



RuBackup

**Система резервного копирования
и восстановления данных**

ОСНОВНЫЕ СВЕДЕНИЯ

ВЕРСИЯ 2.5.7.0.0, 23.05.2025

Содержание

Поддерживаемые продукты	3
Преимущества	4
1. Функции	5
1.1. Надежность и производительность	5
1.2. Автоматизация	5
1.3. Управляемость	5
1.4. Безопасность	6
2. Ключевые понятия	7
3. Архитектура	10
4. Многопользовательская модель	13
5. Основные компоненты RuBackup	16
5.1. Клиент резервного копирования	16
5.1.1. Запуск клиента RuBackup	16
5.2. Сервер резервного копирования	18
5.2.1. Запуск сервера RuBackup	18

Система резервного копирования RuBackup — клиент-серверное приложение, которое:

- автоматически выполняет резервное копирование СУБД, виртуальных машин, почтовых систем, файловых систем, подсистемы Linux и службы каталогов;
- восстанавливает данные из резервных копий по запросу.

Полностью российская разработка с возможностью гибкой адаптации под требования заказчика.

Для быстрого создания системы обеспечения сохранности данных используйте программный комплекс [RuBackup OneClick](#). Продукт ориентирован на малый, средний бизнес и территориально распределенные организации.

Поддерживаемые продукты

СУБД	Системы виртуализации (безагентный способ)	Почтовые системы
Tantor Special Edition (с использованием модуля PostgreSQL) Arenadata Greenplum Microsoft SQL Server MySQL Oracle Database PostgreSQL + в кластере Patroni Postgres Pro SAP HANA	ПК СВ «Брест» VMmanager АЭРОДИСК vAir P-Виртуализация РУСТЭК Basis DynamiX Enterprise KVM Microsoft Hyper-V OpenStack oVirt/zVirt/REDVirt Proxmox VE ROSA Space VM Tionix VMware vSphere	RuPost CommuniGate Pro VK Workmail
Файловые системы	Подсистема Linux	
Linux (Ext4, Ext3, Ext2, XFS, ZFS, BTRFS) Windows (NTFS)	LVM Linux	
Службы каталогов	Хранилища	
FreelPA ALD Pro MS AD	Файловые хранилища Блочные устройства Облачные хранилища Ленточные библиотеки Клиентские хранилища	

Преимущества

- Лучшая производительность среди российских решений
- Сертификат соответствия ФСТЭК России №4879
- Единственное решение с многопоточностью — на всех этапах позволяет выполнять самые жесткие требования к срокам RPO и RTO
- Широкие возможности интеграции — полнофункциональный REST API, толстый клиент и Web, CLI, документация для интеграторов и клиентов
- Надежность и масштабируемость — встроенные алгоритмы кластеризации и балансировки нагрузки между узлами СРК, резервирование собственных компонентов СРК
- Глубокая интеграция с Postgres — поддержка инкрементальных и дифференциальных копий (PTRACK, DELTA, PAGE), использование механизмов работы с томами (LVM и аппаратные снапшоты)

Быстрый старт

Начните работу в корпоративной среде

Глава 1. Функции

1.1. Надежность и производительность

- Полное, инкрементальное и дифференциальное [резервное копирование](#)
- [Хранение](#) резервных копий в СХД, ленточных библиотеках, облаке S3
- Автоматическая [верификация](#) резервных копий (размер файлов, md5sum, электронная подпись)
- [Сжатие](#) резервных копий на клиенте СРК или на сервере
- [Срочное резервное копирование](#) по инициативе клиента СРК или администратора
- Параллелизм — количество одновременных сессий ограничено только аппаратными характеристиками сервера. Параллельные сессии доступны как для СРК в целом, так и для отдельного клиента

1.2. Автоматизация

- Аналитика — построение плана резервного копирования с прогнозированием требуемых ресурсов
- Экономия дискового пространства — автоматическое перемещение резервных копий на другие носители и удаление устаревших копий
- Балансировка нагрузки — распределение копий по разным хранилищам в зависимости от выбранной политики
- Глобальное расписание — автоматическое создание резервных копий клиентских устройств
- Локальное расписание — клиенты могут управлять резервным копированием самостоятельно
- Стратегии резервного копирования — автоматические групповые операции с клиентами СРК

1.3. Управляемость

- Полноценное управление СРК из [командной строки](#) (CLI)
- Графические приложения для [клиента](#) и для [администратора](#) СРК
- [Веб-приложение](#) для администратора СРК, адаптированное под мобильные устройства
- Взаимодействие с СРК через [REST API](#)

1.4. Безопасность

- [Многопользовательская модель](#) администрирования
- Локальный лист запретов (с regex) для каждого клиента, ограничивающий доступную для копирования информацию
- Шифрование резервных копий по [алгоритмам](#) ГОСТ 34-12-2015 (Kuznyechik), Anubis, ARIA, CAST6, Camellia, Kalyna, MARS, AES, Serpent, Simon, SM4, Speck, Treefish, Twofish
- Интеграция с [хранилищем секретов](#)
- [Рассылка уведомлений](#) пользователям о событиях в СРК
- [Протоколирование](#) всех действий администратора и пользователей в базе данных и системном журнале

Глава 2. Ключевые понятия

Серверная группировка RuBackup состоит из основного сервера, необязательного резервного сервера и медиасерверов. В простейшем случае медиасервером является основной сервер резервного копирования (а также резервный сервер, при наличии).

Клиент системы резервного копирования — это отдельный сервер, компьютер или виртуальная машина, на которой установлено клиентское ПО RuBackup для выполнения резервного копирования. Для удобства клиенты могут быть объединены в **группы клиентов**.

На программном уровне сервером RuBackup называется также фоновый процесс (сервис) на сервере СРК, а клиентом RuBackup — фоновое клиентское ПО.

Хранение данных резервных копий (архивов) реализовано в виде хранилищ (storage). Каждое **хранилище** входит в определенный **пул**. Пул — это логическое объединение однотипных устройств хранения резервных копий. Каждый **пул** принадлежит определенному **медиасерверу**. Таким образом, организация хранения данных резервных копий имеет следующую структуру:

Медиасервер → Пул → Хранилище

Метаданные резервных копий хранятся в **репозитории**. Непосредственно **резервные копии** располагаются в **хранилищах** резервных копий, которые ассоциированы с **пулами** хранения резервных копий. Хранилища бывают пяти типов:

1. файловая система;
2. ленточная библиотека;
3. облако;
4. блочные устройства;
5. определяемые клиентом.

Все действия СРК реализованы в виде **задач**, которые объединены в **очереди задач**, в зависимости от типа.

Периодические задания резервного копирования и восстановления данных реализованы в виде **правил глобального расписания**, которые входят в **глобальное расписание** резервного копирования ([Рисунок 1](#)).

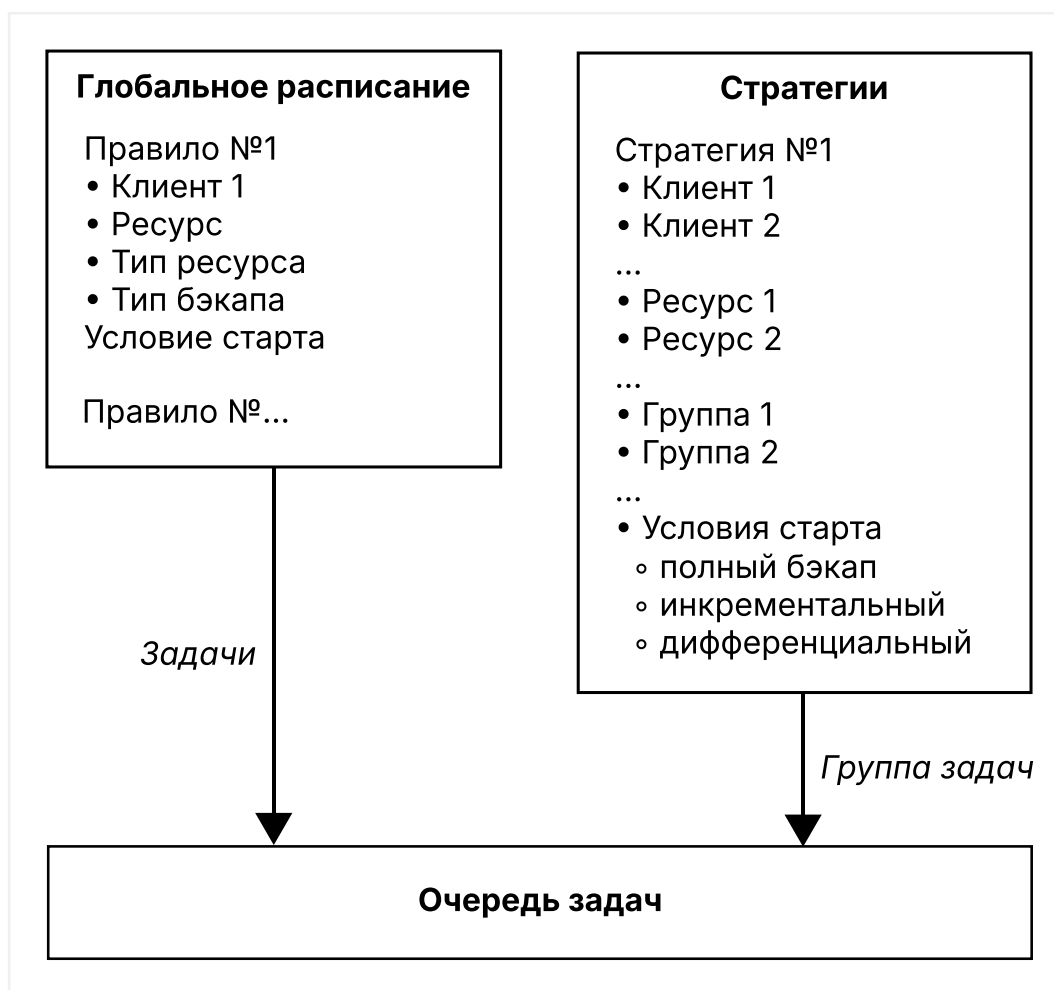


Рисунок 1.

Одновременные действия над группами ресурсов реализованы в виде **стратегий**, которые создают **задачи** резервного копирования в соответствии с **расписаниями** для всех ресурсов и клиентов, которые их касаются.

Система уведомлений RuBackup использует **пользователей** и **группы пользователей** RuBackup для уведомления о событиях системы резервного копирования.

Автономный режим работы клиента — использование клиента СРК RuBackup без сервера резервного копирования. При этом сохраняется возможность использования некоторых клиентских функциональных модулей для создания резервных копий. Чтобы узнать, поддерживается ли использование модуля в автономном режиме, запустите исполнимый файл модуля с опцией `--autonomous` и проверьте код возврата.

Пример 1. Команда проверки поддержки автономного режима для модуля `rb_module_filesystem`

```
sudo /opt/rubackup/modules/rb_module_filesystem -autonomous
```

Пример 2. Команда проверки кода возврата

```
echo $?
```

Код возврата «0» говорит о том, что модуль поддерживает автономный режим, другие коды возврата говорят о том, что автономный режим не поддерживается.

Неинтерактивный режим работы — режим для сценариев массового развертывания, например при использовании Ansible.



Резервный сервер и медиасервер не функционируют с тестовой лицензией!

Глава 3. Архитектура

В минимальной конфигурации СРК RuBackup представляет собой один сервер резервного копирования и один клиент резервного копирования, установленный на том же хосте, на котором работает сервер резервного копирования.

Сервер резервного копирования представляет собой системное фоновое приложение (служба, демон), внутри которого одновременно выполняются множество потоков, отвечающих за разные функции системы резервного копирования.

В простейшем случае единственный сервер резервного копирования взаимодействует с клиентами, координирует задания СРК и хранит резервные копии на доступных ему ресурсах: файловых системах, картриджах ленточных библиотек и облачных сервисах.

В случае обслуживания высококритичных сервисов, система резервного копирования может быть дополнена резервным сервером. В случае отказа основного сервера, резервный сервер автоматически поддержит функционал основного сервера RuBackup, а клиенты системы резервного копирования автоматически подключатся к резервному серверу. После восстановления функционирования основного сервера, клиенты подключатся обратно к основному серверу.

Взаимодействие между системой резервного копирования и ее клиентами обеспечивает основной сервер резервного копирования RuBackup, либо резервный сервер, если он функционирует в режиме замещения основного сервера.

И основной, и резервный серверы включают в себя функционал медиасервера. Медиасервер предназначен для хранения резервных копий, получения их от клиентов и передачи клиентам файлов резервных копий по запросу.

При увеличении количества клиентов, а также при увеличении количества ресурсов, на которых предполагается хранить резервные копии, могут возникнуть задачи распределения нагрузки. В этом случае в серверную группировку могут быть добавлены медиасерверы, с помощью которых можно перераспределить задачи резервного копирования на несколько серверов резервного копирования или построить иерархическую систему хранения резервных копий.

Система резервного копирования RuBackup может выполнять полное, инкрементальное и дифференциальное (разностное) резервное копирование информационных ресурсов разных типов: отдельные файлы и каталоги, блочные устройства, на которых располагаются сырые данные или файловые системы, логические тома LVM, виртуальные машины и базы данных. Функционал резервного копирования и восстановления ресурсов разных типов реализован в соответствующих модулях на клиенте.

Полное резервное копирование — это создание резервной копии всех данных из

исходного набора, независимо от того, изменялись ли данные с момента выполнения последней полной резервной копии.

Дифференциальное (разностное) резервное копирование сохраняет только данные, измененные со времени выполнения предыдущего полного резервного копирования.

Инкрементальное резервное копирование сохраняет только данные, измененные со времени выполнения предыдущей инкрементальной резервной копии, а при отсутствии таковой — со времени выполнения последней полной резервной копии.

Резервное копирование выполняется по заранее заданным правилам в глобальном расписании RuBackup, а также в соответствии с правилами локального расписания клиента, если это разрешено клиенту администратором RuBackup. Также клиенту доступно срочное резервное копирование тех или иных файлов, но в этом случае выполняется полное резервное копирование выбранного ресурса.

Восстановление резервной копии возможно по инициативе клиента. Для восстановления данных пользователь должен ввести пароль, позволяющий выполнить восстановление.

Резервное копирование может быть выполнено с применением сжатия на стороне клиента или на стороне сервера RuBackup. Возможно произвести защитное преобразование резервной копии выбранным алгоритмом. При необходимости резервная копия может быть подписана цифровой подписью на стороне клиента для последующего контроля и предупреждения угрозы ее подмены.

Система резервного копирования может быть настроена таким образом, что резервные копии будут перемещаться на другие устройства хранения (например с дискового устройства хранения на картридж ленточной библиотеки) по достижении определенного срока хранения. Устаревшие резервные копии могут быть удалены из СРК автоматически или сообщение о том, что их следует удалить, будет отправлено администраторам СРК. Время от времени может выполняться проверка резервных копий по разным критериям.

Общий объем резервных копий, хранящихся в системе резервного копирования, может быть ограничен для клиента СРК, или для правила резервного копирования, или для стратегии резервного копирования.

Правила резервного копирования глобального расписания RuBackup имеют определенные время и даты начала и окончания действия. При необходимости правило можно выключить или вновь включить в работу.

Внутренние автоматические работы с резервными копиями — перемещение, удаление, проверка — осуществляются в заранее определенное сервисное окно, чтобы данные операции не пересекались с операциями резервного копирования.

Особое внимание в системе резервного копирования RuBackup уделено вопросам разграничения доступа к резервным копиям. Ключи для защитного преобразования резервных копий располагаются на клиенте и не могут быть скопированы при выполнении резервного копирования (исключаются принудительно из резервных копий). Чтобы восстановить резервную копию требуется ввести пароль, который задается при начале работы клиента с системой резервного копирования. В базе данных системы резервного копирования пароли клиентов не хранятся в чистом виде, но в виде хешей.

Управление системой резервного копирования может осуществляться как с помощью оконных средств администрирования, так и с использованием утилит командной строки.

Базовая конфигурация RuBackup, как клиента, так и сервера, содержится в конфигурационном файле `/opt/rubackup/etc/config.file`. Этот файл содержит информацию об основном и резервном серверах резервного копирования и режиме работы узла (основной сервер, резервный сервер, медиасервер или клиент) и т.п.

Глобальные настройки системы резервного копирования, а также информация о клиентах СРК, глобальном расписании, стратегиях, репозитории резервных копий и пр. хранятся в базе данных rubackup в СУБД PostgreSQL. Для изменения большинства параметров конфигурации СРК не требуется изменять какие-либо сложные конфигурационные файлы и останавливать функционирование СРК. Изменения производятся online с помощью штатных средств администрирования RuBackup.

Клиент RuBackup имеет модульную архитектуру. Клиент RuBackup отвечает за взаимодействие с сервером RuBackup с одной стороны, и с модулями резервного копирования и восстановления с другой стороны. Собственно процедуры резервного копирования и восстановления реализованы в модулях RuBackup. Модуль RuBackup — это утилита, которая отвечает за резервное копирование и восстановление ресурса определенного типа (например, блочных устройств или базы данных) и упаковку резервных копий.

API модуль RuBackup является открытым и может быть использован для разработки модулей третьими лицами. Модули подробно описаны в соответствующих документах.

Глава 4. Многопользовательская модель

В СРК RuBackup реализован многопользовательский режим работы, т. е. назначение типа пользователя и предоставление ему набора полномочий для выполнения определенных рабочих задач в соответствии с его ролью.

В СРК RuBackup предусмотрены следующие типы пользователей:

1. суперпользователь (владелец базы данных RuBackup);
2. супервайзер;
3. сопровождающий;
4. администратор;
5. аудитор.

Суперпользователь является привилегированным администратором, которому позволены любые действия в СРК. Суперпользователь создаётся при конфигурации основного сервера. Имя суперпользователя и пароль задаются также при конфигурации. Чтобы поменять пароль суперпользователя в конфигурационном файле сервера, используйте команду:

```
rb_init -passwd
```

```
root@rbs:~# rb_init --passwd
RuBackup initialization utility
Copyright 2018-2022: LLC "RUBACKUP"
Исключительные права принадлежат ООО "РУБЭКАП"
Author is Andrey Kuznetsov
Version: 2.0 Build: 48024de
password found in /opt/rubackup/etc/config.file

Please enter old password:
Enter new password:
Repeat password:
  Copy old config file to: /opt/rubackup/etc/config.file.old.2024-Jan-18H16-
05-32
Password was changed successfully
root@rbs:~#
```

Для смены пароля в служебной базе данных rubackup:

1. Подключитесь к базе данных, используя пользователя rubackup или postgres, с помощью команды:

```
sudo -u rubackup psql
```

или

```
sudo -u postgres psql
```

2. Выполните команду:

```
sql ALTER USER rubackup PASSWORD '<new-password>';
```

Суперпользователь создается при создании базы данных rubackup и является владельцем базы данных. Таким образом, в списке пользователей СРК пользователя Суперпользователя увидеть нельзя, также как и нельзя создать еще одного пользователя с таким же именем.

У суперпользователя есть следующие возможности:

- добавлять новых пользователей в систему. При этом выбранная группа пользователя влияет только на задачи уведомления. Чтобы пользователь мог получить административные привилегии в СРК, его нужно добавить в супервайзеры, сопровождающие или администраторы;
- менять пароль для других пользователей с помощью RBM.

Супервайзер может выполнять действия, доступные Суперпользователю, за исключением:

- любых действий с пользователями кроме назначения ролей Сопровождающего и Администратора;
- изменения глобальной конфигурации СРК.

Сопровождающий отвечает за медиасервер и может управлять устройствами хранения на этом медиасервере.

Администратор отвечает за группу клиентов и может выполнять их настройки и действия, связанные с клиентами, входящими в группу. Администратор в дереве объектов видит только своих клиентов, и имеет доступ к правилам глобального расписания, резервным копиям и задачам только своих клиентов.

Аудитор — роль, предназначенная для сотрудников информационной безопасности. Аудитору доступен просмотр всех настроек и информации в СРК (кроме настроек глобальной конфигурации) без возможности редактирования. Также аудитору доступны для просмотра все журналы, включая «Журнал событий ИБ».

Порядок назначения типов пользователя, их поиска и удаления можно найти в [RBManual:ROOT:page\\$users.pdf](#).

Глава 5. Основные компоненты RuBackup

5.1. Клиент резервного копирования

Клиент резервного копирования RuBackup представляет собой фоновое приложение (сервис, демон), взаимодействующее с сервером RuBackup.

Расположение	<code>/opt/rubackup/bin/rubackup_client</code>
Запуск	<code>rubackup_client start</code>
Остановка	<code>rubackup_client stop</code>
Перезагрузка	<code>rubackup_client restart</code>
Текущий статус (результат 0 — клиент работает, 1 — не работает)	<code>rubackup_client status</code>
Получить HWID	<code>rubackup_client hwid</code>

5.1.1. Запуск клиента RuBackup

Для штатной эксплуатации рекомендуется запускать клиент RuBackup как сервис. Для этого выполните следующие действия:

1. Включите сервис клиента RuBackup:

```
sudo systemctl enable \
/opt/rubackup/etc/systemd/system/rubackup_client.service
```

2. Перезагрузите `systemctl`:

```
sudo systemctl daemon-reload
```

3. Запустите сервис `rubackup_client`:

```
sudo systemctl start rubackup_client
```

Уточнить статус клиента RuBackup можно при помощи команды:

```
sudo systemctl status rubackup_client
```

```
rubackup_client.service - RuBackup client
```

```
Loaded: loaded (/etc/systemd/system/rubackup_client.service; enabled;
vendor preset: enabled)
```

```
Active: active (running) since Mon 2023-02-20 11:17:59 UTC; 6 days ago
```

```
Process: 1760 ExecStart=/opt/rubackup/bin/rubackup_client start
(code=exited, status=0/SUCCESS)
```

```
Main PID: 1763 (rubackup_client)
```

```
Tasks: 3 (limit: 4610)
```

```
Memory: 60.9M
```

```
CGroup: /system.slice/rubackup_client.service
```

```
1763 /opt/rubackup/bin/rubackup_client start
```

```
фев 20 12:18:07 rb-primary rubackup_client[1763]: [2023-02-20 12:18:07]
```

```
Info: Removing obsolete snapshot file:
```

```
/rubackup-tmp/rb-
```

```
primary_TaskID_1_NORuleOrStrategy_0_D2023_2_20H12_14_16_BackupType_1_Resource
Type_>
```

```
фев 20 12:18:07 rb-primary rubackup_client[1763]: [2023-02-20 12:18:07]
```

```
Info: bool RbModuleUniversal::run_rbfd_command(const string&,
std::string&, pid_t&, std::string&):rbfd command: /opt/rubackup/bin/rbfd -a >
```

```
фев 20 12:18:07 rb-primary rubackup_client[1763]: Rbfd PID: 62636
```

```
фев 20 12:18:07 rb-primary rubackup_client[1763]: Set status for task
ID: 3 from: Start_Transfer to: Transmission
```

```
фев 20 12:18:10 rb-primary rubackup_client[1763]: [193B blob data]
```

```
фев 20 12:18:10 rb-primary rubackup_client[1763]: [2023-02-20 12:18:10]
```

```
Info: The archive '1' has been unpacked successfully
```

```
фев 20 12:18:10 rb-primary rubackup_client[1763]: Final progress: 7832
100%
```

```
фев 20 12:18:10 rb-primary rubackup_client[1763]: Set status for task
ID: 3 from: Transmission to: Finish_Transfer
```

```
фев 20 12:18:10 rb-primary rubackup_client[1763]: Set status for task  
ID: 3 from: Finish_Transfer to: Done
```

```
фев 20 12:18:10 rb-primary rubackup_client[1763]: Task w
```

5.2. Сервер резервного копирования

Сервер резервного копирования RuBackup представляет собой фоновое приложение (сервис, демон).

Расположение	<code>/opt/rubackup/bin/rubackup_server</code>
Запуск	<code>rubackup_server start</code>
Остановка	<code>rubackup_server stop</code>
Перезагрузка	<code>rubackup_server restart</code>
Текущий статус (результат 0 — сервер работает, 1 — не работает)	<code>rubackup_server status</code>
Получить HWID	<code>rubackup_server hwid</code>

5.2.1. Запуск сервера RuBackup

Для штатной эксплуатации рекомендуется запускать сервер RuBackup как сервис. Для этого выполните следующие действия:

1. Включите сервис клиента RuBackup:

```
sudo systemctl enable \  
/opt/rubackup/etc/systemd/system/rubackup_client.service
```

2. Включите сервис сервера RuBackup:

```
sudo systemctl enable \  
/opt/rubackup/etc/systemd/system/rubackup_server.service
```

3. Перезагрузите `systemctl`:

```
sudo systemctl daemon-reload
```

4. Запустите сервис `rubackup_client`:

```
sudo systemctl start rubackup_client
```

5. Запустите сервис `rubackup_server`:

```
sudo systemctl start rubackup_server
```

Уточнить статус сервера RuBackup можно при помощи команды:

```
sudo systemctl status rubackup_server
```

```
rubackup_server.service - RuBackup server
```

```
Loaded: loaded (/etc/systemd/system/rubackup_server.service; enabled;  
vendor preset: enabled)
```

```
Active: active (running) since Mon 2023-02-20 11:19:36 UTC; 6 days ago
```

```
Process: 1897 ExecStart=/opt/rubackup/bin/rubackup_server start  
(code=exited, status=0/SUCCESS)
```

```
Main PID: 1912 (rubackup_server)
```

```
Tasks: 29 (limit: 4610)
```

```
Memory: 254.0M
```

```
CGroup: /system.slice/rubackup_server.service
```

```
1912 /opt/rubackup/bin/rubackup_server start
```

```
фев 27 07:26:20 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1  
has no any file system
```

```
фев 27 07:26:21 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1  
has no any file system
```

```
фев 27 07:26:22 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1  
has no any file system
```

```
фев 27 07:26:23 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1  
has no any file system
```

```
фев 27 07:26:24 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1
has no any file system

фев 27 07:26:25 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1
has no any file system

фев 27 07:26:26 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1
has no any file system

фев 27 07:26:27 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1
has no any file system

фев 27 07:26:28 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1
has no any file system

фев 27 07:26:29 rb-primary rubackup_server[1912]: Warning: Pool: Cloud1
has no any file system
```

Если у вас возникает проблема запуска сервиса RuBackup, и служебная база данных RuBackup в PostgreSQL установлена на отдельном сервере (например, при добавлении в конфигурацию резервного или медиасервера), выполните следующие действия:

1. Удалите зависимости `postgresql.service` в параметрах Requires и After в разделе Unit в юнит-файле:

```
/opt/rubackup/etc/systemd/system/rubackup_server.service
```

2. Перезагрузите `systemctl`:

```
sudo systemctl daemon-reload
```