



RuBackup

**Система резервного копирования
и восстановления данных**

RUPOST

ВЕРСИЯ 2.5.7.0.0, 08.07.2025

Содержание

1. Назначение	3
2. Требования к программным средствам	4
3. Резервируемые данные	5
4. Типы резервного копирования	6
5. Способы восстановления данных	7
6. Комплект поставки	8
7. Ограничения	9
8. Установка	10
8.1. Установка модуля	10
8.1.1. Подготовка к установке модуля	10
8.1.2. Установка	10
8.1.3. После установки	11
8.1.4. Критерий успешности установки	11
9. Подготовка к работе	12
9.1. Подготовка «холодного» хранилища	12
9.2. Подготовка СУБД PostgreSQL	13
9.2.1. Подготовка сервера с СУБД PostgreSQL	13
9.2.2. Создание пользователя СУБД для безопасного выполнения резервной копии PostgreSQL	15
10. Работа с данными	17
10.1. Резервное копирование	18
10.1.1. Создание правила глобального расписания	18
10.1.2. Регулярное резервное копирование	19
10.1.3. Срочное резервное копирование по правилу	19
10.1.4. Срочное резервное копирование (без правила)	19
10.2. Восстановление данных	21
10.2.1. Централизованное восстановление резервных копий с помощью ПО Менеджер Администратора RuBackup (RBM)	21
10.2.2. Восстановление на целевом ресурсе	22
Восстановление почтовых ящиков пользователей	22
Восстановление архивов почтовых ящиков пользователей	22
Восстановление удаленных сообщений пользователей	22
Восстановление базы данных	22
10.3. Просмотр очереди задач	23
11. Приложения	24

11.1. Файл настроек <code>rb_module_rupost.conf</code>	24
11.2. Тонкие настройки модуля RuPost для резервного копирования	26
11.3. Тонкие настройки модуля RuPost для восстановления резервной копии . .	28

Глава 1. Назначение

Модуль *RuPost* позволяет СРК *RuBackup* выполнять резервное копирование и восстановление компонентов корпоративной почтовой системы *RuPost*.

Глава 2. Требования к программным средствам

Для выполнения резервного копирования и восстановления почтовой системы *RuPost* с помощью модуля *RuPost* на узле необходимо предустановленное программное обеспечение:

- 64-битная операционная система *Astra Linux SE 1.7*;
- Клиент резервного копирования *RuBackup*;
- Модуль *RuPost*;
- Служебная база данных почтовой системы *RuPost*;



Если служебная БД имеет конфигурацию *Patroni*, клиент РК должен быть установлен на каждом хосте кластера *Patroni*.

- NFS-клиент для монтирования холодного хранилища почтовой системы *RuPost* версии 3.2.

Для управления резервным копированием и восстановлением почтовой системы *RuPost* рекомендуем использовать программное обеспечение *Менеджер Администратора RuBackup (RBM)* (см. [RuBackup Manager \(RBM\)](#)).

Глава 3. Резервируемые данные

Резервное копирование выполняется для:

- данных пользователей из холодного хранилища почтовой системы:
 - почтовые ящики;
 - архивы почтовых ящиков;
 - удаленные сообщения;
 - календари и расписания.
- служебной базы данных почтовой системы.

Глава 4. Типы резервного копирования

Модуль *RuPost* поддерживает полное и инкрементальное резервное копирование почтовой системы *RuPost*.

При инкрементальном резервном копировании создаётся инкрементальная РК компонентов из холодного хранилища и полная резервная копия служебной базы данных почтовой системы *RuPost*.

Подробнее о типах резервного копирования см. в [Архитектура и инфраструктура](#).

В этом документе приведены инструкции по созданию РК с помощью программы *Менеджер Администратора RuBackup* (см. [Раздел 10.1](#)).

Глава 5. Способы восстановления данных

Модуль *RuPost* поддерживает следующие способы восстановления резервных копий:

- централизованное восстановление данных из резервной копии без развертывания. Резервная копия будет распакована в каталог распаковки;
- централизованное восстановление данных из резервной копии с развертыванием. Компоненты почтовой системы будут распакованы в холодное хранилище, минуя временный каталог.



СРК не восстанавливает компоненты почтовой системы *RuPost* непосредственно в почтовую систему. Данные восстанавливаются только в папки **Restore** холодного хранилища. Дальнейшее развертывание в почтовую систему осуществляется силами администратора *RuPost Server*.

Служебная база данных будет распакована во временный каталог распаковки, а затем развернута в почтовой системе *RuPost*.



Развёртывание базы данных возможно только в конфигурации *standalone*. В кластере *Patroni* подмену базы и перезапуск кластера осуществляет администратор базы данных.

В этом документе приведены инструкции по восстановлению РК с помощью программы *Менеджер Администратора RuBackup* (см. [Раздел 10.2](#)).

Глава 6. Комплект поставки

Дистрибутив модуля *RuPost* СРК *RuBackup* поставляется в виде deb-пакета с именем `rubackup-rupost-<version>_amd64.deb`, где `<version>` - номер версии поставляемого модуля.

Глава 7. Ограничения

- СРК не обеспечивает синхронность данных при резервном копировании данных из холодного хранилища и соответствующих метаданных в служебной БД *RuPost*. При восстановлении РК возможна ситуация, когда данные в служебной БД *RuPost* являются более новыми и не соответствуют данным, находящимся в холодном хранилище.
- Запуск резервного копирования осуществляется вручную. Время начала резервного копирования устанавливается таким образом, чтобы к моменту старта резервного копирования *RuPost* завершил репликацию данных в холодное хранилище.
- Модуль *RuPost* не поддерживает взаимодействие с СРК при помощи веб-приложения *Tucana*.
- Не поддерживается взаимодействие с СРК с помощью утилит командой строки.
- Резервные копии компонентов почтовой системы *RuPost* текущей версии могут быть восстановлены в будущие версии почтовой системы при условии обратной совместимости данных версий почтового клиента *RuPost*.
- Со стороны СРК не предусмотрена возможность контроля объема свободного места в каталогах временного хранилища резервных копий и в каталогах распаковки.
- Не предусмотрено использование блочных устройств, ленточных библиотек и облачных хранилищ в качестве хранилищ резервных копий.
- Модуль *RuPost* не поддерживает резервное копирование и восстановление СУБД *PostgreSQL Pro*.
- Если после резервного копирования статус письма был изменен с *непрочитанное* на *прочитанное*, при восстановлении статус будет соответствовать сохраненному в резервной копии, то есть *непрочитанное*.

Глава 8. Установка

8.1. Установка модуля

8.1.1. Подготовка к установке модуля

На узле, где будет произведена установка модуля для резервного копирования данных почтовой системы *RuPost*, предварительно должен быть развёрнут (см. [Установка пакетов](#)) и настроен (см. [Настройка клиента РК](#)) клиент резервного копирования, подключенный к основному серверу СРК *RuBackup*.

Если вы планируете восстановление почтовой системы через *Менеджер администратора RuBackup (RBM)*, включите на узле клиента резервного копирования функцию централизованного восстановления.

Функция централизованного восстановления управляется параметром `centralized-recovery` конфигурационного файла `/opt/rubackup/etc/config.file` на узле клиента:

Включение централизованного восстановления (`/opt/rubackup/etc/config.file`)

```
centralized-recovery  yes
```

8.1.2. Установка

На подготовленном узле клиента резервного копирования:

1. Остановите сервис клиента резервного копирования *RuBackup*:

```
sudo systemctl stop rubackup_client.service
```

2. Выполните установку модуля *RuPost* СРК *RuBackup*:

```
sudo dpkg -i rubackup-rupost-<version>_amd64.deb
```

где `<version>` — номер версии модуля *RuPost* СРК *RuBackup*.

В ходе установки модуля выполняются распаковка и настройка пакета модуля *RuPost rubackup-rupost*.

В результате установки пакета модуля *RuPost* в систему будут добавлены файлы, приведенные в [таблице](#):

Таблица 1. Перечень устанавливаемых в систему файлов

Расположение	Назначение
<code>/opt/rubackup/etc/rb_module_rupost.conf</code>	Файл настроек модуля <i>RuPost</i> СРК <i>RuBackup</i>
<code>/opt/rubackup/modules/rb_module_rupost</code>	Исполняемый файл модуля <i>RuPost</i> СРК <i>RuBackup</i>

8.1.3. После установки

После установки модуля:

1. Настройте параметры в файле настроек `/opt/rubackup/etc/rb_module_rupost.conf` модуля *RuPost* (см. [Раздел 11.1](#)).
2. Перезапустите сервис клиента резервного копирования *RuBackup*:

```
sudo systemctl restart rubackup_client.service
```

8.1.4. Критерий успешности установки

Об успешной установке и настройке модуля *RuPost* свидетельствует запись о его успешной проверке клиентом резервного копирования (... module 'RuPost' was checked successfully) в журнале событий `/opt/rubackup/log/RuBackup.log`.

Если в журнале событий `/opt/rubackup/log/RuBackup.log` администратор СРК видит ошибку, сообщающую о неправильной конфигурации модуля *RuPost*, проверьте настройки в файле `/opt/rubackup/etc/rb_module_rupost.conf`, выполнив на узле клиента резервного копирования команду:

```
/opt/rubackup/modules/rb_module_rupost -t
```

Если ошибка не поддается анализу, то обратитесь в сервис технической поддержки *RuBackup* с предоставлением всей необходимой информации по возникшей проблеме на официальном сайте <https://support.rubackup.ru/bugzilla>.

Глава 9. Подготовка к работе

С точки зрения резервного копирования и восстановления, система RuPost состоит из СУБД PostgreSQL и «холодного» хранилища писем.

Модуль должен располагаться на узле с СУБД PostgreSQL. Если СУБД развёрнута в кластере Patroni, то модуль должен быть установлен и настроен на каждом узле кластера.

9.1. Подготовка «холодного» хранилища

«Холодное» хранилище писем должно быть примонтировано к узлу с модулем с помощью NFS (к каждому узлу кластера).

Предварительно на узле с модулем должен быть установлен клиент NFS:

```
sudo apt update

sudo apt install nfs-common
```

Чтобы примонтировать «Холодное хранилище» на системе на базе Deb (Ubuntu, Debian, Mint и др.) используйте команду:

```
sudo mount -t nfs ip_address:/nfs_src_folder destination_folder
```

Где `ip_address` - это IP-адрес узла с «холодным» хранилищем писем,

`nfs_src_folder` - исходное расположение «холодного» хранилища писем,

`destination_folder` - точка монтирования «холодного» хранилища писем на узле с модулем.

Для того, чтобы монтирование осталось после перезагрузки узла с модулем, необходимо в файл `/etc/fstab` добавить строку:

```
ip_address:/nfs_src_folder destination_folder nfs auto 0 0
```

В файле конфигурации модуля «`/opt/rubackup/etc/rb_module_rupost.conf`» необходимо указать путь до «холодного» хранилища:

```
cold_storage destination_folder
```

9.2. Подготовка СУБД PostgreSQL

Подготовка СУБД PostgreSQL к выполнению резервного копирования при помощи СРК RuBackup включает в себя:

1. Подготовка сервера с СУБД PostgreSQL;
2. Создание пользователя СУБД для безопасного выполнения резервной копии PostgreSQL.

9.2.1. Подготовка сервера с СУБД PostgreSQL

Для подготовки сервера с СУБД PostgreSQL необходимо выполнить следующие шаги:

1. Для обеспечения доступа пользователя **rubakup_backuper** к СУБД измените метод доступа в конфигурационном файле СУБД PostgreSQL `/etc/postgresql/12/main/pg_hba.conf` (расположение файла может отличаться в зависимости от дистрибутива Linux и версии PostgreSQL) на `md5`.
2. Для непрерывного архивирования и восстановления СУБД PostgreSQL необходимо включить архивирование WAL, для чего:
 - в конфигурационном файле СУБД PostgreSQL `/etc/postgresql/12/main/postgresql.conf` (расположение файла может отличаться в зависимости от дистрибутива Linux и версии PostgreSQL) настройте следующие параметры:

```
wal_level = replica
archive_mode = on
archive_command = 'cp %p /opt/rubakup/mnt/postgresql_archives/%f'
```

- там же установите значение параметра `data_directory` (если оно не определено), иначе модуль резервного копирования не сможет определить местоположение файлов СУБД:

```
data_directory = '/var/lib/postgresql/12/main'
```

- в файле `postgresql.conf` для версий PostgreSQL 12 и более новых должна быть прописана строка, определяющая порядок развертывания СУБД из резервной копии:

```
restore_command = 'cp /opt/rubakup/mnt/postgresql_archives/%f %p'
```

3. После внесения изменений в конфигурационный файл перезапустите

PostgreSQL командой:

```
sudo service postgresql restart
```

Значение параметра `archive_command` должно содержать каталог в файловой системе сервера PostgreSQL, в который будут копироваться архивируемые сегменты WAL.

В настройках RuBackup для каждой СУБД PostgreSQL в файле `/opt/rubackup/etc/rb_module_postgresql.conf` определен параметр `archive_catalog`, содержащий значение каталога, в котором предполагается временное хранение архивных WAL-файлов. Значение этого параметра по умолчанию:

```
/opt/rubackup/mnt/postgresql_archives/
```

При планировании установки СРК RuBackup вы можете назначить для хранения архивных WAL-файлов выделенное хранилище требуемого размера и сделать на него ссылку на том сервере PostgreSQL, где это требуется.

Объем необходимого пространства под архивные WAL-файлы зависит от нагруженности базы данных и периодичности бэкапов, а также от значения параметра `auto_remove_wal` в конфигурационном файле.



Указанный каталог должен быть доступен для записи и чтения пользователю `postgres`, а также пользователю, под контролем которого работает клиент RuBackup!

Для этого воспользуйтесь командой:

```
sudo chown postgres:postgres /opt/rubackup/mnt/postgresql_archives/
```

Для правильной работы клиента RuBackup параметр `archive_catalog` в конфигурации RuBackup и параметр `archive_command` в конфигурационном файле PostgreSQL должны иметь одинаковое значение для одной и той же СУБД.

После изменения параметров конфигурационного файла необходимо перезагрузить PostgreSQL при помощи команды:

```
sudo systemctl restart postgresql
```

9.2.2. Создание пользователя СУБД для безопасного выполнения резервной копии PostgreSQL

Для выполнения операции создания базовой резервной копии пользователь должен обладать правами на выполнение функций начала и окончания резервного копирования экземпляра PostgreSQL. Для настройки выполните следующие действия:

1. Вызовите psql при помощи команды:

```
sudo -u postgres psql
```

2. В psql создайте пользователя rubakup_backuper (в качестве пароля укажите желаемый пароль вместо 12345):

```
create user rubakup_backuper password '12345';
alter role rubakup_backuper with login;
```



В PostgreSQL версии 14 и ниже используются функции `pg_stop_backup` и `pg_start_backup`, а в версии 15 и выше - `pg_backup_stop` и `pg_backup_start`.

```
grant execute on function pg_backup_start to rubakup_backuper;
grant execute on function pg_backup_stop(bool, bool) to rubakup_backuper;
grant execute on function pg_switch_wal to rubakup_backuper;
grant pg_read_all_settings to rubakup_backuper;
```

Вместо пользователя `rubakup_backuper` вы можете создать пользователя с другим именем и с таким же набором прав. В файле конфигурации модуля `/opt/rubakup/etc/rb_module_rupost.conf` необходимо указать имя пользователя и его пароль:

```
cold_storage destination_folder
username rubakup_backuper
password 12345
port 5432
archive_catalog /opt/rubakup/mnt/postgresql_archives
pg_ctl /usr/lib/postgresql/12/bin/pg_ctl
auto_remove_wal yes
direct_restore yes
postgresql_admin postgres
```

Для параметра `pg_ctl` необходимо указать абсолютный путь для используемой версии PostgreSQL.

Глава 10. Работа с данными

Доступно полное или инкрементальное резервное копирование. При выборе инкрементального типа резервного копирования необходимо наличие полной резервной копии, в случае её отсутствия будет выполнено полное резервное копирование. Создание полной резервной копии более длительное, чем создание инкрементальной резервной копии.

При восстановлении инкрементальной резервной копии происходит распаковка архивов в несколько итераций, начиная с полной резервной копии. Поэтому при восстановлении компонентов холодного хранилища почтовой системы в холодном хранилище должно быть достаточно свободного места для распаковки каждой из резервных копий в цепочке. При восстановлении базы данных на узле, где установлен модуль *RuPost*, должно быть достаточно свободного места для распаковки восстанавливаемой базы данных. При восстановлении без развертывания в каталоге, указанном пользователем, должно быть достаточно свободного места для распаковки каждой из резервных копий в цепочке. Так, например, если ресурс на момент создания инкрементальной резервной копии имеет размер 5 ТБ, то в каталоге для распаковки должно быть не менее 5 ТБ+10% свободной памяти.

Схема процесса резервного копирования и восстановления данных почтовой системы *RuPost* приведена на [рисунке](#).

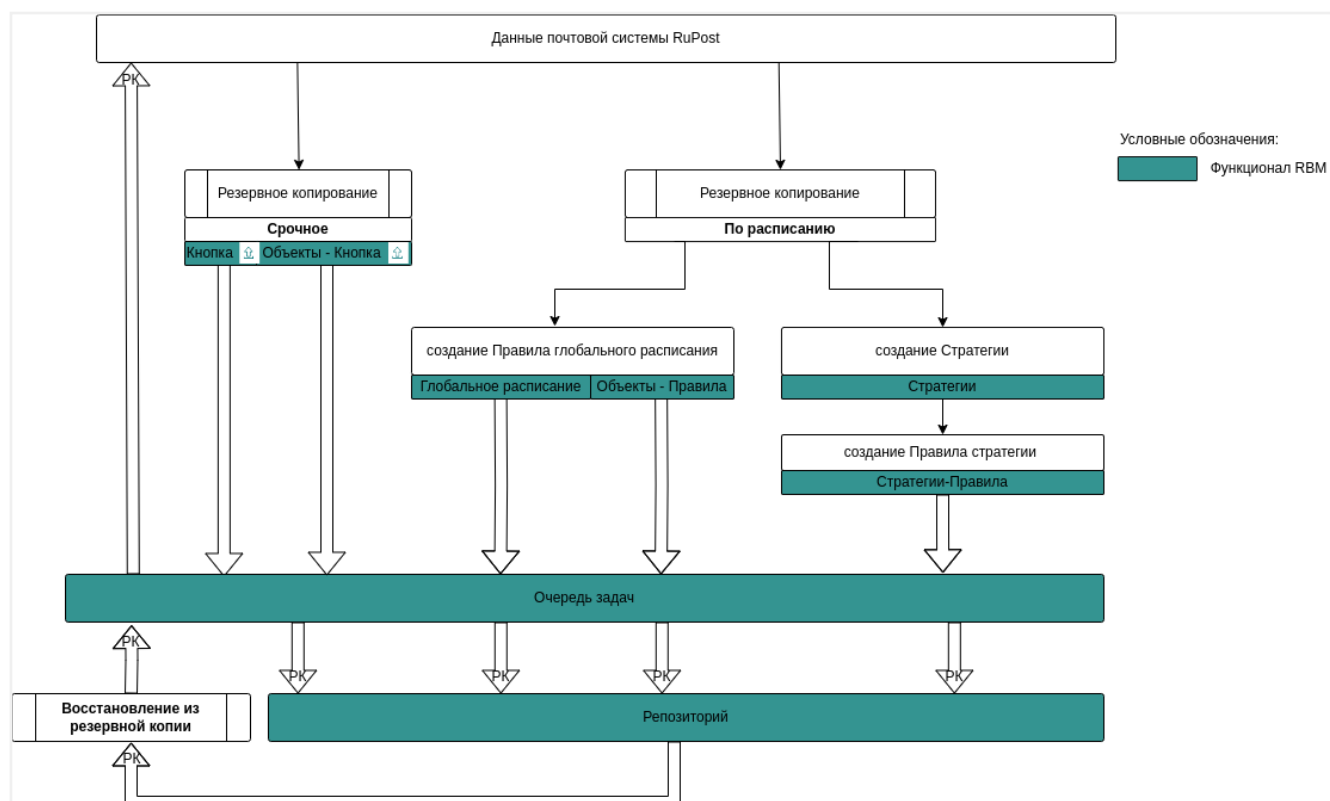


Рисунок 1. Структурная схема резервного копирования и восстановления данных из резервной копии

10.1. Резервное копирование

Подробную информацию о выполнении резервного копирования с помощью ПО *Менеджер Администратора RuBackup (RBM)* см. в [RuBackup Manager \(RBM\)](#).

10.1.1. Создание правила глобального расписания

Для выполнения регулярного (периодического) или срочного (разового) резервного копирования по правилу необходимо создать правило в глобальном расписании.

Правило содержит в себе настройки, необходимые для резервного копирования требуемого ресурса.

1. Перейдите в раздел  **Глобальное расписание** и нажмите  (**Добавить**). Откроется форма создания нового правила глобального расписания.
2. Введите имя создаваемого правила в **Параметры правила**.
3. Из списка **Клиент** выберите узел клиента, на котором установлена почтовая система *RuPost*.
4. Из списка **Тип ресурса** выберите тип резервируемого ресурса **RuPost**. Для выбора резервируемых компонентов почтовой системы нажмите **[...]** и определите *тонкие настройки* модуля *RuPost* (см. [Раздел 11.2](#)).
5. Нажмите **[...]** в поле **Ресурс** и выберите из списка резервируемый компонент почтовой системы.
6. Из списка **Тип РК** выберите тип резервной копии. Возможные значения:
 - полная,
 - инкрементальная.
7. При необходимости задайте дополнительные параметры правила (см. [RBManual:ROOT:page\\$add_global-schedule.pdf](#)), в том числе периодичность его выполнения.
8. Нажмите  **Применить** для сохранения правила.

Созданное правило появится в списке правил:

- раздел  **Объекты** → вкладка **Правила**,
- раздел  **Глобальное расписание**.

Если при создании правила установлен флаг **Включить после создания** , созданное правило будет иметь статус *run*.

Если при создании правила флаг **Включить после создания**  снят, созданное правило будет иметь статус *wait*.

10.1.2. Регулярное резервное копирование

Регулярное резервное копирование выполняется для каждого *включенного* правила в соответствии с заданной этим правилом периодичностью.

Правило в статусе *run* ожидает выполнения в заданную в нём дату и время.

Правило в статусе *wait* приостановлено и не будет выполняться.

1. Перейдите в раздел  **Глобальное расписание**.
2. Если нужное правило резервного копирования есть в списке, проверьте его статус.

Если правило в статусе *run*, ожидайте его выполнения.

Если правило в статусе *wait*, выделите его в списке и установите на панели инструментов флаг **Включить** , или выберите **Выполнить** в контекстном меню этого правила. Правило перейдет в статус *run*.

3. Если правило с нужными настройками отсутствует в списке правил, создайте правило (см. [Раздел 10.1.1](#)). Убедитесь, что в создаваемом правиле установлен флаг **Включить после создания** .

Включенное правило резервного копирования будет выполнено в ближайшую заданную дату и время.

10.1.3. Срочное резервное копирование по правилу

Срочное резервное копирование позволяет выполнить правило резервного копирования немедленно. Срочное резервное копирование по правилу требует настроенного правила в глобальном расписании.

1. Перейдите в раздел  **Глобальное расписание**.
2. Если правило с нужными настройками отсутствует в списке правил, создайте правило (см. [Раздел 10.1.1](#)). Убедитесь, что в создаваемом правиле снят флаг **Включить после создания** .
3. Если нужное правило резервного копирования есть в списке правил, выделите его в списке и нажмите  (**Выполнить**) в панели инструментов, или выберите **Выполнить** в контекстном меню этого правила.

Срочное резервное копирование создает задачу в разделе  **Очередь задач**.

10.1.4. Срочное резервное копирование (без правила)

Срочное резервное копирование (без правила) немедленно выполняет резервное копирование выбранного ресурса, но не создает правила в глобальном расписании.

1. Для выполнения срочного резервного копирования в *RBM* выберите один из способов:

- в разделе **Объекты** выберите клиента резервного копирования, на котором установлена почтовая система *RuPost*, и нажмите  (**Срочное РК**);
- нажмите  (**Срочное РК**) на верхней панели *RBM*.

Откроется форма срочного резервного копирования.

2. Из списка **Клиент** выберите узел клиента, на котором установлена почтовая система *RuPost*.
3. Из списка **Тип ресурса** выберите тип резервируемого ресурса *RuPost*. Для выбора резервируемых компонентов почтовой системы нажмите [...] и определите тонкие настройки модуля *RuPost* (см. [Раздел 11.2](#)).
4. Нажмите [...] в поле **Ресурс** и выберите резервируемый компонент почтовой системы *RuPost*.
5. Из списка **Тип РК** выберите тип резервной копии. Возможные значения:
 - полная,
 - инкрементальная.
6. Из списка **Пул** выберите пул для сохранения резервной копии.
7. Из списка **Защитное преобразование** при необходимости выберите алгоритм защитного преобразования (возможные значения см. в [Алгоритмы защитного преобразования](#)).
8. В **Приоритет** укажите приоритет выполнения срочного резервного копирования. Возможные значения: от 100 до 1000. Чем выше значение, тем выше приоритет выполнения срочного резервного копирования.
9. В **Срок хранения** укажите срок хранения резервной копии, созданной в результате выполнения срочного резервного копирования.
10. В **Транспортировочный буфер** укажите размер транспортировочного буфера (в байтах, килобайтах, мегабайтах или гигабайтах). В транспортировочный буфер помещаются блоки резервной копии, после чего буфер передается на сервер. От размера буфера зависит количество передаваемых блоков за один раз: чем больше размер буфера, тем быстрее данные передаются на сервер, но при этом задействуется больший объем оперативной памяти. Размер транспортировочного буфера указывается в диапазоне от 50 Мб до 1 Гб (по умолчанию 100 Мб).
11. Нажмите **Общие настройки модуля** и определите настройки для многопоточного резервного копирования (см. [RBMManual:ROOT:page\\$clients-list.pdf](#)).
12. После указания необходимых параметров нажмите кнопку  **Применить** — будет создана задача на резервное копирование в разделе  **Очередь задач**.

Если во время выполнения задачи на резервное копирование какие-либо файлы ресурса изменили свой размер, то они будут включены в РК, а задача завершится со статусом *Done*.

Если во время выполнения задачи на резервное копирование какой-либо файл был удален, то при восстановлении будет восстановлена та часть файла, которая успела войти в РК.

10.2. Восстановление данных

Подробную информацию о выполнении восстановления резервных копий с помощью ПО Менеджер Администратора RuBackup (RBM) см. в [RuBackup Manager \(RBM\)](#).

10.2.1. Централизованное восстановление резервных копий с помощью ПО Менеджер Администратора RuBackup (RBM)

Для централизованного восстановления компонентов почтовой системы *RuPost* на клиенте резервного копирования в *RBM*:

1. Перейдите в раздел  **Репозиторий** и выделите требуемую резервную копию. Нажмите кнопку  **Восстановить**, или выберите **Восстановить** в контекстном меню резервной копии.

Откроется форма **Централизованное восстановление**.

2. Из списка **Восстановить на клиента** выберите узел клиента резервного копирования.
3. В **Каталог распаковки** нажмите [...] и укажите временный каталог для распаковки резервной копии.
4. Для выбора компонентов почтовой системы, которые нужно восстановить, в **Параметры восстановления для модуля** нажмите [...] и определите тонкие настройки модуля *RuPost* (см. [Раздел 11.3](#)).
5. Нажмите **Общие настройки модуля** и определите настройки для многопоточного восстановления резервной копии (см. [RBManual:ROOT:page\\$repository_recovery.pdf](#)).
6. Включите **Восстановить на целевом ресурсе** для восстановления РК на целевом ресурсе. Компоненты почтовой системы будут распакованы в холодное хранилище, минуя временный каталог. Служебная база данных будет распакована в **Каталог распаковки**, а затем развернута в почтовой системе *RuPost*.

Если флаг **Восстановить на целевом ресурсе** *выключен*, то резервные копии компонентов почтовой системы и служебной базы данных распаковываются во временный каталог, указанный в параметре **Каталог распаковки**, автоматическое развертывание не происходит.

7. Нажмите  **Применить** для запуска восстановления резервной копии.

Будет создана задача восстановления РК в разделе  **Очередь задач**.

10.2.2. Восстановление на целевом ресурсе

Восстановление почтовых ящиков пользователей

При восстановлении РК почтовые ящики пользователей распаковываются в папку `.Restore`, которая расположена в холодном хранилище для почтовых ящиков. После успешной распаковки папка `.Restore` переименовывается в `Restore`.



Если папка `Restore` уже существует, то в процессе развертывания почтовых ящиков из папки будет удалено все содержимое.

Дальнейшее развертывание в почтовую систему осуществляется силами администратора почтовой системы.

Восстановление архивов почтовых ящиков пользователей

При восстановлении РК архивы почтовых ящиков пользователей распаковываются в папку `.Restore`, которая расположена в холодном хранилище для архивов почтовых ящиков. После успешной распаковки папка `.Restore` переименовывается в `Restore`.



Если папка `Restore` уже существует, то в процессе развертывания архивов почтовых ящиков из папки будет удалено все содержимое.

Дальнейшее развертывание в почтовую систему осуществляется силами администратора почтовой системы.

Восстановление удаленных сообщений пользователей

При восстановлении РК удаленные сообщения пользователей распаковываются в папку `.Restore`, которая расположена в холодном хранилище для удаленных сообщений. После успешной распаковки папка `.Restore` переименовывается в `Restore`.



Если папка `Restore` уже существует, то в процессе развертывания удаленных сообщений из папки будет удалено все содержимое.

Дальнейшее развертывание в почтовую систему осуществляется силами администратора почтовой системы.

Восстановление базы данных

При восстановлении РК база данных распаковывается в выбранный пользователем временный каталог. После успешной распаковки происходит развёртывание

базы данных: существующая база данных заменяется на восстанавливаемую, СУБД перезапускается.



Развёртывание базы данных возможно только в конфигурации *standalone*. В кластере *Patroni* подмену базы и перезапуск кластера осуществляет администратор базы данных.

10.3. Просмотр очереди задач

Проверьте ход выполнения задач на клиенте резервного копирования одним из способов в разделе  **Очередь задач** через *RBM* (см. [RBMManual:ROOT:page\\$task_queue.pdf](#)). При успешном завершении соответствующая задача перейдёт в статус *Done* (*выполнено*).

Глава 11. Приложения

11.1. Файл настроек `rb_module_rupost.conf`

В [таблице](#) описаны параметры модуля *RuPost* для взаимодействия с корпоративной почтовой системой *RuPost*.

Таблица 2. Параметры файла настроек `/opt/rubackup/etc/rb_module_rupost.conf` модуля *RuPost*

Параметр	Описание
username	Имя пользователя в СУБД <i>PostgreSQL</i> , обладающего правами выполнять резервное копирование По умолчанию <code>rubackup_backuper</code>
password	Пароль для пользователя, указанного в параметре <code>username</code>
host	IP-адрес или доменное имя локального хоста, на котором СУБД принимает подключения. Используется для взаимодействия с СУБД, резервное копирование которой выполняется. Параметр необязательный, т.е. его можно не указывать в конфигурационном файле По умолчанию <code>localhost</code>
port	Порт для соединения с СУБД. Параметр необязательный, т.е. его можно не указывать в конфигурационном файле По умолчанию <code>5432</code>
use_secret_storage	Использование хранилища секретов <i>HashiCorp vault v1.16.3</i> Возможные значения <code>yes</code> , <code>no</code> По умолчанию <code>no</code>
archive_catalog	Каталог для хранения архивных WAL По умолчанию <code>/opt/rubackup/mnt/postgresql_archives</code>
pg_ctl	Используется для запуска и остановки СУБД <i>PostgreSQL</i> во время восстановления с разворачиванием. Местонахождение <code>pg_ctl</code> зависит от используемой версии По умолчанию <code>/usr/lib/postgresql/12/bin/pg_ctl</code>
pg_binary	Используется при вызове утилиты <code>pg_ctl</code> для запуска <i>PostgreSQL</i> во время восстановления с разворачиванием. Указывает путь к исполняемому файлу <i>postgres</i> . Параметр <code>pg_binary</code> не является обязательным. По умолчанию исполняемый файл <i>postgres</i> берется из того же каталога, из которого запускался <code>pg_ctl</code> , а если найти файл невозможно, то из жёстко заданного каталога инсталляции
pg_log	Используется при вызове утилиты <code>pg_ctl</code> для запуска <i>PostgreSQL</i> во время восстановления с разворачиванием. В указанный файл будет направляться вывод сообщений сервера. Файл создаётся, если он ещё не существует. Параметр не является обязательным

Параметр	Описание
<code>postgresql_service_name</code>	Служебное имя базы данных По умолчанию <code>postgresql</code>
<code>pg_waldump</code>	Путь до утилиты <code>pg_waldump</code>. Параметр необходимо задать для работы под-типа инкрементального резервного копирования <code>page</code> (при использовании подмодуля <code>postgresql</code>). Местонахождение <code>pg_waldump</code> зависит от используемой версии <i>PostgreSQL</i> По умолчанию <code>/usr/lib/postgresql/12/bin/pg_waldump</code>
<code>num_threads_for_wal_processing</code>	Количество процессов, выделенных для обработки архивных WAL файлов По умолчанию <code>8</code>
<code>auto_remove_wal</code>	В случае значения yes архивные WAL будут удалены из каталога <code>archive_catalog</code> после выполнения резервного копирования (если они включены в резервную копию) Возможные значения <code>yes, no</code> По умолчанию <code>yes</code>
<code>postgresql_admin</code>	Login администратора <i>PostgreSQL</i> в операционной системе По умолчанию <code>postgres</code>
<code>execute_only_on_leader</code>	В случае значения <code>yes</code> резервное копирование выполняется только на лидере кластера <i>Patroni</i>. В случае активации параметра модуль возвращает отрицательный ответ серверу на запрос о наличии ресурса, если хост, на котором производится проверка, не является лидером кластера <i>Patroni</i>. Параметр применяется только при работе в кластере <i>Patroni</i> и используется только в версии модуля 2.0 и ниже. В конфигурационном файле модуля версии 2.1 параметр заменён на <code>patroni_node_type_for_backup</code> Возможные значения <code>yes, no</code> По умолчанию <code>no</code>
<code>patroni_node_type_for_backup</code>	В случае указания значения leader, ресурс будет доступен только при условии, что узел на котором установлен модуль с таким значением имеет роль <i>leader</i> в кластере <i>Patroni</i>. В случае указания значения sync, ресурс будет доступен только при условии, что узел, на котором установлен модуль с таким значением, имеет роль <i>sync standby</i> в кластере <i>Patroni</i>. В случае указания значения async, ресурс будет доступен только при условии, что узел, на котором установлен модуль с таким значением, имеет роль <i>replica</i> в кластере <i>Patroni</i>. Параметр <code>patroni_node_type_for_backup</code> заменяет в версии модуля 2.1 параметр <code>execute_only_on_leader</code>.
<code>wal_wait_timeout</code>	Период ожидания окончания архивации последнего WAL-файла, сгенерированного во время создания резервной копии По умолчанию <code>10</code>

Параметр	Описание
wal_check_period	Период проверки окончания архивации последнего WAL-файла, сгенерированного во время создания резервной копии
	По умолчанию 1
patroni_host	IP-адрес, на котором <i>Patroni</i> принимает входящие запросы <i>Rest API</i> . Параметр необязательный (т.е. его можно не указывать в конфигурационном файле) и необходим только для взаимодействия модуля с <i>Rest API</i> локального процесса <i>Patroni</i> . Если значение параметра не указано, будет предпринята попытка автоматически определить значение для этого параметра через утилиту <code>lsf</code>
	По умолчанию localhost
patroni_port	Порт, на котором локальный процесс <i>Patroni</i> слушает запросы <i>Rest API</i> . Параметр необязательный (т.е. его можно не указывать в конфигурационном файле) и необходим только для взаимодействия модуля с <i>Rest API</i> локального процесса <i>Patroni</i> . Если значение параметра не указано, будет предпринята попытка автоматически определить значение для этого параметра через утилиту <code>lsf</code>
	По умолчанию 8008
cold_storage_path	Директория холодного хранилища почтовых ящиков
	По умолчанию /mnt/cold_storage/mail
cold_storage_archive_path	Директория холодного хранилища архивов почтовых ящиков
	По умолчанию /mnt/cold_storage/archive
cold_storage_record_path	Директория холодного хранилища удаленных писем
	По умолчанию /mnt/cold_storage/record
allow_work_with_incompatible_versions	Позволяет использовать модуль, если версия платформы несовместима с CPK <i>RuBackup</i>
	Возможные значения yes, no
	По умолчанию no

11.2. Тонкие настройки модуля RuPost для резервного копирования

В [таблице](#) описаны тонкие настройки модуля *RuPost* для резервного копирования (см. [Раздел 10.1](#)).

Таблица 3. Тонкие настройки модуля *RuPost* для резервного копирования

Параметр	Описание
backup_maildir	Выполнение резервного копирования почтовых ящиков пользователей из холодного хранилища почтовой системы При значении: • <code>true</code> резервное копирование выполняется; • <code>false</code> резервное копирование не выполняется. По умолчанию <code>true</code>
backup_archive	Выполнение резервного копирования архива почтовых ящиков пользователей из холодного хранилища почтовой системы При значении: • <code>true</code> резервное копирование выполняется; • <code>false</code> резервное копирование не выполняется. По умолчанию <code>false</code>
backup_record	Выполнение резервного копирования удаленных сообщений пользователей из холодного хранилища почтовой системы При значении: • <code>true</code> резервное копирование выполняется; • <code>false</code> резервное копирование не выполняется. По умолчанию <code>false</code>
backup_database	Выполнение резервного копирования служебной базы данных почтовой системы При значении: • <code>true</code> резервное копирование выполняется; • <code>false</code> резервное копирование не выполняется. По умолчанию <code>true</code>
file_list	Создание списка файлов резервных копий для возможности их отдельного восстановления При значении: • <code>true</code> списки файлов РК создаются; • <code>false</code> списки файлов РК не создаются. По умолчанию <code>true</code>

Параметр	Описание
connection_monitoring	Отслеживание связи с базой данных во время выполнения резервного копирования этой базы данных При значении: <code>true</code> связь с БД отслеживается; <code>false</code> связь с БД не отслеживается. По умолчанию <code>true</code>

Кнопка **Значения по умолчанию** очищает параметры и устанавливает для них значения по умолчанию.

Кнопка **ОК** сохраняет настройки.

11.3. Тонкие настройки модуля RuPost для восстановления резервной копии

В [таблице](#) описаны тонкие настройки модуля *RuPost* для восстановления резервной копии (см. [Раздел 10.2](#)).

Таблица 4. Тонкие настройки модуля *RuPost* для восстановления резервной копии

Параметр	Описание
Использовать настройки по умолчанию	Использование значений по умолчанию При значении: <code>true</code> для параметров используются значения по умолчанию. Восстановление происходит для почтовых ящиков, архива почтовых ящиков, удаленных сообщений и служебной базы данных; <code>false</code> значения параметров можно изменить. По умолчанию <code>true</code>
restore_maildir	Выполнение восстановления почтовых ящиков пользователей почтовой системы из РК При значении: <code>true</code> восстановление выполняется; <code>false</code> восстановление не выполняется. По умолчанию <code>true</code>

Параметр	Описание
restore_archive	Выполнение восстановления архива почтовых ящиков пользователей почтовой системы из РК При значении: • <code>true</code> восстановление выполняется; • <code>false</code> восстановление не выполняется. По умолчанию <code>true</code>
restore_record	Выполнение восстановления удаленных сообщений пользователей почтовой системы из РК При значении: • <code>true</code> восстановление выполняется; • <code>false</code> восстановление не выполняется. По умолчанию <code>true</code>
restore_database	Выполнение восстановления служебной базы данных из РК При значении: • <code>true</code> восстановление выполняется; • <code>false</code> восстановление не выполняется. По умолчанию <code>true</code>

Кнопка **OK** сохраняет настройки.