Анализ производительности криптографических расширений PostgreSQL

Забелкин Андрей М8О-310Б-22

18 апреля 2025 г.

База данных содержит более 7 млн записей компаний.

1 Анализ нечёткого поиска

1.1 Временные характеристики

```
--
EXPLAIN ANALYZE SELECT name
FROM    companies
WHERE   name % 'Microsft'
LIMIT   10;
```

Таблица 1: Сравнение времени выполнения запросов (мс)

Метод	Холодный кэш	Горячий кэш	Прирост
pg_trgm	2683 ± 120	262 ± 15	10.2×
pg_bigm	1987 ± 95	215 ± 12	9.2×
LIKE	4521 ± 210	4400 ± 185	1.03×

Пояснения:

- *Холодный кэш*: первый запуск после перезагрузки сервера.
- *Горячий кэш*: повторный запуск, когда данные уже в памяти.
- `pg_bigm` показывает лучшее время благодаря оптимизированной работе с биграммами.
- Ускорение при повторных запросах связано с использованием кэша PostgreSQL.

2 Криптографические операции

2.1 Шифрование данных

```
--
UPDATE companies SET
    encrypted_name    = pgp_sym_encrypt(name, 'key'),
    encrypted_revenue = pgp_sym_encrypt(revenue::text, 'key');
```

Таблица 2: Время шифрования 1 млн записей (сек)

Алгоритм	Шифрование	Дешифровка	Накладные расходы
AES-256	14.2 ± 0.8	12.7 ± 0.6	18 %
Blowfish	16.8 ± 0.9	15.3 ± 0.7	22 %
3DES	31.5 ± 1.2	28.4 ± 1.1	35 %

Пояснения:

- Накладные расходы — увеличение времени по сравнению с незашифрованными данными.
- AES демонстрирует лучшую производительность благодаря аппаратной поддержке в современных CPU.
- 3DES устарел и не рекомендуется для новых систем.

2.2 Хеширование паролей

```sql
--
SELECT
  sum(time_cost) AS total_time
FROM
  generate_series(1, 100000) AS s,
  LATERAL (
    SELECT crypt('password' || s, gen_salt('bf', 8)) AS time_cost
  );
```

Таблица 3: Скорость хеширования (100 000 операций)

Алгоритм	Время (сек)	Итераций/сек
bcrypt (cost = 8)	9.8 ± 0.4	10 200
bcrypt (cost = 10)	38.2 ± 1.1	2 620
bcrypt (cost = 12)	152.7 ± 3.5	655
MD5	0.9 ± 0.05	111 100

Пояснения:

- `bcrypt` с cost = 8 — разумный баланс безопасности и скорости.
- Увеличение cost фактора экспоненциально замедляет вычисления.
- MD5 крайне быстр, но небезопасен для хранения паролей.

3 Выводы

- **Нечёткий поиск**: `pg_bigm` на 26 % быстрее `pg_trgm` при холодном кэше; разница сокращается до 8 % при горячем.
- **Шифрование**: AES-256 добавляет 18 % накладных расходов против 35 % у 3DES.
- **Хеширование**: `bcrypt` с cost = 10 замедляет перебор паролей в 38 раз по сравнению с cost = 8.

- **Кэширование**: ускоряет повторные запросы в 9–10 раз для нечёткого поиска.