768" --target-action=promote restore
2025-05-22 10:45:21.322 P00 INFO: restore command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --delta --exec-id=2252-beede78c --log-level-console=info --log-level-file=info --pgl-path=/var/lib/pgsql/14/data/ --repol-path=/data/backups --stanza=demo --target=0/17000768 --target-action=promote --type=lsn
2025-05-22 10:45:21.336 P00 INFO: repol: restore backup set 20250521-175835P 20250521-180010I, recovery will start at 2025-05-21 18:00:10
2025-05-22 10:45:21.337 P00 INFO: remove invalid files/links/paths from '/var/lib/pgsql/14/data'
2025-05-22 10:45:21.707 P00 INFO: write updated /var/lib/pgsql/14/data/postgresql.auto.conf
2025-05-22 10:45:21.713 P00 INFO: restore global/pg_control (performed last to ensure aborted restores cannot be started)
2025-05-22 10:45:21.714 P00 INFO: restore size = 34.7MB, file total = 1247
2025-05-22 10:45:21.714 P00 INFO: restore command end: completed successfully (393ms)
```

To verify the database again login postgresql & use the database then check the data.

```
[root@pgBackRest_Implement ~]# systemctl start postgresql-14
[root@pgBackRest_Implement ~]# su - postgres
Last login: Thu May 22 10:45:10 +06 2025 on pts/0
[postgres@pgBackRest_Implement ~]$ 
[postgres@pgBackRest_Implement ~]$ psql
psql (14.18)
Type "help" for help.

postgres=# \c student
You are now connected to database "student" as user "postgres".
student=# select * from personal_information;
 id | full_name
----+-----
  1 | Nahid
  2 | Opu
  3 | Karim
  4 | Lablu
  5 | Jamil
(5 rows)
```

To check the log for PITR.

When PITR running, showing in the log at the last stage "write updated /var/lib/pgsql/14/data/postgresql.auto.conf". To show the details log.

```
cat /var/lib/pgsql/14/data/postgresql.auto.conf
systemctl start postgresql-14
```

```
[root@pgBackRest_Implement ~]# cat /var/lib/pgsql/14/data/postgresql.auto.conf
# Do not edit this file manually!
# It will be overwritten by the ALTER SYSTEM command.

# Recovery settings generated by pgBackRest restore on 2025-05-21 17:24:04
restore_command = 'pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo archive-get %f "%p"'

# Recovery settings generated by pgBackRest restore on 2025-05-22 10:45:21
restore_command = 'pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo archive-get %f "%p"'
recovery_target_lsn = '0/17000768'
recovery_target_action = 'promote'
```

Then Check the data that I have deleted, that will be recover.

## 6.8 pgBackRest Retention Policy Configuration in pgbackrest.conf file

Firstly repo2 backup needs to be created.

```
mkdir -p /data/backups2
```

```
chown postgres:postgres /data/backups2/
```

```
[root@pgBackRest_Implement ~]# mkdir -p /data/backups2
[root@pgBackRest_Implement ~]# chown postgres:postgres /data/backups2/
```

Here is the path: /etc/pgbackrest/pgbackrest.conf

*[global]*

*repo1-host=192.168.56.108*

*repo1-path=/data/backups*

*repo2-path=/data/backups2*

*repo1-retention-full=2 [Minimum 2 Full backup remains]*

*repo1-retention-diff=2 [Minimum 2 Differential backup remains]*

*repo2-retention-full-type=time*

*repo2-retention-full=1 [Minimum 1 Full backup remains]*

*repo2-retention-diff=1 [Only 1 Differential backup needs to be kept]*

*archive-async=y*

*spool-path=/var/spool/pgbackrest*

*repo1-host-user=postgres*

*log-level-file=detail*

*start-fast=y*

*[demo]*

*pg1-path=/var/lib/pgsql/14/data/*

*pg1-host-user=postgres*

You need to run stanza-create for repo2 too. This will initialize both repo1 and repo2. Please execute below command.

```
pgbackrest --stanza=demo --log-level-console=info stanza-create
```

To Verify the stanza create for both repos or not

```
pgbackrest --stanza=demo check
```



## 6.9 To execute take repo wise full and differential backup

*pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info backup --type=diff --repo=2*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info backup --type=diff --repo=2
P00 WARN: configuration file contains invalid option 'repo1-retention-full'
P00 WARN: configuration file contains invalid option 'repo1-retention-diff'
P00 WARN: configuration file contains invalid option 'repo2-retention-full-type'
P00 WARN: configuration file contains invalid option 'repo2-retention-full'
P00 WARN: configuration file contains invalid option 'repo2-retention-diff'
2025-05-22 11:27:34.996 P00 INFO: backup command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=2804-Sf858926 --log-level-console=info --log-level-f
ile=detail --pg1-path=/var/lib/pgsql/14/data/ --repo=2 --repo1-path=/data/backups --repo2-path=/data/backups2 --stanza=demo --start-fast --type=diff
2025-05-22 11:27:34.996 P00 WARN: option 'repo1-retention-full' is not set for 'repo1-retention-full-type=count', the repository may run out of space
HINT: to retain full backups indefinitely (without warning), set option 'repo1-retention-full' to the maximum.
2025-05-22 11:27:34.996 P00 WARN: option 'repo2-retention-full' is not set for 'repo2-retention-full-type=count', the repository may run out of space
HINT: to retain full backups indefinitely (without warning), set option 'repo2-retention-full' to the maximum.
2025-05-22 11:27:35.018 P00 INFO: last backup label = 20250522-112648F, version = 2.55.1
2025-05-22 11:27:35.018 P00 INFO: execute non-exclusive backup start: backup begins after the requested immediate checkpoint completes
2025-05-22 11:27:35.216 P00 INFO: backup start archive = 000000060000000000000001D, lsn = 0/1D000028
2025-05-22 11:27:35.217 P00 INFO: check archive for prior segment 000000060000000000000001C
2025-05-22 11:27:36.146 P00 INFO: execute non-exclusive backup stop and wait for all WAL segments to archive
2025-05-22 11:27:36.189 P00 INFO: backup stop archive = 000000060000000000000001D, lsn = 0/1D000138
2025-05-22 11:27:36.205 P00 INFO: check archive for segment(s) 000000060000000000000001D:000000060000000000000001D
2025-05-22 11:27:36.529 P00 INFO: new backup label = 20250522-112648F_20250522-112735D
2025-05-22 11:27:36.598 P00 INFO: diff backup size = 110.2KB, file total = 1248
2025-05-22 11:27:36.598 P00 INFO: backup command end: completed successfully (1603ms)
2025-05-22 11:27:36.598 P00 INFO: expire command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=2804-Sf858926 --log-level-console=info --log-level-f
ile=detail --repo=2 --repo1-path=/data/backups --repo2-path=/data/backups2 --stanza=demo
2025-05-22 11:27:36.598 P00 INFO: option 'repo2-retention-archive' is not set - archive logs will not be expired
2025-05-22 11:27:36.598 P00 INFO: expire command end: completed successfully (0ms)
```

## 6.10 To information see repo wise

*pgbackrest info --repo=2*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest info --repo=2
P00 WARN: configuration file contains invalid option 'repo1-retention-full'
P00 WARN: configuration file contains invalid option 'repo1-retention-diff'
P00 WARN: configuration file contains invalid option 'repo2-retention-full-type'
P00 WARN: configuration file contains invalid option 'repo2-retention-full'
P00 WARN: configuration file contains invalid option 'repo2-retention-diff'
stanza: demo
status: ok
cipher: none

db (current)
wal archive min/max (14): 0000000600000000000000017/000000060000000000000001D

full backup: 20250522-112648F
timestamp start/stop: 2025-05-22 11:26:48+06 / 2025-05-22 11:26:52+06
wal start/stop: 000000060000000000000001B / 000000060000000000000001B
database size: 34.8MB, database backup size: 34.8MB
repo2: backup set size: 4.3MB, backup size: 4.3MB

diff backup: 20250522-112648F_20250522-112735D
timestamp start/stop: 2025-05-22 11:27:35+06 / 2025-05-22 11:27:36+06
wal start/stop: 000000060000000000000001D / 000000060000000000000001D
database size: 34.8MB, database backup size: 110.2KB
repo2: backup set size: 4.3MB, backup size: 4.2KB
backup reference total: 1 full
```

## 6.11 To Backup Expire

*pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info expire --set=20250522-112648F\_20250522-112735D*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info expire --set=20250522-112648F_20250522-112735D
P00 WARN: configuration file contains invalid option 'repo1-retention-full'
P00 WARN: configuration file contains invalid option 'repo1-retention-diff'
P00 WARN: configuration file contains invalid option 'repo2-retention-full-type'
P00 WARN: configuration file contains invalid option 'repo2-retention-full'
P00 WARN: configuration file contains invalid option 'repo2-retention-diff'
2025-05-22 16:32:43.764 P00 INFO: expire command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=4198-05108a0c --log-level-console=info --log-level-f
ile=detail --repo1-path=/data/backups --repo2-path=/data/backups2 --set=20250522-112648F_20250522-112735D --stanza=demo
2025-05-22 16:32:43.765 P00 WARN: option 'repo1-retention-full' is not set for 'repo1-retention-full-type=count', the repository may run out of space
HINT: to retain full backups indefinitely (without warning), set option 'repo1-retention-full' to the maximum.
2025-05-22 16:32:43.765 P00 WARN: option 'repo2-retention-full' is not set for 'repo2-retention-full-type=count', the repository may run out of space
HINT: to retain full backups indefinitely (without warning), set option 'repo2-retention-full' to the maximum.
2025-05-22 16:32:43.768 P00 INFO: option 'repo2-retention-archive' is not set - archive logs will not be expired
2025-05-22 16:32:43.769 P00 WARN: repo2: expiring latest backup 20250522-112648F_20250522-112735D - the ability to perform point-in-time-recovery (PITR) may be affect
ed
HINT: non-default settings for 'repo2-retention-archive'/'repo2-retention-archive-type' (even in prior expiries) can cause gaps in th
e WAL.
2025-05-22 16:32:43.771 P00 INFO: repo2: expire adhoc backup 20250522-112648F_20250522-112735D
2025-05-22 16:32:43.775 P00 INFO: repo2: remove expired backup 20250522-112648F_20250522-112735D
2025-05-22 16:32:43.775 P00 INFO: option 'repo2-retention-archive' is not set - archive logs will not be expired
2025-05-22 16:32:43.775 P00 INFO: expire command end: completed successfully (11ms)
```

## 6.12 Another way to expire the backup and older backup remove by executing this command

*pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info expire --repo2-retention-full=1*

```
[postgres@pgBackRest_Implement ~]$ pgbackrest --config=/etc/pgbackrest/pgbackrest.conf --stanza=demo --log-level-console=info expire --repo2-retention-full=1
P00 WARN: configuration file contains invalid option 'repo1-retention-full'
P00 WARN: configuration file contains invalid option 'repo1-retention-diff'
P00 WARN: configuration file contains invalid option 'repo2-retention-full-type'
P00 WARN: configuration file contains invalid option 'repo2-retention-full'
P00 WARN: configuration file contains invalid option 'repo2-retention-diff'
2025-05-22 16:36:51.962 P00 INFO: expire command begin 2.55.1: --config=/etc/pgbackrest/pgbackrest.conf --exec-id=4208-1ee2e544 --log-level-console=info --log-level-file=detail --repo1-path=/data/backups --repo2-path=/data/backups2 --repo2-retention-full=1 --stanza=demo
2025-05-22 16:36:51.962 P00 WARN: option 'repo1-retention-full' is not set for 'repo1-retention-full-type=count', the repository may run out of space
HINT: to retain full backups indefinitely (without warning), set option 'repo1-retention-full' to the maximum.
2025-05-22 16:36:51.965 P00 INFO: option 'repo1-retention-archive' is not set - archive logs will not be expired
2025-05-22 16:36:51.967 P00 INFO: repo2: 14-1 remove archive; start = 00000006000000000000000017, stop = 0000000600000000000000001A
2025-05-22 16:36:51.967 P00 INFO: expire command end: completed successfully (7ms)
```

## 7. Conclusion

With proper configuration, pgBackRest ensures reliable, automated, and efficient PostgreSQL backups and restores, minimizing downtime and data loss. Implementing it with correct stanza setup, retention policies, and recovery strategies (PITR and delta restore) enhances database resilience and meets critical data protection requirements.