СУБД Ассоциация разработчиков

Система управления базами данных «RuDB»

Руководство по установке и настройке

Версия 26.09, сборка 20260902

Ассоциация разработчиков СУБД
2026

Содержание

1. Совместимые операционные системы 3
2. Требования системы 3
  2.1. Минимальные системные требования 3
  2.2. Рекомендуемые системные требования 3
3. Процесс установки 4
  3.1. Установка в ОС Astra Linux 4
  3.2. Установка в ОС Ubuntu 4
  3.3. Установка в ОС ALT Linux 5
  3.4. Установка в ОС ОСнова 5
4. Результат установки СУБД 5
5. Файловая структура 6
  5.1. Исполняемые файлы 6
  5.2. Каталог базы данных 7
6. Сервисы RuDB 10
  6.1. Сервис rudb-18 10
  6.2. Сервис dgrid 10

1.Совместимые операционные системы

Дистрибутивы выпускается для следующих ОС:

- ALT Linux – версия 11
- Astra Linux – версия 1.8
- Ubuntu – версия 24
- ОСнова - версия 3.1 onyx

2.Требования системы

2.1. Минимальные системные требования

Процессор

- Многопоточность/ядра CPU: минимум 2 ядра, лучше — 4+ ядер.
- Тактовая частота процессора: от 2 ГГц и выше.

Оперативная память (RAM)

- Минимум: 8 ГБ RAM.
- Оптимальное значение: от 16 ГБ RAM и выше.

Хранилище (HDD/SDD)

- Дисковая подсистема должна поддерживать высокую производительность ввода-вывода (I/*).
- Рекомендуемый объем хранилища зависит от объема данных и рабочих нагрузок, но минимально около 50–100 ГБ свободного пространства.
- Для больших объемов данных желательно использование SSD-дисков или RAIDмассивов для повышения производительности операций чтения-записи.

2.2. Рекомендуемые системные требования

Процессор

- Количество ядер: 8+ (желательно больше, особенно для масштабируемых решений).
- Тактовая частота процессора: от 2,5 ГГц и выше.
- Гиперпоточность (Hyper-Threading), поддержка AVX-инструкций и технологии виртуализации (VT-x / AMD-V).

Оперативная память (RAM)

- Минимум: 32 ГБ RAM.

- Желательно: 64 ГБ и выше, в зависимости от размера базы данных и сложности приложений.

Хранилище (SSD/HDD)

- Быстрая дисковая подсистема с высокой пропускной способностью IOPS (операций ввода-вывода в секунду). Рекомендуется NVMe или быстрый SSD.
- Объем диска должен соответствовать размеру баз данных плюс запас на будущее расширение и хранение временных файлов (лог-файлы, транзакционные журналы, временные файлы сортировки и индексов).
- Использовать аппаратные RAID-конфигурации уровня 10 или программные аналоги для увеличения отказоустойчивости и скорости записи/чтения.

3.Процесс установки

Предварительно перед установкой необходимо проверить, что порт 5432 не занят на сервере, на котором производится установка.

Проверка осуществляется с помощью команды (для всех дистрибутивов ОС Linux):

```
sudo netstat -tulpn
```

Если порт занят, возникнут ошибки при установке СУБД.

В случае невозможности освободить данный порт – после неудачной установки необходимо сменить используемый порт СУБД в файле postgresql.conf и перезапустить все сервисы СУБД.

3.1. Установка в ОС Astra Linux

Шаги установки (удаления) пакета:

1. Подключить стандартный репозиторий пакетов текущей ОС.
2. Загрузить необходимую версию пакета СУБД RuDB.
3. Установить пакет:

   ```
   sudo apt install ./rudb-26.09-26090201-astra1.8_x86_64.deb;
   ```

При необходимости удаления использовать команду:

```
sudo apt remove rudb-18;
```

3.2. Установка в ОС Ubuntu

Шаги установки (удаления) пакета:

1. Подключить стандартный репозиторий пакетов текущей ОС.

2. Загрузить необходимую версию пакета СУБД RuDB.
3. Установить пакет:

```
sudo apt install ./rudb-26.09-26090201-ubuntu24_x86_64.deb;
```

При необходимости удаления использовать команду:

```
sudo apt remove rudb-18;
```

3.3. Установка в ОС ALT Linux

Шаги установки (удаления) пакета:

1. Подключить стандартный репозиторий пакетов текущей ОС.
2. Загрузить необходимую версию пакета СУБД RuDB.
3. Установить пакет:

```
apt-get install ./rudb-26.09-26090201-alt11_x86_64.rpm;
```

При необходимости удаления использовать команду:

```
apt-get remove rudb-18;
```

Внимание! Каталог с базами данных в этой ОС не будет удален ни при удалении, ни при переустановке rpm-пакета. Если необходима полная переустановка, выполните удаление каталога данных вручную.

3.4. Установка в ОС ОСнова

Шаги установки (удаления) пакета:

1. Подключить стандартный репозиторий пакетов текущей ОС.
2. Загрузить необходимую версию пакета СУБД RuDB.
3. Установить пакет:

```
sudo apt install ./rudb-26.09-26090201-onyx3.1_x86_64.deb;
```

При необходимости удаления использовать команду:

```
sudo apt remove rudb-18;
```

4.Результат установки СУБД

После успешной установки сервер RuDB запускается и становится доступен на порте 5432.

Внимание! Данный порт не будет доступен, если подключение к СУБД происходит извне! Для удаленного (не локального) взаимодействия любого приложения с СУБД требуется дополнить список разрешенных IP в файле pg_hba.conf. Подробнее об этом файле читайте далее в следующем подразделе "Файловая структура".

Все исполняемые модули размещены в каталоге:

/opt/rudb-18

Базы данных размещены в каталоге:

/var/lib/rudb-18/data

В фоне начинают работу 2 сервиса: rudb-18, dgrid. По умолчанию создается БД:

- rudb - суперпользователь "postgres" с паролем "postgres";

5.Файловая структура

5.1. Исполняемые файлы

Актуально для всех ОС*

Все исполняемые модули размещены в каталоге:

/opt/rudb-18/bin

В данном каталоге расположены файлы, упрощающие работу с диалектом PostgreSQL:

1. **postgres**: Основной серверный процесс БД. Он отвечает за обработку запросов и управление базой данных.
2. **pg_ctl**: Утилита для управления сервером БД. Позволяет запускать, останавливать и перезапускать сервер.
3. **initdb**: Утилита для инициализации новой базы данных. Создает кластер баз данных и настраивает его.
4. **createdb**: Утилита для создания новой базы данных в кластере.
5. **createuser**: Утилита для создания нового пользователя (роли).
6. **dropdb**: Утилита для удаления базы данных из кластера.
7. **dropuser**: Утилита для удаления пользователя (роли).
8. **pg_dump**: Утилита для создания резервной копии базы данных.
9. **pg_restore**: Утилита для восстановления базы данных из резервной копии, созданной с помощью pg_dump.
10. **pg_basebackup**: Утилита для создания полной резервной копии кластера.

5.2. Каталог базы данных

Все базы данных размещены в каталоге:

/var/lib/rudb-18/data

Данный каталог содержит конфигурационные файлы. Среди них присутствуют два важных файла:

1. postgresql.conf
2. pg_hba.conf

Рассмотрим их назначение.

Файл postgresql.conf

Это главный файл конфигурации БД, содержащий глобальные настройки, влияющие на работу всей системы баз данных. Здесь определяются ключевые параметры производительности, памяти, журналирования, репликации и другие аспекты функционирования БД.

Структура файла

Файл организован в виде секций, каждая из которых охватывает определённую группу параметров:

- Общие настройки
- Параметры файловой системы • Журналирование
- Производительность и память
- Репликация и синхронизация
- Безопасность и контроль доступа

Параметры

Применение изменений

Изменённые параметры вступают в силу после перезагрузки основного сервиса СУБД:

```
sudo systemctl restart rudb-18
```

Или с помощью SQL-команды:

```
SELECT pg_reload_conf();
```

Некоторые параметры требуют полной перезагрузки службы, что указывается комментарием рядом с параметром в файле конфигурации.

Файл pg_hba.conf

Это главный файл конфигурации pg_hba.conf используется в СУБД для определения правил аутентификации клиентов. Эти правила задают, каким пользователям разрешено подключаться к базе данных, откуда они могут подключаться и какой метод аутентификации использовать. Формат файла:

TYPE	DATABASE	USER	ADDRESS	METHOD
local	all	all		trust
local	all	all		md5
host	all	postgres	127.0.0.1/32	trust
host	all	all	127.0.0.1/32	md5

Каждая строка файла соответствует одному правилу аутентификации.

Описание полей:

- **Type** – определяет тип подключения:
- **local**: локальные подключения через Unix сокеты.
- **host**: TCP/IP подключения с шифрованием или без.
- **hostssl**: требует SSL-шифрования для подключения.
- **hostnossl**: запрещает использование SSL-шифрования.
- **Database** – база данных, к которой применяется правило. Можно указывать конкретные имена баз данных, группы баз данных (обозначаются звёздочкой *) или слово all для всех баз данных.
- **User** – пользователь, к которому применяется данное правило. Аналогично предыдущему полю, допустимы индивидуальные имена пользователей, группы пользователей (со звёздочками) или all для всех пользователей.
- **Address** – адрес компьютера или диапазон IP-адресов, которым разрешен доступ. Может задаваться в формате CIDR (например, 192.168.1.0/24), IPv6-адрес или отдельное значение для конкретного IP-адреса.
- **Method** – способ аутентификации:
- **trust**: полное доверие без проверки пароля.
- **md5**: проверка MD5-хэша пароля.
- **password**: проверка открытого текста пароля.
- **scram-sha-256**: современный метод аутентификации с безопасным алгоритмом SHA-256. Примечание: данный способ недоступен при работе через протокол TDS.
- **ident**: идентификация пользователя по системе ОС.
- **ldap**: аутентификация через LDAP-сервер.
- **cert**: аутентификация сертификатом.

- **Options** — необязательные параметры, такие как срок жизни сеанса или информация о LDAP-сервере.

Примеры записей в файле pg_hba.conf:

1. Разрешить локальному пользователю postgres подключение ко всем базам данных без пароля

```
local all postgres trust
```

2. Любые пользователи с IP 192.168.1.* могут подключаться к базе данных mydb используя md5-аутентификацию

```
host mydb all 192.168.1.0/24 md5
```

3. Только пользователь admin может подключаться к базе datasecure с зашифрованным каналом и методом scram-sha-256

```
host mydb all 192
hostssl datasecure admin 172.16.0.0/16 scram-sha-256
```

4. Все пользователи могут подключаться к любым базам данных без шифрования, используя стандартный пароль

```
host mydb all 192
host all all 0.0.0.0/0 password
```

Порядок применения правил:

Правила применяются последовательно сверху вниз. Первое подходящее правило для текущего подключения считается применённым. Поэтому важно располагать более строгие ограничения выше менее строгих.

Изменение настроек:

После изменения файла pg_hba.conf необходимо перезагрузить конфигурацию БД командой:

```
sudo systemctl reload rudb-18
```

или вручную через SQL-команду:

```
SELECT pg_reload_conf();
```

6.Сервисы RuDB

6.1. Сервис rudb-18

rudb-18 – это основной сервис СУБД, обеспечивающий работоспособность всех её функциональных возможностей. Данный сервис является ядром СУБД.

Ядро СУБД представляет собой центральный компонент, обеспечивающий выполнение основных функций обработки данных и управление ими. Оно отвечает за обработку запросов пользователей, обеспечение целостности данных, оптимизацию производительности и поддержку параллельной работы множества клиентов одновременно.

6.2. Сервис dgrid

dgrid – это сервис встроенной in-memory-компоненты DGrid, предназначенной для организации хранения данных в оперативной памяти внутри процесса СУБД. Компонента предоставляет механизмы для оперативного добавления, выборки и проверки записей, используя память как основное хранилище. Это решение предназначено для использования в качестве высокопроизводительного кэша или буфера для рабочих данных внутри базы данных, что позволяет ускорить обработку часто используемой информации и снизить нагрузку на основную табличную структуру. Подробнее о работе с компонентой читайте далее в отдельном разделе.