



RuBackup

Система резервного копирования
и восстановления данных

RUPOST

ВЕРСИЯ 2.9.0.0.0

Содержание

1. Назначение	2
2. Требования к программным средствам	3
3. Резервируемые данные	4
4. Типы резервного копирования	5
5. Типы восстановления данных	6
6. Способы восстановления данных	7
7. Комплект поставки	8
8. Ограничения	9
9. Установка модуля	10
9.1. Подготовка к установке модуля	10
9.2. Установка	10
9.3. После установки	11
9.4. Критерий успешности установки	11
10. Подготовка к работе	13
10.1. Подготовка холодного хранилища	13
10.2. Подготовка СУБД PostgreSQL	14
10.2.1. Создание и настройка прав пользователя СУБД	14
10.2.2. Доступ пользователя к СУБД	15
10.2.3. Режим архивирования журналов транзакций	16
11. Работа с данными	18
11.1. Резервное копирование в RBM	19
11.2. Восстановление резервных копий в RBM	20
11.2.1. Полное восстановление	20
11.2.2. Гранулярное восстановление	20
11.2.3. Восстановление на целевом ресурсе	21
Восстановление почтовых ящиков пользователей	21
Восстановление архивов почтовых ящиков пользователей	22
Восстановление удаленных писем пользователей	22
Восстановление базы данных	22
12. Файл настроек модуля	23
13. Файл настроек хранилищ	27
14. Тонкие настройки модуля RuPost для резервного копирования	29
15. Тонкие настройки модуля RuPost для восстановления резервной копии	32

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 версий 4.0, 4.1 и 4.1.1 с помощью модуля RuPost на узле необходимо предустановленное программное обеспечение:

- 64-битная операционная система Astra Linux SE 1.7 или Astra Linux SE 1.8;
- клиент резервного копирования RuBackup;
- модуль RuPost;
- служебная база данных почтовой системы RuPost (опционально, если планируется делать резервное копирование БД);



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

- NFS-клиент для монтирования холодного^[1] хранилища почтовой системы RuPost.

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

[1] Резервное хранилище почтовой системы RuPost, которое используется как источник данных для резервного копирования компонентов почтовой системы.

Глава 3. Резервируемые данные

Резервное копирование выполняется для:

- компонентов холодного^[1 - Глава 2] хранилища почтовой системы:
 - почтовые ящики;
 - архивы почтовых ящиков;
 - удаленные письма;
 - календари и расписания.
- служебной базы данных почтовой системы.

Глава 4. Типы резервного копирования

Модуль RuPost поддерживает полное, инкрементальное и дифференциальное резервное копирование почтовой системы RuPost.

При инкрементальном резервном копировании создаются:

- инкрементальная резервная копия компонентов из холодного^[1 - Глава 2] хранилища;
- полная резервная копия служебной базы данных почтовой системы RuPost.

При дифференциальном резервном копировании создаются:

- дифференциальная резервная копия компонентов из холодного^[1 - Глава 2] хранилища;
- полная резервная копия служебной базы данных почтовой системы RuPost.

Глава 5. Типы восстановления данных

СРК RuBackup поддерживает полное и гранулярное восстановление почтовой системы RuPost.

Полное восстановление доступно для компонентов холодного^[1 - Глава 2] хранилища почтовой системы RuPost и для служебной БД почтовой системы RuPost.

Гранулярное восстановление доступно только для компонентов холодного^[1 - Глава 2] хранилища почтовой системы RuPost.


Для каждого отдельного пользователя почтовой системы RuPost можно восстановить:

- почтовый ящик;
- отдельные папки с письмами из почтового ящика;
- отдельные письма из папок.


Глава 6. Способы восстановления данных

Модуль RuPost поддерживает следующие способы восстановления резервных копий:

- централизованное восстановление данных без развертывания. Резервная копия будет распакована в каталог распаковки;
- централизованное восстановление данных с развертыванием. Компоненты почтовой системы будут распакованы в холодное^[1 - Глава 2] хранилище, минуя временный каталог.

 СРК не восстанавливает компоненты почтовой системы RuPost напрямую в почтовую систему. Данные восстанавливаются только в папки `Restore` холодного^[1 - Глава 2] 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.

Глава 7. 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 8. Ограничения

- СРК не обеспечивает синхронность данных при резервном копировании данных из холодного ^[1 - Глава 2] 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 ^[1 - Глава 2] хранилище.
- Модуль RuPost не поддерживает взаимодействие с СРК при помощи веб-приложения Tusana.
- 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.
- Для хранения резервных копий допускаются только файловые хранилища и блочные устройства.
- 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 9. Установка модуля

9.1. 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
```

9.2. 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 RuPost СРК RuBackup
<code>/opt/rubackup/modules/rb_module_rupost</code>	Исполняемый файл модуля RuPost СРК RuBackup

9.3. После установки

После установки модуля:

1. Настройте параметры в [Файле настроек](#) `rb_module_rupost.conf`.
2. [Подготовьте и настройте](#) компоненты холодного ^[1 - Глава 2] хранилища почтовой системы RuPost.
3. Перезапустите сервис клиента резервного копирования RuBackup:

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

9.4. Критерий успешности установки

Об успешной установке и настройке модуля RuPost свидетельствует запись о его успешной проверке клиентом резервного копирования в журнале событий `/opt/rubackup/log/RuBackup.log`:

Пример записи об успешной установке и настройке модуля (если служебная БД доступна)

```
...
Module version: 2.9.0
RuPost version: "4.0.0"
Database availability: "Available"
```



Даже если служебная база данных почтовой системы RuPost недоступна (проблемы с сетью или указаны некорректные данные для подключения к служебной БД в конфигурационном файле `/opt/rubackup/etc/rb_module_rupost.conf`), можно сделать резервную копию компонентов холодного ^[1 - Глава 2] хранилища почтовой системы RuPost.

Пример записи об успешной установке и настройке модуля (если служебная БД недоступна)

```
...
Module version: 2.9.0
RuPost version: "4.0.0"
Database availability: "Unavailable"
```



Если один из типов холодных^[1 - Глава 2] хранилищ почтовой системы RuPost не будет указан в `rupost_cold_storages.json`, модуль запустится, но будет выведено предупреждение о недоступности конкретно этого типа хранилища.

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/>.

Глава 10. Подготовка к работе

С точки зрения резервного копирования и восстановления почтовая система RuPost состоит из СУБД PostgreSQL и холодного^[1 - Глава 2] хранилища почтовой системы.

Модуль должен иметь доступ к холодному^[1 - Глава 2] хранилищу.

Модуль должен располагаться на узле с СУБД PostgreSQL только в том случае, если необходимо осуществлять резервное копирование служебной базы данных почтовой системы RuPost.

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

- При создании [Срочной резервной копии](#) в качестве клиента необходимо выбрать ноду кластера Patroni, которая является лидером в текущий момент.
- Если для работы с кластером Patroni используется группа клиентов с атрибутами **Разделяемая группа** и **Кластерная группа** (см. [Группы клиентов](#)), то при создании [Правил глобального расписания](#) или [Стратегий](#) можно выбирать любую ноду кластера Patroni. Группа клиентов автоматически перенаправит задачу на ноду кластера, которая является лидером в текущий момент.

10.1. Подготовка холодного хранилища

Холодное^[1 - Глава 2] хранилище писем должно быть примонтировано к узлу с модулем с помощью NFS, для этого:

1. На узле с модулем установите клиент NFS:

```
sudo apt update

sudo apt install nfs-common
```

2. Примонтируйте холодное^[1 - Глава 2] хранилище:

```
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.

3. Чтобы после перезагрузки узла с модулем холодное хранилище оставалось примонтированным, добавьте в файл `/etc/fstab` строку:

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

4. Создайте [Глава 13](#) и настройте параметры для компонентов холодного хранилища почтовой системы RuPost.^[1 - Глава 2]

10.2. Подготовка СУБД PostgreSQL

Подготовка СУБД PostgreSQL к выполнению резервного копирования с помощью модуля RuPost включает в себя:

1. Создание и настройку пользователя СУБД для безопасного выполнения резервной копии PostgreSQL.
2. Доступ пользователя к СУБД PostgreSQL.
3. Настройку режима архивирования журналов транзакций.

10.2.1. Создание и настройка прав пользователя СУБД

Если для доступа к СУБД используется учетная запись не администратора СУБД, то для создания базовой резервной копии пользователь должен обладать правами на вызов функций начала и окончания резервного копирования экземпляра СУБД.

1. Вызовите `psql` от имени администратора СУБД:

```
sudo -u postgres psql
```

2. Создайте в СУБД пользователя `rubackup_backuper`.

```
create user rubackup_backuper password '12345'; ❶  
alter role rubackup_backuper with login;
```

- ❶ Вместо `12345` укажите желаемый пароль.

3. Настройте права пользователя в СУБД для доступа.

Пример 1. Предоставление прав на выполнение функций в PostgreSQL версии 15 и выше

```
grant execute on function pg_backup_start(text, bool) to  
rubackup_backuper; ❶
```

```
grant execute on function pg_backup_stop(bool) to rubakup_backuper; ①
grant execute on function pg_switch_wal() to rubakup_backuper;
grant pg_read_all_settings to rubakup_backuper;
```

- ① 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`.

В файле [настроек модуля](#) необходимо указать имя созданного пользователя (`username`) и его пароль (`password`).

10.2.2. Доступ пользователя к СУБД

Правила доступа к СУБД задаются в файле `pg_hba.conf`.

Файл `pg_hba.conf`:

- устанавливает метод подключения (`local` — локально, `host` — по TCP/IP);
- ограничивает доступ базой данных или задачей (`replication`);
- предоставляет доступ пользователю с заданных адресов указанным методом аутентификации.



В зависимости от версии СУБД PostgreSQL месторасположение файла может отличаться.

Добавьте в файл `pg_hba.conf` строки, указывающие на право пользователя `rubakup_backuper` на локальный (`localhost`, `127.0.0.1`) доступ по TCP/IP к СУБД для резервного копирования и репликации. Закомментируйте строки, предоставляющие доступ неограниченному кругу лиц (`host all all`).

```
host backupdb rubakup_backuper 127.0.0.1 md5
host replication rubakup_backuper 127.0.0.1 md5
```

Пример 2. Пример файла `pg_hba.conf`

```
# TYPE DATABASE USER ADDRESS METHOD

# "local" is for Unix domain socket connections only
local all all md5

# IPv4 local connections:
# host all all 127.0.0.1/32 md5
```

```
# IPv6 local connections
# host    all             all                ::1/128           md5

# Allow replication connections from localhost, by a user with the
# replication privilege
local    replication all                md5
host     replication all                127.0.0.1/32     md5
host     replication all                ::1/128          md5
host     backupdb      rubackup_backuper 127.0.0.1/32     md5
host     replication rubackup_backuper 127.0.0.1/32     md5
```

Если вы хотите подключаться из локальной сети (192.168.*.*) непосредственно к СУБД, добавьте в `pg_hba.conf` строку:

```
host    any    any    192.168.1.1/32 md5
```



Для доступа к СУБД необходимо открыть порт 5432 (или иной заданный в `postgresql.conf`) в фаерволе узла клиента. Пользователь, для которого требуется доступ к СУБД, должен существовать и иметь необходимые полномочия.



В файле настроек модуля необходимо указать абсолютный путь для используемой версии PostgreSQL (`pg_ctl`).

10.2.3. Режим архивирования журналов транзакций

СУБД на основе PostgreSQL хранят журналы транзакций. Архивирование и резервное копирование этих журналов позволяет восстановить БД из любого неконсистентного состояния путем последовательного применения сохраненных транзакций.

Режим архивирования журналов транзакций настраивается в файле `postgresql.conf`.



В зависимости от версии СУБД PostgreSQL месторасположение файла может отличаться.

Чтобы включить режим архивирования журналов транзакций в `postgresql.conf`:

1. установите параметру `wal_level` значение `replica`;
2. установите параметру `archive_mode` значение `yes` или `always`;
3. задайте команды сохранения (`archive_command`) и восстановления

(`restore_command`) журналов транзакций.

Пример 3. Настройки архивирования журналов транзакций в `postgresql.conf`

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

Допускается сжатие архивных файлов транзакций во время архивирования или восстановления из архива.

Пример 4. Команды сжатия и распаковки архивных журналов транзакций

```
archive_command = 'gzip < %p >
/opt/rubackup/mnt/postgresql_archives/%f.gz'
restore_command = 'gunzip < /opt/rubackup/mnt/postgresql_archives/%f.gz >
%p'
```

Путь к архиву в файле настроек модуля должен совпадать с путём, указанным в конфигурационном файле СУБД PostgreSQL (параметр `archive_command`). В [файле настроек](#) модуля (параметр `archive_catalog`) укажите на папку, хранящую архив журналов транзакций,

Глава 11. Работа с данными

Доступно полное, инкрементальное и дифференциальное резервное копирование. При выборе инкрементального или дифференциального 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 или дифференциальной резервной копии.

При восстановлении инкрементальной резервной копии происходит распаковка архивов в несколько итераций, начиная с полной резервной копии. Поэтому при восстановлении компонентов холодного ^[1 - Глава 2] 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](#).

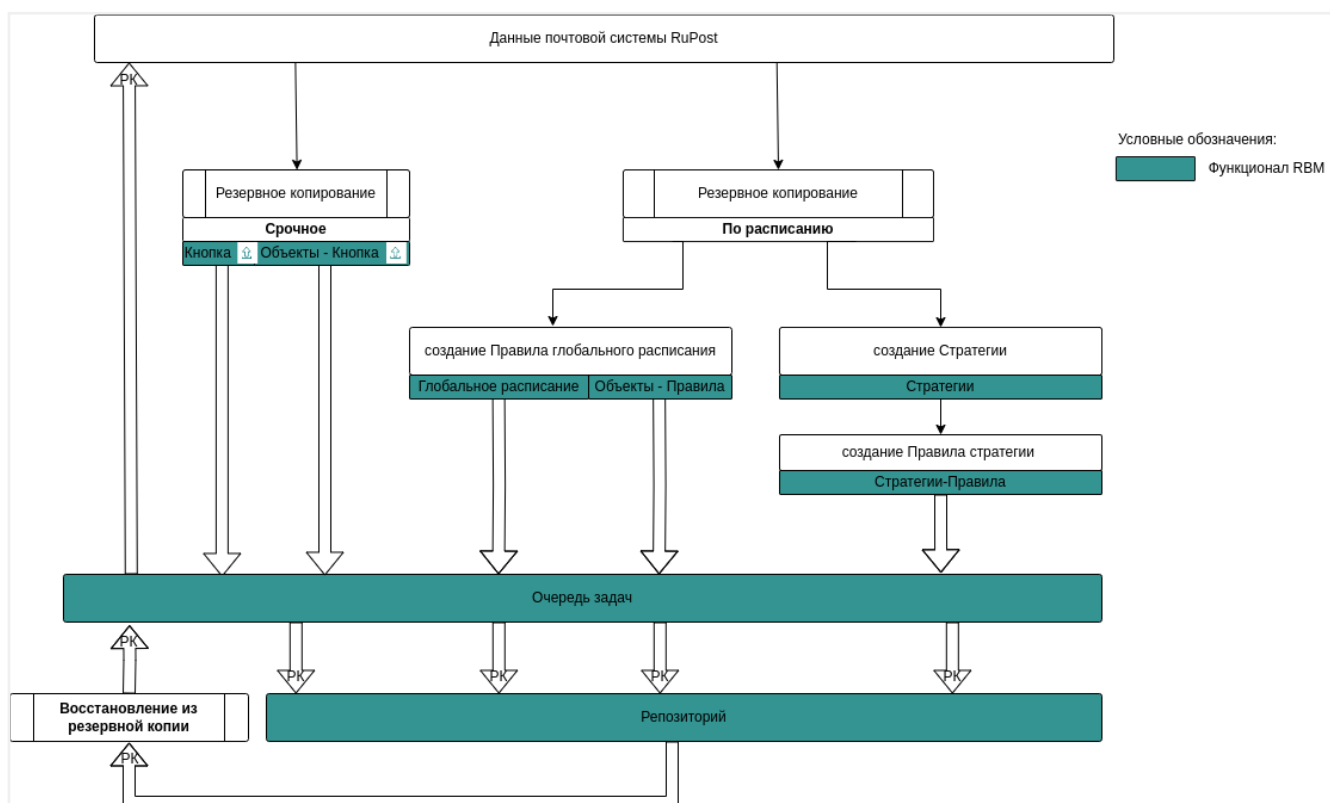


Рисунок 1. Структурная схема резервного копирования и восстановления данных из резервной копии

11.1. Резервное копирование в RBM

Выберите способ выполнения резервного копирования в приложении [Менеджер администратора RuBackup \(RBM\)](#) и произведите настройку, следуя указаниям из соответствующего документа:

- [Срочное резервное копирование](#)
- [Глобальное расписание](#)
- [Стратегии](#)



Для резервного копирования служебной базы данных и компонентов холодного^[1 - Глава 2] хранилища почтовой системы RuPost необходимо создавать отдельные правила глобального расписания.

Для срочного резервного копирования:

1. Из списка **Клиент** выберите клиента, на котором установлена почтовая система RuPost.
2. Из списка **Тип ресурса** выберите тип резервируемого ресурса RuPost. Для выбора резервируемого хранилища почтовой системы нажмите [...] и определите [тонкие настройки модуля](#).
3. Нажмите [...] в поле **Ресурс** и выберите резервируемый компонент почтовой

системы RuPost.

4. Из списка **Тип РК** выберите тип резервной копии.
5. Нажмите **✓ Применить** для сохранения изменений.

Для возврата к значениям по умолчанию нажмите **Значения по умолчанию**.

11.2. Восстановление резервных копий в RBM

В приложении **Менеджер администратора RuBackup (RBM)** произведите настройку, следуя указаниям из документа **Раздел «Репозиторий»**.

11.2.1. Полное восстановление

Для полного восстановления компонентов почтовой системы RuPost на клиенте резервного копирования:

1. Из списка **Восстановить на клиенте** выберите узел клиента резервного копирования.
2. В **Каталог распаковки** нажмите **[...]** и укажите временный каталог для распаковки резервной копии.
3. В **Параметры восстановления для модуля** нажмите **[...]** и в **тонких настройках модуля для восстановления резервной копии** определите компоненты почтовой системы, для которых нужно выполнить восстановление.
4. Установите флаг **☑ Восстановить на целевом ресурсе** для восстановления РК на целевом ресурсе. Компоненты почтовой системы будут распакованы в холодное хранилище^[1 - Глава 2], минуя временный каталог. Службная база данных будет распакована в **Каталог распаковки**, а затем развернута в почтовой системе RuPost.

Если флаг **Восстановить на целевом ресурсе** **☐**, то развертывание не выполняется (резервные копии почтовых ящиков, архивов почтовых ящиков, удаленных писем, календарей и расписаний из холодного^[1 - Глава 2] хранилища почтовой системы RuPost распаковываются во временный каталог, указанный в параметре **Каталог распаковки**).

11.2.2. Гранулярное восстановление

Для гранулярного восстановления компонентов холодного^[1 - Глава 2] хранилища почтовой системы RuPost на клиенте резервного копирования:

1. Из списка **Восстановить на клиенте** выберите узел клиента резервного копирования.
2. В **Каталог распаковки** нажмите **[...]** и укажите временный каталог для распаковки резервной копии.

3. В **Параметры восстановления для модуля** нажмите [...] и в **тонких настройках модуля для восстановления резервной копии** определите компоненты почтовой системы, для которых нужно выполнить восстановление.
4. Установите флаг **Восстановить на целевом ресурсе** для восстановления РК на целевом ресурсе. Компоненты почтовой системы будут распакованы в холодное хранилище^[1 - Глава 2], минуя временный каталог. Служебная база данных будет распакована в **Каталог распаковки**, а затем развернута в почтовой системе RuPost.

Если флаг **Восстановить на целевом ресурсе** **О**, то развертывание не выполняется (резервные копии почтовых ящиков, архивов почтовых ящиков, удаленных писем, календарей и расписаний из холодного^[1 - Глава 2] хранилища почтовой системы RuPost распаковываются во временный каталог, указанный в параметре **Каталог распаковки**).

5. В блоке **Гранулярное восстановление** нажмите **+** (**Добавить объекты**). Откроется форма со списком объектов гранулярного восстановления.
6. Выберите необходимого пользователя или конкретные папки пользователя из списка^[1].
 - Для восстановления календарей или расписаний пользователя, выберите папку с названием **Sogo**.
 - Для восстановления отдельных писем выберите необходимого пользователя и папку, нажмите **Выбрать письма**.



При выполнении гранулярного восстановления значения параметров **restore_maildir**, **restore_archive**, **restore_record**, **restore_database** в **тонких настройках модуля для восстановления резервной копии**, если они заданы, не учитываются.

7. Нажмите **✓ Применить**. В блоке **Гранулярное восстановление** отобразятся выбранные для восстановления компоненты из холодного^[1 - Глава 2] хранилища. Список можно отредактировать или полностью очистить.

11.2.3. Восстановление на целевом ресурсе

Восстановление почтовых ящиков пользователей

При восстановлении РК почтовые ящики пользователей распаковываются в папку **.Restore**, которая расположена в холодном^[1 - Глава 2] 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^[1 - Глава 2] 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 писем пользователей

При восстановлении РК удаленные письма пользователей распаковываются в папку `.Restore`, которая расположена в холодном^[1 - Глава 2] хранилище для удаленных писем. После успешной распаковки папка `.Restore` переименовывается в `Restore`.



Если папка `Restore` уже существует, то в процессе развертывания удаленных писем 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.

[1] Доступна пагинация по пользователям и папкам.

Глава 12. Файл настроек модуля






Начиная с версии 2.8 при обновлении модуля RuPost параметры в [файле настроек модуля](#) и [файле настроек хранилищ](#) сохраняют ранее установленные значения.

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
<code>cold_storages_json_path</code>	<p>Путь к JSON-файлу с конфигурацией холодных^[1 - Глава 2] хранилищ (почтовых ящиков, архивов почтовых ящиков, удаленных писем) каждого пространства хранения</p> <p>По умолчанию JSON-файл отсутствует в репозитории, создайте его вручную (см. Глава 13)</p> <p> После перехода на JSON-конфигурацию необходимо заново настроить правила и стратегии резервного копирования с привязкой к ID пространства хранения.</p> <p>Если РК была создана в версии RuBackup до 2.8, то при восстановлении с развёртыванием данные будут восстановлены в первое пространство хранения из Глава 13.</p>
<code>username</code>	<p>Имя пользователя в СУБД <i>PostgreSQL</i>, обладающего правами выполнять резервное копирование</p> <p>По умолчанию</p> <p><code>rubackup_backuper</code></p>
<code>password</code>	<p>Пароль для пользователя, указанного в параметре <code>username</code></p>
<code>host</code>	<p>IP-адрес или доменное имя хоста, на котором СУБД принимает подключения. Используется для взаимодействия с СУБД, резервное копирование которой выполняется.</p> <p>По умолчанию</p> <p><code>localhost</code></p>
<code>port</code>	<p>Порт для соединения с СУБД</p> <p>По умолчанию</p> <p><code>5432</code></p>
<code>postgresql_admin</code>	<p>Имя пользователя — администратора PostgreSQL</p> <p>По умолчанию</p> <p><code>postgres</code></p>

Параметр	Описание
archive_catalog	<p>Каталог для хранения архивных WAL</p> <p>По умолчанию</p> <p><code>/opt/rubackup/mnt/postgresql_archives</code></p>
pg_ctl	<p>Путь до утилиты <code>pg_ctl</code></p> <p>По умолчанию</p> <p><code>/usr/lib/postgresql/12/bin/pg_ctl</code></p> <p>Параметр используется при восстановлении с развертыванием для остановки и запуска сервиса СУБД</p> <p> Для версии ОС Astra Linux SE 1.8 необходимо закомментировать параметр <code>pg_ctl</code> и использовать параметр <code>postgresql_service_name</code>. По умолчанию параметр <code>postgresql_service_name</code> отсутствует в конфигурационном файле, его необходимо добавить вручную и указать в значении имя сервиса СУБД PostgreSQL.</p> <p> Используется, если не задано значение параметра <code>postgresql_service_name</code></p>
postgresql_service_name	<p>Имя <code>systemd</code> службы СУБД</p> <p>По умолчанию</p> <p><code>postgresql</code></p> <p>Параметр используется при восстановлении с развертыванием для остановки и запуска сервиса СУБД через <code>systemctl</code></p> <p> Используется, если не задано значение параметра <code>pg_ctl</code></p>
num_threads_for_wal_processing	<p>Количество процессов, выделенных для обработки архивных WAL файлов</p> <p>По умолчанию</p> <p>8</p>
pg_binary	<p>Используется при вызове утилиты <code>pg_ctl</code> для запуска PostgreSQL во время восстановления с развертыванием. Указывает путь к исполняемому файлу <code>postgres</code>.</p> <p>По умолчанию исполняемый файл <code>postgres</code> берется из того же каталога, из которого запускался <code>pg_ctl</code>, а если найти файл невозможно, то из жёстко заданного каталога инсталляции</p>
pg_log	<p>Используется при вызове утилиты <code>pg_ctl</code> для запуска PostgreSQL во время восстановления с развертыванием. В указанный файл будет направляться вывод сообщений сервера. Файл создаётся, если он ещё не существует.</p>

Параметр	Описание
auto_remove_wal	<p>В случае значения <code>yes</code> архивные WAL будут удалены из каталога <code>archive_catalog</code> после выполнения резервного копирования (если они включены в резервную копию)</p> <p>Возможные значения</p> <p><code>yes</code>, <code>no</code></p> <p>По умолчанию</p> <p><code>yes</code></p>
wal_wait_timeout	<p>Период ожидания окончания архивации последнего WAL-файла, сгенерированного во время создания резервной копии</p> <p>По умолчанию</p> <p><code>10</code></p>
wal_check_period	<p>Период проверки окончания архивации последнего WAL-файла, сгенерированного во время создания резервной копии</p> <p>По умолчанию</p> <p><code>1</code></p>
pid_wait_timeout	<p>Период проверки выключения сервиса PostgreSQL</p> <p>По умолчанию</p> <p><code>10</code></p>
patroni_host	<p>IP-адрес, на котором Patroni принимает входящие запросы REST API. Параметр необходим для взаимодействия модуля с REST API локального процесса Patroni.</p> <p>Если значение параметра не указано, будет предпринята попытка автоматически определить значение для этого параметра через утилиту <code>lsnf</code></p> <p>По умолчанию</p> <p><code>localhost</code></p>
patroni_port	<p>Порт, на котором локальный процесс Patroni слушает запросы REST API. Параметр необходим для взаимодействия модуля с REST API локального процесса Patroni.</p> <p>Если значение параметра не указано, будет предпринята попытка автоматически определить значение для этого параметра через утилиту <code>lsnf</code></p> <p>По умолчанию</p> <p><code>8008</code></p>





Параметр	Описание
patroni_node_type_for_backup	<p>leader</p> <p>Ресурс будет доступен только при условии, что узел на котором установлен модуль с таким значением имеет роль <i>leader</i> в кластере Patroni.</p> <p>sync</p> <p>Ресурс будет доступен только при условии, что узел, на котором установлен модуль с таким значением, имеет роль <i>sync standby</i> в кластере Patroni.</p> <p>async</p> <p>Ресурс будет доступен только при условии, что узел, на котором установлен модуль с таким значением, имеет роль <i>replica</i> в кластере Patroni.</p>
allow_work_with_incompatible_versions	<p>Позволяет использовать модуль, если версия платформы несовместима с CPK RuBackup</p> <p>Возможные значения</p> <p>yes, no</p> <p>По умолчанию</p> <p>no</p>

Глава 13. Файл настроек хранилищ

 Начиная с версии 2.8 при обновлении модуля RuPost параметры в [файле настроек модуля](#) и [файле настроек хранилищ](#) сохраняют ранее установленные значения.

В [таблице](#) описаны параметры модуля RuPost для взаимодействия с несколькими пространствами хранения корпоративной почтовой системы RuPost.

Таблица 3. Параметры файла настроек `rupost_cold_storages.json` модуля RuPost

Параметр	Описание
id	Уникальный ID каждого пространства хранения  Значение не может быть пустым
name	Имя пространства хранения
path	Путь к директории холодного ^[1 - Глава 2] хранилища почтовых ящиков  Значение параметра должно быть уникальным для каждого пространства хранения
archive_path	Путь к директории холодного ^[1 - Глава 2] хранилища архивов почтовых ящиков  Значение параметра должно быть уникальным для каждого пространства хранения
deleted_records_path	Путь к директории холодного ^[1 - Глава 2] хранилища удаленных писем  Значение параметра должно быть уникальным для каждого пространства хранения

Пример 5. Структура `rupost_cold_storages.json`

```
{
  "storages": [
    {
      "id": "storage_id_1",
      "name": "storage_name_1",
      "path": "/cold_storage_1",
      "archive_path": "/archive_path_1",
      "deleted_records_path": "/deleted_record_path_1"
    },
    {
      "id": "storage_id_2",
      "name": "storage_name_2",
      "path": "/cold_storage_2",
```


```
"archive_path": "/archive_path_2",  
"deleted_records_path": "/deleted_record_path_2"  
}  
]  
}
```


Глава 14. Тонкие настройки модуля RuPost для резервного копирования

В [таблице](#) описаны тонкие настройки модуля RuPost для резервного копирования (см. [Раздел 11.1](#)).

Таблица 4. Тонкие настройки модуля RuPost для резервного копирования

Параметр	Описание
backup_maldir	Выполнение резервного копирования почтовых ящиков пользователей из холодного ^[1 - Глава 2] хранилища почтовой системы true Резервное копирование выполняется. false Резервное копирование не выполняется. По умолчанию true
backup_archive	Выполнение резервного копирования архива почтовых ящиков пользователей из холодного ^[1 - Глава 2] хранилища почтовой системы true Резервное копирование выполняется. false Резервное копирование не выполняется. По умолчанию false
backup_record	Выполнение резервного копирования удаленных писем пользователей из холодного ^[1 - Глава 2] хранилища почтовой системы true Резервное копирование выполняется. false Резервное копирование не выполняется. По умолчанию false

Параметр	Описание
backup_database	<p>Выполнение резервного копирования служебной базы данных почтовой системы</p> <p>true Резервное копирование выполняется.</p> <p>false Резервное копирование не выполняется.</p> <p>По умолчанию false</p>
connection_monitoring	<p>Отслеживание связи с базой данных во время выполнения резервного копирования этой базы данных</p> <p>true Связь с БД отслеживается.</p> <p>false Связь с БД не отслеживается.</p> <p>По умолчанию true</p> <p> Если резервное копирование служебной базы данных не происходит, то значение параметра <code>connection_monitoring</code> игнорируется.</p>
messages_details	<p>Извлечение атрибутов писем (<i>детализация</i>) во время резервного копирования. От значения этого параметра при выполнении РК зависит, какие данные будут доступны для просмотра при гранулярном восстановлении из этой РК.</p> <p>true Из писем извлекаются все доступные поля: <i>от кого, кому, тема, копия, скрытая копия, размер, дата письма и ID письма</i>.</p> <p>false Из писем извлекаются поля: <i>размер, дата письма и ID письма</i>.</p> <p>По умолчанию true</p>

Параметр	Описание
<code>thread_count</code>	<p>Количество потоков выполнения детализации писем при резервном копировании.</p> <p> Не используется, если у параметра <code>messages_details</code> задано значение <code>false</code></p> <p>Возможные значения</p> <p>Целое число >0</p> <p>По умолчанию</p> <p>10</p>


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

Кнопка **OK** сохраняет настройки.

Глава 15. Тонкие настройки модуля RuPost для восстановления резервной копии

В [таблице](#) описаны тонкие настройки модуля RuPost для восстановления резервной копии (см. [Раздел 11.2](#)).

Таблица 5. Тонкие настройки модуля RuPost для восстановления резервной копии

Параметр	Описание
Использовать настройки по умолчанию	<p>Использование значений по умолчанию</p> <p>true</p> <p>Для параметров используются значения по умолчанию. Восстановление из РК происходит для почтовых ящиков, архива почтовых ящиков, удаленных писем и служебной базы данных всех пользователей почтовой системы.</p> <p>false</p> <p>Значения параметров можно изменить.</p> <p>По умолчанию</p> <p>true</p>
restore_maildir	<p>Выполнение восстановления почтовых ящиков всех пользователей почтовой системы</p> <p>true</p> <p>Восстановление выполняется.</p> <p>false</p> <p>Восстановление не выполняется.</p> <p>По умолчанию</p> <p>true</p> <p> Параметр используется только для полного восстановления почтовых ящиков.</p>

Параметр	Описание
restore_archive	<p>Выполнение восстановления архива почтовых ящиков всех пользователей почтовой системы</p> <p>true Восстановление выполняется.</p> <p>false Восстановление не выполняется.</p> <p>По умолчанию <code>true</code></p> <p> Параметр используется только для полного восстановления архивов почтовых ящиков.</p>
restore_record	<p>Выполнение восстановления удаленных писем всех пользователей почтовой системы</p> <p>true Восстановление выполняется.</p> <p>false Восстановление не выполняется.</p> <p>По умолчанию <code>true</code></p> <p> Параметр используется только для полного восстановления удаленных писем.</p>
restore_database	<p>Выполнение восстановления служебной базы данных из РК</p> <p>true Восстановление выполняется.</p> <p>false Восстановление не выполняется.</p> <p>По умолчанию <code>false</code></p> <p> Параметр используется только для полного восстановления служебной базы данных.</p>

Кнопка **OK** сохраняет настройки.