

Установка и конфигурирование СУБД

Алексей Федин

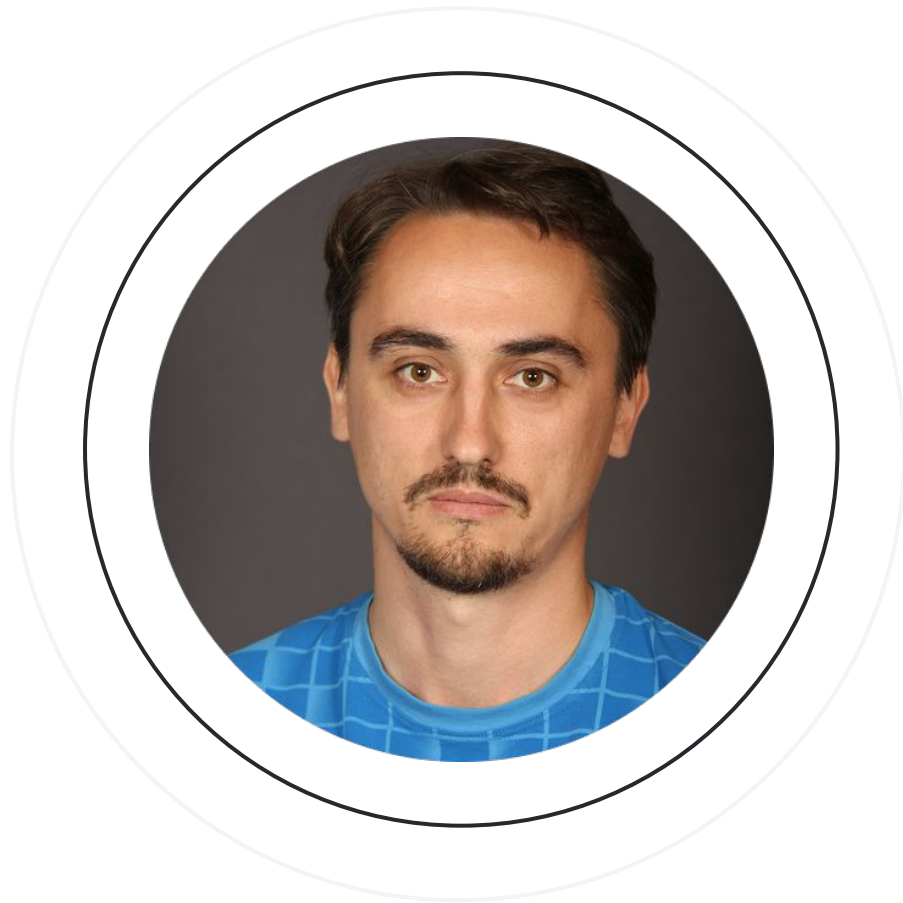
Системный аналитик в ООО «Открытые решения»



Алексей Федин

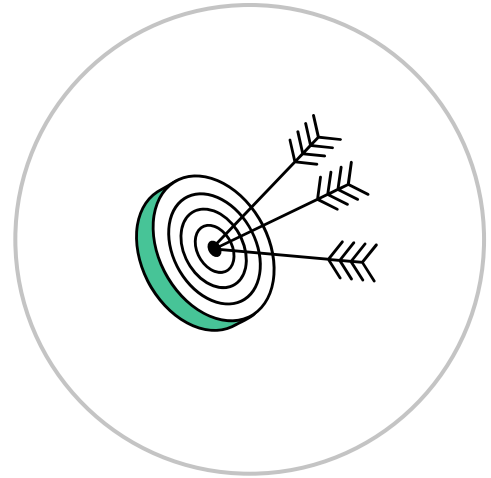
О спикере:

- системный аналитик
- работает в IT с 2002 года
- опыт администрирования и работы с сетями более 10 лет
- с 2008 года занимается проектированием и наладкой решений информационной безопасности в промышленности
- работал в компаниях: «PTСофт», Positive Technologies, iGrids, ElcomSoft



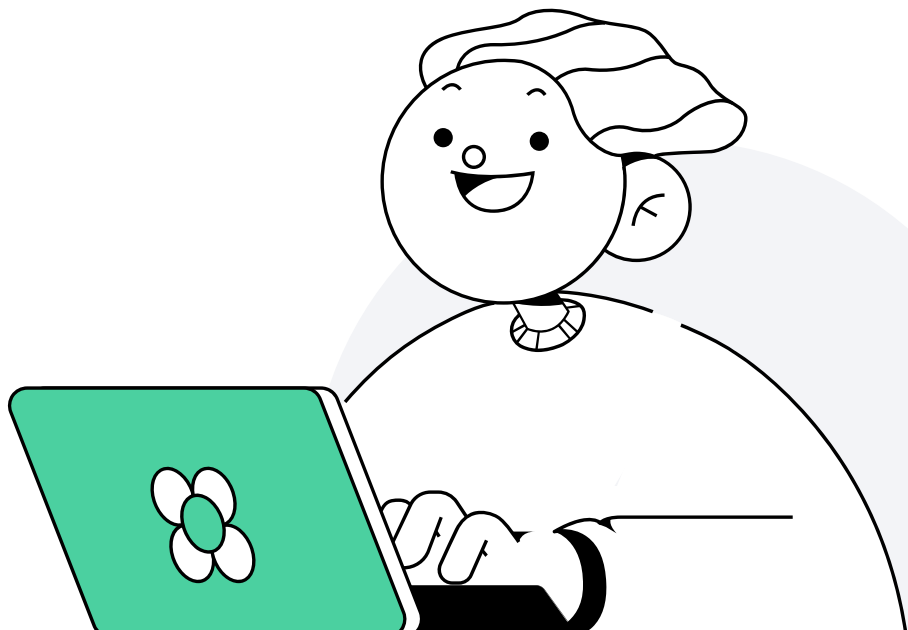
Цели занятия

- Узнать, как установить PostgreSQL
- Выполнить настройки после установки
- Рассмотреть сетевой доступ к БД
- Узнать, что такое балансировщик



План занятия

- 1 Выбор типа установщика
- 2 Настройки после установки
- 3 Доступ по сети
- 4 Балансировщик
- 5 Итоги
- 6 Домашнее задание



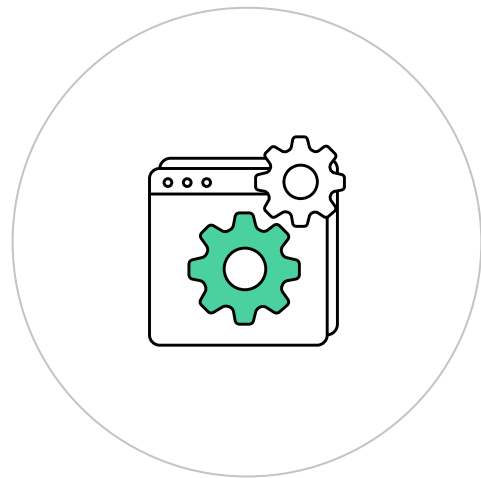
Выбор типа установщика



1

Тип установки

- 1 Исходный код
- 2 Установочный файл
- 3 Установка из репозитория
- 4 One-click installer
- 5 Docker



Исходный код

```
./configure
make
su
make install
adduser postgres
mkdir -p /usr/local/pgsql/data
chown postgres /usr/local/pgsql/data
su - postgres
/usr/local/pgsql/bin/initdb -D /usr/local/pgsql/data
/usr/local/pgsql/bin/pg_ctl -D /usr/local/pgsql/data -l logfile start
/usr/local/pgsql/bin/createdb test
/usr/local/pgsql/bin/psql test
```

[Полная инструкция](#)

Установочный файл

- Загрузите [установочный файл](#)
- Готовые [стеки](#) ПО: LAPP, MAPP или WAPP
(Linux/Mac/Windows + Apache + PHP + PostgreSQL)



Установка из репозитория

Debian:

```
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 --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key
add -

sudo apt-get update

sudo apt-get -y install postgresql
```

Red hat:

```
yum install postgresql-server

dnf install postgresql-server
```

One-click installer

Скачайте [установщик](#)
с графическим интерфейсом



Docker

Официальный [образ](#).

Установка:

```
docker pull postgres
```

Запуск:

```
docker run --name test_db -p 5432:5432 -e POSTGRES_PASSWORD=mysecretpassword  
-d postgres
```

Консоль:

```
sudo docker exec -it test_db psql -U postgres
```

Редакции

Распространение:

- **fork** — дистрибутив, несовместимый с оригинальной версией, proprietary
- **patch** — совместимость сохранена, open source
- **add-on** — значительные добавления в код, меняющие поведение СУБД
- **redistribution** — распространение исходного дистрибутива под другим именем или лицензией

Примеры:

- Postgres Professional (fork)
- Mammoth PostgreSQL (fork)
- SEPostgres (patch)
- PostGIS (add-on)
- HadoopDB (add-on)

Ограничения PostgreSQL

Максимальный размер базы данных	Нет ограничений
Максимальный размер таблицы	32 ТБ
Максимальный размер записи	1,6 ТБ
Максимальный размер поля	1 ГБ
Максимум записей в таблице	Нет ограничений
Максимум полей в записи	250 — 1 600, в зависимости от типов полей
Максимум индексов в таблице	Нет ограничений

Настройки после установки



2

Linux

Если есть проблемы с доступом, то измените права владения на каталог:

```
chown -R postgres:postgres /usr/local/pgsql/data
```

Установка пароля

Зайдите под пользователем под умолчанием:

```
sudo -u postgres psql
```

Установите пароль:

```
ALTER USER postgres with encrypted password 'my_password';
```

Перезапустите сервер:

```
sudo systemctl restart postgresql.service
```


Настройки памяти

- **shared_buffers** — **15–25% RAM** для Linux и **512 МБ** для Windows
- **effective_cache_size** — до 75% RAM
- **work_mem** — вычисляется как **RAM × 0,25 / max_connections**
- **maintenance_work_mem** — **RAM / 20** для одного autovacuum worker

Настройки планировщика

- **seq_page_cost** — стоимость выборки с диска при последовательном чтении
- **random_page_cost** — стоимость выборки с диска при произвольном чтении. Можно установить значение между **2,0** и **3,0**
- **cpu_tuple_cost** — стоимость обработки строки. По умолчанию установлено как **0,01**, но можно попробовать значение **0,03**

Настройки WAL

- **wal_buffers** — по умолчанию вычисляется как **shared_buffers / 32**, но не больше 16 МБ. Попробуйте установить значение в **32 МБ**
- **wal_level** — установить как **archive**. Для потоковой репликации — **hot_standby** или **replica**
- **bgwriter_lru_maxpages** — по умолчанию 100, но лучше установить **1 000**
- **fsync** — установите в **enabled**

Настройки autovacuum

- **autovacuum** — установите в **on**
- **autovacuum_max_workers** — по умолчанию установлено 3, но лучше подобрать значение в диапазоне **от 6 до 12**

Доступ по сети



3

Предварительные настройки

На межсетевом экране не забывайте открыть порт **TCP 5432**.

Проверьте пути к конфигурационным файлам:

```
SHOW config_file;
```

```
SHOW hba_file;
```

Настройка postgresql.conf

Разрешите подключение к серверу:

```
listen_addresses = '*'
```

```
listen_addresses = localhost, 192.168.0.100
```

Настройка pg_hba.conf

Подключение	БД	Пользователь	IP	Аутентификация
host	all	all	192.168.0.1/24	md5
host	all	all	0.0.0.0/0	md5
host	all	all	::/0	radius
host	postgres	postgres	all	ldap
hostnossl	all	all	.example.com	reject

[Документация](#)

Проверка подключения

Проверьте, что сетевая служба доступна:

```
telnet 192.168.0.1 5432
```

Проверьте подключение:

```
psql -U postgres -h 192.168.0.1 -d postgres
```

Балансировщик



4

Репликация и балансировка

Репликация — это механизм, позволяющий синхронизировать содержимое нескольких БД

Балансировка — это механизм, позволяющий распределять запросы к СУБД по разным копиям базы данных с целью увеличения производительности



PgBouncer

PgBouncer — балансировщик, управляющий пулом соединений postgres. Клиенты подключаются к PgBouncer, как к обычной БД, и он создаёт подключение к реальному серверу.

Установка:

```
sudo apt install pgbouncer
```

Откуда можно загрузить установочные файлы:

- [источник 1](#)
- [источник 2](#)

Под Windows запускается только через MinGW

PgBouncer

PgBouncer поддерживает несколько видов пулов:

- **Пул сеансов** (по умолчанию) — клиент подключается к БД, и ему назначается собственное подключение на всё время работы. Когда клиент отключается, это подключение возвращается в пул
- **Пул транзакций** — клиент подключается к БД, и ему назначается подключение только на время транзакции. Когда транзакция завершается, это подключение возвращается в пул
- **Пул операторов** — клиент подключается к БД, и ему назначается подключение только на время запроса. В этом режиме невозможны транзакции с несколькими операторами

Pgpool-II

Pgpool — ПО, работающее между сервером и клиентом Postgres.

Установка:

```
sudo apt install pgpool2
```

Откуда можно загрузить установочные файлы:

- [источник 1](#)
- [источник 2](#)

Под Windows запускается только через MinGW

Возможности pgpool-II

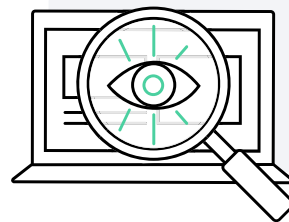
- Пул соединений
- Репликация
- Балансировка нагрузки
- Ограничение количества соединений
- Watchdog
- Кэширование запросов в памяти
- Параллельные запросы

PgBouncer vs pgpool

	PgBouncer	Pgpool
Процессы	Запускает один процесс	Запускает один процесс (fork) на клиента
Виды пулов	Сессия, транзакция, оператор	Сессия
HA	—	+
Балансировка нагрузки	—	+
Мультикластер	+	—
Ограничение соединений	Per-pool, per-database, per-user, per-client	Общее количество соединений
Очередь соединений	+	—

Демонстрация работы

Работа с балансировщиком



Итоги

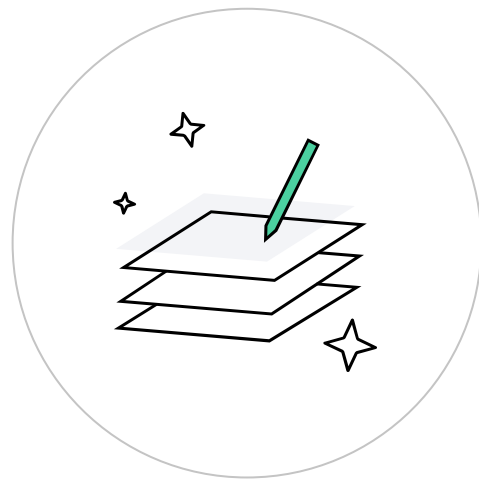
- Узнали, как установить PostgreSQL
- Выполнили настройки после установки
- Настроили сетевой доступ к БД
- Узнали, что такое балансировщик



Домашнее задание

Давайте посмотрим ваше домашнее задание

- 1 Вопросы по домашней работе задавайте в чате группы
- 2 Задачи можно сдавать по частям
- 3 Зачёт по домашней работе ставят после того, как приняты все задачи



Задавайте вопросы и пишите отзыв о лекции

Алексей Федин

Системный аналитик в ООО «Открытые решения»

