
Konfiguracja bazy danych

Hubert Albanowsski, Filip Opac

01 lip 2024

Contents

1	Lokalizacja i struktura katalogów	1
1.1	Lokalizacja:	1
1.2	Struktura katalogów:	1
2	Tabele - rozmiar, planowanie i monitorowanie	3
2.1	Rozmiar	3
2.2	Planowanie	3
2.3	Monitorowanie	4
3	Podstawowe parametry konfiguracyjne	5
3.1	Plik postgresql.conf	5
3.2	Plik pg_hba.conf	6
3.3	Plik pg_ident.conf	6
4	Indices and tables	7

Lokalizacja i struktura katalogów

1.1 Lokalizacja:

1) Katalog danych:

- `/var/lib/postgresql/<wersja>/main` na systemach Debian/Ubuntu
- `/var/lib/pgsql/<wersja>/data` na systemach Red Hat/CentOS.
- Zawiera wszystkie dane, pliki konfiguracyjne, logi i pliki kontrolne.

2) Katalog Konfiguracyjny:

- Pliki konfiguracyjne zwykle znajdują się w katalogu danych, choć zdarza się, że mogą znajdować się w innym katalogu np `/etc/postgresql/<wersja>/main`

3) Katalog logów:

- Domyślnie `/var/log/postgresql` na Debianie/Ubuntu oraz `/var/lib/pgsql/<wersja>/data/pg_log` na Red Hat/CentOS
- zawiera logi PostgreSQL

1.2 Struktura katalogów:

- `base/`: Zawiera dane użytkownika dla każdej bazy danych.
- `global/`: Przechowuje dane globalne, np. tabele systemowe.
- `pg_xlog/` lub `pg_wal/` (od wersji 10): Zawiera dzienniki Write-Ahead Log (WAL).
- `pg_clog/` lub `pg_xact/`: Przechowuje dane dotyczące transakcji.
- `pg_tblspc/`: Linki symboliczne do tabel przestrzeni.
- `pg_multixact/`: Dane dotyczące wielokrotnych transakcji.

- `pg_subtrans/`: Dane dotyczące podrzędnych transakcji.
- `pg_stat/`: Dane statystyczne.
- `pg_snapshots/`: Przechowuje dane dotyczące snapshotów.

Tabele - rozmiar, planowanie i monitorowanie

2.1 Rozmiar

Rozmiar tabel można monitorować za pomocą poniższych zapytań SQL:

- Dla jednej tabeli:

```
SELECT pg_size_pretty(pg_total_relation_size('nazwa_tabeli'));
```

- Dla wszystkich tabel:

```
SELECT relname AS "Table", pg_size_pretty(pg_total_relation_size(relid)) AS "Size"
FROM pg_catalog.pg_statio_user_tables
ORDER BY pg_total_relation_size(relid) DESC;
```

2.2 Planowanie

Częścią planowania tabel jest:

- 1) **Normalizacja** - Rozbijanie tabel na mniejsze by zmniejszyć powtarzanie danych i zależności.
- 2) **Denormalizacja** - Łączenie tabel w większe i wprowadzanie redundancji w celu przyspieszenia czasu zapytań.
- 3) **Indeksowanie** - tworzenie indeksów w celu przyspieszenia przeszukiwań tabeli.

```
CREATE INDEX idx_nazwa_kolumny ON nazwa_tabeli(nazwa_kolumny);
```

2.3 Monitorowanie

Dodatkowo możliwe jest monitorowanie wydajności tabel za pomocą:

- **pg_stat_user_table:**

```
SELECT relname, seq_scan, seq_tup_read, idx_scan, idx_tup_fetch, n_tup_ins, n_tup_upd, n_
↳ tup_del
FROM pg_stat_user_tables;
```

- **pg_stat_activity:**

```
SELECT pid, username, datname, state, query_start, query
FROM pg_stat_activity;
```

Podstawowe parametry konfiguracyjne

3.1 Plik postgresql.conf

Plik *postgresql.conf* zawiera ustawienia dotyczące wydajności, logowania, sieci i wielu innych aspektów.

Kluczowe ustawienia:

1) Słuchanie połączeń:

```
listen_addresses = 'localhost' # Adresy IP, na których PostgreSQL będzie nasłuchiwać
↪połączeń
port = 5432                    # Port, na którym PostgreSQL będzie nasłuchiwać połączeń
```

2) Pamięć i wydajność:

```
shared_buffers = 128MB        # Ilość pamięci RAM przeznaczona na buforowanie danych
work_mem = 4MB                # Ilość pamięci RAM na operacje sortowania i agregacji
↪na użytkownika
maintenance_work_mem = 64MB   # Ilość pamięci RAM na operacje utrzymawcze (np.
↪VACUUM, CREATE INDEX)
```

3) Autovacuum:

```
autovacuum = on               # Automatyczne czyszczenie i analiza tabel
autovacuum_naptime = 1min     # Częstotliwość uruchamiania procesu autovacuum
```

3.2 Plik pg_hba.conf

Plik **pg_hba.conf** odpowiada za kontrolę dostępu do bazy danych PostgreSQL.

Przykład konfiguracji:

#	TYPE	DATABASE	USER	ADDRESS	METHOD
# Zezwól lokalnym użytkownikom na połączenie					
local	all		all		md5
# Zezwól zdalnym użytkownikom z sieci 192.168.1.0/24 na połączenie					
host	all		all	192.168.1.0/24	md5

3.3 Plik pg_ident.conf

Plik **pg_ident.conf** pozwala mapować systemowych użytkowników do użytkowników PostgreSQL.

Przykład konfiguracji:

#	MAPNAME	SYSTEM-USERNAME	PG-USERNAME
mymap		johndoe	john
mymap		janedoe	jane

W pliku **pg_hba.conf** można użyć tej mapy:

host	all	all	127.0.0.1/32	ident map=mymap
------	-----	-----	--------------	-----------------

CHAPTER 4

Indices and tables

- `genindex`
- `modindex`
- `search`