

RuBackup

Система резервного копирования и восстановления данных

Руководство по установке и обновлению серверов резервного копирования и Linux-клиентов RuBackup



Содержание

| | |
|---|----|
| Введение..... | 5 |
| Перед установкой..... | 6 |
| Поддерживаемые версии источников данных..... | 6 |
| Платформы виртуализации..... | 6 |
| Базы данных..... | 7 |
| Бизнес-приложения..... | 7 |
| Файловые системы и др..... | 8 |
| Системные требования..... | 9 |
| Установка "Все в одном"..... | 9 |
| Сервер RuBackup..... | 10 |
| База данных RuBackup..... | 12 |
| Медиасервер..... | 13 |
| Клиент резервного копирования..... | 14 |
| Модули резервного копирования..... | 17 |
| Менеджер администратора RuBackup (RBM)..... | 23 |
| REST API..... | 23 |
| Сетевые порты..... | 24 |
| Особенности установки пакетов в Linux..... | 27 |
| Подготовка кластера СУБД PostgreSQL к установке служебной базы данных RuBackup..... | 28 |
| Установка "Все в одном"..... | 31 |
| Подготовка к установке..... | 31 |
| Необходимые пакеты..... | 31 |
| Пакеты для ОС без графической оболочки..... | 32 |
| Инсталляция пакетов RuBackup..... | 32 |
| Установка лицензии..... | 33 |
| Парольная политика для локальных учетных записей RuBackup..... | 35 |
| Настройка СРК RuBackup в формате «Все в одном»..... | 35 |
| Настройка пользователей на сервере RuBackup..... | 42 |

| | |
|--|----|
| Запуск сервера RuBackup..... | 43 |
| Настройка ограничения на количество открытых файловых дескрипторов на хосте с сервером RuBackup..... | 45 |
| Ручной запуск..... | 48 |
| Запуск сервиса сервера RuBackup..... | 48 |
| Запуск сервера в терминальном режиме..... | 49 |
| Настройка хранилища резервных копий..... | 49 |
| Развернутая установка..... | 50 |
| Установка основного сервера..... | 50 |
| Подготовка к установке основного сервера..... | 50 |
| Инсталляция основного сервера RuBackup..... | 51 |
| Установка лицензии..... | 53 |
| Настройка основного сервера..... | 54 |
| Настройка пользователей на сервере RuBackup..... | 61 |
| Запуск основного сервера RuBackup..... | 62 |
| Запуск основного сервера в терминальном режиме..... | 65 |
| Настройка хранилища резервных копий..... | 65 |
| Установка резервного сервера..... | 66 |
| Подготовка к установке резервного сервера..... | 66 |
| Инсталляция резервного сервера..... | 67 |
| Установка лицензии..... | 68 |
| Настройка резервного сервера..... | 69 |
| Настройка пользователей на резервном сервере RuBackup..... | 73 |
| Запуск резервного сервера RuBackup..... | 74 |
| Запуск резервного сервера в терминальном режиме..... | 77 |
| Настройка хранилища резервных копий..... | 78 |
| Установка медиасервера..... | 79 |
| Подготовка к установке медиасервера..... | 79 |
| Инсталляция медиасервера RuBackup..... | 80 |
| Установка лицензии..... | 81 |

| | |
|---|-----|
| Настройка медиасервера..... | 82 |
| Настройка пользователей на медиасервере RuBackup..... | 87 |
| Запуск медиасервера RuBackup..... | 87 |
| Запуск медиасервера в терминальном режиме..... | 91 |
| Настройка хранилища резервных копий..... | 91 |
| Установка клиента..... | 92 |
| Подготовка к установке клиента..... | 92 |
| Пакеты для ОС без графической оболочки..... | 92 |
| Инсталляция клиента RuBackup..... | 92 |
| Настройка клиента RuBackup..... | 94 |
| Настройка пользователей на клиенте RuBackup..... | 96 |
| Запуск клиента RuBackup..... | 98 |
| Дополнительные настройки..... | 100 |
| Установка RBM на удаленном хосте..... | 100 |
| Подготовка к установке..... | 100 |
| Удаление клиента RuBackup..... | 103 |
| Удаление сервера RuBackup..... | 103 |
| Мастер-ключ..... | 104 |
| Неинтерактивный режим работы..... | 105 |
| Установка обновлений RuBackup..... | 106 |
| Подготовка к обновлению..... | 107 |
| Обновление..... | 107 |
| Восстановление базы данных..... | 111 |
| Установка нового модуля..... | 114 |
| Установка нового модуля вместе с обновлением СРК..... | 114 |
| Установка нового модуля без обновления СРК..... | 114 |
| Приложение А..... | 115 |
| Приложение Б..... | 119 |

Введение

Система резервного копирования и восстановления данных RuBackup (далее – Система, СРК) – системное клиент-серверное приложение, предназначенное для автоматизированного выполнения процедур резервного копирования данных серверов, виртуальных машин, баз данных и приложений в центрах обработки данных, а также для восстановления данных из резервных копий по расписанию, запросу пользователя или системного администратора.

RuBackup является мощным и гибким средством автоматизации, предназначенным для защиты информации центра обработки данных и корпоративной сети предприятия.

Перед развертыванием системы резервного копирования в вашем центре обработки данных необходимо провести планирование необходимых ресурсов, которые потребуются для ее работы. Следует учесть для каких данных требуется выполнять резервные копии, как часто, какие временные окна допустимы для проведения операций резервного копирования данных, какое допустимое время восстановления данных должно быть в случае их утраты по основному месту хранения и много других нюансов.

Настоящее руководство описывает базовые шаги установки сервера и клиента резервного копирования и предназначено для системных администраторов, отвечающих за внедрение и сопровождение СРК .

Принципы работы СРК и вопросы ее администрирования изложены в документе «RuBackup. Руководство системного администратора».

Перед установкой

Поддерживаемые версии источников данных

Платформы виртуализации

- ISPsystem VMmanager 2023
- ПК СВ "Брест" 3.3 (кроме 3.3.1)
- ПК СВ "Брест" 3.2
- ПК СВ "Брест" 3.1
- ПК СВ "Брест" 2.12
- RUSTACK 2021.2
- RUSTACK 2.6
- АЭРОДИСК VAIR 3.8
- АЭРОДИСК VAIR 3.7
- VMware vSphere 7.0.3.00600
- OpenStack 2023.1 Antelope (SLIRP)
- OpenStack Zed
- OpenStack Yoga
- OpenStack Xena
- OpenStack Neutron
- zVirt
- DynamiX (экспериментальный модуль, прошел дизайн-тестирование)
- KVM (экспериментальный модуль, прошел дизайн-тестирование)
- Tionix (экспериментальный модуль, прошел дизайн-тестирование)
- ECP Veil (экспериментальный модуль, прошел дизайн-тестирование)

- oVirt (экспериментальный модуль, прошел дизайн-тестирование)
- REDVirt (экспериментальный модуль, прошел дизайн-тестирование)
- P-Виртуализация (экспериментальный модуль, прошел дизайн-тестирование)

Базы данных

- Tanor Special Edition 15
- PostgreSQL 16
- PostgreSQL 15
- PostgreSQL 14
- PostgreSQL 13
- PostgreSQL 12
- PostgreSQL 11
- PostgreSQL 10
- PostgreSQL 9
- Patroni 3.0
- Patroni 2.1
- Postgres Pro 15
- Postgres Pro 14
- Postgres Pro 13
- Postgres Pro 12
- Postgres Pro 11
- Greenplum Database 6

Бизнес-приложения

- CommuniGate Pro 6.3
- FreeIPA 4.9
- FreeIPA 4.8
- FreeIPA 4.7
- Mailion (экспериментальный модуль, прошел дизайн-тестирование)

Файловые системы и др.

Linux:

- Ext4
- Ext3
- Ext2
- XFS
- LVM2

Системные требования

В данном разделе представлены актуальные системные требования для всех компонентов группировки RuBackup, конфигурационной базы данных RuBackup, доступных модулей резервного копирования, Менеджера администратора RuBackup, а также для установки "Все в одном". Данный раздел документа обновляется по мере выхода новых модулей резервного копирования и добавления поддержки операционных систем путем выпуска обновленных сборок RuBackup. Настоятельно рекомендуется следовать нижеуказанным рекомендациям для обеспечения должного быстродействия решения.

Установка "Все в одном"

Оборудование

Таблица 1 — Оборудование

| Аппаратный компонент | Значение | Примечание |
|-----------------------|----------|---|
| Процессор | 4 ядра | |
| Оперативная память | от 4 ГБ | Если клиент резервного копирования используется на одной машине с остальными компонентами RuBackup, рассчитать необходимое количество оперативной памяти для операций клиента резервного копирования можно по формуле из раздела "Клиент резервного копирования". |
| Дисковое пространство | 480 ГБ | Без учета совокупного объема хранимых резервных копий, в случае когда хранение производится непосредственно на медиасerverе. |

Операционные системы

- Astra Linux 1.8
- Astra Linux 1.7.5
- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04

- CentOS 8
- CentOS 7
- Alt Linux 10
- РЕД ОС 7.3

Полные списки операционных систем, поддерживаемых различными компонентами, такими как Сервер RuBackup, Медиасервер, а также Клиент резервного копирования, могут быть шире, чем указано в списке операционных систем для установки "Все в одном". Для получения детальной информации по поддерживаемым операционным системам для каждого из компонентов группировки RuBackup обратитесь к соответствующим разделам ниже.

Список операционных систем, поддерживаемых различными модулями резервного копирования, может отличаться от списка операционных систем для установки "Все в одном". Для получения детальной информации по поддерживаемым операционным системам для каждого из модулей резервного копирования обратитесь к разделу "Модули резервного копирования".

Сервер RuBackup

Оборудование

Рекомендуемая конфигурация сервера RuBackup зависит от совокупного объема хранимых данных. Для планирования конфигурации сервера воспользуйтесь таблицей ниже.

| Аппаратный компонент | | Объем хранимых данных | | | Примечание |
|-----------------------|--------------------------------|--|--------|--------|---|
| Процессор | | 48 ТБ | 96 ТБ | 144 ТБ | Рекомендуемые модели: Intel Xeon 4210, AMD EPYC 7000 или более современные |
| | | 10 ядер, 20 потоков (2 потока на 1 ядро или более) | | | |
| Оперативная память | | 128 ГБ | 256 ГБ | 256 ГБ | |
| Дисковое пространство | Твердотельный накопитель (SSD) | RAID 1, 2 диска по 480 ГБ каждый | | | Объем дискового пространства для установки операционной системы и компонентов RuBackup, за исключением конфигурационной базы данных RuBackup. |
| | Твердотельный накопитель, | 3.84 ТБ | | | |
| | | | | | <ul style="list-style-type: none"> • Рекомендуется в случае развертывания |

| | | | | | |
|--|--|--|--|---|--|
| | <p>подключенный через шину PCI Express (NVMe SSD)</p> | | | | <p>инстанса PostgreSQL для конфигурационной базы данных RuBackup на той же машине, где установлен сервер RuBackup.</p> <ul style="list-style-type: none"> • Диски NVMe SSD позволяют повысить производительность операций в фильтре Блума и скорость обработки данных при выполнении процессов дедупликации. • 3.84 Тб предусматривают потенциальный рост объемов обрабатываемых данных. • Для обеспечения максимального уровня отказоустойчивости и быстродействия при промышленной эксплуатации рекомендуется использовать в качестве конфигурационной базы RuBackup СУБД PostgreSQL в отказоустойчивой конфигурации, например, с использованием решения Patroni, развернутом на отдельностоящих машинах. |
| | <p>Жесткий диск (HDD) или флэш-накопитель (flash drive)</p> | <p>RAID 50, 12 дисков по 4 ТБ каждый</p> | <p>RAID 50, 12 дисков по 8 ТБ каждый</p> | <p>RAID 50, 12 дисков по 12 ТБ каждый</p> | <ul style="list-style-type: none"> • Рекомендуется в случае активного использования машины с основным сервером в качестве медиасервера, для возможности расширения дискового |

| | | | | | |
|-------------|--|--|--|--|---|
| | | | | | <p>пространства под хранение резервных копий.</p> <ul style="list-style-type: none"> • В случае хранения данных на опосредованных СХД, данный компонент не используется. |
| Сеть | | Два сетевых адаптера с пропускной способностью 10 Гб каждый, с 2 портами (dual port) | | | |

Операционные системы

- Astra Linux 1.8
- Astra Linux 1.7.5
- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10
- РЕД ОС 7.3

База данных RuBackup

Оборудование

Таблица 2 — Оборудование

| Аппаратный компонент | Значение |
|-----------------------|----------|
| Процессор | 4 ядра |
| Оперативная память | 64 ГБ |
| Дисковое пространство | 3,84 ТБ |

Для обеспечения максимального уровня отказоустойчивости и быстродействия при промышленной эксплуатации, рекомендуется использовать в качестве конфигурационной базы RuBackup СУБД PostgreSQL в отказоустойчивой конфигурации с использованием решения Patroni, развернутом на отдельно стоящих машинах, с совокупным объемом дискового пространства 3.84 ТБ, построенного с использованием твердотельных накопителей, подключенных через шину PCI Express (NVMe SSD).

Поддерживаемые версии СУБД в качестве базы данных RuBackup

- Tantor Special Edition 15
- PostgreSQL 16
- PostgreSQL 15
- PostgreSQL 14
- PostgreSQL 13
- PostgreSQL 12
- PostgreSQL 11
- Patroni 3.0

Медиасервер

Оборудование

Рекомендуемая конфигурация медиасервера зависит от совокупного объема хранимых данных и схожа с конфигурацией сервера RuBackup. Для расчета конфигурации медиасервера воспользуйтесь таблицей ниже.

Таблица 3 — Оборудование

| Аппаратный компонент | Объем хранимых данных | | | Примечание |
|----------------------|--|--------|--------|--|
| | 48 ТБ | 96 ТБ | 144 ТБ | |
| Процессор | 10 ядер, 20 потоков (2 потока на 1 ядро или более) | | | Рекомендуемые модели: Intel Xeon 4210, AMD EPYC 7000 или более современные |
| Оперативная память | 128 ГБ | 256 ГБ | 256 ГБ | |

| | | | | | |
|------------------------------|---|--|-----------------------------------|------------------------------------|--|
| Дисковое пространство | Твердотельный накопитель (SSD) | RAID 1, 2 диска по 480 ГБ каждый | | | Объём дискового пространства для установки операционной системы и компонентов RuBackup. |
| | Жесткий диск (HDD) или флэш-накопитель (flash drive) | RAID 50, 12 дисков по 4 ТБ каждый | RAID 50, 12 дисков по 8 ТБ каждый | RAID 50, 12 дисков по 12 ТБ каждый | |
| Сеть | | Два сетевых адаптера с пропускной способностью 10 Гб каждый, с 2 портами (dual port) | | | <ul style="list-style-type: none"> Для возможности расширения дискового пространства под хранение резервных копий. В случае хранения данных на опосредованных СХД, данный компонент не используется. |

Операционные системы

- Astra Linux 1.8
- Astra Linux 1.7.5
- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 7
- CentOS 8
- Alt Linux 10

Клиент резервного копирования

Оборудование

Процессор

1 ядро

Оперативная память

Объем оперативной памяти (в байтах) для одного обрабатываемого ресурса рассчитывается по следующей формуле:

Оперативная память для одного ресурса = 1ГБ + 4% от размера целевого ресурса.

Для расчета общего количества оперативной памяти для клиента резервного копирования необходимо сложить получившиеся значения оперативной памяти для всех задач резервного копирования, выполняемых одновременно.

Оперативная память для клиента = Оперативная память для ресурса №1 + Оперативная память для ресурса №2 + ... + Оперативная память для ресурса №N

Для пула типа "Block device" размера блока может быть задан при создании пула. Значением по умолчанию является 131072 Б. Для получения более подробной информации по настройке пулов обратитесь к секции "Пулы" раздела "Хранилища" Руководства системного администратора RuBackup.

Для пулов типов "File system", "Tape library", "Cloud" размер блока является фиксированным и равен 16384 Б.

Для всех типов пулов длина ключа хеш-функции зависит от выбранной хеш-функции в настройках пула. Например, для хеш-функции SHA1 длина ключа составляет 20 Б.

Дисковое пространство

- **Резервное копирование:** объём свободного дискового пространства, составляющий не менее 3% от совокупного объёма данных, резервное копирование которых осуществляется одновременно.
- **Восстановление данных:** объём свободного дискового пространства должен быть не менее совокупного объёма одновременно восстанавливаемых данных с использованием данного клиента.
- **Многопоточное резервное копирование:** объём свободного дискового пространства зависит от выбранных параметров: количества потоков, размера блока и длины хеша. Чем больше используется потоков, тем больше требуемый объём. Чем меньше выбранный размер блока, тем больше требуется доступного пространства на диске. Чем больше длина хеша, тем больше требуется памяти.
- **Расчёт требуемого объёма:** Приблизительный расчёт требуемого объёма доступного пространства в многопоточном режиме можно оценить как $(worker_parallelism *)\%$ от ресурса. Это означает, что для каждого рабочего потока, который будет использоваться при многопоточной обработке данных, потребуется определённый объём доступного пространства на диске.

* - Worker_parallelism относится к возможности одновременного выполнения нескольких процессов в рамках одного приложения.

Более точный расчёт максимально требуемого объёма свободного пространства на диске при бекапе можно оценить по формуле:

$$V = \frac{\text{Объём ресурса}}{\text{Размер блока}} \times (\text{Размер хеша} + 20) \times (K + 1) + \text{Размер метаданных}$$

где:

- K = 1 при однопоточном режиме;
- K = worker_parallelism, если заданы многопоточный режим (enable_multithreading) и слабая дедупликация (enable_flexible_dedup).

Эта формула позволяет рассчитать объём памяти, необходимый для хранения данных во время процесса бэкапа.

В формуле используются следующие обозначения:

- Объём ресурса — общий объём данных, подлежащих бэкапу;
- Размер блока — размер блока данных, используемого для обработки данных во время бэкапа;
- Размер хеша — размер хеша, используемого для идентификации данных;
- worker_parallelism — количество рабочих потоков, используемых для выполнения бэкапа;
- enable_multithreading — флаг, указывающий на использование многопоточности;
- enable_flexible_dedup — флаг, указывающий на использование гибкой дедупликации;
- 20 — максимальный размер сериализованной позиции в файле;
- 1 — временная база для вычисления сигнатуры или отправки хешей на сервер;
- размер метаданных – это 0.02 * объём ресурса.

Ниже представлена таблица с примерами расчетов.

| Ресурс | Хеш | Блок | K | Размер метаданных | ДискПространство (ГБ) |
|--------------|-----|---------|-----|-------------------|-----------------------|
| 536870912000 | 64 | 8192 | 8 | 10737418240 | 56 |
| 536870912000 | 64 | 8192 | 32 | 10737418240 | 179 |
| 536870912000 | 64 | 8192 | 64 | 10737418240 | 343 |
| 536870912000 | 64 | 8192 | 128 | 10737418240 | 671 |
| 536870912000 | 64 | 1048576 | 8 | 10737418240 | 10 |
| 536870912000 | 64 | 1048576 | 32 | 10737418240 | 11 |

| | | | | | |
|---------------|----|---------|-----|-------------|------|
| 536870912000 | 64 | 1048576 | 64 | 10737418240 | 12 |
| 536870912000 | 64 | 1048576 | 128 | 10737418240 | 15 |
| 1099511627776 | 64 | 8192 | 8 | 21990232555 | 114 |
| 1099511627776 | 64 | 8192 | 32 | 21990232555 | 366 |
| 1099511627776 | 64 | 8192 | 64 | 21990232555 | 702 |
| 1099511627776 | 64 | 8192 | 128 | 21990232555 | 1374 |
| 1099511627776 | 64 | 1048576 | 8 | 21990232555 | 21 |
| 1099511627776 | 64 | 1048576 | 32 | 21990232555 | 23 |
| 1099511627776 | 64 | 1048576 | 64 | 21990232555 | 25 |
| 1099511627776 | 64 | 1048576 | 128 | 21990232555 | 31 |

Операционные системы

- Astra Linux 1.8
- Astra Linux 1.7.5
- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10
- Debian 10
- Red Hat Enterprise Linux 7
- Red Hat Enterprise Linux 8
- РЕД ОС 7.3

Модули резервного копирования

Операционные системы

Платформы виртуализации

Модуль для ISPsystem VMmanager

- Astra Linux 1.7
- Astra Linux 1.6
- CentOS 8
- CentOS 7
- Alt Linux 10

Модуль для ПК СВ "Брест"

- Astra Linux 1.7
- Astra Linux 1.6

Модуль для РУСТЭК

- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04

Модуль для АЭРОДИСК VAIR

- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- Debian 10

Модуль для VMware vSphere

- Ubuntu 20.04

Модуль для zVirt

- CentOS 8

Модуль для oVirt (экспериментальный модуль, прошел дизайн-тестирование)

- CentOS 8

Модуль для REDVirt (экспериментальный модуль, прошел дизайн-тестирование)

- CentOS 8

Модуль для P-Виртуализация (экспериментальный модуль, прошел дизайн-тестирование)

- CentOS 7

Модуль для OpenStack (экспериментальный модуль, прошел дизайн-тестирование)

- Astra Linux 1.7

Модуль для Tionix (экспериментальный модуль, прошел дизайн-тестирование)

- Alt Linux 10

Модуль для DynamiX (экспериментальный модуль, прошел дизайн-тестирование)

Модуль для KVM (экспериментальный модуль, прошел дизайн-тестирование)

Модуль для ECP Veil (экспериментальный модуль, прошел дизайн-тестирование)

Базы данных

Модуль для PostgreSQL и Patroni (Universal)

- Astra Linux 1.7
- Astra Linux 1.7.5
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04

- CentOS 8
- CentOS 7
- Alt Linux 10
- Red Hat Enterprise Linux 7
- Red Hat Enterprise Linux 8
- РЕД ОС 7.3

Модуль для PostgreSQL (резервное копирование и восстановление индивидуальных баз данных и таблиц)

- Astra Linux 1.8
- Astra Linux 1.7.5
- Astra Linux 1.7
- Astra Linux 1.6
- Debian 10
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10
- Red Hat Enterprise Linux 7
- Red Hat Enterprise Linux 8
- РЕД ОС 7.3

Модуль для Postgres Pro

- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7

- Alt Linux 10
- РЕД ОС 7.3

Модуль для Greenplum

- CentOS 7

Бизнес-приложения

Модуль для CommuniGate Pro

- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10

Модуль для FreeIPA

- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10

Модуль для Mailion (экспериментальный модуль, прошел дизайн-тестирование)

Файловые системы и др.

Модуль файловых систем Linux (входит в состав клиента резервного копирования)

- Astra Linux 1.8
- Astra Linux 1.7.5
- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10
- Debian 10
- Red Hat Enterprise Linux 7
- Red Hat Enterprise Linux 8
- РЕД ОС 7.3
- ОС РОСА «КОБАЛЬТ» 7.9

Модуль для LVM (входит в состав клиента резервного копирования)

- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10
- Debian 10
- Red Hat Enterprise Linux 7
- Red Hat Enterprise Linux 8

Менеджер администратора RuBackup (RBM)

Оборудование

Таблица 4 — Оборудование

| Аппаратный компонент | Значение |
|-----------------------|----------|
| Процессор | 4 ядра |
| Оперативная память | 4 ГБ |
| Дисковое пространство | 30 ГБ |

Операционные системы:

- Microsoft Windows (экспериментальный режим, прошел дизайн-тестирование)
- Astra Linux 1.8
- Astra Linux 1.7.5
- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10

REST API

Операционные системы

- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7

- Alt Linux 10

Сетевые порты

Безопасное соединение компонентов СРК RuBackup и обмен информацией между ними подразумевает техническую возможность коммуникации по сети. Перед установкой продукта необходимо обеспечить взаимодействие компонентов СРК путем открытия соответствующих портов для входящего и исходящего трафика между серверами, на которых установлены компоненты СРК.

В таблице 5 представлены компоненты СРК RuBackup, которые принимают входящие соединения по указанным портам и протоколам

Таблица 5 — Сетевые порты

| Компонент | | Целевой сервис | Протокол | Порт | Описание |
|-------------------|--|----------------|----------|--------|---|
| от | до | | | | |
| Основной сервер | Медиа сервер | rubackup-cmd | TCP | 9991 | Управление операциями на медиа сервере |
| | | rubackup-media | TCP | 9993 | Управление операциями с данными |
| Основной сервер | База данных RuBackup на отдельной машине | postgresql | TCP | 5432** | Сохранение конфигурационной и оперативной информации |
| Резервный сервер* | Основной сервер | rubackup-cmd | TCP | 9991 | Обеспечение отказоустойчивости |
| | | rubackup-media | TCP | 9993 | Передача данных между медиа серверами в составе основного и резервного серверов |
| Резервный сервер* | База данных RuBackup на отдельной машине | postgresql | TCP | 5432 | Сохранение конфигурационной и оперативной информации |
| Медиа сервер | Медиа сервер | rubackup-media | TCP | 9993 | Передача данных между медиа серверами |
| Медиа сервер | Резервный сервер* | rubackup-cmd | TCP | 9991 | Управление операциями на медиа сервере |
| | | rubackup-media | TCP | 9993 | Управление операциями с данными |
| Медиа сервер | База данных RuBackup на отдельной машине | postgresql | TCP | 5432** | Сохранение конфигурационной и оперативной информации |

| Клиент резервного копирования | Основной сервер | rubackup-cmd | TCP | 9991 | Управление операциями на клиенте резервного копирования |
|---|--|----------------|----------|---------|---|
| Клиент резервного копирования | Медиа сервер | rubackup-media | TCP | 9993 | Передача данных между медиа сервером и клиентом |
| Клиент резервного копирования | Резервный сервер* | rubackup-cmd | TCP | 9991 | Управление операциями на клиенте резервного копирования |
| | | rubackup-media | TCP | 9993 | Передача данных между медиа сервером и клиентом |
| Компонент | | Целевой сервис | Протокол | Порт | Описание |
| от | до | | | | |
| RuBackup REST API | Основной сервер | rubackup-rbm | TCP | 9995 | Отправка запросов на сервер и получение информации |
| RuBackup REST API*** | База данных RuBackup на отдельной машине | postgresql | TCP | 5432** | Получение информации из базы данных |
| RuBackup REST API | Резервный сервер* | rubackup-rbm | TCP | 9995 | Отправка запросов на сервер и получение информации |
| Менеджер RuBackup (RBM) на отдельной машине | База данных RuBackup на отдельной машине | postgresql | TCP | 5432** | Сохранение конфигурационной и оперативной информации |
| Менеджер RuBackup (RBM) на отдельной машине | Основной сервер | rubackup-rbm | TCP | 9995 | Управление операциями RuBackup |
| Менеджер RuBackup (RBM) на отдельной машине | Резервный сервер* | rubackup-rbm | TCP | 9995 | Управление операциями RuBackup |
| Клиент, посылающий запрос через Rubackup REST API | Основной сервер | rubackup-api | HTTPS | 443**** | Управление операциями RuBackup через REST API |
| Клиент, посылающий запрос через Rubackup REST API | Резервный сервер* | rubackup-api | HTTPS | 443**** | Управление операциями RuBackup через REST API |

* При наличии резервного сервера.

** Если база данных сконфигурирована с использованием нестандартного порта, то для подключения к ней продукта RuBackup порт может быть изменен вручную в конфигурационном файле **`/opt/rubackup/etc/config.file`**.

*** Если база данных использует нестандартный порт, то для подключения к ней RuBackup REST API порт можно поменять вручную в конфигурационном файле **`/opt/rubackup/etc/rubackup_api.conf`**

**** Если необходимо использовать нестандартный порт для подключения к RuBackup REST API, его можно поменять вручную в конфигурационном файле **`/opt/rubackup/etc/rubackup_api.conf`**.

Особенности установки пакетов в Linux

Дистрибутивы сервера и клиента RuBackup могут поставляться в виде deb и rpm-пакетов. Для разных дистрибутивов Linux, по причине их отличий друг от друга, предусмотрены специально подготовленные пакеты RuBackup.

Перед установкой клиентского и серверного пакетов необходимо установить пакет **rubackup-common** необходимой версии, подходящий для Вашего дистрибутива Linux. Например:

```
$ sudo dpkg -i rubackup-common_2.0.0-1_amd64.deb
```

или

```
$ sudo rpm -i rubackup-common-2.0.0-1.el7.x86_64.rpm
```

Перечень клиентских и серверных пакетов, устанавливаемых в различных операционных системах, представлен в приложении (см. Приложение Б).

В зависимости от типа используемого пакетного менеджера в Вашем дистрибутиве Linux, процедура установки и удаления пакетов может использовать команды `dpkg`, `rpm`, `apt`, `yum` и пр. В настоящем руководстве процедуры установки описаны для пакетного менеджера, который оперирует пакетами deb. Например, процедура установки пакета клиента RuBackup выглядит следующим образом:

```
$ sudo dpkg -i rubackup-client_2.0.0-1_amd64.deb
```

Для установки клиента RuBackup в ОС с пакетным менеджером, который оперирует rpm-пакетами, вместо вышеуказанной команды следует выполнить команду:

```
$ sudo rpm -i rubackup-client-2.0.0-1.el8.x86_64.rpm
```

Процедуры удаления пакетов в настоящем руководстве описаны для пакетного менеджера, который оперирует пакетами deb. Например, процедура удаления пакета клиента RuBackup выглядит следующим образом:

```
$ sudo apt remove rubackup-client
```

Для удаления клиента RuBackup в операционной системе с пакетным менеджером, который оперирует rpm-пакетами, вместо вышеуказанной команды следует выполнить:

```
$ sudo yum remove rubackup-client
```

Либо:

```
$ sudo rpm -e rubackup-client
```

Некоторые операционные системы, такие как Alt Linux, используют пакетную систему rpm, но вместо yum используют apt. Перед установкой или удалением пакетов RuBackup следует уточнить, какие команды необходимо использовать для вашего дистрибутива Linux.

Подготовка кластера СУБД PostgreSQL к установке служебной базы данных RuBackup

СУБД PostgreSQL используется для хранения метаданных резервных копий и конфигурационных параметров системы резервного копирования RuBackup.

В начале процедуры подготовки кластера СУБД PostgreSQL установите пакет `postgresql` для инсталляции PostgreSQL:

```
$ sudo apt install postgresql
```

и пакет `postgresql-contrib`, содержащий дополнительные модули:

```
$ sudo apt install postgresql-contrib
```

При установке сервера PostgreSQL в ОС Astra Linux SE 1.6 необходимо установить пакет `postgresql-contrib-9.6!`

Перед установкой сервера RuBackup в конфигурационный файл `pg_hba.conf` необходимо добавить возможность подключения к СУБД для всех серверов, которые будут входить в серверную группировку RuBackup. Например:

```
local all postgres peer
# 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
host all all 192.168.0.50/24 md5
host all all 192.168.0.51/24 md5
host all all 192.168.0.52/24 md5
host all all 192.168.0.53/24 md5
```

При этом можно оставить строку `local all postgres peer`.

В файле postgresql.conf необходимо настроить listener:

```
#-----  
# CONNECTIONS AND AUTHENTICATION  
#-----  
  
# - Connection Settings -  
  
#listen_addresses = 'localhost'   # what IP address(es) to listen on;  
listen_addresses = '*'  
                                # comma-separated list of addresses;  
                                # defaults to 'localhost'; use '*' for all  
                                # (change requires restart)  
port = 5432                      # (change requires restart)  
max_connections = 100            # (change requires restart)
```

Примечания:

1. Возможность подключения к СУБД для всех серверов, которые будут входить в серверную группировку RuBackup, можно добавить и после установки сервера RuBackup в конфигурационный файл pg_hba.conf, после чего необходимо перезапустить PostgreSQL.

2. Размеры параметров в файле postgresql.conf shared_buffers ~50 % от размера оперативной памяти. При использовании дедупликации рекомендуется минимальный объем оперативной памяти сервера 64 GB effective_cache_size ~70 % от размера оперативной памяти work_mem 32 MB.

3. max_parallel_workers – не менее 50 % от количества процессорных ядер, если сервер СУБД совмещен с сервером RuBackup и 100 %, если сервер СУБД является выделенным.

После внесения этих изменений необходимо:

1. Перезагрузить сервис postgresql:

```
$ sudo service postgresql restart
```

2. Проверить подключение к СУБД:

```
$ sudo -u postgres psql
```

```
psql (12.5 (Ubuntu 12.5-0ubuntu0.20.04.1))
```

```
Type "help" for help.
```

```
postgres=#
```

3. Задать пароль для пользователя базы данных postgres :

```
$ sudo -u postgres psql  
psql (12.5 (Ubuntu 12.5-0ubuntu0.20.04.1))  
Type "help" for help.
```

```
postgres=# alter user postgres password '12345';  
ALTER ROLE  
postgres=#
```

При установке СУБД PostgreSQL в ОС Astra Linux Special Edition с максимальным уровнем защищенности («Смоленск»), чтобы избежать ошибок получения мандатного контроля целостности для пользователя, необходимо:

Вариант 1 (рекомендуемый). В файле /etc/parsec/mswitch.conf для параметра zero_if_notfound установите значение yes. После этого перезагрузите сервис postgresql:

```
$ sudo systemctl restart postgresql
```

Вариант 2.

1. Создайте пользователя rubackup в ОС Astra Linux Special Edition:

```
$ sudo useradd --system --no-user-group rubackup
```

2. Установите права пользователя rubackup, а также разрешить пользователю postgres чтение мандатных меток:

```
$ sudo pdpl-user -l 0:0 rubackup
```

```
$ setfacl -d -m u:postgres:r /etc/parsec/macdb
```

```
$ setfacl -R -m u:postgres:r /etc/parsec/macdb
```

Установка "Все в одном"

Развертывание RuBackup в формате "Все в одном" подразумевает установку всех компонентов группировки RuBackup на одной физической или виртуальной машине. Данный тип установки может быть использован в случаях, когда необходимо провести пилотную эксплуатацию функциональных возможностей решения или требуется защищать небольшие объемы данных, например, при использовании для демонстраций функциональных возможностей, в домашнем окружении, в "песочнице" или в небольшой организации.

Внимание! Развертывание RuBackup в формате "Все в одном" не предполагает использование резервного сервера и дополнительных медиасерверов. Чтобы включить данные компоненты в состав RuBackup, перейдите к разделу Развернутая установка .

Подготовка к установке

Необходимые пакеты

Перед установкой "Все в одном" необходимо, чтобы в системе были установлены следующие пакеты:

| Название пакета | Описание |
|--|--|
| mailutils (или bsd-mailx) | Утилиты электронной почты |
| libcurl4 (возможно libcurl3 в зависимости от ОС) | Клиентская библиотека передачи данных по URL |
| libqt5sql5-psql | Необходим для ОС Astra Linux |
| libsasl2-2, libldap-2.4-2 | Необходимы для операционных систем Ubuntu 18.04, Ubuntu 20.04, Astra Linux Special Edition 1.6, Astra Linux Special Edition 1.7, Debian 10 |
| openldap, cyrus-sasl | Необходимы для операционных систем РЕД ОС 7.3, CentOS 8, CentOS 7 |
| libldap2, libsasl2-3 | Необходимы для операционной системы ALT Linux 10 |

Внимание! Названия пакетов могут отличаться в зависимости от дистрибутива Linux.

Чтобы система уведомлений RuBackup работала корректно, необходимо настроить отправку электронной почты с сервера RuBackup. Для отправки электронной почты сервер RuBackup использует утилиту /usr/bin/mail.

При использовании ленточной библиотеки с сервером резервного копирования, настройку см. в руководстве «Работа с ленточной библиотекой».

Пакеты для ОС без графической оболочки

Если вы устанавливаете сервер RuBackup на ОС без графической оболочки, то, чтобы использовать Менеджер администратора RuBackup (RBM), установите следующие пакеты:

```
$ sudo apt install libgl1-mesa-dev  
$ sudo apt install libxkbcommon-x11-0  
$ sudo apt install libfontconfig1
```

Инсталляция пакетов RuBackup

Для установки СРК RuBackup «Все в одном» выполните следующие действия:

1. Авторизуйтесь под пользователем root, командой:

```
$ sudo -i
```

2. Настройте переменные среды для пользователя root. Для этого добавьте следующие строки в файл /root/.bashrc:

```
PATH=$PATH:/opt/rubackup/bin  
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib  
export PATH  
export LD_LIBRARY_PATH
```

Эти переменные также можно определить в файле /etc/environment.

3. Для перехода в каталог /root/, выполните команду:

```
cd /root
```

4. Перезагрузите переменные окружения, командой:

```
# . .bashrc
```

5. Установите пакет **rubackup-common**, как представлено на примере:

```
$ sudo dpkg -i rubackup-common_2.1.0-1_amd64.deb
```

6. Для отображения пользовательского интерфейса установите пакет **rubackup-common-gui**, как представлено на примере:

rubackup-common-gui_2.1.0-1_amd64.deb

7. Установите пакет **rubackup-client**, как представлено на примере:

```
$ sudo dpkg -i rubackup-client_2.1.0-1_amd64.deb
```

8. Установите пакет **rubackup-rbc**, как представлено на примере:

```
$ sudo dpkg -i rubackup-rbc_2.1.0-1_amd64.deb
```

9. Установите пакет **rubackup-server**, как представлено на примере:

```
$ sudo dpkg -i rubackup-server_2.1.0-1_amd64.deb
```

10. Установите пакет **rubackup-rbm**, как представлено на примере:

```
$ sudo dpkg -i rubackup-rbm_2.1.0-1_amd64.deb
```

Имя файла пакета может отличаться в зависимости от сборки.

По окончании установки пакета **rubackup-common** будет создана локальная группа **rubackup**, в которую следует добавить всех пользователей, которые будут работать с RuBackup.

При установке сервера RuBackup в ОС Astra Linux Special Edition 1.6 или 1.7 с активированным режимом защитной программной среды, после установки пакета **rubackup-server** выполните шаги:

1. Выполните команду:

```
$ sudo update-initramfs -u -k all
```

2. Перезагрузите операционную систему:

```
$ sudo reboot
```

Установка лицензии

Сервер RuBackup содержит в себе тестовую лицензию на выполнение резервного копирования общим объемом резервных копий 1 ТБ. При первом запуске сервер RuBackup получит лицензионный файл от глобального лицензионного сервера RuBackup. Если выход в Интернет с сервера невозможен, обратитесь к своему поставщику с указанием *hardware ID* для получения лицензионного файла.

Hardware ID можно узнать при помощи следующей команды:

```
# rubackup_server hwid
```

RuBackup hardware ID:

```
5253096d055899485ed2787eccfc57ae54ff04e76104856726c913732aa0c2  
b8
```

Файлы лицензии требуется переименовать как rubackup.lic и разместить в /opt/rubackup/etc/rubackup.lic, заменив тестовую лицензию на выданную.

Для того чтобы получить информацию о лицензии, выполните следующие действия:

1. Выполните команду:

```
# rubackup_server license
```

2. Перейдите в журнальный файл /opt/rubackup/log/RuBackup.log. В журнальном файле отобразится информация о лицензии:

```
Server: 'dima' description: 'Primary RuBackup server'  
Node type: Primary RuBackup server  
HWID: 3aa1a74636919cbc7ab0b0c012339e868171157f4adb47c4a3959b75a04ab2c3  
License start date: 2024-02-20, end date: 2025-02-20  
The license issued to the customer: localhost  
License type: 'Backend', Maximum capacity: 1 TB  
Used: 0.000000 TB
```

- Server, description - Имя узла сервера и описание
- Node type (primary, secondary, media) - Роль сервера (основной, резервный, медиасервер)
- HWID - Hardware ID
- License start date, end date - Дата начала и дата окончания действия лицензии
- The license issued to the customer - Пользователь, владеющий лицензией
- License type, Maximum capacity - Тип лицензии, максимальный объем данных, который позволяет резервировать лицензия данного типа
- Used - Объем зарезервированных данных

Парольная политика для локальных учетных записей RuBackup

Основные аспекты парольной политики для учетных записей RuBackup следующие:

- **Длина пароля и использование спецсимволов:** Пароль обязательно должен иметь длину не менее 12 символов и содержать минимум 1 цифру, 1 заглавную букву и минимум 1 специальный символ. Это повышает сложность пароля и делает его более устойчивым к подбору.
- **Рекомендуемая замена транспортного пароля (выданного администратором RuBackup пользователю) при первом входе в локальную учетную запись:** Рекомендуется обновление временного (транспортного) пароля после первого входа в учетную запись, Это гарантирует, что пользователь использует уникальный и надёжный пароль, который не был известен ранее.
- **Регулярная смена паролей:** Рекомендуется регулярно, не менее одного раза в 3 месяца, менять пароли, чтобы минимизировать риск компрометации.
- **Пароль не отображается на экране при вводе:** Это предотвращает визуальный перехват пароля другими людьми, которые могут находиться рядом.
- **Хранение паролей в зашифрованном виде или хранение хэша пароля:** Система использует алгоритм шифрования для преобразования пароля в зашифрованный текст, который затем сохраняется в базе данных. Для проверки пароля при входе в систему, система расшифровывает предоставленный пользователем пароль и сравнивает его с сохранённым зашифрованным паролем.
- **Уникальность паролей:** Не рекомендуется повторное использование паролей. Необходимо использовать уникальные пароли для разных учетных записей, чтобы уменьшить риск компрометации учетной записи.

Настройка СРК RuBackup в формате «Все в одном»

Внимание! Процедура настройки сервера также выполняет настройку клиента. После настройки сервера RuBackup не следует выполнять на нем настройку клиента, так как это повлечет замену серверных настроек клиентскими и сервер перестанет работать.

Первоначальная настройка сервера RuBackup осуществляется с помощью интерактивной утилиты **rb_init**. Утилита **rb_init** добавит необходимые сетевые сервисы в файл `/etc/services`. Выполните следующие действия:

1. Запустите `rb_init` от пользователя `root` и примите лицензионное соглашение. Для этого нажмите клавишу **y**

```
root@rubackup-primary:~# rb_init
RuBackup initialization utility
Copyright 2018-2022: LLC "RUBACKUP"
Исключительные права принадлежат ООО "РУБЭКАП"
Author is Andrey Kuznetsov
Version: 2.0 Build: 4
```

You MUST agree with the End User License Agreement (EULA) before installing RuBackup (y[es]/n[o]/r[ead]/q[uit])

2. Выберите сценарий конфигурирования основного (primary) сервера. Для этого нажмите клавишу **p**.

```
Do you want to configure RuBackup server (primary, secondary, media)
or client (p/s/m/c/q)?
```

```
Primary RuBackup server configuration...
```

3. Укажите адрес сервера СУБД PostgreSQL (по умолчанию при нажатии клавиши Enter в качестве адреса сервера используется localhost):

```
Enter hostname or IP address of PostgreSQL server
[ localhost ]:
```

4. Укажите пароль пользователя базы данных postgres:

```
Please enter password for "postgres" database user:
```

5. Введите имя суперпользователя RuBackup (по умолчанию при нажатии клавиши Enter в качестве имени суперпользователя используется rubackup):

Внимание! В имени суперпользователя запрещено использовать следующие символы: пробел, \, \$, #, ` , /, ?, *, ., ,, ;, :, %, ^, &, <, >

```
Enter name of RuBackup superuser [ rubackup ]:
```

6. Создайте суперпользователя базы данных и задайте пароль для суперпользователя базы данных rubackup:

```
Database user "rubackup" doesn't exist. Do you want to create
database user "rubackup" (y/n)? y
```

```
Please enter password for "rubackup" database user:
Repeat password:
```

```
Create PostgreSQL user rubackup      User rubackup was created.
```

```
Attention!!! -----> Use this login to authenticate to RBM as
RuBackup superuser.
```

7. Введите имя базы данных (по умолчанию при нажатии клавиши Enter в качестве имени базы данных используется rubackup):

Внимание! В имени базы данных запрещено использовать следующие символы:
пробел, \, \$, #, `, /, ?, *, ,, ,, ,, ;, :, %, ^, &, <, >

```
Enter RuBackup database name [ rubackup ]:
```

```
Database "rubackup" doesn't exist. Do you want to create database
"rubackup" on "localhost" host (y/n)? y
```

```
Creating new "rubackup" database...
```

```
Execute: /opt/rubackup/rc/init/bloom.sql: success
```

```
Execute: /opt/rubackup/rc/init/server.sql: success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_aerodisk-vm.sql:
success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_brest_template.sql:
success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_brest_vm.sql:
success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_btrfs.sql: success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_ceph_rbd.sql:
success
```

Execute:

```
/opt/rubackup/rc/modules/rb_module_communicate_pro.sql: success
```

Execute:

```
/opt/rubackup/rc/modules/rb_module_communicate_pro_mail.sql:  
success
```

Execute:

```
/opt/rubackup/rc/modules/rb_module_docker_container.sql: success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_docker_image.sql:  
success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_docker_volume.sql:  
success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_filesystem.sql:  
success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_filesystem_win.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_freeipa.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_iscsi_vmmanager.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_jatoba1.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_kvm.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_lvm.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_lxd_container.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_lxd_image.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_mariadb101.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_mssql.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_mssql_files.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_mysql.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_opennebula_template.sql:
success
  Execute: /opt/rubackup/rc/modules
/rb_module_zfs_volume.sql: success/rb_module_opennebula_vm.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_oracle_dp_db.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_oracle_dp_table.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_oracle_rman.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_ovirt.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_pg_dump_database.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_pg_dump_table.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_pg_superb.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_postgres_pro.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_postgresql.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_proxmox_container.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_proxmox_vm.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_redis.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_rescue_image.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_rustack
/opt/rubackup/rc/modules/rb_module_rvirt_vm.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_s3_cloud_bucket.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_sqlite.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_tarantool.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_vcenter.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_veil_vm.sql: success
```

Execute: /opt/rubackup/rc/modules/rb_module_vmware.sql: success
Execute: /opt/rubackup/rc/modules/rb_module_zfs.sql: success
Execute: /opt/rubackup/rc/modules/rb_module_zfs_volume.sql: success

Do you want to add a required file system to the 'Default' pool in the configuration? (y/n)?y

Enter path: /default_pool

Path "/default_pool" doesn't exist. Do you want to create it? (y/n)y

RuBackup client configuration: client-server mode

Hostname of primary server: primary-server

8. Установка «Все в одном» не предполагает наличие резервного сервера. Выберите n.

Will you use secondary server (y/n)? n

9. Далее будет выполняться настройка клиента RuBackup. Выберите сетевой интерфейс, посредством которого клиенту RuBackup разрешено взаимодействовать с системой резервного копирования:

Choose client net interface ID for use: **1**

Selected interface: ens18

10. Укажите, можно ли будет администратору системы СРК RuBackup восстанавливать копии, сделанные для данного клиента:

Do you allow centralized recovery (y/n)? **y**

11. Укажите, будет ли использоваться непрерывная удаленная репликация на этом клиенте:

Do you plan to use continuous remote replication to apply remote replicas on this client (y/n)?y

Enter local backup directory path [/tmp]:

12. Укажите директорию для временных операций с файлами резервных копий:

Enter local backup directory path [/tmp] : **/rubackup-tmp**

Would you like to create /rubackup-tmp (y/n)?y

Create RuBackup master key...

13. Укажите количество потоков для одновременной обработки задач резервного копирования на основном сервере (каждый поток имеет отдельное соединение со служебной базой данных СРК):

Set amount threads parallelizm for server [8]:

14. Укажите количество потоков для одновременной обработки задач резервного копирования на медиасервере (каждый поток имеет отдельное соединение со служебной базой данных СРК):

Set amount threads parallelizm media server [8]:

15. Укажите, хотите ли вы создать ключи электронно-цифровой подписи:

Will you use digital signature (y/n)?y

Create new secret key

Create new public key

16. Укажите, хотите ли вы установить верхний предел оперативной памяти, которая может использоваться при резервном копировании на клиенте (точность верхней границы объема памяти не гарантируется):

Do you want to set a soft memory threshold? (y/n)?y

Enter the allowed amount of memory for backup in GB (integer value):

17. Укажите объем оперативной памяти, который может использоваться при резервном копировании на клиенте в ГБ (целое число):

Enter the allowed amount of memory for backup in GB (integer value):

18. Укажите, хотите ли вы включить системный мониторинг для данного клиента:

Do you want to enable system monitoring of this client (y/n)?
y

19. Укажите, хотите ли вы включить аудит безопасности:

Do you want to enable RuBackup security audit ([y]es, [n]o, [q]uit)(y/n/q)? y

20. Укажите, какой тип аудита вы хотите включить:

Choose security audit type ([e]ssential only, [t]asks (additionally to essential), [q]uit)(e/t/q)?e

Внимание! По окончании работы утилиты `rb_init` будет сформирован главный конфигурационный файл `/opt/rubackup/etc/config.file`. В этом файле параметр `server-inet-interfaces` определяет сетевые интерфейсы, посредством которых серверу резервного копирования разрешено взаимодействовать с клиентами. В списке интерфейсов необходимо оставить только те, которые необходимы, и удалить все лишние интерфейсы, если они присутствуют (`vnet`, `virbr` и т.п.).

Настройка пользователей на сервере RuBackup

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup и Менеджера администратора RuBackup (RBM), должны:

- входить в группу `rubackup`,
- иметь правильно настроенные переменные среды.

Группа **`rubackup`** была создана в процессе установки пакета `rubackup-common`.

Чтобы настроить пользователя для возможности работы с RuBackup, выполните следующие действия:

1. Добавьте пользователя в группу `rubackup` при помощи команды:

```
$ sudo usermod -a -G rubackup пользователь
```

После этого введите команду:

```
$ sg rubackup
```

2. Настройте для *пользователя* следующие переменные среды. Для этого добавьте следующие строки в файл `/home/пользователь/.bashrc`:

```
PATH=$PATH:/opt/rubackup/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib
export PATH
export LD_LIBRARY_PATH
```

3. Перезагрузите переменные окружения:

```
$ . .bashrc
```

Запуск сервера RuBackup

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

1. Добавьте сервис клиента RuBackup в автозапуск при загрузке системы:

```
$ sudo systemctl enable rubackup_client.service
```

2. Добавьте сервис сервера RuBackup в автозапуск при загрузке системы:

```
$ sudo systemctl enable rubackup_server.service
```

3. Чтобы служба `systemd` перезагрузила настройки, введите команду:

```
$ 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 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 Thu 2023-02-09 13:24:23 MSK; 2 weeks 1 days ago
```

Main PID: 3038534 (rubackup_client)

Tasks: 3 (limit: 4610)

Memory: 16.5M

CGroup: /system.slice/rubackup_client.service

└─3038534 /opt/rubackup/bin/rubackup_client start

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Check additional RuBackup modules:

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Try to check module: 'File system' ...

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Execute OS command: /opt/rubackup/modules/rb_module_filesystem -t 2>&1

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Module version: 2.0

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: ... module 'File system' was checked successfully. Module supports archiving

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Try to check module: 'LVM logical volume' ...

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Execute OS command: /opt/rubackup/modules/rb_module_lvm -t 2>&1

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Module version: 2.0

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: LVM version: 2.03.07(2)

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: ... module 'LVM logical volume' was checked successfully

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 2022-04-18 16:05:18 MSK; 2min 31s ago
     Process: 27631 ExecStart=/opt/rubackup/bin/rubackup_server start
   (code=exited, status=0/SUCCESS)
    Main PID: 27635 (rubackup_server)
       Tasks: 29 (limit: 4628)
      Memory: 7.3M
     CGroup: /system.slice/rubackup_server.service
            └─27635 /opt/rubackup/bin/rubackup_server start
```

```
april 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup monitoring
connected to the database rubackup@l>
april 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup
media_server_TLQ_loop connected to the database>
april 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup Media Net Server
connected to the database ruba>
april 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup Media Net Server
will use 8 threads
april 18 16:05:19 ubuntu rubackup_server[27635]: Media server heartbeat:
Connected to RuBackup server: 12>
april 18 16:05:20 ubuntu rubackup_server[27635]: Media server ubuntu has
added to alive RuBackup server g>
april 18 16:05:24 ubuntu rubackup_server[27635]: RuBackup Media Server
internal tasks loop connected to t>
april 18 16:05:24 ubuntu rubackup_server[27635]: RuBackup media server cloud
loop connected to the databa>
april 18 16:06:00 ubuntu rubackup_server[27635]: RuBackup db_service
connected to the database rubackup@l>
april 18 16:06:00 ubuntu rubackup_server[27635]: RuBackup db_watcher
connected to the database rubackup@l>
lines 1-20/20 (END)
```

Настройка ограничения на количество открытых файловых дескрипторов на хосте с сервером RuBackup

Если число клиентов/медиа серверов в группировке растёт и/или на клиентах включена функция многопоточной передачи данных, то при увеличении количества входящих соединений сервер RuBackup может достичь предела выделенных лимитов на открытые файловые дескрипторы. Сетевые соединения тоже используют такие

дескрипторы. Ограничения устанавливает администратор хоста, на котором запущен сервер RuBackup.

Достижение этого ограничения приводит к ошибкам при выполнении бэкапа\ восстановления. Иногда сервер RuBackup может аварийно завершить работу. Признаком недостатка файловых дескрипторов является наличие следующего сообщения в журналах сервера или в системных журналах:

error:Too many open files

Для решения проблемы администратору хоста необходимо увеличить максимальное число (лимит) открытых дескрипторов, а затем перезапустить сервер. В зависимости от того, как на хосте запускается сервер, максимальное число (лимит) открытых дескрипторов меняется по разному.

Проверьте по формулам, описанным ниже, число нужных вам файловых дескрипторов и убедитесь, что на сервере их достаточно.

Важно! Чтобы рассчитать необходимое количество файловых дескрипторов, учтите следующее:

- В режиме простоя сервер использует около 100 файловых дескрипторов.
- Каждый подключённый клиент или медиасервер добавляет по два открытых файловых дескриптора на сервере.
- Выполнение любой задачи на стороне клиента при выключенном «сетевом параллелизме» (`network_parallelism`) требует двух дополнительных файловых дескриптора на сервере.
- При включённом «сетевом параллелизме» (`network_parallelism`) клиент открывает N соединений к серверу, где N — значение, заданное для параметра `network_parallelism`. В рамках каждого сетевого соединения, как правило, на стороне сервера требуется запросить информацию из базы данных, поэтому требуемое число открытых файловых дескрипторов будет $N*2$.

Общая формула для расчёта необходимого количества файловых дескрипторов:

- Если сетевой параллелизм выключен: $100 + MC * 2 + KL * 2 + KL * 2$
- Если сетевой параллелизм включён: $100 + MC * 2 + KL * 2 + KL * N$

Где:

- MC — число медиасерверов.
- KL — число клиентов.
- N — значение, заданное для параметра `network_parallelism`.

Пример расчета 1

Рассмотрим пример расчёта необходимого количества файловых дескрипторов для системы, состоящей из одного сервера RuBackup, двух медиасерверов и 50 клиентов. Предположим, что сетевой параллелизм отключён.

Необходимое количество файловых дескрипторов рассчитывается следующим образом:

- для сервера RuBackup потребуется 100 дескрипторов;
- для двух медиасерверов — 4 дескриптора;
- для 50 клиентов в простое — 100 дескрипторов;
- для всех 50 клиентов с задачами одновременно — ещё 100 дескрипторов.

Таким образом, общее количество необходимых файловых дескрипторов составляет 304.

Стандартное значение лимита в 1024 будет достаточным.

Пример расчета 2

Рассмотрим пример расчёта необходимого количества файловых дескрипторов для системы, состоящей из одного сервера RuBackup, двух медиасерверов и 50 клиентов. Предположим, что сетевой параллелизм включён со значением 40.

Необходимое количество файловых дескрипторов рассчитывается следующим образом:

- для сервера RuBackup потребуется 100 дескрипторов;
- для двух медиасерверов — 4 дескриптора;
- для 50 клиентов в простое — 100 дескрипторов;
- для всех 50 клиентов с задачами одновременно — ещё 2000 дескрипторов.

Таким образом, общее количество необходимых файловых дескрипторов составляет 2204.

Стандартное значение лимита в 1024 будет недостаточным для такой системы, поэтому рекомендуется увеличить лимит. Желательно установить лимит в 3000 файловых дескрипторов для запаса.

Сервер может быть запущен двумя способами:

1. Ручным запуском;
2. Запуском сервиса сервера RuBackup.

Для каждого из этих способов существует собственное отдельное ограничение на число открытых файловых дескрипторов. Если необходимо установить определенное число файловых дескрипторов, выполните по одному из перечисленных способов:

Ручной запуск

Если вы вводите команды:

rubackup_server start/stop

Значит установка лимита файловых дескрипторов будет проводится ручным запуском и необходимо ввести команды, описанные ниже.

По умолчанию установлено ограничение на число открытых файловых дескрипторов - 1024 файла,

2. Проверьте текущий лимит командой от имени пользователя root:

ulimit -n

3. Для временного изменения значения сессии для пользователя root выполните команду:

ulimit -n N

где N — желаемое значение.

В рамках этой же сессии можно запустить сервер, после завершения сессии изменения будут отменены.

3. Для постоянного изменения значения необходимо в файл `/etc/security/limits.conf` добавьте следующие строки:

```
root    hard  nofile    N
root    soft  nofile    N
```

где N — желаемое значение.

4. Закройте сессию и откройте новую, а также проверьте значение лимита командой: **ulimit -n**.

5. Из этой-же сессии перезапустите сервер.

Запуск сервиса сервера RuBackup

1. Если вы запускаете сервис сервера RuBackup командами:

systemctl start/stop rubackup_server

Значит установка лимита файловых дескрипторов будет проводится как сервис и необходимо ввести команды, описанные ниже.

По умолчанию значение ограничения количества открытых файлов задаётся в службе `systemd`, стандартное значение — 1024 файла.

2. Для изменения значения добавьте следующую строку в секцию `[Service]` файла `/etc/systemd/system/rubackup_server.service`:

LimitNOFILE=N

где N — желаемое значение.

3. Загрузите обновленный конфигурационный файл сервиса в службу systemd командой:

systemctl daemon-reload

4. Перезапустите сервис сервера RuBackup, командами:

systemctl stop rubackup_server

systemctl start rubackup_server

Запуск сервера в терминальном режиме

В том случае, если планируется тестирование RuBackup, рекомендуется запускать сервер RuBackup в терминальном режиме с помощью команды:

rubackup_server start

Остановить сервер RuBackup можно с помощью команды:

rubackup_server stop

Настройка хранилища резервных копий

Если в процессе настройки сервера при помощи утилиты `rb_init` не был назначен каталог для хранения резервных копий для пула **Default**, то после настройки сервера RuBackup в журнальном файле `/opt/rubackup/log/RuBackup.log` появятся записи о том, что в пуле `Default` нет ни одной файловой системы для хранения резервных копий:

```
Thu Sep 19 12:40:30 2019: Warning: Pool: Default has no any file system
```

Необходимо назначить для пула **Default** хотя бы один каталог для хранения резервных копий. Это можно сделать при помощи утилиты командной строки или Менеджера администратора RuBackup (RBM):

1. Настройка хранилища с помощью `rb_local_filesystem`

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup, должны входить в группу `rubackup`. Чтобы добавить пользователей в группу, внесите изменения в файл `/etc/group`.

Чтобы назначить локальный каталог в качестве хранилища резервных копий, следует выполнить команду:

```
$ rb_local_filesystems -a /rubackup1 -p 1
```

В этом примере в качестве хранилища добавляется каталог /rubackup1.

2. Настройка хранилища с помощью rb_local_filesystem

Внимание! Настройка хранилища с помощью RBM производится, если хранилища не настроены утилитой rb_init в процессе первоначальной настройки.

Порядок настройки хранилища изложен в документе «Руководство системного администратора RuBackup».

Развернутая установка

Для использования RuBackup в продуктивных окружениях среднего и промышленного масштаба, а также для проведения нагрузочных испытаний, рекомендуется разворачивать компоненты RuBackup, включая служебную базу данных RuBackup, на отдельных машинах с рекомендуемой конфигурацией. Это позволит достичь максимальных показателей производительности и выполнить резервное копирование, восстановление и удаленную репликацию данных в кратчайшие сроки.

Установка основного сервера

Подготовка к установке основного сервера

Необходимые пакеты

Перед установкой сервера RuBackup необходимо, чтобы в системе были установлены следующие пакеты.

| Название пакета | Описание |
|--|--|
| mailutils (или bsd-mailx) | Утилиты электронной почты |
| libcurl4 (возможно libcurl3 в зависимости от ОС) | Клиентская библиотека передачи данных по URL |

| | |
|---------------------------|--|
| libqt5sql5-psql | Необходим для ОС Astra Linux |
| libsasl2-2, libldap-2.4-2 | Необходимы для операционных систем Ubuntu 18.04, Ubuntu 20.04, Astra Linux Special Edition 1.6, Astra Linux Special Edition 1.7, Debian 10 |
| openldap, cyrus-sasl | Необходимы для операционных систем РЕД ОС 7.3, CentOS 8, CentOS 7 |
| libldap2, libsasl2-3 | Необходимы для операционной системы ALT Linux 10 |

Внимание! Названия пакетов могут отличаться в зависимости от дистрибутива Linux.

Чтобы система уведомлений RuBackup работала корректно, необходимо настроить отправку электронной почты с сервера RuBackup. Для отправки электронной почты сервер RuBackup использует утилиту /usr/bin/mail.

При использовании ленточной библиотеки с сервером резервного копирования, настройку см. в руководстве «Работа с ленточной библиотекой».

Пакеты для ОС без графической оболочки

Если вы устанавливаете сервер RuBackup на ОС без графической оболочки, то, чтобы использовать Менеджер администратора RuBackup (RBM), установите следующие пакеты:

```
$ sudo apt install libgl1-mesa-dev  
$ sudo apt install libxkbcommon-x11-0  
$ sudo apt install libfontconfig1
```

Инсталляция основного сервера RuBackup

Для инсталляции основного сервера RuBackup следует выполнить следующие действия:

1. Авторизуйтесь под пользователем root:

```
$ sudo -i
```

2. Настройте следующие переменные среды для пользователя root. Для этого добавьте следующие строки в файл /root/.bashrc:

```
PATH=$PATH:/opt/rubackup/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib
export PATH
export LD_LIBRARY_PATH
```

Эти переменные также можно определить в файле `/etc/environment`.

3. Перейдите в каталог **/root/**, для этого выполните:

```
cd /root
```

4. Перезагрузите переменные окружения:

```
# . .bashrc
```

5. Установите пакет **rubackup-common**. Пример:

```
$ sudo dpkg -i rubackup-common_2.0.0-1_amd64.deb
```

6. Установите пакет **rubackup-client**. Пример:

```
$ sudo dpkg -i rubackup-client_2.0.0-1_amd64.deb
```

7. Установите пакет **rubackup-server**. Пример:

```
$ sudo dpkg -i rubackup-server_2.0.0-1_amd64.deb
```

8. Установите пакет **rubackup-rbm**. Пример:

```
$ sudo dpkg -i rubackup-rbm_2.0.0-1_amd64.deb
```

Имя файла пакета может отличаться в зависимости от сборки.

В процессе установки пакета `rubackup-common` будет создана локальная группа `rubackup`, в которую следует добавить всех пользователей, которые будут работать с RuBackup.

При установке сервера RuBackup в ОС Astra Linux Special Edition 1.6 или 1.7 с активированным режимом защитной программной среды, после установки пакета **rubackup-server** выполните шаги:

1. Выполните команду:

```
$ sudo update-initramfs -u -k all
```

2. Перезагрузите операционную систему:

```
$ sudo reboot
```

Установка лицензии

Внимание! Для работы основного сервера требуется отдельная лицензия.

Сервер RuBackup содержит в себе тестовую лицензию на выполнение резервного копирования общим объемом резервных копий 1 ТБ. При первом запуске сервер RuBackup получит лицензионный файл от глобального лицензионного сервера RuBackup. Если выход в Интернет с сервера невозможен, обратитесь к своему поставщику с указанием *hardware ID* для получения лицензионного файла.

Hardware ID можно узнать при помощи следующей команды:

```
# rubackup_server hwid
```

```
RuBackup hardware ID:
```

```
5253096d055899485ed2787eccfc57ae54ff04e76104856726c913732aa0  
c2b8
```

Файлы лицензии требуется переименовать как `rubackup.lic` и разместить в `/opt/rubackup/etc/rubackup.lic`, заменив тестовую лицензию на выданную.

Для того чтобы получить информацию о лицензии, выполните следующие действия:

1. Выполните команду:

```
# rubackup_server license
```

2. Перейдите в журнальный файл /opt/rubackup/log/RuBackup.log. В журнальном файле отобразится информация о лицензии:

```
Server: 'dima' description: 'Primary RuBackup server'  
Node type: Primary RuBackup server  
HWID: 3aa1a74636919cbc7ab0b0c012339e868171157f4adb47c4a3959b75a04ab2c3  
License start date: 2024-02-20, end date: 2025-02-20  
The license issued to the customer: localhost  
License type: 'Backend', Maximum capacity: 1 TB  
Used: 0.000000 TB
```

- Server, description - Имя узла сервера и описание
- Node type (primary, secondary, media) - Роль сервера (основной, резервный, медиасервер)
- HWID - Hardware ID
- License start date, end date - Дата начала и дата окончания действия лицензии
- The license issued to the customer - Пользователь, владеющий лицензией
- License type, Maximum capacity - Тип лицензии, максимальный объем данных, который позволяет резервировать лицензия данного типа
- Used - Объем зарезервированных данных

Настройка основного сервера

Внимание! Процедура настройки основного сервера также выполняет настройку клиента. После настройки сервера RuBackup не следует выполнять на нем настройку клиента, так как это повлечет замену серверных настроек клиентскими и сервер перестанет работать.

Первоначальная настройка сервера RuBackup осуществляется с помощью интерактивной утилиты **rb_init**. Утилита **rb_init** добавит необходимые сетевые сервисы в файл /etc/services. Выполните следующие действия:

1. Запустите **rb_init** (от пользователя root).

```
root@rubackup-primary:~# rb_init  
RuBackup initialization utility  
Copyright 2018-2022: LLC "RUBACKUP"  
Исключительные права принадлежат ООО "РУБЭКАП"  
Author is Andrey Kuznetsov  
Version: 2.0 Build: 4
```



You MUST agree with the End User License Agreement (EULA) before installing RuBackup (y[es]/n[o]/r[ead]/q[uit])

2. Примите лицензионное соглашение. Для этого нажмите клавишу **y**.

You MUST agree with the End User License Agreement (EULA) before installing RuBackup (y[es]/n[o]/r[ead]/q[uit])

Do you want to configure RuBackup server (primary, secondary, media) or client (p/s/m/c/q)?

3. Выберите сценарий конфигурирования основного (primary) сервера. Для этого нажмите клавишу **p**.

Do you want to configure RuBackup server (primary, secondary, media) or client (p/s/m/c/q)?

Primary RuBackup server configuration...

4. Укажите адрес сервера СУБД PostgreSQL (по умолчанию при нажатии клавиши Enter в качестве адреса сервера используется localhost):

Enter hostname or IP address of PostgreSQL server
[localhost]:

5. Укажите пароль пользователя базы данных postgres:

Please enter password for "postgres" database user:

6. Введите имя суперпользователя RuBackup (по умолчанию при нажатии Enter используется имя суперпользователя - rubackup):

Внимание! В имени суперпользователя запрещено использовать следующие символы: пробел, \, \$, #, ` , /, ?, *, ., ,, ;, :, %, ^, &, <, >

Enter name of RuBackup superuser [rubackup]:

7. Создайте суперпользователя базы данных и задайте пароль для суперпользователя базы данных rubackup:

```
Database user "rubackup" doesn't exist. Do you want to create
database user "rubackup" (y/n)? y
```

```
Please enter password for "rubackup" database user:
Repeat password:
```

```
Create PostgreSQL user rubackup      User rubackup was created.
```

```
Attention!!! -----> Use this login to authenticate to RBM as
RuBackup superuser.
```

8. Введите имя базы данных (по умолчанию при нажатии клавиши Enter в качестве имени базы данных используется rubackup):

Внимание! В имени базы данных запрещено использовать следующие символы:
пробел, \, \$, #, `, /, ?, *, ,, ,, ,, ;, :, %, ^, &, <, >

```
Enter RuBackup database name [ rubackup ]:
```

```
Database "rubackup" doesn't exist. Do you want to create database
"rubackup" on "localhost" host (y/n)? y
```

```
Creating new "rubackup" database...
```

```
Execute: /opt/rubackup/rc/init/bloom.sql: success
```

```
Execute: /opt/rubackup/rc/init/server.sql: success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_aerodisk-vm.sql:
success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_brest_template.sql:
success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_brest_vm.sql:
success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_btrfs.sql: success
```

```
Execute: /opt/rubackup/rc/modules/rb_module_ceph_rbd.sql:
success
```


Execute:

/opt/rubackup/rc/modules/rb_module_communicate_pro.sql: success

Execute:

/opt/rubackup/rc/modules/rb_module_communicate_pro_mail.sql:
success

Execute:

/opt/rubackup/rc/modules/rb_module_docker_container.sql: success

Execute: /opt/rubackup/rc/modules/rb_module_docker_image.sql:
success

Execute: /opt/rubackup/rc/modules/rb_module_docker_volume.sql:
success

Execute: /opt/rubackup/rc/modules/rb_module_filesystem.sql:
success

```
Execute: /opt/rubackup/rc/modules/rb_module_filesystem_win.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_freeipa.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_iscsi_vmmanager.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_jatoba1.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_kvm.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_lvm.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_lxd_container.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_lxd_image.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_mariadb101.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_mssql.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_mssql_files.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_mysql.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_opennebula_template.sql:
success
  Execute: /opt/rubackup/rc/modules
/rb_module_zfs_volume.sql: success/rb_module_opennebula_vm.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_oracle_dp_db.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_oracle_dp_table.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_oracle_rman.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_ovirt.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_pg_dump_database.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_pg_dump_table.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_pg_superb.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_postgres_pro.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_postgresql.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_proxmox_container.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_proxmox_vm.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_redis.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_rescue_image.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_rustack
/opt/rubackup/rc/modules/rb_module_rvirt_vm.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_s3_cloud_bucket.sql:
success
  Execute: /opt/rubackup/rc/modules/rb_module_sqlite.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_tarantool.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_vcenter.sql: success
  Execute: /opt/rubackup/rc/modules/rb_module_veil_vm.sql: success
```

Execute: /opt/rubackup/rc/modules/rb_module_vmware.sql: success
Execute: /opt/rubackup/rc/modules/rb_module_zfs.sql: success
Execute: /opt/rubackup/rc/modules/rb_module_zfs_volume.sql: success

9. Добавьте локальное файловое хранилище для дефолтного пула:
Do you want to add a required file system to the 'Default' pool in the configuration? (y/n)?y

10. Введите путь к директории, которая будет ассоциирована с дефолтным пулом:

Enter path: /default_pool

Path "/default_pool" doesn't exist. Do you want to create it? (y/n)y

RuBackup client configuration: client-server mode

Hostname of primary server: primary-server

11. Если в конфигурации подразумевается резервный (secondary) сервер, то выберите эту возможность:

Will you use secondary server (y/n)?

12. Укажите адрес резервного сервера:

Hostname of secondary server: **rb-secondary**

13. Далее будет выполняться настройка клиента RuBackup. Выберите сетевой интерфейс, посредством которого клиенту RuBackup разрешено взаимодействовать с системой резервного копирования:

Choose client net interface ID for use: **1**

Selected interface: ens18

14. Укажите, можно ли будет администратору системы CPK RuBackup восстанавливать копии, сделанные для данного клиента:

Do you allow centralized recovery (y/n)? **y**

15. Укажите, будет ли использоваться непрерывная удаленная репликация на этом клиенте:

Do you plan to use continuous remote replication to apply remote replicas on this client (y/n)?y

Enter local backup directory path [/tmp]:

16. Укажите директорию для временных операций с файлами резервных копий:

```
Enter local backup directory path [/tmp] : /rubackup-tmp
Would you like to create /rubackup-tmp (y/n)?y
Create RuBackup master key...
```

17. Укажите количество потоков для одновременной обработки задач резервного копирования на основном сервере (каждый поток имеет отдельное соединение со служебной базой данных СРК):

```
Set amount threads parallelizm for server [ 8 ]:
```

18. Укажите количество потоков для одновременной обработки задач резервного копирования на медиасervere (каждый поток имеет отдельное соединение со служебной базой данных СРК):

```
Set amount threads parallelizm media server [ 8 ]:
```

19. Укажите, хотите ли вы создать ключи электронно-цифровой подписи:

```
Will you use digital signature (y/n)?y
Create new secret key
Create new public key
```

20. Укажите, хотите ли вы установить верхний предел оперативной памяти, которая может использоваться при резервном копировании на клиенте (точность верхней границы объема памяти не гарантируется):

```
Do you want to set a soft memory threshold? (y/n)?y
```

```
Enter the allowed amount of memory for backup in GB (integer
value):
```

21. Укажите объем оперативной памяти, который может использоваться при резервном копировании на клиенте в ГБ (целое число):

```
Enter the allowed amount of memory for backup in GB (integer
value):
```

22. Укажите, хотите ли вы включить системный мониторинг для данного клиента:

```
Do you want to enable system monitoring of this client (y/n)?
```

23. Укажите, хотите ли вы включить аудит безопасности:

Do you want to enable RuBackup security audit ([y]es, [n]o, [q]uit)(y/n/q)? y

24. Укажите, какой тип аудита вы хотите включить:

Choose security audit type ([e]ssential only, [t]asks (additionally to essential), [q]uit)
(e/t/q)?e

Внимание! По окончании работы утилиты `rb_init` будет сформирован главный конфигурационный файл `/opt/rubackup/etc/config.file`. В этом файле параметр `server-inet-interfaces` определяет сетевые интерфейсы, посредством которых серверу резервного копирования разрешено взаимодействовать с клиентами. В списке интерфейсов необходимо оставить только те, которые необходимы, и удалить все лишние интерфейсы, если они присутствуют (`vnet`, `virbr` и т.п.).

Настройка пользователей на сервере RuBackup

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup и оконного Менеджера Администратора (RBM), должны:

- иметь правильно настроенные переменные среды,
- входить в группу `rubackup`.

Группа **`rubackup`** была создана в процессе установки пакета `rubackup-common`.

Чтобы настроить пользователя для возможности работы с RuBackup, выполните следующие действия:

1. Добавьте пользователя в группу `rubackup` при помощи команды:

```
$ sudo usermod -a -G rubackup пользователь
```

После этого введите команду:

```
$ sg rubackup
```

2. Настройте для *пользователя* следующие переменные среды. Для этого добавьте следующие строки в файл `/home/пользователь/.bashrc`:

```
PATH=$PATH:/opt/rubackup/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib
export PATH
export LD_LIBRARY_PATH
```

3. Перезагрузите переменные окружения:

```
$ . .bashrc
```

Запуск основного сервера RuBackup

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

1. Добавьте сервис клиента RuBackup в автозапуск при загрузке системы:

```
$ sudo systemctl enable rubackup_client.service
```

2. Добавьте сервис сервера RuBackup в автозапуск при загрузке системы:

```
$ sudo systemctl enable rubackup_server.service
```

3. Чтобы служба `systemd` перезагрузила настройки, введите команду:

```
$ 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 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 Thu 2023-02-09 13:24:23 MSK; 2 weeks 1 days ago
```

Main PID: 3038534 (rubackup_client)

Tasks: 3 (limit: 4610)

Memory: 16.5M

CGroup: /system.slice/rubackup_client.service

└─3038534 /opt/rubackup/bin/rubackup_client start

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Check additional RuBackup modules:

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Try to check module: 'File system' ...

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Execute OS command: /opt/rubackup/modules/rb_module_filesystem -t 2>&1

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Module version: 2.0

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: ... module 'File system' was checked successfully. Module supports archiving

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Try to check module: 'LVM logical volume' ...

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Execute OS command: /opt/rubackup/modules/rb_module_lvm -t 2>&1

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Module version: 2.0

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: LVM version: 2.03.07(2)

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: ... module 'LVM logical volume' was checked successfully

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 2022-04-18 16:05:18 MSK; 2min 31s ago

Process: 27631 ExecStart=/opt/rubackup/bin/rubackup_server start (code=exited, status=0/SUCCESS)

Main PID: 27635 (rubackup_server)

Tasks: 29 (limit: 4628)

Memory: 7.3M

CGroup: /system.slice/rubackup_server.service

└─27635 /opt/rubackup/bin/rubackup_server start

apn 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup monitoring connected to the database rubackup@l>

apn 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup media_server_TLQ_loop connected to the database>

apn 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup Media Net Server connected to the database ruba>

apn 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup Media Net Server will use 8 threads

apn 18 16:05:19 ubuntu rubackup_server[27635]: Media server heartbeat: Connected to RuBackup server: 12>

apn 18 16:05:20 ubuntu rubackup_server[27635]: Media server ubuntu has added to alive RuBackup server g>

apn 18 16:05:24 ubuntu rubackup_server[27635]: RuBackup Media Server internal tasks loop connected to t>

apn 18 16:05:24 ubuntu rubackup_server[27635]: RuBackup media server cloud loop connected to the databa>

apn 18 16:06:00 ubuntu rubackup_server[27635]: RuBackup db_service connected to the database rubackup@l>

apn 18 16:06:00 ubuntu rubackup_server[27635]: RuBackup db_watcher connected to the database rubackup@l>

lines 1-20/20 (END)

Внимание! Если у вас возникает проблема запуска сервиса сервера 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:


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

2. Перезагрузите systemctl:
\$ sudo systemctl daemon-reload

Запуск основного сервера в терминальном режиме

В том случае, если планируется тестирование RuBackup, рекомендуется запускать основной сервер RuBackup в терминальном режиме с помощью команды:

```
# rubackup_server start
```

Остановить сервер RuBackup можно с помощью команды:

```
# rubackup_server stop
```

Настройка хранилища резервных копий

Если в процессе настройки сервера при помощи утилиты `rb_init` не был назначен каталог для хранения резервных копий для пула **Default**, то после настройки основного сервера RuBackup в журнальном файле `/opt/rubackup/log/RuBackup.log` появятся записи о том, что в пуле `Default` нет ни одной файловой системы для хранения резервных копий:

```
Thu Sep 19 12:40:30 2019: Warning: Pool: Default has no any file system
```

Необходимо назначить для пула **Default** хотя бы один каталог для хранения резервных копий. Это можно сделать при помощи утилиты командной строки или Менеджера администратора RuBackup (RBM):

1. Настройка хранилища с помощью `rb_local_filesystem`

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup, должны входить в группу `rubackup`. Чтобы добавить пользователей в группу, внесите изменения в файл `/etc/group`.

Чтобы назначить локальный каталог в качестве хранилища резервных копий, следует выполнить команду:

```
$ rb_local_filesystems -a /rubackup1 -p 1
```

В этом примере в качестве хранилища добавляется каталог `/rubackup1`.

2. Настройка хранилища с помощью `rb_local_filesystem`

Внимание! Настройка хранилища с помощью RBM производится, если хранилища не настроены утилитой `rb_init` в процессе первоначальной настройки.

Порядок настройки хранилища изложен в документе «Руководство системного администратора RuBackup».

Установка резервного сервера

Подготовка к установке резервного сервера

Необходимые пакеты

Перед установкой сервера RuBackup необходимо, чтобы в системе были установлены следующие пакеты.

| Название пакета | Описание |
|--|--|
| mailutils (или bsd-mailx) | Утилиты электронной почты |
| libcurl4 (возможно libcurl3 в зависимости от ОС) | Клиентская библиотека передачи данных по URL |
| libqt5sql5-psql | Необходим для ОС Astra Linux |
| libsasl2-2, libldap-2.4-2 | Необходимы для операционных систем Ubuntu 18.04, Ubuntu 20.04, Astra Linux Special Edition 1.6, Astra Linux Special Edition 1.7, Debian 10 |
| openldap, cyrus-sasl | Необходимы для операционных систем РЕД ОС 7.3, CentOS 8, CentOS 7 |
| libldap2, libsasl2-3 | Необходимы для операционной системы ALT Linux 10 |

Внимание! Названия пакетов могут отличаться в зависимости от дистрибутива Linux.

Чтобы система уведомлений RuBackup работала корректно, необходимо настроить отправку электронной почты с сервера RuBackup. Для отправки электронной почты сервер RuBackup использует утилиту `/usr/bin/mail`.

При использовании ленточной библиотеки с сервером резервного копирования, настройку см. в руководстве «Работа с ленточной библиотекой».

Пакеты для ОС без графической оболочки

Если вы устанавливаете сервер RuBackup на ОС без графической оболочки, то, чтобы использовать Менеджер администратора RuBackup (RBM), установите следующие пакеты:

```
$ sudo apt install libgl1-mesa-dev  
$ sudo apt install libxkbcommon-x11-0  
$ sudo apt install libfontconfig1
```

Инсталляция резервного сервера

Для инсталляции резервного сервера RuBackup следует выполнить следующие действия:

1. Авторизуйтесь под пользователем root:

```
$ sudo -i
```

2. Настройте следующие переменные среды для пользователя root. Для этого добавьте следующие строки в файл /root/.bashrc:

```
PATH=$PATH:/opt/rubackup/bin  
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib  
export PATH  
export LD_LIBRARY_PATH
```

Эти переменные также можно определить в файле /etc/environment.

3. Перейдите в каталог **/root/**, для этого выполните:

```
cd /root
```

4. Перезагрузите переменные окружения:

```
$ . .bashrc
```

5. Установите пакет **rubackup-common**. Пример:

```
$ sudo dpkg -i rubackup-common_2.0.0-1_amd64.deb
```

6. Установите пакет **rubackup-client**. Пример:

```
$ sudo dpkg -i rubackup-client_2.0.0-1_amd64.deb
```

7. Установите пакет **rubackup-server**. Пример:

```
$ sudo dpkg -i rubackup-server_2.0.0-1_amd64.deb
```

8. Установите пакет **rubackup-rbm**. Пример:

```
$ sudo dpkg -i rubackup-rbm_2.0.0-1_amd64.deb
```

Имя файла пакета может отличаться в зависимости от сборки.

При установке сервера RuBackup в ОС Astra Linux Special Edition 1.6 или 1.7 с активированным режимом защитной программной среды, после установки пакета **rubackup-server** выполните следующие шаги:

1. Выполните команду:

```
$ sudo update-initramfs -u -k all
```

2. Перезагрузите операционную систему:

```
$ sudo reboot
```

Установка лицензии

Внимание! Для работы резервного сервера требуется отдельная лицензия.

Сервер RuBackup содержит в себе тестовую лицензию на выполнение резервного копирования общим объемом резервных копий 1 ТБ. При первом запуске сервер RuBackup получит лицензионный файл от глобального лицензионного сервера RuBackup. Если выход в Интернет с сервера невозможен, обратитесь к своему поставщику с указанием *hardware ID* для получения лицензионного файла.

Hardware ID можно узнать при помощи следующей команды:

```
# rubackup_server hwid
```

```
RuBackup hardware ID:
```

```
5253096d055899485ed2787eccfc57ae54ff04e76104856726c913732aa0  
c2b8
```

Файлы лицензии требуется переименовать как `rubackup.lic` и разместить в `/opt/rubackup/etc/rubackup.lic`, заменив тестовую лицензию на выданную.

Для того чтобы получить информацию о лицензии, выполните следующие действия:

1. Выполните команду:

```
# rubackup_server license
```

2. Перейдите в журнальный файл /opt/rubackup/log/RuBackup.log. В журнальном файле отобразится информация о лицензии:

```
Server: 'dima' description: 'Primary RuBackup server'  
Node type: Primary RuBackup server  
HWID: 3aa1a74636919cbc7ab0b0c012339e868171157f4adb47c4a3959b75a04ab2c3  
License start date: 2024-02-20, end date: 2025-02-20  
The license issued to the customer: localhost  
License type: 'Backend', Maximum capacity: 1 TB  
Used: 0.000000 TB
```

- Server, description - Имя узла сервера и описание
- Node type (primary, secondary, media) - Роль сервера (основной, резервный, медиасервер)
- HWID - Hardware ID
- License start date, end date - Дата начала и дата окончания действия лицензии
- The license issued to the customer - Пользователь, владеющий лицензией
- License type, Maximum capacity - Тип лицензии, максимальный объем данных, который позволяет резервировать лицензия данного типа
- Used - Объем зарезервированных данных

Настройка резервного сервера

Резервный сервер не является обязательным компонентом СРК RuBackup и используется для повышения отказоустойчивости системы.

Необходимо чтобы основной и резервный сервер могли верифицировать друг друга по hostname. Для этого необходимо произвести соответствующие настройки в /etc/hosts на обоих узлах.

Необходимо добавить IP-адрес резервного сервера в pg_hba.conf СУБД PostgreSQL, содержащую БД rubackup.

Первоначальная настройка резервного сервера RuBackup осуществляется с помощью интерактивной утилиты **rb_init**. Выполните следующие действия:

1. Запустите rb_init от имени суперпользователя (с правами root).

```
root@ secondary-server~# rb_init  
RuBackup initialization utility
```

Copyright 2018-2022: LLC "RUBACKUP"

Исключительные права принадлежат ООО "РУБЭКАП"

Author is Andrey Kuznetsov

Version: 2.0 Build: 4

You MUST agree with the End User License Agreement (EULA)
before installing RuBackup (y[es]/n[o]/r[ead]/q[uit])

2. Примите лицензионное соглашение. Для этого нажмите клавишу **y**.

3. Выберите сценарий конфигурирования резервного (secondary) сервера. Для этого нажмите клавишу **s**.

```
Do you want to configure RuBackup server (primary, secondary,  
media) or client (p/s/m/c/q)?s
```

```
Secondary server configuration...
```

```
Enter hostname or IP address of PostgreSQL server
```

```
[ localhost ]:
```

4. Введите адрес сервера, на котором располагается база данных RuBackup, и пароль (по умолчанию при нажатии клавиши Enter в качестве адреса сервера используется localhost):

```
Enter hostname or IP address of PostgreSQL server [ localhost ]: primary-  
server
```

```
Enter name of RuBackup superuser [ rubakup ]:
```

5. Укажите имя суперпользователя RuBackup (по умолчанию при нажатии клавиши Enter в качестве имени суперпользователя используется rubakup):

Внимание! В имени суперпользователя запрещено использовать следующие символы:
пробел, \, \$, #, `, /, ?, *, ., ,, ;, :, %, ^, &, <, >

```
Enter name of RuBackup superuser [ rubakup ]: rubakup
```

```
Please enter password for "rubakup" database user:
```

6. Введите пароль для суперпользователя Rubakup:

Please enter password for "rubackup" database user:
Enter RuBackup database name [rubackup]:

7. Введите имя базы данных (по умолчанию при нажатии клавиши Enter в качестве имени базы данных используется rubackup):

Внимание! В имени базы данных запрещено использовать следующие символы:
пробел, \, \$, #, ` , /, ?, *, ., ,, ;, :, %, ^, &, <, >

Enter RuBackup database name [rubackup]:
RuBackup client configuration: client-server mode

Hostname of primary server: primary-server
Hostname of secondary server: secondary-server

Choose client net interface ID for use:

8. Выберите сетевой интерфейс, посредством которого клиенту RuBackup разрешено взаимодействовать с системой резервного копирования:

Choose client net interface ID for use:
Possible interfaces for RuBackup client communication:

lo [0]
ens18 [1]

Choose client net interface ID for use: **1**

Selected interface: ens18

Do you allow centralized recovery (y/n)?

9. Укажите можно ли будет администратору системы СРК RuBackup восстанавливать копии, сделанные для данного клиента:

Do you allow centralized recovery (y/n)?y

Do you plan to use continuous remote replication to apply remote replicas on this client (y/n)?

10. Укажите, будет ли использоваться непрерывная удаленная репликация на этом клиенте:

Do you plan to use continuous remote replication to apply remote replicas on this client (y/n)?y

Enter local backup directory path [/tmp]

11. Укажите директорию для временных операций с файлами резервных копий:

Enter local backup directory path [/tmp] : **/rubackup-tmp**

Would you like to create /rubackup-tmp (y/n)?y

Create RuBackup master key...

12. Создайте мастер-ключ для защитного преобразования резервных копий (необходимо ввести не менее 10 символов в поле Passphrase):

Passphrase:

Will you use digital signature (y/n)?

13. Укажите, хотите ли вы создать ключи электронно-цифровой подписи:

Will you use digital signature (y/n)?y

Create new secret key

Create new public key

14. Укажите, хотите ли вы установить верхний предел оперативной памяти, которая может использоваться при резервном копировании на клиенте (точность верхней границы объема памяти не гарантируется):

Do you want to set a soft memory threshold? (y/n)?y

Enter the allowed amount of memory for backup in GB (integer value):

15. Укажите объем оперативной памяти, который может использоваться при резервном копировании на клиенте в ГБ (целое число):

Enter the allowed amount of memory for backup in GB (integer value):

16. Укажите, хотите ли вы включить системный мониторинг для данного клиента:

Do you want to enable system monitoring of this client (y/n)?

По окончании работы `rb_init` запустите клиентский и серверный сервисы резервного копирования. Следуйте инструкции из раздела «Настройка пользователей на резервном сервере RuBackup».

По завершении настройки резервного сервера необходимо:

1. Авторизовать резервный сервер при первом запуске в системе резервного копирования в RBM (см. подробности в «Руководстве системного администратора RuBackup»).

Внимание! После запуска резервного сервера необходимо соблюсти порядок авторизации. Сначала нужно авторизовать в системе клиент и только потом резервный сервер. В противном случае будет добавлено два клиента, что приведет к ошибкам.

2. Назначить резервному серверу хотя бы один пул типа «Файловая система» для хранения резервных копий и каталог для хранения резервных копий.

Эти задачи можно выполнить в оконном Менеджере Администратора RBM (см. «Руководство системного администратора RuBackup»).

Настройка пользователей на резервном сервере RuBackup

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup и оконного Менеджера Администратора (RBM), должны:

- иметь правильно настроенные переменные среды,
- входить в группу `rubackup`.

Группа **`rubackup`** была создана в процессе установки пакета `rubackup-common`.

Чтобы настроить пользователя для возможности работы с RuBackup, выполните следующие действия:

1. Добавьте пользователя в группу rubackup при помощи команды:

```
$ sudo usermod -a -G rubackup пользователь
```

После этого введите команду:

```
$ sg rubackup
```

2. Настройте для *пользователя* следующие переменные среды. Для этого добавьте следующие строки в файл `/home/пользователь/.bashrc`:

```
PATH=$PATH:/opt/rubackup/bin  
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib  
export PATH  
export LD_LIBRARY_PATH
```

3. Перезагрузите переменные окружения:

```
$ . .bashrc
```

Запуск резервного сервера RuBackup

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

1. Добавьте сервис клиента RuBackup в автозапуск при загрузке системы:

```
$ sudo systemctl enable rubackup_client.service
```

2. Добавьте сервис сервера RuBackup в автозапуск при загрузке системы:

```
$ sudo systemctl enable rubackup_server.service
```

3. Чтобы служба systemd перезагрузила настройки, введите команду:

```
$ 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 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 Thu 2023-02-09 13:24:23 MSK; 2 weeks 1 days ago

Main PID: 3038534 (rubackup_client)

Tasks: 3 (limit: 4610)

Memory: 16.5M

CGroup: /system.slice/rubackup_client.service

└─3038534 /opt/rubackup/bin/rubackup_client start

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Check additional RuBackup modules:

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Try to check module: 'File system' ...

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Execute OS command: /opt/rubackup/modules/rb_module_filesystem -t 2>&1

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Module version: 2.0

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: ... module 'File system' was checked successfully. Module supports archiving

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Try to check module: 'LVM logical volume' ...

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Execute OS command: /opt/rubackup/modules/rb_module_lvm -t 2>&1

Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Module version: 2.0

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: LVM
version: 2.03.07(2)
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: ...
module 'LVM logical volume' was checked successfully
```

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 2022-04-18 16:05:18 MSK; 2min 31s ago
```

```
Process: 27631 ExecStart=/opt/rubackup/bin/rubackup_server start (code=exited,
status=0/SUCCESS)
```

```
Main PID: 27635 (rubackup_server)
```

```
Tasks: 29 (limit: 4628)
```

```
Memory: 7.3M
```

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

```
└─27635 /opt/rubackup/bin/rubackup_server start
```

```
anp 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup monitoring
connected to the database rubackup@|>
```

```
anp 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup
media_server_TLQ_loop connected to the database>
```

```
anp 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup Media Net Server
connected to the database ruba>
```

```
anp 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup Media Net Server
will use 8 threads
```

```
anp 18 16:05:19 ubuntu rubackup_server[27635]: Media server heartbeat:
Connected to RuBackup server: 12>
```

```
anp 18 16:05:20 ubuntu rubackup_server[27635]: Media server ubuntu has
added to alive RuBackup server g>
```

```
anp 18 16:05:24 ubuntu rubackup_server[27635]: RuBackup Media Server
internal tasks loop connected to t>
```

```
anp 18 16:05:24 ubuntu rubackup_server[27635]: RuBackup media server cloud
loop connected to the databa>
```

```
anp 18 16:06:00 ubuntu rubackup_server[27635]: RuBackup db_service
connected to the database rubackup@|>
```

```
anp 18 16:06:00 ubuntu rubackup_server[27635]: RuBackup db_watcher
connected to the database rubackup@|>
```

```
lines 1-20/20 (END)
```

Внимание! Если у вас возникает проблема запуска сервиса сервера RuBackup и служебная база данных RuBackup в PostgreSQL установлена при этом 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:

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

2. Перезагрузите systemctl:

```
$ sudo systemctl daemon-reload
```

Запуск резервного сервера в терминальном режиме

В том случае, если планируется тестирование RuBackup, рекомендуется запускать резервный сервер RuBackup в терминальном режиме с помощью команды:

```
# rubackup_server start
```

Остановить сервер RuBackup можно с помощью команды:

```
# rubackup_server stop
```

Настройка хранилища резервных копий

Если в процессе настройки сервера при помощи утилиты `rb_init` не был назначен каталог для хранения резервных копий для пула **Default**, то после настройки основного сервера RuBackup в журнальном файле `/opt/rubackup/log/RuBackup.log` появятся записи о том, что в пуле Default нет ни одной файловой системы для хранения резервных копий:

```
Thu Sep 19 12:40:30 2019: Warning: Pool: Default has no any file system
```

Необходимо назначить для пула **Default** хотя бы один каталог для хранения резервных копий. Это можно сделать при помощи утилиты командной строки или оконного менеджера системного администратора системы резервного копирования RBM:

1. Настройка хранилища с помощью `rb_local_filesystem`

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup, должны входить в группу `rubackup`. Чтобы добавить пользователей в группу, внесите изменения в файл `/etc/group`.

Чтобы назначить локальный каталог в качестве хранилища резервных копий, следует выполнить команду:

```
$ rb_local_filesystems -a /rubackup1 -p 1
```

В этом примере в качестве хранилища добавляется каталог `/rubackup1`.

2. Настройка хранилища с помощью `rb_local_filesystem`

Внимание! Настройка хранилища с помощью RBM производится, если хранилища не настроены утилитой `rb_init` в процессе первоначальной настройки.

Порядок настройки хранилища изложен в документе «Руководство системного администратора RuBackup».

Установка медиасервера

Подготовка к установке медиасервера

Необходимые пакеты

Перед установкой медиасервера RuBackup необходимо, чтобы в системе были установлены следующие пакеты.

| Название пакета | Описание |
|--|--|
| mailutils (или bsd-mailx) | Утилиты электронной почты |
| libcurl4 (возможно libcurl3 в зависимости от ОС) | Клиентская библиотека передачи данных по URL |
| libqt5sql5-psql | Необходим для ОС AstraLinux |
| libsasl2-2, libldap-2.4-2 | Необходимы для операционных систем Ubuntu 18.04, Ubuntu 20.04, Astra Linux Special Edition 1.6, Astra Linux Special Edition 1.7, Debian 10 |
| openldap, cyrus-sasl | Необходимы для операционных систем РЕД ОС 7.3, CentOS 8, CentOS 7 |
| libldap2, libsasl2-3 | Необходимы для операционной системы ALT Linux 10 |

Внимание! Названия пакетов могут отличаться в зависимости от дистрибутива Linux.

Чтобы система уведомлений RuBackup работала корректно, необходимо настроить отправку электронной почты с сервера RuBackup. Для отправки электронной почты сервер RuBackup использует утилиту /usr/bin/mail.

При использовании ленточной библиотеки с сервером резервного копирования, настройку см. в руководстве «Работа с ленточной библиотекой».

Пакеты для ОС без графической оболочки

Если вы устанавливаете сервер RuBackup на ОС без графической оболочки, то, чтобы использовать Менеджер администратора RuBackup (RBM), установите следующие пакеты:

```
$ sudo apt install libgl1-mesa-dev  
$ sudo apt install libxkbcommon-x11-0  
$ sudo apt install libfontconfig1
```

Инсталляция медиасервера RuBackup

Для инсталляции медиасервера RuBackup следует выполнить следующие действия:

1. Авторизуйтесь под пользователем root:

```
$ sudo -i
```

2. Настройте следующие переменные среды для пользователя root. Для этого добавьте следующие строки в файл `/root/.bashrc`:

```
PATH=$PATH:/opt/rubackup/bin  
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib  
export PATH  
export LD_LIBRARY_PATH
```

Эти переменные также можно определить в файле `/etc/environment`.

3. Перейдите в каталог `/root/`, для этого выполните:

```
cd /root
```

4. Перезагрузите переменные окружения:

```
$ . .bashrc
```

5. Установите пакет `rubackup-common`. Пример:

```
$ sudo dpkg -i rubackup-common_2.0.0-1_amd64.deb
```


6. Установите пакет **rubackup-client**. Пример:

```
$ sudo dpkg -i rubackup-client_2.0.0-1_amd64.deb
```

7. Установите пакет **rubackup-server**. Пример:

```
$ sudo dpkg -i rubackup-server_2.0.0-1_amd64.deb
```

8. Установите пакет **rubackup-rbm**. Пример:

```
$ sudo dpkg -i rubackup-rbm_2.0.0-1_amd64.deb
```

При установке сервера RuBackup в ОС Astra Linux Special Edition 1.6 или 1.7 с активированным режимом защитной программной среды, после установки пакета **rubackup-server** следует:

1. Выполнить команду:

```
$ sudo update-initramfs -u -k all
```

2. Перезагрузить операционную систему:

```
$ sudo reboot
```

Установка лицензии

Внимание! Для работы медиасервера требуется отдельная лицензия.

Сервер RuBackup содержит в себе тестовую лицензию на выполнение резервного копирования общим объемом резервных копий 1 ТБ. При первом запуске сервер RuBackup получит лицензионный файл от глобального лицензионного сервера RuBackup. Если выход в Интернет с сервера невозможен, обратитесь к своему поставщику с указанием *hardware ID* для получения лицензионного файла.

Hardware ID можно узнать при помощи следующей команды:

```
# rubackup_server hwid
```

RuBackup hardware ID:

```
5253096d055899485ed2787eccfc57ae54ff04e76104856726c913732aa0  
c2b8
```

Файлы лицензии требуется переименовать как **rubackup.lic** и разместить в **/opt/rubackup/etc/rubackup.lic**, заменив тестовую лицензию на выданную.

Для того чтобы получить информацию о лицензии, выполните следующие действия:

1. Выполните команду:

```
# rubackup_server license
```

2. Перейдите в журнальный файл /opt/rubackup/log/RuBackup.log. В журнальном файле отобразится информация о лицензии:

```
Server: 'dima' description: 'Primary RuBackup server'
Node type: Primary RuBackup server
HWID: 3aa1a74636919cbc7ab0b0c012339e868171157f4adb47c4a3959b75a04ab2c3
License start date: 2024-02-20, end date: 2025-02-20
The license issued to the customer: localhost
License type: 'Backend', Maximum capacity: 1 TB
Used: 0.000000 TB
```

- Server, description - Имя узла сервера и описание
- Node type (primary, secondary, media) - Роль сервера (основной, резервный, медиасервер)
- HWID - Hardware ID
- License start date, end date - Дата начала и дата окончания действия лицензии
- The license issued to the customer - Пользователь, владеющий лицензией
- License type, Maximum capacity - Тип лицензии, максимальный объем данных, который позволяет резервировать лицензия данного типа
- Used - Объем зарезервированных данных

Настройка медиасервера

Необходимо чтобы медиасервер имел возможность определить IP-адрес основного сервера по hostname основного сервера. Для этого необходимо произвести соответствующие настройки в /etc/hosts на медиасервере.

Необходимо добавить IP-адрес резервного сервера в pg_hba.conf СУБД PostgreSQL, содержащую БД rubackup.

Первоначальная настройка медиасервера RuBackup осуществляется с помощью интерактивной утилиты **rb_init**. Выполните следующие действия:

1. Запустите `rb_init` от имени суперпользователя (с правами `root`).

```
root@rubackup-media:~# rb_init
RuBackup initialization utility
Copyright 2018-2022: LLC "RUBACKUP"
Исключительные права принадлежат ООО "РУБЭКАП"
Author is Andrey Kuznetsov
Version: 2.0 Build: 4
```

You MUST agree with the End User License Agreement (EULA) before installing RuBackup (y[es]/n[o]/r[ead]/q[uit])

2. Примите лицензионное соглашение. Для этого нажмите клавишу `y`.

You MUST agree with the End User License Agreement (EULA) before installing RuBackup (y[es]/n[o]/r[ead]/q[uit])

```
RuBackup command service was added to /etc/services
RuBackup license service was added to /etc/services
RuBackup media service was added to /etc/services
Do you want to configure RuBackup server (primary, secondary, media) or
client (p/s/m/c/q)?
```

3. Выберите сценарий конфигурирования медиасервера. Для этого нажмите клавишу `m`.

Do you want to configure RuBackup server (primary, secondary, media) or client (p/s/m/c/q)? **m**

Media server configuration...

Enter hostname or IP address of PostgreSQL server [localhost]:

4. Введите адрес сервера, на котором располагается база данных RuBackup, и пароль (по умолчанию при нажатии клавиши `Enter` в качестве адрес сервера используется `localhost`).

Enter hostname or IP address of PostgreSQL server [localhost]: **primary-server**

Enter name of RuBackup superuser [rubackup]:

5. Укажите имя суперпользователя RuBackup:

Внимание! В имени суперпользователя запрещено использовать следующие символы: пробел, \, \$, #, ` , /, ?, *, ., ,, ;, :, %, ^, &, <, >

```
Enter name of RuBackup superuser [ rubackup ]: rubackup
Please enter password for "rubackup" database user:
```

6. Введите пароль для суперпользователя RuBackup:

```
Please enter password for "rubackup" database user:
Enter RuBackup database name [ rubackup ]:
```

7. Введите имя базы данных (по умолчанию при нажатии клавиши Enter в качестве имени базы данных используется rubackup):

Внимание! В имени суперпользователя запрещено использовать следующие символы: пробел, \, \$, #, ` , /, ?, *, ., ,, ;, :, %, ^, &, <, >

```
Enter RuBackup database name [ rubackup ]:
RuBackup client configuration: client-server mode
```

```
Hostname of primary server: primary-server
```

8. Укажите, хотите ли вы создать резервный сервер:

```
Will you use secondary server (y/n)?y
Hostname of secondary server: test
```

```
Choose client net interface ID for use:
```

9. Выберите сетевой интерфейс, посредством которого клиенту RuBackup разрешено взаимодействовать с системой резервного копирования:

```
Possible interfaces for RuBackup client communication:
```

```
lo [0]
enps18 [1]
```

```
Choose client net interface ID for use: 1
```

```
Selected interface: enp1s0
```

```
Do you allow centralized recovery (y/n)?
```

10. Укажите, можно ли будет администратору системы СРК RuBackup восстанавливать копии, сделанные для данного клиента

Do you allow centralized recovery (y/n)?y

Do you plan to use continuous remote replication to apply remote replicas on this client (y/n)?

11. Укажите, будет ли использоваться непрерывная удаленная репликация на этом клиенте:

Do you plan to use continuous remote replication to apply remote replicas on this client (y/n)?y

Enter local backup directory path [/tmp] :/rubackup-tmp

12. Укажите директорию для временных операций с файлами резервных копий:

Enter local backup directory path [/tmp] :**rubackup-tmp**

13. Укажите количество потоков для одновременной обработки задач резервного копирования на основном сервере (каждый поток имеет отдельное соединение со служебной базой данных СРК):

Set amount threads parallelizm for server [8]:

14. Укажите количество потоков для одновременной обработки задач резервного копирования на медиасervere (каждый поток имеет отдельное соединение со служебной базой данных СРК):

Set amount threads parallelizm media server [8]:

Create RuBackup master key...

15. Укажите, хотите ли вы создать ключи электронно-цифровой подписи:

Will you use digital signature (y/n)?y

Create new secret key

Create new public key

16. Укажите, хотите ли вы установить верхний предел оперативной памяти, которая может использоваться при резервном копировании на клиенте (точность верхней границы объема памяти не гарантируется):

```
Do you want to set a soft memory threshold? (y/n)?y
```

```
Enter the allowed amount of memory for backup in GB (integer value):
```

17. Укажите объем оперативной памяти, который может использоваться при резервном копировании на клиенте в ГБ (целое число):

```
Enter the allowed amount of memory for backup in GB (integer value):
```

По окончании работы `rb_init` запустите клиентский и серверный сервисы резервного копирования. Следуйте инструкции из раздела «Настройка пользователей на медиасервере RuBackup».

По завершении настройки медиасервера необходимо:

1. Авторизовать медиасервер при первом запуске в системе резервного копирования в RBM (см. подробности в «Руководстве системного администратора RuBackup»).

Внимание! После запуска медиасервера необходимо соблюсти порядок авторизации! Сначала нужно авторизовать в системе клиента и только потом медиасервер. В противном случае будет добавлено два клиента, что приведет к ошибкам.

2. Перезагрузить медиасервер:

```
$ sudo systemctl restart rubackup_server
```

3. Медиасерверу нужно назначить хотя бы один пул типа «Файловая система» для хранения резервных копий и каталог для хранения резервных копий.

Эти задачи можно выполнить в оконном Менеджере Администратора RBM (см. «Руководство администратора RuBackup»).

Настройка пользователей на медиасervere RuBackup

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup и оконного Менеджера Администратора (RBM), должны:

- иметь правильно настроенные переменные среды,
- входить в группу `rubackup`.

Группа **rubackup** была создана в процессе установки пакета `rubackup-common`.

Чтобы настроить пользователя для возможности работы с RuBackup, выполните следующие действия:

1. Добавьте пользователя в группу `rubackup` при помощи команды:

```
$ sudo usermod -a -G rubackup пользователь
```

После этого введите команду:

```
$ sg rubackup
```

2. Настройте для *пользователя* следующие переменные среды (добавьте следующие строки в файл `/home/пользователь/.bashrc`):

```
PATH=$PATH:/opt/rubackup/bin  
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib  
export PATH  
export LD_LIBRARY_PATH
```

3. Перезагрузите переменные окружения:

```
$ . .bashrc
```

Запуск медиасervera RuBackup

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

1. Добавьте сервис клиента RuBackup в автозапуск при загрузке системы:

```
$ sudo systemctl enable rubackup_client.service
```

2. Добавьте сервис сервера RuBackup в автозапуск при загрузке системы:

```
$ sudo systemctl enable rubackup_server.service
```

3. Чтобы служба systemd перезагрузила настройки, введите команду:

```
$ 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 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 Thu 2023-02-09 13:24:23 MSK; 2 weeks 1 days ago
```

```
Main PID: 3038534 (rubackup_client)
```

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

```
Memory: 16.5M
```

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

```
└─3038534 /opt/rubackup/bin/rubackup_client start
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Check additional RuBackup modules:
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Try to check module: 'File system' ...
```



```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Execute OS command: /opt/rubackup/modules/rb_module_filesystem -t 2>&1
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Module version: 2.0
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: ... module 'File system' was checked successfully. Module supports archiving
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Try to check module: 'LVM logical volume' ...
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Execute OS command: /opt/rubackup/modules/rb_module_lvm -t 2>&1
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: Module version: 2.0
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: LVM version: 2.03.07(2)
```

```
Feb 10 19:54:12 rb-primary rubackup_client[3038534]: ... module 'LVM logical volume' was checked successfully
```

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 2022-04-18 16:05:18 MSK; 2min 31s ago

Process: 27631 ExecStart=/opt/rubackup/bin/rubackup_server start (code=exited, status=0/SUCCESS)

Main PID: 27635 (rubackup_server)

Tasks: 29 (limit: 4628)

Memory: 7.3M

CGroup: /system.slice/rubackup_server.service

└─27635 /opt/rubackup/bin/rubackup_server start

```
apn 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup monitoring
connected to the database rubackup@l>
apn 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup
media_server_TLQ_loop connected to the database>
apn 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup Media Net Server
connected to the database ruba>
apn 18 16:05:19 ubuntu rubackup_server[27635]: RuBackup Media Net Server
will use 8 threads
apn 18 16:05:19 ubuntu rubackup_server[27635]: Media server heartbeat:
Connected to RuBackup server: 12>
apn 18 16:05:20 ubuntu rubackup_server[27635]: Media server ubuntu has
added to alive RuBackup server g>
apn 18 16:05:24 ubuntu rubackup_server[27635]: RuBackup Media Server
internal tasks loop connected to t>
apn 18 16:05:24 ubuntu rubackup_server[27635]: RuBackup media server cloud
loop connected to the databa>
apn 18 16:06:00 ubuntu rubackup_server[27635]: RuBackup db_service
connected to the database rubackup@l>
apn 18 16:06:00 ubuntu rubackup_server[27635]: RuBackup db_watcher
connected to the database rubackup@l>
lines 1-20/20 (END)
```

Внимание! Если у вас возникает проблема запуска сервиса сервера 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:

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

2. Перезагрузите systemctl:

```
$ sudo systemctl daemon-reload
```

Запуск медиасервера в терминальном режиме

В том случае, если планируется тестирование RuBackup, рекомендуется запускать сервер RuBackup в терминальном режиме с помощью команды:

```
# rubackup_server start
```

Остановить сервер RuBackup можно с помощью команды:

```
# rubackup_server stop
```

Настройка хранилища резервных копий

В процессе настройки медиасервера при помощи утилиты rb_init не назначается каталог хранения резервных копий для пула **Default**, поэтому в журнальном файле /opt/rubackup/log/RuBackup.log появится запись о том, что в пуле Default нет ни одной файловой системы для хранения резервных копий:

```
Thu Sep 19 12:40:30 2019: Warning: Pool: Default has no any file system
```

Необходимо назначить для пула **Default** хотя бы один каталог для хранения резервных копий. Это можно сделать при помощи утилиты командной строки или Менеджера администратора RuBackup (RBM):

1. Настройка хранилища с помощью rb_local_filesystem

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup, должны входить в группу rubackup. Чтобы добавить пользователей в группу, внесите изменения в файл /etc/group.

Чтобы назначить локальный каталог в качестве хранилища резервных копий, следует выполнить команду:

```
$ rb_local_filesystems -a /rubackup1 -p 1
```

В этом примере в качестве хранилища добавляется каталог /rubackup1.

2. Настройка хранилища с помощью rb_local_filesystem

Внимание! Настройка хранилища с помощью RBM производится, если хранилища не настроены утилитой rb_init в процессе первоначальной настройки.

Порядок настройки хранилища изложен в документе «Руководство системного администратора RuBackup».

Установка клиента

Подготовка к установке клиента

Перед установкой клиента RuBackup необходимо провести настройку, описанную в этом разделе

Внимание! Перед установкой убедитесь, что сетевое имя узла отличается от «localhost».

Пакеты для ОС без графической оболочки

Если вы устанавливаете клиент RuBackup на ОС без графической оболочки, то для возможности использовать Менеджер администратора RuBackup (RBM) необходимо установить следующие пакеты:

```
$ sudo apt install libgl1-mesa-dev  
$ sudo apt install libxkbcommon-x11-0  
$ sudo apt install libfontconfig1  
$ sudo apt install libqt5gui5
```

В зависимости от используемой ОС, кроме указанных выше вам могут потребоваться дополнительные пакеты. При необходимости, обратитесь в службу технической поддержки RuBackup по адресу электронной почты support@rubackup.ru.

Инсталляция клиента RuBackup

Для установки клиента RuBackup следует выполнить следующие действия:

- 1 Авторизуйтесь под пользователем root:

```
$ sudo -i
```

- 2 Настройте следующие переменные среды для пользователя root. Для этого добавьте следующие строки в файл /root/.bashrc:

```
PATH=$PATH:/opt/rubackup/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib
export PATH
export LD_LIBRARY_PATH
```

Эти переменные также можно определить в файле `/etc/environment`.

3. Перейдите в каталог **/root/**, для этого выполните:

```
cd /root
```

4. Перезагрузите переменные окружения:

```
# . .bashrc
```

5. Установите пакет **rubackup-common**. Пример:

```
$ sudo dpkg -i rubackup-common_2.0.0-1_amd64.deb
```

6. Установите пакет **rubackup-client**. Пример:

```
$ sudo dpkg -i rubackup-client_2.0.0-1_amd64.deb
```

Имя файла пакета может отличаться в зависимости от сборки.

При установке клиента RuBackup в ОС Astra Linux Special Edition 1.6 или 1.7 с активированным режимом защитной программной среды, после установки пакета **rubackup-client** следует:

1. Добавить в файл `/etc/digisig/digisig_initramfs.conf` строки:

```
DIGSIG_ENFORCE=1
DIGSIG_LOAD_KEYS=1
```

2. Выполнить команду:

```
$ sudo update-initramfs -u -k all
```

3. Перезагрузить операционную систему:

```
$ sudo reboot
```

Настройка клиента RuBackup

Внимание! Для клиента RuBackup должно быть настроено корректное разрешение имени основного сервера. Если клиент RuBackup не сможет определить IP-адрес по имени основного сервера, то он прекратит свою работу. Используйте корректные настройки DNS или файла /etc/hosts.

Первоначальная настройка клиента RuBackup осуществляется с помощью интерактивной утилиты **rb_init**. Для настройки выполните следующие действия:

1. Запустите **rb_init** (от пользователя **root**).

rb_init

```
root@rb-client01:~# rb_init
RuBackup initialization utility
Copyright 2018-2022: LLC "RUBACKUP"
Исключительные права принадлежат ООО "РУБЭКАП"
Author is Andrey Kuznetsov
Version: 2.0 Build: 0
```

2. Примите лицензионное соглашение. Для этого нажмите клавишу **y**.

```
You MUST agree with the End User License Agreement (EULA) before
installing RuBackup (y[es]/n[o]/r[ead]/q[uit])y
RuBackup client configuration...
Choose client mode: client-server or autonomous (c/a)?
```

3. Выберите сценарий конфигурирования клиента: клиент-сервер или автономный. Для выбора связки клиент-сервер введите **c**.

```
Choose client mode: client-server or autonomous (c/a)?c
RuBackupc client configuration: client-server mode
```

4. Укажите адрес основного (primary) сервера СРК:

```
Hostname of primary server: rb-primary
```

5. Если в конфигурации подразумевается резервный (secondary) сервер, то выберите эту возможность:

```
Will you use secondary server (y/n)?
```

6. Укажите имя резервного сервера:

```
Hostname of secondary server: rb-secondary
```

7. Укажите ID сетевого интерфейса, посредством которого клиенту RuBackup разрешено взаимодействовать с системой резервного копирования:

Possible interfaces for RuBackup client communication:

lo [0]

enp18 [1]

Choose client net interface ID for use: **1**

Selected interface: enp18

Do you allow centralized recovery (y/n)?

8. Укажите, можно ли будет администратору системы СРК RuBackup восстанавливать копии, сделанные для данного клиента:

Do you allow centralized recovery (y/n)? y

Do you plan to use continuous remote replication to apply remote replicas on this client (y/n)?

9. Укажите, будет ли использоваться непрерывная удаленная репликация на этом клиенте:

Do you plan to use continuous remote replication to apply remote replicas on this client (y/n)?y

10. Укажите локальный каталог для временного хранения файлов с метаданными, создаваемых при операциях резервного копирования:

Enter local backup directory path [/tmp] : /rubackup-tmp

Would you like to create /rubackup-tmp (y/n)?

11. Подтвердите создание каталога для временных файлов:

Would you like to create /rubackup-tmp (y/n)?y

Create RuBackup master key...

12. Укажите хотите ли вы создать ключи электронно цифровой подписи:

Will you use digital signature (y/n)?y

Create new secret key

Create new public key

13. Укажите, хотите ли вы установить верхний предел оперативной памяти, которая может использоваться при резервном копировании на клиенте (точность верхней границы объема памяти не гарантируется):

Do you want to set a soft memory threshold? (y/n)?y

Enter the allowed amount of memory for backup in GB (integer value):

14. Укажите объем оперативной памяти, который может использоваться при резервном копировании на клиенте в ГБ (целое число):

Enter the allowed amount of memory for backup in GB (integer value):

15. Укажите возможность подключения системы мониторинга:

Do you want to enable system monitoring of this client (y/n)?y

Примечание: порядок установки автономного режима работы клиента приведен в документе «Автономный режим работы клиента Rubackup».

По окончании работы `rb_init` клиент будет сконфигурирован. После этого необходимо добавить пользователя в группу `rubackup` и запустить клиентский процесс (см. разделы «Настройка пользователей на клиенте RuBackup» и «Запуск клиента RuBackup»).

Также необходимо авторизовать клиента в СРК. Это может сделать системный администратор RuBackup при помощи Менеджера администратора RuBackup (RBM) либо утилит командной строки. Процесс авторизации клиента описан в документе «Руководство системного администратора RuBackup».

Настройка пользователей на клиенте RuBackup

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup и Менеджера клиента RuBackup (RBC), должны иметь правильно настроенные переменные среды.

Группа **rubackup** создаётся в процессе установки пакета `rubackup-common`.

Чтобы настроить пользователя для возможности работы с RuBackup, выполните следующие действия:

1. Добавьте пользователя в группу `rubackup` при помощи команды:

```
$ sudo usermod -a -G rubackup пользователь
```

После этого введите команду **`sg rubackup`**.

2. Настройте для пользователя следующие переменные среды. Для этого добавьте следующие строки в файл `/home/пользователь/.bashrc`:

```
PATH=$PATH:/opt/rubackup/bin  
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib  
export PATH  
export LD_LIBRARY_PATH
```

3. Перезагрузите переменные окружения:

```
$ . .bashrc
```

Важно! Т.к. команда используется для конкретного файла, её необходимо выполнять из директории, где расположен этот файл, либо указать полный путь до файла.

4. Настроенный таким образом пользователь сможет запускать утилиты командной строки и графический менеджер клиента RuBackup.

Запуск клиента RuBackup

В том случае, если планируется тестирование RuBackup, рекомендуется запускать клиент RuBackup в терминальном режиме с помощью команды:

```
# rubackup_client start
```

Остановить клиент RuBackup можно с помощью команды:

```
# rubackup_client stop
```

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

1. Включите сервис клиента RuBackup:

```
$ sudo systemctl enable 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 Fri 2022-04-15 09:45:34 MSK; 3 days ago

Process: 3234 ExecStart=/opt/rubackup/bin/rubackup_client start (code=exited, status=0/SUCCESS)

Main PID: 3237 (rubackup_client)

Tasks: 2 (limit: 4628)

Memory: 173.9M

CGroup: /system.slice/rubackup_client.service

└─3237 /opt/rubackup/bin/rubackup_client start

anp 15 12:49:12 ubuntu rubackup_client[3237]: Create digital signature for /rubackup-tmp/ubuntu_TaskID_>

anp 15 12:49:12 ubuntu rubackup_client[3237]: Execute OS command: /usr/bin/openssl dgst -sign /opt/ruba>

anp 15 12:49:12 ubuntu rubackup_client[3237]: Create digital signature for /rubackup-tmp/ubuntu_TaskID_>

anp 15 12:49:12 ubuntu rubackup_client[3237]: Digital signatures were transfered successfully for task >

anp 15 12:49:12 ubuntu rubackup_client[3237]: Remove obsoleted file: /rubackup-tmp/ubuntu_TaskID_1_NORu>

anp 15 12:49:12 ubuntu rubackup_client[3237]: Remove obsoleted file: /rubackup-tmp/ubuntu_TaskID_1_NORu>

anp 15 12:49:12 ubuntu rubackup_client[3237]: Remove obsoleted signature file: /rubackup-tmp/ubuntu_Tas>

anp 15 12:49:12 ubuntu rubackup_client[3237]: Remove obsoleted signature file: /rubackup-tmp/ubuntu_Tas>

anp 15 12:49:12 ubuntu rubackup_client[3237]: Set status for task ID: 1 from: Transmission to: Done

anp 15 12:49:12 ubuntu rubackup_client[3237]: Task was done. ID: 1

Дополнительные настройки

Установка RBM на удаленном хосте

СРК Rubackup предоставляет возможность установки оконного Менеджера Администратора RuBackup (RBM) на удаленном хосте.

Пользователи, от имени которых будет осуществляться запуск оконного Менеджера Администратора (RBM) на удаленном хосте, должны входить в группу rubackup.

Подготовка к установке

- 1) Подготовка сервера Rubackup

Для подготовки сервера Rubackup необходимо настроить файл «/etc/postgresql/12/main/pg_hba.conf» и в строке с IPv4 прописать адрес хоста, на котором будет установлен удалённый RBM:

```
# 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
host all all 192.168.113.21/32 md5
host all all 192.168.113.30/32 md5
host all all 192.168.113.31/32 md5
host all all 192.168.113.33/32 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 peer
host replication all 127.0.0.1/32 md5
host replication all ::1/128 md5
"/etc/postgresql/12/main/pg_hba.conf" 107L, 5080C
```

Примечание – Путь настройки файла может отличаться в зависимости от версии postgresql.

- 2) Подготовка узла, с которого будет выполняться вход в RBM

Для подготовки узла необходимо выполнить следующие действия:

1. Скачайте пакет драйверов

```
$ sudo apt install libqt5sql5-psql
```

2. Настройте файлы «/root/.bashrc» и «/home/<user_name>/.bashrc»

```
PATH=$PATH:/opt/rubackup/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib
export PATH
export LD_LIBRARY_PATH
-- ВСТАВКА --
```

```
PATH=$PATH:/opt/rubackup/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/rubackup/lib
export PATH
export LD_LIBRARY_PATH
```

3. Выполните команду:

```
$ . .bashrc
```

4. Если ОС без графической оболочки, установите следующие пакеты:

```
$ sudo apt install libgl1-mesa-dev
$ sudo apt install libxkbcommon-x11-0
$ sudo apt install libfontconfig1
```

5. Установите пакеты rubackup-common и rubackup-rbm. Пример:

```
$ sudo dpkg -i rubackup-common_2.0.0-1_amd64.deb
$ sudo dpkg -i rubackup-rbm_2.0.0-1_amd64.deb
```

При установке пакет rubackup-rbm может попросить установить дополнительные зависимости - это необходимо сделать для корректной работы приложения.

После совершения этих действий RBM будет готов к запуску.

Для запуска Менеджера Администратора RuBackup используйте команду:

```
$ rbm&
```

или

```
opt/rubackup/bin/rbm
```

После этого в открывшееся окно «Аутентификация» введите наименование сервера RuBackup, имя пользователя и пароль (Рисунок 1).

Примечание: при настройке многопользовательского режима RuBackup в RBM возможно зайти под определённой ролью: суперпользователь, супервайзер, сопровождающий или администратор.

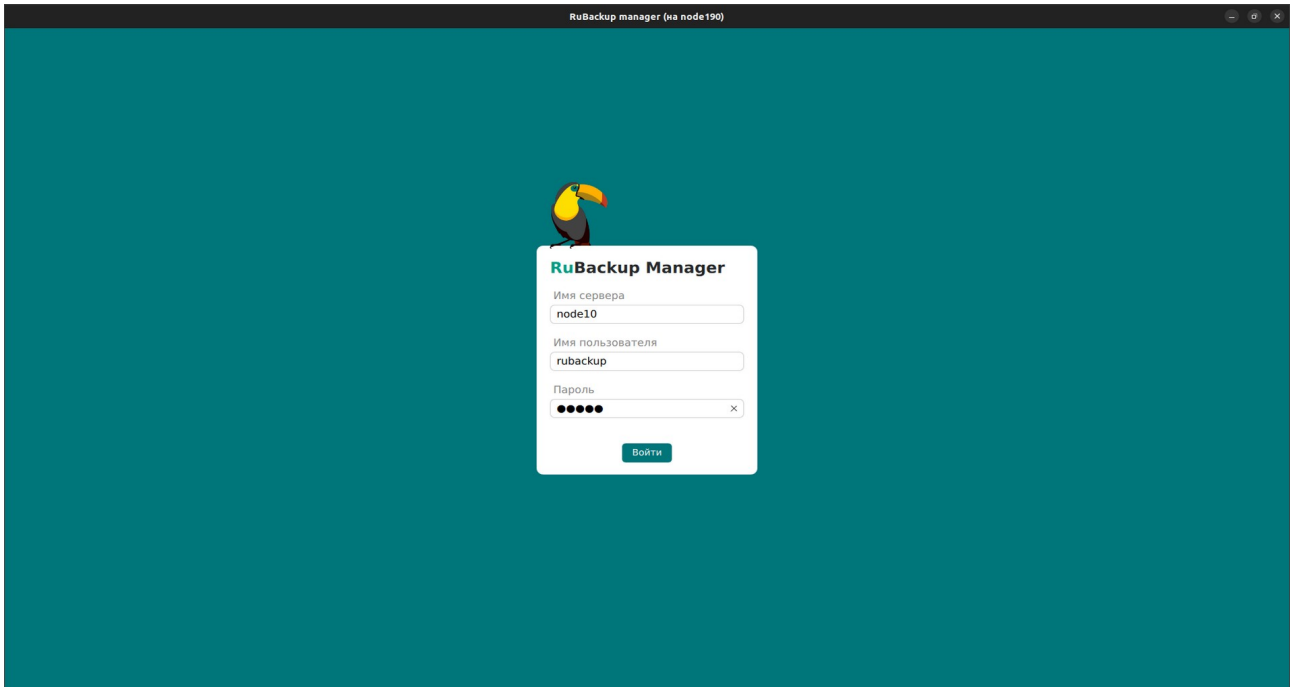


Рисунок 1

Удаление клиента RuBackup

Удалите клиент RuBackup при помощи команд:

Linux deb-пакет:

```
$ sudo apt remove rubackup-client
```

Linux rpm-пакет:

```
$ sudo rpm -e rubackup-client
```

Если необходимо удалить клиент RuBackup из конфигурации СРК, это может сделать системный администратор RuBackup при помощи Менеджера администратора RuBackup.

После удаления клиента RuBackup в ОС Astra Linux Special Edition 1.6 или 1.7 с активированным режимом защитной программной среды:

1. Выполните команду:

```
$ sudo update-initramfs -u -k all
```

2. Перезагрузите операционную систему:

```
$ sudo reboot
```

Удаление сервера RuBackup

Чтобы удалить сервер RuBackup, выполните следующие действия:

1. Если сервис был запущен не через `systemctl`, а вручную, выполните следующую команду:

```
$ rubackup_server stop
```

2. Удалите сервер RuBackup.

Для удаления Linux deb пакета команда:

```
$ sudo apt remove rubackup-server
```

Для удаления Linux rpm пакета команда:

```
$ sudo rpm -e rubackup-server
```

При необходимости удалить медиасервер RuBackup из конфигурации СРК можно при помощи Менеджера администратора RuBackup (RBM).

Мастер-ключ

В ходе настройки будет создан мастер-ключ для защитного преобразования резервных копий и ключи для электронной подписи, если предполагается использовать электронную подпись.

Внимание! При потере ключа вы не сможете восстановить данные из резервной копии, если последняя была преобразована с помощью защитных алгоритмов.

Важно! Ключи рекомендуется после создания скопировать на внешний носитель, а также распечатать бумажную копию и убрать её в надежное место.

Мастер-ключ рекомендуется распечатать при помощи утилиты hexdump, так как она может содержать неотображаемые на экране символы. Например:

```
$ hexdump/opt/rubackup/keys/master-key
```

```
0000000 79d1 4749 7335 e387 9f74 c67e 55a7 20ff
```

```
0000010 6284 54as 83a3 2053 4818 e183 1528 a343
```

```
0000020
```

Важно! Секретный ключ необходимо хранить в месте, доступном только тем, кто должен иметь возможность произвести обратное преобразование файла. Потеря ключа делает невозможным обратное преобразование файла.

Неинтерактивный режим работы

Неинтерактивный режим работы необходим для выполнения сценариев массового развертывания, например, при использовании Ansible — программного решения для удаленного управления конфигурациями серверов.

Администратор имеет возможность конфигурировать СРК RuBackup в `bash/shell` однострочной командой и, как следствие, использовать эту команду в скриптах для автоматизации процесса.

Настройка СРК RuBackup осуществляется с помощью интерактивной утилиты **`rb_init`** (**неинтерактивный режим**). Описание утилиты приведено в документе «Утилиты командной строки».

Установка обновлений RuBackup

При установке обновлений серверной группировки RuBackup необходимо обновить:

- Основной сервер.
- Резервный сервер (при наличии).
- Все медиасервера (при наличии).

При этом должны быть обновлены все компоненты серверной группировки RuBackup.

Внимание! Начиная с версии 2.1, обратная совместимость клиентской и серверной частей СРК Rubackup возможна только в случае N версии серверной части и N или $(N-1)$ версии клиентской части, где N — номер версии оперативного обновления публичного релиза 2.1.

Обновление необходимо выполнять последовательно по всем версиям, начиная со следующей версии относительно текущей и до последней доступной версии, включая все промежуточные.

Обновление можно выполнить:

- Обновив пакеты вручную (см. разделы Подготовка к обновлению, Обновление).
- Автоматически.

Для автоматического обновления пакетов группировки RuBackup необходимо:

- 1.Скачать архив обновления, содержащий скрипт `script.sh`.
- 2.Разархивировать архив.
- 3.Выполнить скрипт `script.sh`:

\$ sudo ./script.sh

При выполнении скрипта будут проверены версии установленных пакетов и новых пакетов. Если версия новых пакетов старше версии установленных пакетов на один релиз, то обновление будет выполнено. В противном случае обновление выполнено не будет.

В результате автоматического обновления будут обновлены пакеты RuBackup, которые были расположены в одной папке со скриптом. Обновленный сервер будет работать только с теми клиентами одной кластерной группы, которые были обновлены, или со всеми клиентами, кластерной группы в случае, если ни один из них не был обновлён.

Подготовка к обновлению

Перед обновлением СРК RuBackup создайте любым удобным способом резервную копию текущей конфигурации СРК, чтобы иметь возможность восстановить Систему при возникновении ошибок. Один из возможных сценариев — произвести резервное копирование директории /opt/rubackup/etc на всех компонентах СРК и сделать резервную копию базы данных rubackup. Для этого на узле сервера PostgreSQL, содержащего служебную базу данных, выполните:

```
$ pg_dump -U rubackup rubackup > rubackup_database_dump.sql
```

Чтобы сохранить настройки модулей, создайте резервную копию конфигурационных файлов на всех машинах, на которых установлены компоненты СРК RuBackup:

```
$ sudo tar cvfz RuBackupConfig.tar.gz /opt/rubackup/etc/
```

Обновление

Чтобы обновить RuBackup, выполните следующие действия:

1. Закройте окно RBM на АРМ администратора СРК или ином узле, используемом для запуска графического Менеджера администратора RuBackup (RBM).

2. Скачайте свежую сборку с официального сайта www.rubackup.ru.

3. Распакуйте следующие пакеты на соответствующих узлах, например, с помощью команды **\$ unzip файл_архива.zip**:

- rubackup-common, rubackup-client и модули на клиенте;
- rubackup-common, rubackup-client, rubackup-server на серверах;
- rubackup-common и rubackup-rbm на АРМ администратора.

4. Установите распакованные пакеты с помощью команд в зависимости от типа используемого пакетного менеджера в Вашем дистрибутиве Linux, например, для пакетного менеджера, оперирующего deb-пакетами:

```
$ sudo dpkg -i rubackup-common_2.0.0-1_amd64.deb
```

```
$ sudo dpkg -i rubackup-client_2.0.0-1_amd64.deb
```

```
$ sudo dpkg -i rubackup-server_2.0.0-1_amd64.deb
```

```
$ sudo dpkg -i rubackup-rbm_2.0.0-1_amd64.deb
```

Или для пакетного менеджера, оперирующего rpm-пакетами:

```
$ sudo rpm -U rubackup-common_2.0.0-1.el7.x86_64.rpm
$ sudo rpm -U rubackup-client_2.0.0-1.el7.x86_64.rpm
$ sudo rpm -U rubackup-server_2.0.0-1.el7.x86_64.rpm
$ sudo rpm -U rubackup-rbm_2.0.0-1.el7.x86_64.rpm
```

Пример пакета модуля:

```
$ sudo dpkg -i rubackup-postgresql_2.0.0-1_amd64.deb
или
$ sudo rpm -U rubackup-postgresql_2.0.0-1.el7.x86_64.rpm
```

5. На основном сервере RuBackup вызовите утилиту `rb_update` одним из двух способов:

- При вызове `rb_init` (будет создан новый конфигурационный файл);
 - 1) Для выбора обновления существующей базы данных после ввода имени пользователя и названия базы данных выберите действия с базой данных, т.к. она уже существует. Выберите `u` (`upgrade`).

```
You've specified already existing database. Do you want to upgrade(u), drop(d) or keep(k)
existing database (u/d/k)?u
```

- 2) Выберите вариант действий с данными базы данных. Возможно создать резервную копию данных, чтобы восстановить ее при необходимости. Для этого в шаге запроса на создание резервной копии данных существующей базы данных нужно выберите `y` (`yes`).

```
Do you want to dump the database 'rubackup' (pg_dump method) (y/n/q)? y
```

- 3) Выберите формат резервной копии. Доступны два формата: `plain` (текстовый sql-скрипт), `custom` (`custom`-архив, восстановление выполняется с помощью `pg_restore`). Резервная копия в формате `custom` занимает меньше места на диске, по сравнению с форматом `plain`. Также для формата `custom` доступен выбор степени сжатия (0-9) - чем выше степень сжатия, тем меньше `custom`-архив занимает места на диске и тем дольше выполняется процедура резервного копирования базы данных.

```
Enter format for dump file (c[custom], p[plain]) (c/p) [ c ]
```

```
Enter compression level (0-9) for dump file [ 1 ]:
```

```
Enter path for dump file [ /root ]:
```

4) Выберите путь для сохранения резервной копии - по умолчанию это директория, откуда была вызвана утилита.

Enter path for dump file [/root]:

5) После выполнения всех действий начнется обновление базы данных.

При необходимости восстановить базу данных из созданной резервной копии см. раздел «Восстановление базы данных».

- Напрямую с помощью команды (в этом случае будет только обновлена структура служебной базы данных):

```
$ rb_update -H <hostname> -P <port> -D <database_name> -U  
<user_name> -r <rubackup_superuser_password> -p  
<postgres_superuser_password> -I <path_to_new_sql_scripts>  
-R -O -
```

В данной команде укажите значения параметров:

<hostname> – имя хоста базы данных

<port> – номер порта базы данных

<database_name> – имя обновляемой базы данных

<user_name> – имя пользователя базы данных

<rubackup_superuser_password> – пароль от суперпользователя rubackup

<postgres_superuser_password> – пароль от суперпользователя postgres

<path_to_new_sql_scripts> – путь к каталогу с новыми sql-скриптами.

-R – принудительное обновление существующей базы данных (ничего менять не нужно)

-O – вывод процесса обновления в стандартный поток терминала (ничего менять не нужно)

6. На клиентских узлах приведите конфигурационные файлы соответствующих модулей в состояние до обновления пакетов. Исходная конфигурация модулей находится в сохраненном ранее архиве RuBackupConfig.tar.gz.

- Для восстановления разархивируйте архив командой:

```
$ tar xvfz RuBackupConfig.tar.gz
```

- Далее замените файл конфигурации установленного модуля file.conf в папке opt/rubackup/etc, например, командой:

```
$ mv file.conf opt/rubackup/etc, где file.conf — файл конфигурации модуля
```

7. На всех узлах, на которых было установлено обновление, выполните команду:

```
$ sudo systemctl daemon-reload
```

8. Запустите все клиентские и серверные процессы с помощью команд:

а) На клиентских узлах с помощью команды:

```
$ sudo systemctl start rubackup_client.service
```

б) На основном и резервном серверах, а также медиасervere с помощью команд:

```
$ sudo systemctl start rubackup_client.service
```

```
$ sudo systemctl start rubackup_server.service
```

9. Убедитесь, что СРК RuBackup обновилась корректно:

1) Проверьте, запускается ли RBM на APM администратора СРК или ином узле, используемом для запуска Менеджера администратора RuBackup, с помощью команды:

```
$ rbm&
```

2) Удостоверьтесь, что все клиенты и серверы находятся в статусе онлайн:

а) На клиентских узлах с помощью команды:

```
$ sudo systemctl status rubackup_client.service
```

б) На основных и резервных серверах, а также на медиасerverах с помощью команд:

```
$ sudo systemctl status rubackup_client
```

```
$ sudo systemctl status rubackup_server
```

3) Проверьте сохранность резервных копий в RBM, в разделе «Репозиторий», либо через терминал с помощью команды:

```
$ rb_repository -l
```

Проверьте сохранность правил глобального расписания:

```
$ rb_global_schedule -l
```

и сохранность стратегий:

```
$ rb_strategies -l
```

После обновления рекомендуем выполнить базовое резервное копирование и восстановление файла небольшого размера (до 500 МБ) во все типы хранилищ, чтобы убедиться в работоспособности СРК после обновления.

Восстановление базы данных

При необходимости восстановить сделанную во время конфигурации резервную копию базы данных, выполните следующие шаги:

1. Остановите все процессы на всех узлах, подключенных к базе данных, которую необходимо восстановить:

а) На клиентских узлах с помощью команды:

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

б) На основном, резервном и медиасерверах СРК RuBackup с помощью команд:

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

```
$ sudo systemctl stop rubackup_server.service
```

в) На АРМ администратора СРК или ином узле, используемом для запуска Менеджера администратора RuBackup (RBM) – закройте окно RBM.

2. На узле сервера PostgreSQL, содержащего служебную базу данных, подключитесь к СУБД в режиме суперпользователя:

```
$ sudo -iu postgres psql
```

```
(12.5 (Ubuntu 12.5-0ubuntu0.20.04.1))
```

```
Type "help" for help.
```

```
postgres=#
```

3. Удалите базу данных, которую хотите восстановить из резервной копии:

```
postgres=# drop database <database_name>;
```

где <database_name> - это имя базы данных (по умолчанию – rubackup)

4. Создайте новую пустую базу данных с указанным именем и владельцем, в эту базу данных будет происходить восстановление:

```
postgres=# create database <database_name> owner <owner_name>;
```

где <database_name> - имя базы данных,

<owner_name> - имя владельца БД (по умолчанию – rubackup)

5. Отключитесь от СУБД:

```
postgres=# \q
```

6. Запустите восстановление базы данных:

- Для резервной копии, сделанной в формате plain text (файл с расширением sql), выполните в терминале команду:

```
sudo -u <admin_user_name> psql <database_name> <  
<file_name_and_path>.sql
```

где <admin_user_name> - имя пользователя, обладающего правами администратора, по умолчанию это postgres,

<database_name> - имя базы данных,

<file_name_and_path>.sql — путь до файла и имя файла от резервной копии базы данных

- Для резервной копии, сделанной в формате custom (файл с расширением .dump), с помощью утилиты pg_restore в терминале выполните следующую команду:

```
pg_restore -h <hostname> -p <port> -U <admin_user_name> -d  
<database_name> <file_name_and_path>.dump
```

где <hostname> - имя хоста, на котором будет восстановлена БД

<port> -номер порта

<admin_user_name> - имя пользователя, обладающего правами администратора, по умолчанию это postgres,

<database_name> - имя базы данных,

<file_name_and_path>.dump — путь до файла резервной копии базы данных и имя файла

7. Дождитесь окончания восстановления базы данных и убедитесь, что оно прошло без ошибок.

8. Запустите СРК RuBackup:

1. На всех узлах, которые подключены к восстановленной базе данных, выполните:

```
$ sudo systemctl daemon-reload
```

2. Запустите все клиентские и серверные процессы на всех узлах, подключенных к восстановленной базе данных, с помощью команд:

а) На клиентских узлах с помощью команды:

```
$ sudo systemctl start rubackup_client.service
```

б) На узлах основном и резервном серверах, а также на медиасerverе СРК RuBackup с помощью команд:

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

```
$ sudo systemctl stop rubackup_server.service
```

Установка нового модуля

Установка нового модуля вместе с обновлением СРК

На серверных узлах установка новых модулей в рамках обновления версии RuBackup происходит вместе с установкой пакета **rubackup_server**.

1. На клиентских узлах установите пакет нового модуля командой:

```
$ sudo apt install <имя пакета>
```

2. Настройте модуль по его инструкции;
3. Перезапустите клиентский сервис командой:

```
$ sudo systemctl restart rubackup_client
```

Теперь Вы можете выбрать новый модуль (новый тип ресурса) при создании правил резервного копирования.

Установка нового модуля без обновления СРК

Для того чтобы установить еще один модуль к уже имеющейся инсталляции СРК, Вам понадобится sql-скрипт и пакет устанавливаемого модуля.

1. Чтобы передать информацию о новом модуле в БД rubackup, на узле с сервером RuBackup выполните команду:

```
rb_modules -i <путь к sql-файлу>
```

2. Установите пакет модуля на требуемый клиентский узел RuBackup с помощью команды:

```
$ sudo apt install <имя пакета>
```

3. Настройте модуль по его инструкции;
4. Перезапустите клиентский сервис командой:

```
$ sudo systemctl restart rubackup_client
```

Теперь Вы можете выбрать новый модуль (новый тип ресурса) при создании правил резервного копирования.

Приложение А

(справочное)

Настройка публичного репозитория RuBackup

Пакеты СРК RuBackup также можно установить из публичных репозиториях.

Публичные репозитории доступны для операционных систем:

- Astra Linux 1.8
- Astra Linux 1.7
- Astra Linux 1.6
- Debian 10
- Ubuntu 20.04
- Ubuntu 18.04

Deb-based дистрибутивы

1. Добавляем ключ репозитория:

```
$ sudo wget -qO -  
https://edge.astralinux.ru/artifactory/api/security/keypair/g  
c-astra-official-repo-key/public | sudo apt-key add -
```

2. В /etc/apt/sources.list добавляем строку:

```
$ sudo deb https://dl.astralinux.ru/rubackup/repository-deb-  
main/ [OS-VERSION] public
```

Где:

- <https://dl.astralinux.ru/rubackup/repository-deb-main/> - основной адрес репозитория для deb-пакетов;
- [OS-VERSION] - версия ОС:
 - astra_1.8;

- astra_1.7;
 - astra_1.6;
 - debian_10;
 - ubuntu_18.04;
 - ubuntu_20.04.
- public - параметр публичного репозитория.

Пример:

```
GNU nano 3.2 /etc/apt/sources.list
# deb http://download.astralinux.ru/astra/frozen/1.7_x86-64/1.7.2/repository-base 1.7_x86-64 main contrib non-free
deb http://download.astralinux.ru/astra/frozen/1.7_x86-64/1.7.2/repository-base 1.7_x86-64 main non-free contrib
deb http://download.astralinux.ru/astra/frozen/1.7_x86-64/1.7.2/repository-extended 1.7_x86-64 main contrib non-free
deb https://dl.astralinux.ru/rubackup/repository-deb-main/ astra_1.7 public
```

3. Обновляем репозитории и проверяем наличие пакетов, а так же их версии командами (например, RBM).

4. Устанавливаем нужный пакет:

```
$ sudo apt update
```

```
$ sudo apt-cache search rubackup
```

```
$ sudo apt-cache madison rubackup-[PACKET_NAME]
```

```
$ sudo apt install -y rubackup-[PACKET_NAME]
```

Где:

- [PACKET_NAME] - имя нужного нам пакета.

Пример:

```
root@rubackup-primary:~# apt-cache search rubackup
```

```
rubackup-rbm - RuBackup administrator manager
```

```
rubackup-client - RuBackup client
```

```
rubackup-common - RuBackup common files and libs
rubackup-rest-api - RuBackup REST API
rubackup-server - RuBackup server
root@rubackup-primary:~# apt-cache madison rubackup-rbm
rubackup-rbm | 2.1.0~a.242-1 |
https://download.astralinux.ru/rubackup/repository-deb-main
astra_1.7/stable amd64 Packages
rubackup-rbm | 2.1.0~a.242-1 |
https://dl.astralinux.ru/rubackup/repository-deb-main
astra_1.7/stable amd64 Packages
root@rubackup-primary:~# apt install -y rubackup-rbm
```

Если требуется другая версия (например более ранняя), то установку производим с помощью команды:

```
$ sudo apt install rubackup-rbm=2.0.49-1
```

RPM-based дистрибутивы

Вариант 1.

1. В каталоге `/etc/yum.repos.d/` находим файл с текущими репозиториями (например, `CentOS-Base.repo`) и добавляем в конец файла следующий код:

```
#rubackup
[rubackup]
name=rubackup
baseurl=https://dl.astralinux.ru/rubackup/repository-rpm-
main/
gpgcheck=0
```

2. Обновляем репозитории и устанавливаем нужный пакет:

```
# yum update
```

```
# yum install -y rubackup-[PACKET_NAME]
```

Вариант 2.

Устанавливаем пакет напрямую.

```
# yum install -y
```

```
https://dl.astralinux.ru/artifactory/rubackup-rpm-main/[OS-  
VERSION]/public/2.0/[RELEASE-VERSION]/rubackup-[PACKET_NAME]-  
[RELEASE-VERSION].el7.x86_64.rpm
```

Где:

- [OS-VERSION] - версия ОС:
 - astra_1.6;
 - astra_1.7;
 - astra_1.8;
 - debian_10;
 - ubuntu_18.04;
 - ubuntu_20.04.
- [PACKET_NAME] - имя нужного нам пакета;
- [RELEASE-VERSION] - версия релиза.

Приложение Б

(справочное)

Перечень серверных пакетов для различных ОС

| Тип ОС | Сервер | Графика |
|--------------------------|--|---|
| Ubuntu 18.04, 20.04 | Mailutils Libcurl4 Postgresql Postgresql-contrib | Libgl1-mesa-dev Libxkbcommon-x11-0 Libfontconfig1 |
| Debian 10 | Mailutils Libcurl4 Postgresql Postgresql-contrib | |
| CentOS 7 | mailx libcurl epel-release yum-utils postgresql12-server postgresql12 postgresql12-contrib | LibGLU libxkbcommon-x11 libqt5xdg.x86_64 |
| CentOS 8 | mailx libcurl postgresql-server postgresql-contrib | xauth |
| Astra Linux 1.7 | BSD-Mailx Libcurl4 Postgresql Postgresql-contrib-11 | libqt5sql5-psql |
| Astra Linux 1.6 Smolensk | BSD-Mailx Libcurl3 Postgresql Postgresql-contrib-9.6 | libqt5sql5-psql |
| Alt Linux 9 | Mailutils Libcurl Postgresql12 Postgresql12-contrib | |
| Alt Linux 9 Elbrus | Mailutils Libcurl Postgresql12 Postgresql12-contrib | |
| RedOS 7.3 | Mailx | qt5-qtbase-postgresql |

| | | |
|--------------|---|--|
| | Libcurl Postgresql-server Postgresql-contrib | |
| Oracle Linux | Mailx Libcurl Postgresql-server Postgresql-contrib | |