RuBackup
Система резервного копирования и восстановления данных

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

Версия 2.3.0

30.10.2024

Содержание

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
- ПК СВ "Брест"
- RUSTACK
- АЭРОДИСК VAIR
- VMware vSphere
- OpenStack
- zVirt
- Tionix
- ROSA Virtualization
- DynamiX
- KVM
- 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

- Tantor Special Edition
- PostgreSQL

- Patroni
- Postgres Pro
- Greenplum Database

Бизнес-приложения

- CommuniGate Pro
- FreeIPA
- Mailion (экспериментальный модуль, прошел дизайн-тестирование)

Файловые системы и др.

- Linux
- Windows

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 медиасервере.

Операционные системы

- Astra Linux 1.8
- Astra Linux 1.7
- Astra Linux 1.6
- Debian 12

- Debian 10
- Ubuntu 22.04
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Red Hat Enterprise Linux 9
- Rosa Cobalt 7.3
- Rosa Cobalt 7.9
- Rosa Chrome 12
- Alt Linux 10
- РЕД ОС 7.3
- РЕД ОС 8

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.

<table>
<tr><th colspan="2">Аппаратный компонент</th><th colspan="3">Объем хранимых данных</th><th>Примечание</th></tr>
<tr><td colspan="2" rowspan="2">Процессор</td><td>48 ТБ</td><td>96 ТБ</td><td>144 ТБ</td><td rowspan="2">Рекомендуемые модели: Intel Xeon 4210, AMD EPYC 7000 или более современные</td></tr>
<tr><td colspan="3">10 ядер, 20 потоков (2 потока на 1 ядро или более)</td></tr>
<tr><td colspan="2">Оперативная память</td><td>128 ГБ</td><td>256 ГБ</td><td>256 ГБ</td><td></td></tr>
<tr><td rowspan="2">Дисковое пространство</td><td>Твердотельный накопитель (SSD)</td><td colspan="3">RAID 1, 2 диска по 480 ГБ каждый</td><td>Объём дискового пространства для установки операционной системы и компонентов RuBackup, за исключением конфигурационной базы данных RuBackup.</td></tr>
<tr><td>Твердотельный накопитель, подключенный через шину PCI Express (NVMe SSD)</td><td colspan="3">3.84 ТБ</td><td>• Рекомендуется в случае развертывания инстанса PostgreSQL для конфигурационной базы данных RuBackup на той же машине, где установлен сервер RuBackup.
• 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</td></tr>
</table>

Аппаратный компонент		Объем хранимых данных			Примечание
					рекомендуется использовать в качестве конфигурационной базы RuBackup СУБД PostgreSQL в отказоустойчивой конфигурации, например, с использованием решения Patroni, развернутом на отдельностоящих машинах.
	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	• Рекомендуется в случае активного использования машины с основным сервером в качестве медиасервера, для возможности расширения дискового пространства под хранение резервных копий. • 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

- Astra Linux 1.6
- Debian 12
- Debian 10
- Ubuntu 22.04
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Red Hat Enterprise Linux 9
- Rosa Cobalt 7.3
- Rosa Cobalt 7.9
- Rosa Chrome 12
- Alt Linux 10
- РЕД ОС 7.3
- РЕД ОС 8

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

Аппаратный компонент		**Объем хранимых данных**			**Примечание**
		48 ТБ	**96 ТБ**	**144 ТБ**	
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	• Для возможности расширения дискового пространства под хранение резервных копий. • 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
- Astra Linux 1.6
- Debian 12
- Debian 10
- Ubuntu 22.04
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Red Hat Enterprise Linux 9

- Rosa Cobalt 7.3
- Rosa Cobalt 7.9
- Rosa Chrome 12
- Alt Linux 10
- РЕД ОС 7.3
- РЕД ОС 8

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 единовременно.

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 единовременно.

- **Восстановление данных:** объем свободного дискового пространства должен быть не менее совокупного объема единовременно 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:

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

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 * объем 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

- Astra Linux 1.6
- Debian 12
- Debian 10
- Ubuntu 22.04
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10
- Red Hat Enterprise Linux 9
- Rosa Cobalt 7.3
- Rosa Cobalt 7.9
- Rosa Chrome 12
- Alt Linux 10
- РЕД ОС 7.3
- РЕД ОС 8
- Windows Server 2012
- Windows Server 2016
- Windows Server 2019
- Windows Server 2022

Модули резервного копирования

Операционные системы

Платформы виртуализации

Модуль для ISPsystem VMmanager

- Astra Linux 1.7
- Astra Linux 1.6

- Ubuntu 20.04 (в экспериментальном режиме)
- Ubuntu 18.04 (в экспериментальном режиме)
- CentOS 8 (в экспериментальном режиме)
- CentOS 7 (в экспериментальном режиме)
- Alt Linux 10 (в экспериментальном режиме)
- РЕД ОС 7.3 (в экспериментальном режиме)

Модуль для ПК СВ "Брест"

- Astra Linux 1.7
- Astra Linux 1.6

Модуль для РУСТЭК

- Astra Linux 1.8 (в экспериментальном режиме)
- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 22.04 (в экспериментальном режиме)
- Ubuntu 20.04 (в экспериментальном режиме)
- Ubuntu 18.04 (в экспериментальном режиме)

Модуль для АЭРОДИСК VAIR

- Astra Linux 1.7
- Astra Linux 1.6
- Debian 10
- Ubuntu 20.04
- Ubuntu 18.04
- Alt Linux 10 (в экспериментальном режиме)

Модуль для VMware vSphere

- Ubuntu 20.04

Модуль для zVirt

- CentOS 8
- CentOS 7 (в экспериментальном режиме)
- Alt Linux 10 (в экспериментальном режиме)
- РЕД ОС 7.3 (в экспериментальном режиме)
- Rosa Cobalt 7.9 (в экспериментальном режиме)

Модуль для oVirt

- CentOS 8
- CentOS 7 (в экспериментальном режиме)
- Alt Linux 10 (в экспериментальном режиме)
- РЕД ОС 7.3 (в экспериментальном режиме)
- Rosa Cobalt 7.9 (в экспериментальном режиме)

Модуль для REDVirt

- CentOS 8
- CentOS 7 (в экспериментальном режиме)
- Alt Linux 10 (в экспериментальном режиме)
- РЕД ОС 7.3 (в экспериментальном режиме)
- Rosa Cobalt 7.9 (в экспериментальном режиме)

Модуль для P-Виртуализация

- CentOS 7

Модуль для OpenStack

- Astra Linux 1.8 (в экспериментальном режиме)
- Astra Linux 1.7
- Astra Linux 1.6
- Ubuntu 20.04 (в экспериментальном режиме)
- Ubuntu 18.04 (в экспериментальном режиме)
- РЕД ОС 7.3

Модуль для Tionix

- Astra Linux 1.7 (в экспериментальном режиме)
- Astra Linux 1.6 (в экспериментальном режиме)
- Debian 10 (в экспериментальном режиме)
- Ubuntu 20.04 (в экспериментальном режиме)
- Ubuntu 18.04 (в экспериментальном режиме)
- CentOS 8 (в экспериментальном режиме)
- CentOS 7 (в экспериментальном режиме)
- РЕД ОС 7.8 (в экспериментальном режиме)
- Alt Linux 10
- Rosa Cobalt 7.9 (в экспериментальном режиме)

Модуль для DynamiX

- Astra Linux 1.8 (в экспериментальном режиме)
- Astra Linux 1.7
- Astra Linux 1.6 (в экспериментальном режиме)
- Debian 10 (в экспериментальном режиме)
- Ubuntu 20.04 (в экспериментальном режиме)
- Ubuntu 18.04 (в экспериментальном режиме)
- Alt Linux 10 (в экспериментальном режиме)

- CentOS 8 (в экспериментальном режиме)
- CentOS 7 (в экспериментальном режиме)
- РЕД ОС 7.3 (в экспериментальном режиме)
- Rosa Cobalt 7.9 (в экспериментальном режиме)

Модуль для Greenpum

- Alt Linux 10 (в экспериментальном режиме)
- CentOS 8 (в экспериментальном режиме)
- CentOS 7 (в экспериментальном режиме)
- РЕД ОС 7.3 (в экспериментальном режиме)

Модуль для KVM

- Astra Linux 1.8 (в экспериментальном режиме)
- Astra Linux 1.7
- Astra Linux 1.6 (в экспериментальном режиме)
- Debian 10 (в экспериментальном режиме)
- Ubuntu 20.04 (в экспериментальном режиме)
- Ubuntu 18.04 (в экспериментальном режиме)
- Alt Linux 10 (в экспериментальном режиме)
- CentOS 8 (в экспериментальном режиме)
- CentOS 7 (в экспериментальном режиме)
- РЕД ОС 7.3 (в экспериментальном режиме)
- Rosa Cobalt 7.9 (в экспериментальном режиме)

Модуль для ECP VeiL

- Astra Linux 1.8 (в экспериментальном режиме)
- Astra Linux 1.7 (в экспериментальном режиме)
- Astra Linux 1.6 (в экспериментальном режиме)

- Debian 10 (в экспериментальном режиме)
- Ubuntu 20.04 (в экспериментальном режиме)
- Ubuntu 18.04 (в экспериментальном режиме)

Базы данных

Модуль для PostgreSQL и Patroni (Universal)

- Astra Linux 1.8
- Astra Linux 1.7
- Astra Linux 1.6
- Debian 10
- Ubuntu 22.04
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Alt Linux 10
- Red Hat Enterprise Linux 9
- РЕД ОС 7.3

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

- Astra Linux 1.8
- Astra Linux 1.7
- Astra Linux 1.6
- Debian 10
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8

- CentOS 7
- Alt Linux 10
- РЕД ОС 7.3

Модуль для Postgres Pro

- Astra Linux 1.8 (в экспериментальном режиме)
- 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

Бизнес-приложения

Модуль для CommuniGate Pro

- Astra Linux 1.8 (в экспериментальном режиме)
- 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 FreeIPA

- Astra Linux 1.8 (в экспериментальном режиме)

- 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 Mailion

- Astra Linux 1.7

Файловые системы и др.

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

- Astra Linux 1.8
- Astra Linux 1.7
- Astra Linux 1.6
- Debian 10
- Debian 12
- Ubuntu 22.04
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Red Hat Enterprise Linux 9
- Rosa Cobalt 7.3

- Rosa Cobalt 7.9
- Rosa Chrome 12
- Alt Linux 10
- РЕД ОС 7.3
- РЕД ОС 8

Модуль для LVM (входит в состав клиента резервного копирования)

- Astra Linux 1.8
- Astra Linux 1.7
- Astra Linux 1.6
- Debian 10
- Debian 12
- Ubuntu 22.04
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Red Hat Enterprise Linux 9
- Rosa Cobalt 7.3
- Rosa Cobalt 7.9
- Rosa Chrome 12
- Alt Linux 10
- РЕД ОС 7.3
- РЕД ОС 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)
- Astra Linux 1.8
- Astra Linux 1.7
- Astra Linux 1.6
- Debian 12
- Debian 10
- Ubuntu 22.04
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8
- CentOS 7
- Red Hat Enterprise Linux 9
- РЕД ОС 7.3
- РЕД ОС 8
- Rosa Cobalt 7.9
- Rosa Chrome 12
- Alt Linux 10

REST API

Операционные системы:

- Astra Linux 1.8
- Astra Linux 1.7

- Astra Linux 1.6
- Debian 12 (экспериментальный режим)
- Debian 10
- Ubuntu 22.04
- Ubuntu 20.04
- Ubuntu 18.04
- CentOS 8 (экспериментальный режим)
- CentOS 7 (экспериментальный режим)
- Alt Linux 10 (экспериментальный режим)
- Red Hat Enterprise Linux 9
- РЕД ОС 8 (экспериментальный режим)
- РЕД ОС 7.3 (экспериментальный режим)
- Rosa Cobalt 7.9 (экспериментальный режим)
- Rosa Cobalt 7.3 (экспериментальный режим)
- Rosa Chrome 12 (экспериментальный режим)

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	Медиасервер	rubackup-cmd	TCP	9991	Управление операциями на медиасервере

Компонент		Целевой сервис	Протокол	Порт	Описание
от	до				
		rubackup-media	TCP	9993	Управление операциями с данными
Основной сервер	База данных RuBackup на отдельностоящей 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 медиасерверами в составе основного и резервного серверов
Резервный сервер*	База данных RuBackup на отдельностоящей машине	postgresql	TCP	5432	Сохранение конфигурационной и оперативной информации
Медиасервер	Медиасервер	rubackup-media	TCP	9993	Передача данных между медиасерверами
Медиасервер	Резервный сервер*	rubackup-cmd	TCP	9991	Управление операциями на медиасервере
		rubackup-media	TCP	9993	Управление операциями с данными
Медиасервер	База данных RuBackup на отдельностоящей 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	Медиасервер	rubackup-media	TCP	9993	Передача данных между медиасервером и клиентом
Клиент резервного копирования	Резервный сервер*	rubackup-cmd	TCP	9991	Управление операциями на клиенте резервного копирования
		rubackup-media	TCP	9993	Передача данных между медиасервером и клиентом
RuBackup REST API	Основной сервер	rubackup-rbm	TCP	9995	Отправка запросов на сервер и получение информации

Компонент		Целевой сервис	Протокол	Порт	Описание
от	до				
RuBackup REST API***	База данных RuBackup на отдельностоящей 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 отдельно стоящей машине	База данных RuBackup на отдельностоящей машине	postgresql	TCP	5432**	Сохранение конфигурационной и оперативной информации
Менеджер RuBackup (RBM) на отдельно стоящей машине	Основной сервер	rubackup-rbm	TCP	9995	Управление операциями RuBackup
Менеджер RuBackup (RBM) на отдельно стоящей 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`**.

*** Порт для подключения, при необходимости, может быть изменен через переменные окружения в файле **`/opt/rubackup/etc/rubackup_api.env`** (см. в «Руководстве по установке и взаимодействию с программным интерфейсом RuBackup REST API»).

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_<version>_amd64.deb

или

$ sudo rpm -i rubackup-common_<version>.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_<version>_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_<version>.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.

Конфигурирование локали

При запуске элементов СРК RuBackup в ручном режиме через терминал возможно возникновение ошибки локали (ошибка может встретиться в процессе установки СРК и версии ALSE 1.8). Подобное связано с ошибочно сконфигурированной локалью.

Для проверки локали следует запустить команду и убедиться, что команда проходит без ошибок:

$ sudo locale

Если возникли ошибки после проверки статуса локали, то следует применить следующую команду для её реконфигурирования :

$ dpkg-reconfigure locales

Альтернативным вариантом является запуск с явным указанием локали. Например, для получения harware id для сервера команда может выглядеть следующим образом:

$ sudo LANG=C LC_ALL=C opt/rubackup/bin/rubackup_server hwid

Лицензирование СРК RuBackup

Для использования полного функционала системы резервного копирования и восстановления данных RuBackup требуется установить лицензионный файл для каждого развёрнутого серверного компонента — основного, резервного и медиасерверов.

Примечание. Лицензированию подлежит каждый сервер СРК RuBackup. Лицензирование клиентов СРК RuBackup не требуется.

Лицензионный договор (EULA) на право использования программного продукта СРК RuBackup находится в папке */opt/rubackup/copyrights/*, а также доступен для

ознакомления на официальном сайте https://www.rubackup.ru/. Используя программный продукт пользователь принимает условия лицензионного договора.

Типы лицензий

Лицензия на СРК RuBackup может быть нескольких типов, в зависимости от ограничений для лицензиата. Способы лицензирования системы резервного копирования RuBackup приведены в таблице 6.

Все серверные компоненты системы резервного копирования RuBackup подлежат единому типу лицензирования.

Таблица 6 — Типы лицензий СРК RuBackup

Параметр лицензирования / **Тип лицензии**	**Конфигурация**	**Объём резервируемых данных**	**Срок действия**	**Ограничение**
backend	Без ограничений	Суммарный объём всех хранимых резервных копий в системе СРК*	Бессрочная или срочная	При исчерпании объёма лицензии невозможно выполнить резервное копирование, но восстановление данных доступно. Минимальная лицензия — 1 ТБ
frontend	Без ограничений	Суммарный объём полных уникальных резервных копий источников данных**	Бессрочная или срочная	Учитывается только наибольшая резервная копия клиента СРК RuBackup. Минимальная лицензия — 1 ТБ
По конфигурации	Количество клиентов системы резервного копирования, количество сокетов сервера***	Максимальный объём хранимых резервных копий 250 ТБ*	Бессрочная или срочная	Минимальная конфигурация: 1 сервер и 10 клиентов. Для каждого клиента (не зависимо от конфигурации) доступно резервное копирование файловой системы и LVM-томов

Параметр лицензирования Тип лицензии	Конфигурация	Объём резервируемых данных	Срок действия	Ограничение
backend тестовая	1 сервер	1 ТБ	1 год	Получение автоматическое при запуске основного сервера
Временная	По запросу	По запросу	По запросу	Предоставляется по запросу
* учитывается объём всех резервных копий после сжатия и дедупликации, объём хранимых метаданных; ** учитывается объём резервных копий после сжатия, но до дедупликации, если она используется, также учитывается объём хранимых метаданных; *** учитываются только используемые (заполненные) сокеты				

Файл лицензии

- Файл лицензии имеет расширение *.lic* и должен находится в каталоге */opt/rubackup/etc/* с именем файла *rubackup.lic*.
- При запуске СРК RuBackup система программного лицензирования будет осуществлять поиск лицензии в каталоге */opt/rubackup/etc/*.
- Проверка файла лицензии осуществляется каждый час после запуска сервера по следующим параметрам:

– тип сервера СРК RuBackup: основной, резервный, медиа;

– идентификатор хоста лицензируемого сервера *hardware id*;

– в зависимости от типа лицензии:

– суммарный объём резервируемых данных;

– суммарный объём созданных полных резервных копий;

– срок действия;

– количество одномоментно подключенных клиентов резервного копирования.

Получение лицензионного файла

Для получения лицензионного файла сервера (основного, резервного и медиасерверов) у поставщика необходимо:

1. Полностью развернуть серверную группировку запланированной архитектуры системы резервного копирования RuBackup, установив пакеты серверной части СРК RuBackup на хостах.
2. Выполнить конфигурирование серверов, в результате получив конфигурационный файл */opt/rubackup/etc/config.file.*
3. Определить способ генерирования идентификатора *hardware id*, указав значение параметра *use_product_uuid* в конфигурационном файле */opt/rubackup/etc/config.file*.
4. На каждом сервере получить идентификатор *hardware id*, выполнив команду:

rubackup_server hwid

Зафиксировать любым удобным способом для какого типа сервера (основной, резервный, медиа) получен идентификатор.

5. Предоставить поставщику полученные идентификаторы удобным способом и получить лицензионные файлы для серверных компонентов СРК RuBackup на адрес электронной почты пользователя.

Установка лицензионного файла

Установите лицензионный файл на каждом хосте лицензируемого сервера СРК RuBackup.

Для установки лицензионного файла необходимо:

1. Привести имя полученного файла лицензии к виду *rubackup.lic*, выполнив команду, находясь в папке с файлом:

mv <old_filename.lic> rubackup.lic

где *<old_filename.lic>* - текущее имя файла лицензионного ключа.

2. Переместить или заменить[1] (в случае обновления лицензии) файл лицензии в папке */opt/rubackup/etc/*, выполнив команду, находясь в папке с подготовленным файлом лицензионного ключа:

cp rubackup.lic /opt/rubackup/etc/rubackup.lic

Примечание. Активация лицензии произойдёт после запуска сервера.

3. Опционально: при выполнении первичного лицензирования или обновления лицензии в следствии изменения архитектуры СРК RuBackup требуется выполнить конфигурирование каждого компонента с помощью утилиты *rb_init*, соблюдая порядок:

- основной сервер;

1 Рекомендуется сохранить существующий файл лицензии

- резервный сервер;
- медиасервера;
- клиенты системы резервного копирования.

В случае обновления лицензии в следствии изменения срока действия или объёма резервируемых данных конфигурирование компонентов СРК RuBackup не требуется.

4. Опционально: при обновлении лицензии, произведите перезапуск сервера, выполнив в терминале команду:

- если сервер ранее запущен в ручном режиме через терминал:

```
sudo rubackup_server restart
```

- если сервер ранее запущен как сервис:

```
sudo systemctl restart rubackup_server
```

Обновление лицензионного файла

- Обновление лицензионного файла необходимо в случае:

– изменения идентификатора хоста лицензируемого сервера *hardware id*;

– окончание лицензии (по какому-либо параметру лицензирования в зависимости от типа лицензии);

– изменения существующей архитектуры СРК RuBackup, например, установки модулей для расширения возможностей резервного копирования и восстановления данных, при использовании типа лицензии «по конфигурации».

- Для обновления лицензионного файла:

– получите обновлённую лицензию СРК RuBackup у поставщика в соответствии с подразделом «Получение лицензионного файла»;

– произведите обновление установленного лицензионного файла на сервере RuBackup в соответствии с подразделом настоящего документа.

Получение сведений о лицензии

Сведения об установленной лицензии доступны для просмотра в журнале событий на хосте лицензированного сервера и в Менеджере администратора RuBackup.

Просмотр сведений о лицензии в журнале событий

Для просмотра сведений о лицензии в журнале событий RuBackup.log:

1. Добавьте сведения об установленной лицензии на хосте лицензированного сервера СРК RuBackup (после его запуска) в журнал событий, выполнив команду в терминале:

rubackup_server license

Команда добавляет в журнал событий */opt/rubackup/log/RuBackup.log* данные об установленной на сервере лицензии.

2. Для просмотра сведений о лицензии в журнале событий, например, выполните, команду в терминале:

sudo tail -f /opt/rubackup/log/RuBackup.log

В терминале будет выведена следующая информация о лицензии:

- имя хоста, на котором развёрнут сервер, и его описание;
- роль сервера (основной, резервный, медиа);
- идентификатор хоста лицензированного сервера *hwid*;
- дату начала действия лицензии;
- дату окончания действия лицензии;
- тип лицензии;
- максимальный размер резервируемых данных;
- размер использованных резервируемых данных.

Просмотр сведений о лицензии в Менеджере администратора RuBackup

Для просмотра сведений об установленных на серверах СРК RuBakup лицензиях:

1. Запустите Менеджер администратора RuBackup (RBM).
2. Выполните авторизацию пользователя.
3. В верхней панели RBM нажмите кнопку Настройка  и в выпадающем меню (рисунок 1) выберите пункт «Лицензия».

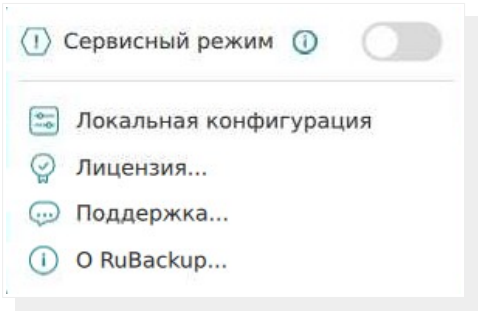


Рисунок 1 — Окно RBM «Настройки»

4. В открывшемся окне «Лицензии» (рисунок 2) приведены сведения об установленных текущих лицензиях серверной части СРК RuBackup, данные будут выведены в соответствии с типом лицензии:

- имя хоста, на котором развёрнут лицензируемый сервер;
- описание хоста, на котором развёрнут лицензируемый сервер;
- тип узла — тип лицензируемого сервера (основной, резервный или медиасервер);
- тип лицензии — возможные значения: backend, frontend, configuration (см. таблицу 6);
- ёмкость — максимальный размер резервируемых данных (ТБ);
- использованная ёмкость — размер использованных резервированных данных (байт);
- дата начала лицензии — дата установки и запуска лицензируемого серверва в формате YYYY.MM.DD, с представлением времени в 24-часовой нотации hh:mm;
- дата окончания действия лицензии — дата аннулирования лицензии и прекращения доступа к функии резервного копирования данных (функция восстановления данных из ранее сделанных резервных копий доступна) в формате YYYY.MM.DD, с представлением времени в 24-часовой нотации hh:mm;
- заказчик, по запросу которого предоставлена лицензия;
- сокеты — количество лицензируемых разъёмов на материнской плате сервера;
- клиенты СРК RuBackup;
- HWID — идентификатор хоста, на котором развёрнут лицензируемый сервер.

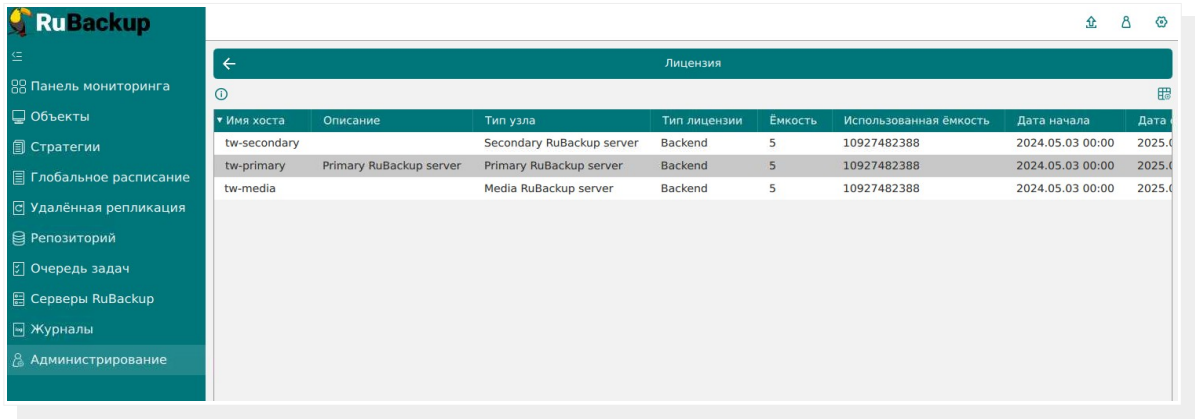


Рисунок 2 — Карточка установленных лицензий СРК RuBackup

5. Для лицензии типа «по конфигурации» возможен просмотр установленных расширений: по двойному нажатию ЛКМ на лицензию или выделив лицензию и нажав появившуюся кнопку ⓘ. В окне «Расширения лицензии» будут выведены все расширения, определяющие, какие именно источники данных можно использовать для создания резервных копий, поддерживаемые соответствующими модулями СРК RuBackup.

Генерирование hardware id

Идентификатор хоста лицензируемого сервера *hardware id* генерируется на основании данных, приведённых в таблице 7.

Таблица 7 — Условия формирования идентификатора hardware id

Версия RuBackup	ОС	Данные для формирования hardware id	Параметр config.file	Значение по умолчанию
Установка версии 2.1 и более поздняя	Linux	данные псевдо-файла *sys/class/dmi/id/product_uuid*, содержащего идентификатор *UUID* материнской платы, установленный производителем платы, и закодированной информации в DMI BIOS	use_product_uuid	true
	Windows	имя хоста *hostname*;		
Установлена версия ранее 2.1	Linux	идентификатор */etc/machine-id* и имя хоста */etc/hostname*	нет	нет

Версия RuBackup	ОС	Данные для формирования hardware id	Параметр config.file	Значение по умолчанию
Обновление установленной версии ранее 2.1			use_product_uuid	false

Уведомление о наступлении ограничения лицензии

- Чтобы обеспечить бесперебойную работу СРК RuBackup, действие лицензий рекомендуется продлевать до истечения параметров лицензирования.
- В случае срочной лицензии система уведомляет клиента об окончании срока действия лицензии не позднее, чем за 45 дней при запуске сервера — в терминале будет выведено предупреждение об истечении срока действия лицензии.
- Также актуальность лицензии всегда можно проверить в консоли Менеджера администратора RuBackup или вывести сведения о текущей лицензии в терминал.

Дистрибутивы установочных пакетов

Для развёртывания компонентов Системы резервного копирования и восстановления данных RuBackup получить актуальные установочные deb/rpm пакеты возможно одним из способов:

• на компакт-диске, полученном от изготовителя;

• из дополнительно подключаемого, публичного репозитория. Сведения по подключению репозитория и установке из него приведены в «Приложение А» настоящего документа;

• скачав актуальные установочные пакеты СРК RuBackup из облачного диска Астры на официальном сайте компании https://disk.astralinux.ru/s/y3Xg57z3JyYtNbg. На диске вы найдёте:

• папки с названиями операционных систем, содержащие совместимые с указанной ОС установочные пакеты для развёртывания компонентов СРК RuBackup:

• Alt Linux 10;

• Astra Linux 1.6;

• Astra Linux 1.7;

- Astra Linux 1.8;
- CentOS 7;
- CentOS 8;
- Debian 10;
- Debian 12;
- RedOS 7.3;
- RedOS 8;
- RHEL 9;
- Rosa Chrome 12;
- Rosa Cobalt 7.3;
- Rosa Cobalt 7.9;
- Ubuntu 18.04;
- Ubuntu 20.04;
- Ubuntu 22.04;

- папку «Experimental», содержащую папки с названиями операционных систем, содержащие совместимые с указанной ОС экспериментальные установочные пакеты для развёртывания компонентов СРК RuBackup, прошедшие только дизайн-тестирование, и папку, содержащую экспериментальные скрипты:

- Alt Linux 10;
- Astra Linux 1.6;
- Astra Linux 1.7;
- Astra Linux 1.8;
- CentOS 7;
- CentOS 8;
- Debian 10;
- Debian 12;
- RedOS 7.3;
- RedOS 8;
- RHEL 9;

• Rosa Chrome 12;

• Rosa Cobalt 7.3;

• Rosa Cobalt 7.9;

• Ubuntu 18.04;

• Ubuntu 20.04;

• Ubuntu 22.04;

• WinOS — папка содержит экспериментальную версию Менеджера администратора RuBackup для ОС MS Windows;

• Scripts — содержит экспериментальные версии скриптов :

• script_block_device_metadata.sh скрипт резервного копирования метаданных дедуплицированного пула;

• upgrade_rubackup_packages.sh скрипт автоматического обновления;

Компания «Рубэкап» будет признательна за обратную связь по работе экспериментальных версий пакетов компонентов СРК RuBackup любым удобным для вас способом — обратившись в службу технической поддержки https://support.rubackup.ru или пользовательскую группу в Telegram https://t.me/rubackup_user_group. Вы помогаете нам становиться лучше!

• папку «Prev_Version», содержащую установочный пакет модуля резервного копирования и восстановления данных кластеров СУБД PostfreSQL для поддержки нового функционала серверной и клиентской группировок релиза 2.1;

• RB_key.iso — специализированный загрузочный образ RuBackup.

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.

4. Параметр default_transaction_isolation должен принимать значение «read committed».

После внесения этих изменений необходимо:

1. Перезагрузить сервис postgresql:

$ sudo service postgresql restart

2. Проверить подключение к СУБД:

$ sudo -u postgres psql

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.

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

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

Настройка SSL соединений

Для повышения безопасности сервера базы данных возможно использование надежного шифрования соединений с базой данных.

Для настройки SSL соединений:

1. Создайте сертификаты для сервера PostgreSQL и его клиентов (postgres-клиентов) (см. пункт настоящего документа).
2. Выполните настройку конфигурационных файлов на сервере PostgreSQL (см. пункт настоящего документа).
3. После установки пакетов компонентов СРК выполните настройку SSL соединений для postgres-клиентов (на хостах, где развёрнуты компоненты СРК), предварительно добавив полученные сертификаты (см. пункты и настоящего документа).

Создание сертификатов

Аутентификация клиента по сертификату позволяет серверу проверить личность подключающегося, подтверждая, что сертификат X.509, представленный клиентом, подписан доверенным центром сертификации (CA).

Сертификаты SSL проверяются и выдаются Центром сертификации.

Если вы не имеете PKI инфраструктуры открытых ключей, то на отдельном хосте, который может выполнять роль Центра сертификации:

1. Создайте директории, в которую будут сгенерированы сертификаты Центра сертификации, сервера PostgreSQL и для всех postgres-клиентов (в зависимости от архитектуры вашей СРК):

- Центра сертификации (*ca*);
- сервера PostgreSQL (*pg-server*);
- основного сервера RuBackup (*rb-server*);
- медиасервера (*rb-media*);
- АРМ администратора, если Менеджер администратора RuBackup (RBM) развёрнут на отдельном хосте (*rb-rbm*),

выполнив команду:

mkdir certs && cd certs && mkdir ca pg-server rb-server rb-media rb-rbm

2. Создайте закрытый ключ Центра сертификации, для этого:

- Перейдите в ранее созданную папку:

cd ca

- Сгенерируйте закрытый ключ для CA (*ca.key*), выполнив команду, например:

openssl genrsa -out ca.key 2048

- Создайте самоподписанный сертификат Центра сертификации (*ca.crt*) сроком действия 1 год, выполнив команду:

openssl req -new -x509 -days 365 -key ca.key -out ca.crt

где CN — это полное имя хоста (FQDN), на котором развёрнут CA.

3. Выпустите сертификат и закрытый ключ для сервера PostgreSQL, для этого:

- Перейдите в ранее созданную папку, выполнив команду:

cd pg-server

- Сгенерируйте закрытый ключ для сервера PostgreSQL */pg-server/server.key*, выполнив команду:

openssl genrsa -out server.key 2048

- Сгенерируйте запрос на сертификат сервера PostgreSQL */pg-server/server.csr*, выполнив команду:

openssl req -new -key server.key -out server.csr

где CN — это полное имя хоста (FQDN), на котором развёрнут сервер PostgreSQL.

- Подпишите запрос на сертификат сервера PostgreSQL закрытым ключом Центра сертификации, выполнив команду:

openssl x509 -req -in server.csr -CA ../ca/ca.crt -CAkey ../ca/ca.key -CAcreateserial -out server.crt -days 365

4. Повторите шаг 3 для каждого postgres-клиента, сгенерировав закрытый ключ (postgresql.key) и выпустив сертификат (postgresql.crt) для всех postgres-клиентов, указав в сертификате соответствующее FQDN хоста, на котором развёрнут компонент СРК.

Настройка SSL соединения на сервере PostgreSQL

Выполните приведённые ниже настройки, чтобы сервер PostgreSQL прослушивал как обычные, так и SSL соединения через один и тот же TCP-порт и согласовывал использование SSL с любым подключающимся postgres-клиентом.

1. Скопируйте , например, в папку */etc/postgresql/16/main* на сервер PostgreSQL из папки */pg-server* Центра сертификации подготовленные:

- сертификат Центра сертификации (*ca.crt*);

- подписанный сертификат сервера PostgreSQL (*server.crt*);
- сгенерированный закрытый ключ сервера PostgreSQL (*server.key*).

2. Для файлов сертификата и закрытого ключа установите полный доступ на чтение и запись только для владельцев, выполнив команду, например:

chmod 600 server.crt server.key ca.crt

Сделайте владельцем файлов пользователя и группу пользователя *postgres*, выполнив команду:

chown postgres:postgres server.crt server.key ca.crt

3. Отредактируйте конфигурационный файл *postgresql.conf*:

- включите поддержку зашифрованных соединений:

ssl = on

- укажите путь к файлу сертификата Центра сертификации (или цепочке сертификатов):

ssl_ca_file = '/etc/postgresql/16/main/ca.crt'

Сертификат CA проверяет, что сертификат postgres-клиента подписан доверенным центром сертификации.

- укажите путь к файлу сертификата сервера PostgreSQL:

ssl_cert_file = '/etc/postgresql/16/main/server.crt'

Сертификат будет отправлен postgres-клиенту для указания подлинности сервера PostgreSQL.

- укажите путь к файлу закрытого ключа сервера PostgreSQL:

ssl_key_file = '/etc/postgresql/16/main/server.key'

Закрытый ключ доказывает, что сертификат сервера PostgreSQL был отправлен владельцем; не указывает, что владелец сертификата заслуживает доверия.

4. Чтобы потребовать от postgres-клиента предоставления доверенного сертификата, отредактируйте конфигурационный файл *pg_hba.conf*:

- добавьте опцию аутентификации *clientcert=verify-ca* или *clientcert=verify-full* в соответствующие *hostssl* строки, где:

– *clientcert=verify-full* сервер PostgreSQL не только проверяет цепочку сертификатов, но также проверяет, совпадает ли имя пользователя или его сопоставление с *CN* предоставленного сертификата;

– *clientcert=verify-ca* сервер проверяет, что сертификат postgres-клиента подписан одним из доверенных центров сертификации.

Также желательно закомментировать все строчки *host*, например:

#host all all 0.0.0.0/0 md5

hostssl all all 0.0.0.0/0 [md5,cert] clientcert=[verify-ca,verify-full] (в старых версиях [0,1])

где:

md5 — запросить пароль пользователя,

cert — аутентификация по сертификату.

Если параметр *clientcert* не указан, сервер проверяет сертификат postgres-клиента по своему файлу CA, только если сертификат postgres-клиента представлен и CA настроен.

5. Произведите настройку карты имён пользователей.

При использовании внешней системы аутентификации, такой как Ident, имя пользователя операционной системы, инициировавшего подключение, может не совпадать с именем пользователя базы данных (роли), который должен использоваться. В этом случае карта имен пользователей может быть применена для сопоставления имени пользователя операционной системы с именем пользователя базы данных

Чтобы использовать сопоставление имен пользователей, отредактируйте:

- конфигурационный файл *pg_hba.conf* — укажите в значении параметра *map=map-name* :

  hostssl all all 0.0.0.0/0 md5 clientcert=verify-full map=sslmap

- конфигурационный файл *pg_ident.conf*, хранящийся в каталоге данных кластера — настройте карты имен пользователей, добавьте, например:

  # MAPNAME SYSTEM-USERNAME PG-USERNAME

  sslmap postgres postgres

  sslmap postgres rubackup

  где:

– в столбце «*SYSTEM-USERNAME*» указываем CN сертификата postgres-клиента;

– в столбце «*PG-USERNAME*» указываем имя пользователя, с которым нужно сопоставить.

6. Для применения изменений, перезапустите сервер , выполнив команду:

sudo systemctl restart postgresql

Настройка SSL соединения на сервере/клиенте СРК

Для подключения к базе данных PostgreSQL через защищённое соединение выполните приведённые ниже настройки на хостах, на которых развёрнуты компоненты СРК (postgres-клиенты):

1. Перенесите из соответствующей postgres-клиенту папки на хосте Центра сертификации подготовленные:

- сертификат Центра сертификации (ca.crt), чтобы клиент СРК мог проверить, что конечный сертификат сервера PostgreSQL был подписан его доверенным корневым сертификатом;
- сертификат сервера/клиента СРК (postgresql.crt);
- сгенерированный закрытый ключ сервера/клиента СРК (postgresql.key).

2. Для файлов сертификата и закрытого ключа установите полный доступ на чтение и запись только для владельцев, выполнив команду, например:

chmod 600 server.crt server.key ca.crt

3. Сделайте владельцем файлов пользователя, от имени которого будет запущен компонент СРК (postgres-клиент), выполнив команду:

chown suser:suser server.crt server.key ca.crt

4. Выполните конфигурирование, запустив утилиту *rb_init* после установки пакетов компонента СРК и настройте защищенное SSL-соединение с сервером PostgreSQL.

   Если ранее сервер/клиент СРК был сконфигурирован, то установите необходимый режим работы SSL, отредактировав значение параметра *SSLMode=verify-full*, указанное на шаге 4 пункта , в конфигурационном файле */opt/rubackup/etc/config.file*.

5. Для применения изменений перезапустите сервер/клиент СРК, выполнив команду соответственно настраиваемому компоненту СРК:

Сервер/ медиасервера RuBackup	**sudo systemctl restart rubackup_server**
Клиент СРК	**sudo systemctl restart rubackup_client**

6. Выполните проверку сертификата:

openssl verify -verbose -CAfile RootCert.pem Intermediate.pem

Настройка SSL соединения на отдельном хосте Менеджера администратора RuBackup

Для подключения к базе PostgreSQL данных через защищённое соединение выполните приведённые ниже настройки на текущем хосте:

1. Перенесите из соответствующей postgres-клиенту папки на хосте Центра сертификации подготовленные:

- сертификат Центра сертификации (ca.crt), чтобы клиент мог проверить, что конечный сертификат сервера PostgreSQL был подписан его доверенным корневым сертификатом;
- сертификат клиента (postgresql.crt);
- сгенерированный закрытый ключ клиента (postgresql.key).

2. Разместите сертификаты и закрытый ключ в каталоге по умолчанию:

Для ОС Linux	~/.postgresql/
Для ОС Windows	%appdata%\postgresql\

3. Для файлов сертификата и закрытого ключа установите полный доступ на чтение и запись только для владельцев, выполнив команду, например:

chmod 600 server.crt server.key ca.crt

4. Для файлов сертификата и закрытого ключа сделайте владельцем файлов пользователя, от имени которого будет запущен компонент СРК (клиент PostgreSQL), выполнив команду:

chown suser:suser server.crt server.key ca.crt

5. После установки пакетов RBM выполните настройку параметра *SSLMode* в конфигурационном файле *~/.rbm2/.rb_gui_main_settings* или в графической утилите RBM в окне «Настройки — Локальная конфигурация» параметр *Режим SSL соединения с PostgreSQL*, установив значение, указанное на шаге 4.

6. Для применения изменений перезапустите настраиваемый клиент, выполнив команду:

opt/rubackup/bin/rbm

7. Выполните проверку сертификата:

openssl verify -verbose -CAfile RootCert.pem Intermediate.pem

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

Перед установкой "Все в одном" необходимо, чтобы в системе были установлены зависимости пакетов Linux (см. Приложение Б).

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».

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_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-common-gui_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-client_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-rbc_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-server_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-rbm_<version>_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:
5253096d055899485ed2787eccfc57ae54ff04e76104856726c913732aa0c2b8
```

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 визуальноый 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**в терминале (процедура настройки приведена ниже) или с помощью мастера настройки RuBackup **rb_init_gui** (процедура настройки приведена в разделе Конфигурирование (или обновление) сервера/клиента резервного копирования RuBackup).

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
```

2. Для продолжения выполнения конфигурирования СРК RuBackup необходимо принять лицензионное соглашение:

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

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. Укажите, необходимо ли использовать защищенное SSL-соединение с базой данных СРК «RuBackup»:

Do you want to use a secure SSL connection to the database 'rubackup' (y/n/q)?

Далее выберите и введите название выбранного режима SSL в соответствии с таблицей 8. По умолчанию выбран режим *require*.

Таблица 8 — *Описание режимов SSL*

sslmode	Защита от прослушивания	Защита от MITM	Утверждение
disable	Нет	Нет	Мне не важна безопасность и я не приемлю издержки, связанные с шифрованием.
allow	Возможно	Нет	Мне не важна безопасность, но я приемлю издержки, связанные с шифрованием, если на этом настаивает сервер
prefer	Возможно	Нет	Мне не важна безопасность, но я предпочитаю шифрование (и приемлю связанные издержки), если это поддерживает сервер
require	Да	Нет	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Я доверяю сети в том, что она обеспечивает подключение к нужному серверу
verify-ca	Да	Зависит от политики ЦС	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу
verify-full	Да	Да	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу и это именно указанный мной сервер

Укажите расположение подготовленных сертификатов:

- в поле *sslrootcert* — укажите расположение сертификата корневого центра сертификации;

- в поле *sslcert* — укажите расположение сертификата основного сервера;
- в поле *sslkey* — укажите расположение закрытого ключа основного сервера.

Если настройка SSL-соединения с БД не требуется, нажмите клавишу <n>. По умолчанию подключение будет установлено с параметром *sslmode=allow*, в этом случае для подключения к БД будут использованы файлы сертификатов и закрытых ключей, которые расположены в папке */opt/rubackup/keys*, При подключении к БД данные будут шифроваться.

Если в конфигурации postgresql SSL выключен, то по умолчанию *sslmode* будет *disable*.

7. 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 ]:
   ```

8. Создайте суперпользователя базы данных и задайте пароль для суперпользователя базы данных (по умолчанию — rubackup):

   ```
   Database user "rubackup" doesn't exist. Do you want to create database
   user "rubackup" (y/n)?
   Please enter password for "rubackup" database user:
   Repeat password:
   ```

9. 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 ]:
```

10. Подтвердите создание служебной базы данных:

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

11. Добавьте локальное файловое хранилище для пула по умолчанию:

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

12. Введите путь к директории, которая будет ассоциирована с пулом по умолчанию. Если указанной директории не существует, то она будет создана:

Enter path:

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

Will you use secondary server (y/n)?

14. 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:

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

Do you allow centralized recovery (y/n)?

16. 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)?

17. Укажите директорию для временных операций с файлами резервных копий (по умолчанию при нажатии клавиши Enter в качестве директории для временных операций с файлами резервных копий используется /tmp). Если указанная директория не существует, то далее будет предложено её создать:

Enter local backup directory path [/tmp] :

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

18. 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]:

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

Set amount threads parallelizm media server [8]:

20. Автоматическое создание мастер-ключа:

Create RuBackup master key...

21. Укажите, хотите ли вы использовать ключи электронно-цифровой подписи:

Will you use digital signature (y/n)?

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 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)?

24. Если используется ограничение верхнего предела оперативной памяти, то 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):

25. Выберите какие публичные имена будут использованы DNS-сервером:
Do you want to use ipv4[1] ipv6[2] or both[3] in DNS requests?

26. Укажите, хотите ли вы включить аудит безопасности:

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

27. Укажите, какой тип аудита вы хотите включить:

- essential only — журналирование всех значимых таблиц, кроме очередей задач и временных таблиц;

• tasks (additionally to essential) — журналирование всех значимых таблиц и задач в очередях.

Позднее возможно включить/отключить данную опцию и изменить выбранный тип аудита с помощью утилиты для работы с журналом событий информационной безопасности rb_security.

```
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.).**

28. Далее получите лицензионный файл у поставщика и произведите установку лицензии в соответствии с подразделом Лицензирование СРК RuBackup.

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

Уточнить статус сервера RuBackup можно при помощи команды:

$ sudo systemctl status rubackup_server

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 серевер 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 выключеном «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: $100+МС*2+КЛ*2+КЛ*2$
- Если сетевой параллелизм включён: $100+МС*2+КЛ*2+КЛ*N$

Где:

- МС — число медиасерверов.
- КЛ — 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 нобходимо 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 нобходимо 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

Перед установкой сервера RuBackup необходимо, чтобы в системе были установлены зависимости пакетов Linux (см. Приложение Б).

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».

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_<version>_amd64.deb
```

6. Установите пакет **rubackup-common-gui**. Пример:

```
$ sudo dpkg -i rubackup-common-gui_<version>_amd64.deb
```

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

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

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

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

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

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

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

```
$ sudo dpkg -i rubackup-rbm_<version>_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
```

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** в терминале (процедура настройки приведена ниже) или с помощью мастера настройки RuBackup **rb_init_gui** (процедура настройки приведена в разделе Конфигурирование (или обновление) сервера/клиента резервного копирования RuBackup).

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
```

2. Для продолжения конфигурирования клиента резервного копирования примите лицензионное соглашение:

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

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)?
```

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. Укажите, необходимо ли использовать защищенное SSL-соединение с базой данных СРК «RuBackup»:

Do you want to use a secure SSL connection to the database 'rubackup' (y/n/q)?

Далее выберите и введите название выбранного режима SSL в соответствии с таблицей 9. По умолчанию выбран режим *require*.

Таблица 9 — *Описание режимов SSL*

sslmode	Защита от прослуши-вания	Защита от MITM	Утверждение
disable	Нет	Нет	Мне не важна безопасность и я не приемлю издержки, связанные с шифрованием.
allow	Возможно	Нет	Мне не важна безопасность, но я приемлю издержки, связанные с шифрованием, если на этом настаивает сервер
prefer	Возможно	Нет	Мне не важна безопасность, но я предпочитаю шифрование (и приемлю связанные издержки), если это поддерживает сервер
require	Да	Нет	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Я доверяю сети в том, что она обеспечивает подключение к нужному серверу
verify-ca	Да	Зависит от политики ЦС	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу
verify-full	Да	Да	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу и это именно указанный мной сервер

Укажите расположение подготовленных сертификатов:

- в поле *sslrootcert* — укажите расположение сертификата корневого центра сертификации;
- в поле *sslcert* — укажите расположение сертификата основного сервера;
- в поле *sslkey* — укажите расположение закрытого ключа основного сервера.

Если настройка SSL-соединения с БД не требуется, нажмите клавишу <n>. По умолчанию подключение будет установлено с параметром *sslmode=allow*, в этом случае

для подключения к БД будут использованы файлы сертификатов и закрытых ключей, которые расположены в папке */opt/rubackup/keys*, При подключении к БД данные будут шифроваться.

Если в конфигурации postgresql SSL выключен, то по умолчанию *sslmode* будет *disable*.

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

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

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)?
```

9. Добавьте локальное файловое хранилище для пула по умолчанию:

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

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

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

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:
```

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:

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

    Do you allow centralized recovery (y/n)?

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

16. Укажите директорию для временных операций с файлами резервных копий (по умолчанию при нажатии клавиши Enter в качестве директории для временных операций с файлами резервных копий используется /tmp). Если указанная директория не существует, то далее будет предложено её создать:

    Enter local backup directory path [/tmp] :

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 медиасервере (каждый поток имеет отдельное соединение со служебной базой данных СРК):

    Set amount threads parallelizm media server [8]:

19. Автоматическое создание мастер-ключа:

    Create RuBackup master key...

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

    Will you use digital signature (y/n)?

21. Укажите, хотите ли вы включить системный мониторинг для данного клиента:
    Do you want to enable system monitoring of this client (y/n)?

22. 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)?`

23. 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):`

24. Выберите какие публичные имена будут использованы DNS-сервером:
    Do you want to use ipv4[1] ipv6[2] or both[3] in DNS requests?

25. Укажите, хотите ли вы включить аудит безопасности:
    Do you want to enable RuBackup security audit ([y]es, [n]o, [q]uit)(y/n/q)?

26. Укажите, какой тип аудита вы хотите включить:

- essential only — журналирование всех значимых таблиц, кроме очередей задач и временных таблиц;
- tasks (additionally to essential) — журналирование всех значимых таблиц и задач в очередях.

Позднее возможно включить/отключить данную опцию и изменить выбранный тип аудита с помощью утилиты для работы с журналом событий информационной безопасности rb_security.

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

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.).**

27. Далее получите лицензионный файл у поставщика и произведите установку лицензии в соответствии с подразделом Лицензирование СРК RuBackup.

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

Уточнить статус сервера RuBackup можно при помощи команды:

$ sudo systemctl status rubackup_server

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

Перед установкой сервера RuBackup необходимо, чтобы в системе были установлены зависимости пакетов Linux (см. Приложение Б)

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».

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_<version>_amd64.deb

6. Установите пакет **rubackup-common-gui**. Пример:

   $ sudo dpkg -i rubackup-common-gui_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-client_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-rbc_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-server_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-rbm_<version>_amd64.deb

где <version> в имени 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

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** в терминале (процедура настройки приведена ниже) или с помощью мастера настройки RuBackup **rb_init_gui** (процедура настройки приведена в разделе Конфигурирование (или обновление) сервера/клиента резервного копирования RuBackup).

Выполните следующие действия:

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

root@ secondary-server~# **rb_init**

2. Для продолжения конфигурирования клиента резервного копирования примите лицензионное соглашение:

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

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)?
```

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]:

5. Укажите, необходимо ли использовать защищенное SSL-соединение с базой данных СРК «RuBackup»:

Do you want to use a secure SSL connection to the database 'rubackup' (y/n/q)?

Далее выберите и введите название выбранного режима SSL в соответствии с таблицей 10. По умолчанию выбран режим *require*.

Таблица 10 — *Описание режимов SSL*

sslmode	Защита от прослушивания	Защита от MITM	Утверждение
disable	Нет	Нет	Мне не важна безопасность и я не приемлю издержки, связанные с шифрованием
allow	Возможно	Нет	Мне не важна безопасность, но я приемлю издержки, связанные с шифрованием, если на этом настаивает сервер
prefer	Возможно	Нет	Мне не важна безопасность, но я предпочитаю шифрование (и приемлю связанные издержки), если это поддерживает сервер

sslmode	Защита от прослуши-вания	Защита от MITM	Утверждение
require	Да	Нет	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Я доверяю сети в том, что она обеспечивает подключение к нужному серверу
verify-ca	Да	Зависит от политики ЦС	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу
verify-full	Да	Да	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу и это именно указанный мной сервер

Укажите расположение подготовленных сертификатов:

- в поле *sslrootcert* — укажите расположение сертификата корневого центра сертификации;
- в поле *sslcert* — укажите расположение сертификата основного сервера;
- в поле *sslkey* — укажите расположение закрытого ключа основного сервера.

Если настройка SSL-соединения с БД не требуется, нажмите клавишу <n>. По умолчанию подключение будет установлено с параметром *sslmode=allow*, в этом случае для подключения к БД будут использованы файлы сертификатов и закрытых ключей, которые расположены в папке */opt/rubackup/keys*, При подключении к БД данные будут шифроваться.

Если в конфигурации postgresql SSL выключен, то по умолчанию *sslmode* будет *disable*.

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

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

7. Введите пароль для суперпользователя Rubackup:

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

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

Enter RuBackup database name [rubackup]:

9. Укажите имя основного сервера Rubackup:

   Hostname of primary server:

10. 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:

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

    Do you allow centralized recovery (y/n)?

12. 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)?

13. Укажите директорию для временных операций с файлами резервных копий (по умолчанию при нажатии клавиши Enter в качестве директории для временных операций с файлами резервных копий используется /tmp). Если указанная директория не существует, то далее будет предложено её создать:

    Enter local backup directory path [/tmp] :

14. Автоматическое создание мастер-ключа:

    Create RuBackup master key...

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

    Will you use digital signature (y/n)?

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

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

17. 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)?

18. 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):

19. Выберите какие публичные имена будут использованы DNS-сервером:

    Do you want to use ipv4[1] ipv6[2] or both[3] in DNS requests?

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.).**

20. Далее получите лицензионный файл у поставщика и произведите установку лицензии в соответствии с подразделом Лицензирование СРК RuBackup.

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

Уточнить статус сервера RuBackup можно при помощи команды:

$ sudo systemctl status rubackup_server

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

Перед установкой медиасервера RuBackup необходимо, чтобы в системе были установлены зависимости пакетов Linux (см. Приложение Б).

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».

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_<version>_amd64.deb

6. Установите пакет **rubackup-common-gui**. Пример:

$ sudo dpkg -i rubackup-common-gui_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-client_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-rbc_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-server_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-rbm_<version>_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

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** в терминале (процедура настройки приведена ниже) или с помощью мастера настройки RuBackup **rb_init_gui** (процедура настройки приведена в разделе Конфигурирование (или обновление) сервера/клиента резервного копирования RuBackup).

Выполните следующие действия:

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

   root@rubackup-media:~# **rb_init**

2. Для продолжения конфигурирования клиента резервного копирования примите лицензионное соглашение:

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

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 (по умолчанию при нажатии клавиши Enter в качестве адрес сервера используется localhost).

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

5. Укажите, необходимо ли использовать защищенное SSL-соединение с базой данных СРК «RuBackup»:

   Do you want to use a secure SSL connection to the database 'rubackup' (y/n/q)?

6. Далее выберите и введите название выбранного режима SSL в соответствии с таблицей 11. По умолчанию выбран режим *require*.

Таблица 11 — *Описание режимов SSL*

sslmode	Защита от прослушивания	Защита от MITM	Утверждение
disable	Нет	Нет	Мне не важна безопасность и я не приемлю издержки, связанные с шифрованием.
allow	Возможно	Нет	Мне не важна безопасность, но я приемлю издержки, связанные с шифрованием, если на этом настаивает сервер

sslmode	Защита от прослуши-вания	Защита от MITM	Утверждение
prefer	Возможно	Нет	Мне не важна безопасность, но я предпочитаю шифрование (и приемлю связанные издержки), если это поддерживает сервер
require	Да	Нет	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Я доверяю сети в том, что она обеспечивает подключение к нужному серверу
verify-ca	Да	Зависит от политики ЦС	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу
verify-full	Да	Да	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу и это именно указанный мной сервер

Укажите расположение подготовленных сертификатов:

- в поле *sslrootcert* — укажите расположение сертификата корневого центра сертификации;
- в поле *sslcert* — укажите расположение сертификата основного сервера;
- в поле *sslkey* — укажите расположение закрытого ключа основного сервера.

Если настройка SSL-соединения с БД не требуется, нажмите клавишу <n>. По умолчанию подключение будет установлено с параметром *sslmode=allow*, в этом случае для подключения к БД будут использованы файлы сертификатов и закрытых ключей, которые расположены в папке */opt/rubackup/keys*, При подключении к БД данные будут шифроваться.

Если в конфигурации postgresql SSL выключен, то по умолчанию *sslmode* будет *disable*.

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

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

8. Введите пароль для суперпользователя RuBackup:

Please enter password for "rubackup" database user:

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

Enter RuBackup database name [rubackup]:

10. Введите имя основного сервера Rubackup:

Hostname of primary server:

11. Укажите используете ли вы резервный сервер Rubackup:

Will you use secondary server:

12. Пр наличии резервного сервера Rubackup введите его имя:

Hostname of secondary server:

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

Choose client net interface ID for use:

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

Do you allow centralized recovery (y/n)?

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)?

16. Укажите директорию для временных операций с файлами резервных копий (по умолчанию при нажатии клавиши Enter в качестве директории для временных операций с файлами резервных копий используется /tmp). Если указанная директория не существует, то далее будет предложено её создать:

Enter local backup directory path [/tmp] :

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 медиасервере (каждый поток имеет отдельное соединение со служебной базой данных СРК):

    Set amount threads parallelizm media server [8]:

19. Автоматическое создание мастер-ключа:

    Create RuBackup master key...

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

    Will you use digital signature (y/n)?

21. Укажите, хотите ли вы включить системный мониторинг для данного сервера:

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

22. 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)?

23. 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):

24. Выберите какие публичные имена будут использованы DNS-сервером:

    `Do you want to use ipv4[1] ipv6[2] or both[3] in DNS requests?`

25. Далее получите лицензионный файл у поставщика и произведите установку лицензии в соответствии с подразделом Лицензирование СРК RuBackup.

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 медиасервере 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 медиасервера 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
```

Уточнить статус сервера RuBackup можно при помощи команды:

$ sudo systemctl status rubackup_server

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_<version>_amd64.deb

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

$ sudo dpkg -i rubackup-client_<version>_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/digsig/digsig_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** в терминале (процедура настройки приведена ниже) или с помощью мастера настройки RuBackup **rb_init_gui** (процедура настройки приведена в разделе Конфигурирование (или обновление) сервера/клиента резервного копирования RuBackup).

Для настройки выполните следующие действия:

1. Запустите rb_init (от пользователя root).

```
# rb_init
```

2. Для продолжения конфигурирования клиента резервного копирования примите лицензионное соглашение:

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

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
```

4. Укажите адрес основного (primary) сервера СРК:

```
Hostname of primary server:
```

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:
```

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

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

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

```
Do you allow centralized recovery (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)?
```

10. Укажите локальный каталог для временного хранения файлов с метаданными, создаваемых при операциях резервного копирования (по умолчанию при нажатии клавиши Enter в качестве директории для временных операций с файлами резервных копий используется /tmp). Если указанная директория не существует, то далее будет предложено её создать:

```
Enter local backup directory path [/tmp] :
```

11. Подтвердите создание каталога для временных файлов:

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

12. Автоматическое создание мастер-ключа:

Create RuBackup master key…

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

    Will you use digital signature (y/n)

14. Укажите, хотите ли вы включить системный мониторинг для данного клиента:
    Do you want to enable system monitoring of this client (y/n)?

15. 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)?

16. 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):

17. Выберите какие публичные имена будут использованы DNS-сервером:

    Do you want to use ipv4[1] ipv6[2] or both[3] in DNS requests?

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

Дополнительные настройки

Конфигурирование (или обновление) сервера/клиента резервного копирования RuBackup

Установка пакета мастера настройки RuBackup

Для конфигурирования сервера или клиента резервного копирования RuBackup с помощью графический интерфейса:

1. Предварительно установите обязательные пакеты сервера RuBackup (rubackup-common, rubackup-client, rubackup-server) или клиента резервного копирования (rubackup-common, rubackup-client), как описано в соответствующих разделах настоящего документа.
2. Установите пакет rubackup-common-gui для поддержки графических интерфейсов RuBackup, например, выполнив команду:

   $ sudo dpkg -i rubackup-common-gui_<version>_amd64.deb
3. Установите пакет графического приложения rubackup-init-gui, например, выполнив команду:

   $ sudo dpkg -i rubackup-init-gui_<version>_amd64.deb

Конфигурирование или обновление сервера/клиента резервного копирования RuBackup

Выполните первичное конфигурирование или обновление пакетов сервера (основного, медиа или резервного) или клиента резервного копирования RuBackup с помощью мастера настройки RuBackup. Для возврата на предыдущий шаг и редактирования выбора используйте кнопку возврата  .

1. Запустите мастер настройки RuBackup (графическое приложение rb_init), выполнив команду:

$ rb_init_gui&

2.	**Основной сервер**	После запуска мастера настройки RuBackup в приветственном окне (рисунок 3):
	Резервный	

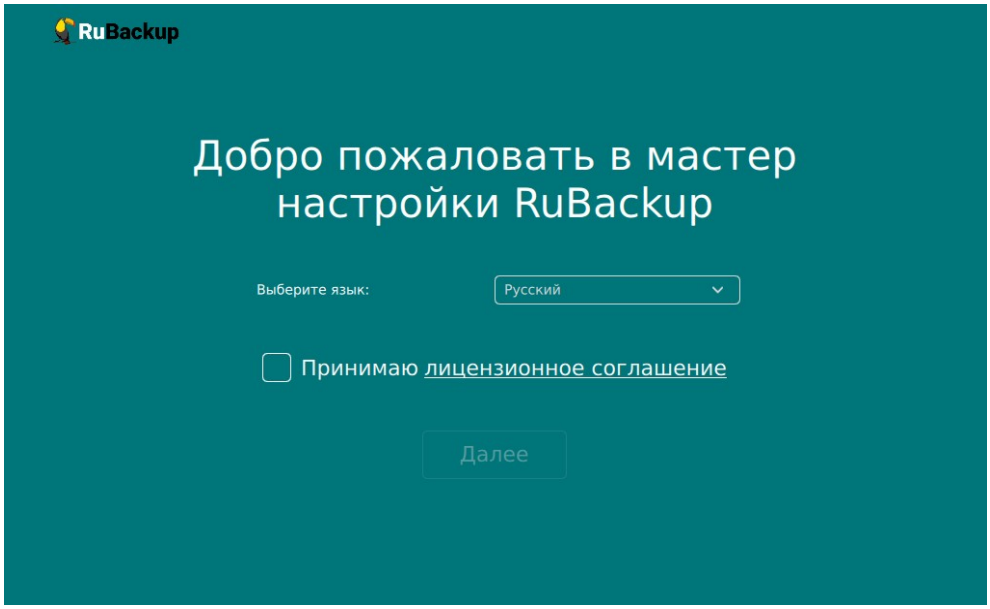
<table>
<tr><td rowspan="2"></td><td>сервер
Медиа-сервер
Клиент РК</td><td>- выберите язык интерфейса приложения из предложенных вариантов (русский или английский);
- примите лицензионное соглашения для продолжения настройки RuBackup, проставив отметку в чек-боксе ☑ Для ознакомления нажмите на активный элемент лицензионное соглашение и в открывшемся окне подтверждения скопируйте в буфер ссылку на лицензионное соглашения для дальнейшего просмотра в браузере;
- нажмите ставшую активной кнопку Далее.</td></tr>
<tr><td colspan="2"> 

Рисунок 3</td></tr>
</table>

<table>
<tr><td rowspan="3">3.</td><td>Основной сервер
Резервный сервер
Медиа-сервер</td><td>В открывшемся окне выберете настраиваемый компонент.
Если на настраиваемом узле установлен пакет rubackup-server, то мастер настройки автоматически предлагает произвести настройку серверного компонента (рисунок 4):
- основной сервер;
- резервный сервер;
- медиасервер.</td></tr>
</table>

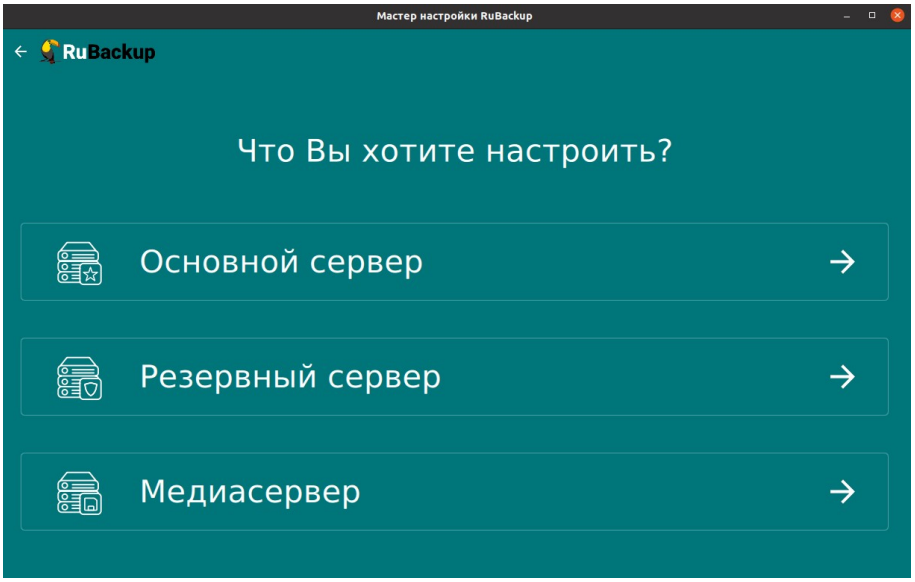


Рисунок 4

Клиент РК	Если на настраиваемом узле отсутствует установленный пакет rubackup-server, то мастер настройки автоматически предлагает произвести настройку клиентского компонента (рисунок 5): - в автономном режиме клиента (использования без серверной части СРК RuBackup. При этом сохраняется возможность использования любых функциональных модулей для создания резервных копий); - в клиент-серверном режиме.

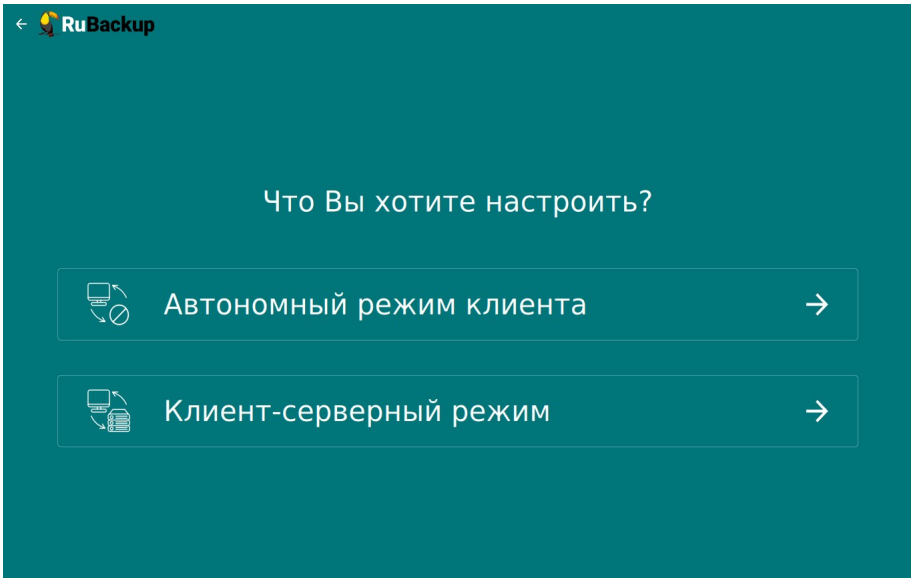


Рисунок 5

<table>
<tr><td>4.</td><td colspan="2">Заполните открывшуюся форму настраиваемого компонента СРК RuBackup.</td></tr>
<tr><td></td><td colspan="2">Блок «Общие параметры»</td></tr>
<tr><td></td><td>Основной сервер</td><td rowspan="3">В поле «Количество сетевых потоков» укажите количество потоков для одновременной обработки задач резервного копирования на основном сервере (каждый поток имеет отдельное соединение со служебной базой данных СРК)</td></tr>
<tr><td></td><td>Резервный сервер</td></tr>
<tr><td></td><td>Медиасервер</td></tr>
<tr><td></td><td>Основной сервер</td><td rowspan="3">В поле «Версия IP для DNS запросов» выберите какие публичные имена будут использованы DNS-сервером.</td></tr>
<tr><td></td><td>Резервный сервер</td></tr>
<tr><td></td><td>Медиасервер</td></tr>
<tr><td></td><td>Основной сервер</td><td rowspan="3">Активируйте переключатель «Перезапись мастер-ключа» для автоматического формирования нового мастер-ключа и перезаписи (при наличии) текущего мастер-ключа.</td></tr>
<tr><td></td><td>Резервный сервер</td></tr>
<tr><td></td><td>Медиасервер</td></tr>
<tr><td></td><td colspan="2">Блок «Параметры сервера»</td></tr>
<tr><td></td><td>Резервный сервер</td><td rowspan="2">В поле «Имя основного сервера» укажите ip-адрес или FQDN основного сервера RuBackup (в соответствии с настройками файла hosts узла основного сервера).</td></tr>
<tr><td></td><td>Медиасервер</td></tr>
<tr><td></td><td>Основной сервер</td><td rowspan="3">В поле «Адрес сервера PostgreSQL»* — укажите адрес, на котором развёрнута СУБД PostgreSQL:
• если СУБД PostgreSQL развёрнута на отдельном от основного сервера узле, то следует указать адрес соответствующего узла;
• если СУБД PostgreSQL и основной сервер развёрнуты на одном узле, то оставьте значение localhost, выбранное по умолчанию</td></tr>
<tr><td></td><td>Резервный сервер</td></tr>
<tr><td></td><td>Медиасервер</td></tr>
<tr><td></td><td>Основной сервер</td><td>В поле «Пароль PostgreSQL»* укажите пароль пользователя базы данных postgres</td></tr>
<tr><td></td><td>Основной сервер</td><td>В поле «Имя суперпользователя RuBackup» укажите имя суперпользователя базы данных rubackup (имя БД по умолчанию). Суперпользователь будет создан в процессе конфигурирования</td></tr>
</table>

		основного сервера.
	Основной сервер **Резервный сервер** **Медиасервер**	В поле «**Пароль пользователя RuBackup**»* укажите пароль для суперпользователя базы данных rubackup (имя БД по умолчанию).
	Основной сервер	В поле «**Имя базы RuBackup**» введите имя базы данных (по умолчанию в качестве имени базы данных используется «rubackup»), которая будет использоваться в качестве служебной БД или будет создана в случае её отсутствия. Внимание! В имени базы данных запрещено использовать следующие символы: пробел, \, $, #, `, /, ?, *, ., ,, ;, :, %, ^, &, <, >
	Основной сервер	При обновлении в поле «**Если база уже существует**» выберите действие с существующей базой данных: • keep — пропустить действие, База данных будет сохранена в текущем состоянии; • drop — удалить существующую базу данных; • upgrade — обновить существующую базу данных. При удалении и обновлении существующей базы данных по умолчанию будет сделана резервная копия данных, если переключатель «**Отключить дамп**» деактивирован, если активировать данный переключатель, то резервное копирование для текущей базы данных перед удалением/обновлением выполнено не будет. Если резервное копирование существующей базу данных будет выполнено, то в поле «**Формат дампа**» выберите тип резервной копии базы данных: • custom archives — custom-архив, восстановление выполняется с помощью pg_restore. Резервная копия в формате custom занимает меньше места на диске, по сравнению с форматом plain. Настройте «Уровень сжатия дампа»; • plain — текстовый sql-скрипт. Для типа резервной копии БД *custom archives* в поле «**Уровень сжатия дампа**» выберите степень сжатия резервной копии базы

		данных (значение от 0 до 9). Чем выше степень сжатия, тем меньше архив занимает места на диске и тем дольше выполняется процедура резервного копирования базы данных. В поле «**Путь к папке дампа**»* выберите путь для сохранения резервной копии - по умолчанию это директория, откуда была вызвана утилита.
	Основной сервер **Резервный сервер** **Медиасервер**	В поле «**Сетевой интерфейс**» выберите сетевой интерфейс, посредством которого клиенту RuBackup разрешено взаимодействовать с системой резервного копирования.
	Основной сервер	В поле «**Путь файловой системы для добавления в «Default»**»* необходимо назначить для пула Default хотя бы один каталог для хранения резервных копий.
	Основной сервер **Резервный сервер** **Медиасервер**	В поле «**Локальный каталог резервного копирования**» укажите локальный каталог для временного хранения файлов с метаданными, создаваемых при операциях резервного копирования (по умолчанию при нажатии клавиши Enter в качестве директории для временных операций с файлами резервных копий используется /tmp). Если указанная директория не существует, то будет создана.
	Основной сервер **Медиасервер**	В поле «**Имя резервного сервера**» укажите ip-адрес или FQDN основного сервера RuBackup (в соответствии с настройками файла hosts узла основного сервера).
	Основной сервер **Резервный сервер** **Медиасервер**	В поле «**Количество параллельных задач**» укажите количество потоков для одновременной обработки задач резервного копирования на медиасервере (каждый поток имеет отдельное соединение со служебной базой данных СРК).
	Основной сервер **Резервный сервер** **Медиасервер**	В поле «**Объём памяти дедупликации, байт**» для ограничения потребления оперативной памяти сервером при дедупликации резервных копий. При использовании дедупликации рекомендуется минимальный объем оперативной памяти сервера 64 GB effective_cache_size ~70 % от размера оперативной памяти work_mem 32 MB.
	Основной	Активируйте переключатель «**Непрерывная удалённая**

	сервер	**репликация»** при необходимости на клиенте. Непрерывная удалённая репликация осуществляется только в хранилище блочного типа.
	Резервный сервер	
	Медиасервер	
	Основной сервер	Активируйте переключатель «**Разрешать централизованное восстановление для клиента**» для восстановления данных из резервной копии с помощью утилиты «Менеджер администратора RuBackup» (RBM), с помощью консольной утилиты *rbfd* или утилиты «Менеджера клиента RuBackup» (RBC). В случае деактивированного переключателя восстановление из резервной копии будет возможно с помощью консольной утилиты *rbfd* или утилиты «Менеджера клиента RuBackup» на узле клиента резервного копирования. Централизованное восстановление данных из резервной копии с помощью утилиты «Менеджер администратора RuBackup» (используемой на любом узле) будет отключено.
	Резервный сервер	
	Медиасервер	
	Основной сервер	Активируйте переключатель «**Создать ключи ЭЦП**», если хотите создать ключи электронно-цифровой подписи. Резервная копия может быть подписана цифровой подписью для последующего контроля и предупреждения угрозы её подмены
	Резервный сервер	
	Медиасервер	
	Основной сервер	Активируйте переключатель «**Перезаписать ключи цифровой подписи**», для создания новой связки ключей, используемых для электронно-цифровой подиси.
	Резервный сервер	
	Медиасервер	
	Основной сервер	Активируйте переключатель «**Аудит безопасности**» для журналирования всех значимых таблиц, кроме очередей задач и временных таблиц; Для расширения регистрируемых событий активируйте переключатель «**Аудит задач**» для журналирования всех значимых таблиц и задач в очередях. Позднее возможно включить/отключить данную опцию и изменить выбранный тип аудита с помощью утилиты для работы с журналом событий информационной безопасности *rb_security*.
	Блок «Настройка SSL»	

	Основной сервер **Резервный сервер** **Медиасервер**	Активируйте переключатель «**Использовать SSL соединение с базой данных**» для настройки безопасного соединения со служебной базой данных RuBackup, и настройте ставшие активными параметры: в поле «**SSL режим работы с Postgres**» — выберите соответствующий режим работы (в зависимости от настроек узла, на котором установлена БД), подробное описание режимов смотри в подразделе «Настройка SSL соединений». Если в конфигурации PostgreSQL SSL выключен, то по умолчанию SSL режим будет *disable*; в поле «**Корневой сертификат**»* — укажите полный путь к сертификату доверенного Центра сертификации (прописав в поле или выбрав по нажатию рядом с полем кнопки ...), который необходимо заранее разместить в папке *opt/rubackup/keys*; в поле «**Сертификат клиента**»* — укажите полный путь к сертификату (открытому ключу) настраиваемого узла, выданный доверенным Центром сертификации (прописав в поле или выбрав по нажатию рядом с полем кнопки), который необходимо заранее разместить в папке opt/rubackup/keys; в поле «**Ключ клиента**»* — укажите полный путь к закрытому ключу сертификата настраиваемого узла, выданный доверенным Центром сертификации (прописав в поле или выбрав по нажатию рядом с полем кнопки), который необходимо заранее разместить в папке opt/rubackup/keys.
	Блок «Параметры автономного клиента»	
	Клиент автономный	В поле «Каталог архивирования»* выберите
	Клиент автономный	В поле «**Метод сжатия**» выберите тип сжатия резервных копий: • none — без сжатия; • fast — многопоточный аналог optimal. • optimal — стандартная утилита сжатия Linux; • best — больший коэффициент сжатия, чем optimal, при большем времени.

	Клиент автономный	В поле «**Тип хранилища резервных копий**» выберите тип каталога для хранения резервных копий: • локальный каталог — каталог расположен на текущем узле клиента резервного копирования. Если выбран этот тип хранилища, то в поле «**Локальный каталог резервного копирования**» укажите полный путь к каталогу (прописав в поле или выбрав по нажатию рядом с полем кнопки ...); • сетевой каталог — общий каталог с сетевым доступом. Если выбран этот тип хранилища, то необходимо: В поле «**Тип сетевого каталога**» выбрать протокол для обеспечения удалённой связи: **nfs** (для ОС UNIX и Linux) или **cifs** (для ОС Windows). В поле «**Предназначенное устройство**» укажите выделенное локальное устройство (например: /dev/sdb) или сетевой ресурс для хранения резервных копий (например: srv://net_share). В поле «**Параметры монтирования**» укажите место монтирования файловых системы LTFS. Для работы с лентами LTO RuBackup использует файловую систему LTFS. По умолчанию точка монтирования — каталог /opt/rubackup/mnt.
* отмечены обязательные для заполнения поля (если они активны)		

5. После заполнения всех полей формы настраиваемого компонента СРК RuBackup нажмите ставшую доступной кнопку Далее. В окне подтверждения для конфигурирования компонента СРК RuBackup подтвердите ваше действие, нажав кнопку «Да» (рисунок 6).

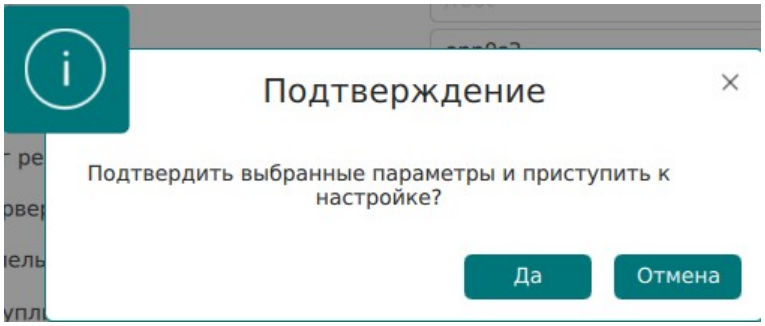


Рисунок 6

6. Далее, если в форме настраиваемого компонента СРК RuBackup указаны папки, которых не существует, то будет выведено подтверждение для их создания (рисунок 7). В окне подтверждения для конфигурирования компонента СРК RuBackup подтвердите ваше действие, нажав кнопку «Да».

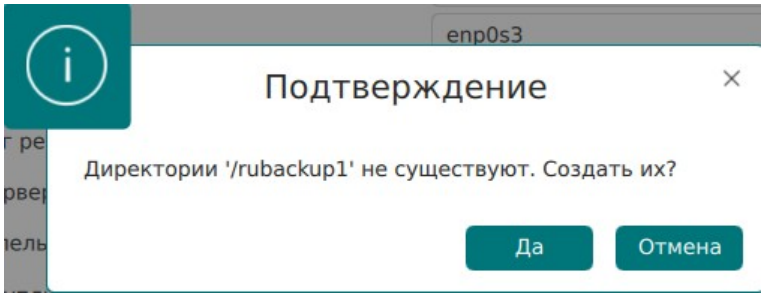


Рисунок 7

7. После подтверждений настройки и создания директорий в случае успешного конфигурирования пользователь будет уведомлён сообщением, пример, которого приведён на рисунке 8, в котором приведена информация о лицензионном соглашении, правообладателе, версии продукта, имя текущего узла с указанием настроенного компонента СРК RuBackup. Также могут быть приведены некоторые рекомендации и предупреждения по настройкам параметров.

В случае обновления будет указано выбранное и выполненное действие для существующей служебной базы данных.

Также указан созданный конфигурационный файл /opt/rubackup/etc/config.file.

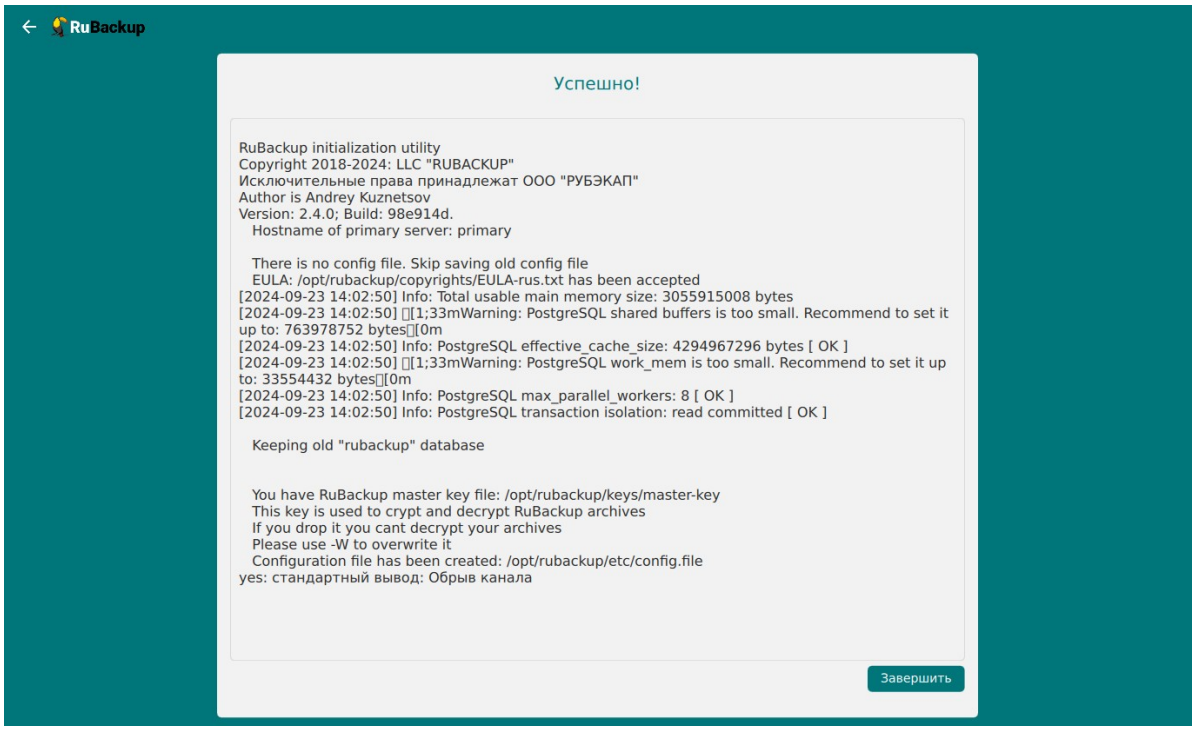


Рисунок 8

По нажатию на кнопку «Завершить» работа приложения будет завершена.

8. По окончании работы **rb_init_gui** необходимо добавить пользователя в группу rubackup и:

- для основного сервера RuBackup запустить клиентский и серверный сервисы резервного копирования. Следуйте инструкции из раздела «Настройка пользователей на сервере RuBackup» и Запуск основного сервера RuBackup.
- для резервного сервера RuBackup запустить клиентский и серверный сервисы резервного копирования. Следуйте инструкции из раздела « Настройка пользователей на резервном сервере RuBackup».
- для медиасервера запустить клиентский и серверный сервисы резервного копирования. Следуйте инструкции из раздела «Настройка пользователей на медиасервере RuBackup».
- для клиента резервного копирования запустить клиентский процесс (см. разделы «Настройка пользователей на клиенте RuBackup» и «Запуск клиента RuBackup»).

Настройка прокси-сервера

При наличии прокси-сервера HAProxy, принимающего запросы к служебной базе данных СРК RuBackup рекомендуется выполнить следующие действия:

1. В файле haproxy.cfg задать значение параметров timeout client и timeout server:
   - Рекомендуемое значение 48h или более;
   - Согласно официальной документации[2] значения параметров timeout client и timeout server должны быть одинаковые.
2. Убедиться, что в настройках служебной СУБД PostgreSQL отсутствуют таймауты, а если присутствуют, то выставить такие же значения как и в настройках HAProxy (см. п. 1).
3. Добавить в файл haproxy.cfg в строку с проверкой хоста PostgreSQL параметр shutdown-sessions, например: "server primary 192.168.122.60:3306 check on-marked-down shutdown-sessions".
4. Завершить все активные задачи в СРК RuBackup.
5. Остановить сервис сервера СРК RuBackup:

   $ sudo systemctl stop rubackup_server
6. Перезапустить PostgreSQL:

   $ sudo systemctl restart postgresql
7. Запустить сервис сервера СРК RuBackup:

   $ sudo systemctl start rubackup_server

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:

2 https://docs.haproxy.org/2.6/configuration.html

```
# 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_<version>_amd64.deb
   $ sudo dpkg -i rubackup-rbm_<version>_amd64.deb
```

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

Результаты установки

В результате установки пакетов Менеджера администратора RuBackup:

- пакеты СРК RuBackup будут развёрнуты в созданную директорию */opt/rubackup*;
- сформирован конфигурационный файл *~/.rbm2/.rb_gui_main_settings*.

Структура директории /opt/rubackup

Структура установленных пакетов приведена в Приложение В.

Настройка параметров конфигурационного файла

Для настройки параметров RBM отредактируйте конфигурационный файл *~/.rbm2/.rb_gui_main_settings*, выполнив команду:

```
sudo nano ~/.rbm2/.rb_gui_main_settings
```

Описание параметров конфигурационного файла *~/.rbm2/.rb_gui_main_settings* приведено в таблице 12.

Таблица 12 — *Описание параметров конфигурации*

Параметр	Значение по умолчанию	Возможные значения	Описание
ExitWithoutConfirmation	**false**	**false**	Выход пользователя из RBM без подтверждения
		true	

ExperimentalLogic	**false**	**false**	Функция экспериментального режима (не протестированные дополнительные возможности RBM)
		true	
Hostname	**localhost**	FQDN, hostname или ip-адрес	Адрес текущего хоста
IdleTimeoutInMinutes	**5**	Целое число от 5 до 29	Время бездействия пользователя для автоматического выхода из RBM (в минутах)
InfoHints	**true**	**false**	Показывать справочные подсказки
		true	
Lang	**Ru**	**Ru**	Язык на элементах графического интерфейса RBM
		En	
LogsLevel	**0**	Уровень логирования	
		0	Нет сообщений
		1	Fatal
		2	Critical | Fatal
		3	Warning | Critical | Fatal
		4	Debug | Warning | Critical | Fatal
		5	Info | Debug | Warning | Critical | Fatal
RecordPerPage	**50**	Целое неотрицательное число	Максимальное количество записей в таблице окна RBM на одной странице
SSLMode*	**allow**	Режим SSL-соединения с СУБД PostgreSQL (см. подраздел)	
		disable	Мне не важна безопасность и я не приемлю издержки, связанные с шифрованием
		allow	Мне не важна безопасность, но я приемлю издержки, связанные с шифрованием, если на этом настаивает сервер

		prefer	Мне не важна безопасность, но я предпочитаю шифрование (и приемлю связанные издержки), если это поддерживает сервер
		require	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Я доверяю сети в том, что она обеспечивает подключение к нужному серверу
		verify-ca	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу
		verify-full	Я хочу, чтобы мои данные шифровались, и я приемлю сопутствующие издержки. Мне нужна уверенность в том, что я подключаюсь к доверенному серверу и это именно указанный мной сервер
SessionIsBlocked	**false**	**false**	Автоматический выход из системы, если пользователь не активен в течении времени, указанного для параметра *IdleTimeoutInMinutes*
		true	
Theme	**default_theme**	**dark2_theme**	Настройка внешнего вида графического интерфейса RBM
		dark_theme	
		default_theme	
		pink_theme	
		vtb_theme	
UpdateTablePeriod	**5**	Целое число от 1 до 999999	Период времени, через который информация на странице будет обновлена (в секундах)
UseMsAdAuthByDefault	**false**	**false**	Использование базы данных MS AD по умолчанию
		true	

Username	**rubackup**	Имя пользователя, входящего в группу *rubackup*	Имя учётной записи пользователя, используемой для входа в RBM и подключения к СУБД PostgreSQL
UsernameWithDomain	**rubackup**	FQDN	Имя учётной записи пользователя, используемой для входа в RBM и подключения к базе данных MS AD
		Имя пользователя, входящего в группу *rubackup*	Если происходит подключение к СУБД PostgreSQL, то укажите значение параметра Username
* - для настройки SSL соединения выполните действия, указанные в подразделе «Настройка SSL соединения на отдельном хосте Менеджера администратора RuBackup» настоящего документа			

После совершения этих действий RBM будет готов к запуску.

Для запуска Менеджера Администратора RuBackup используйте команду:

```
$ rbm&
```

или

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

После этого введите в открывшееся окно «Аутентификация» наименование сервера Rubackup, имя пользователя и пароль (Рисунок 9).

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

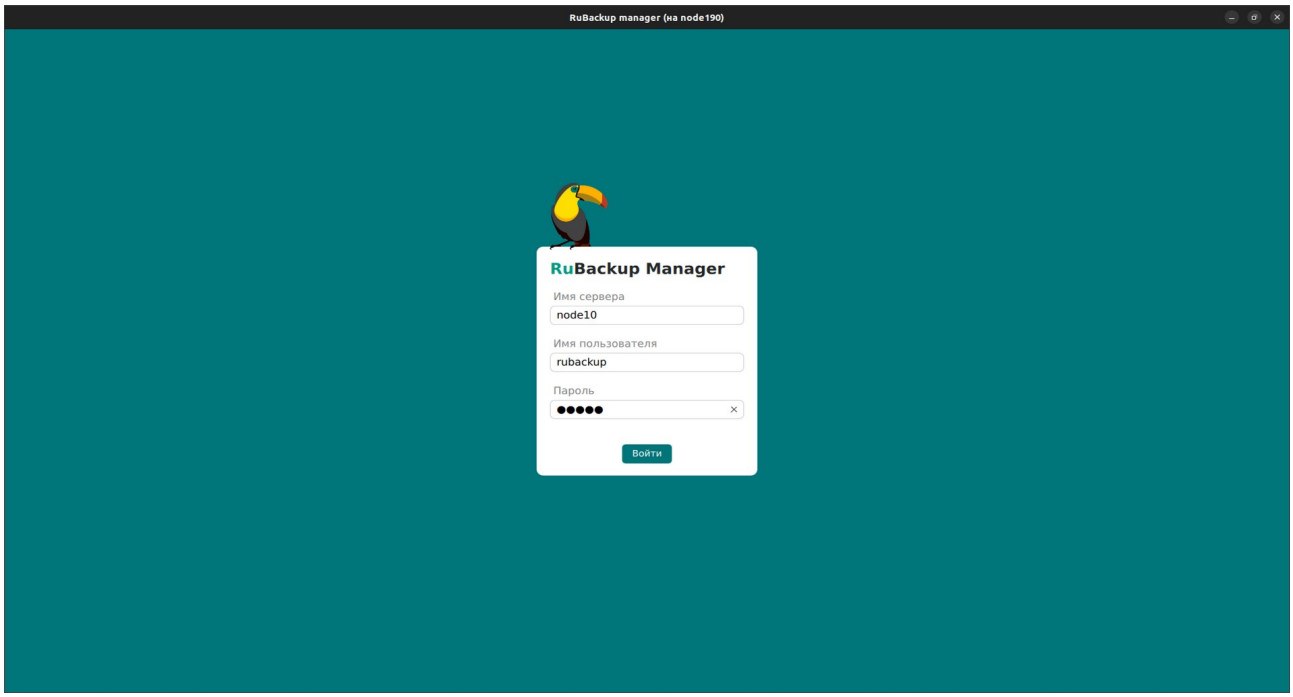

RuBackup manager (на node190)
RuBackup Manager
Имя сервера
node10
Имя пользователя
rubackup
Пароль
Войти

Рисунок 9

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».

Обновление RuBackup

Типы обновлений

СРК Rubackup поддерживает следующие типы обновлений:

- критическое обновление (hot fix). Содержит исправление критических ошибок, не связанных с безопасностью. Пакет критического обновления не является кумулятивным и требует установки предыдущих пакетов критического обновления, выпущенных для конкретного оперативного обновления или публичного релиза;

- оперативное обновление. Содержит исправление ошибок, найденных в продукте с момента последнего публичного релиза;

- публичный релиз. Версия СРК Rubackup, содержащая новый функционал, устранение ошибок и все исправления, выпущенные ранее в критических и оперативных обновлениях.

Версионность обновлений

Обновление необходимо выполнять последовательно по всем версиям, начиная со следующей версии относительно текущей и до последней доступной версии, включая все промежуточные.

Обратная совместимость

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, и $(N-1) \geq 2.1$.

Серверная группировка

К каждому серверу из группировки должно быть применено устанавливаемое обновление.

В случае, если версии обновлений в группе серверов будут различными, то работоспособными остаются те сервера, установленная версия обновления которых, совпадает с номером версии обновления primary сервера.

Клиентская группировка

Обновленный сервер или серверная группа 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 при выполнении условия обратной совместимости.

Установка обновления

Перед установкой любого обновления ознакомьтесь с версией файла, датой выпуска и условиями обратной совместимости. Убедитесь, что версия компонента CPK RuBackup старше версии, указанной в обновлении, не более чем на 1.

Порядок обновления

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.

Режимы установки обновления

Обновление можно выполнить:

- Обновив пакеты вручную.
- Автоматически.

Режим автоматического обновления

- Для автоматического обновления пакетов группировки RuBackup необходимо:

1. Скачать архив обновления, содержащий в папке Experimental/Scripts скрипт upgrade_rubackup_packages.sh.
2. Разархивировать архив.
3. Выполнить скрипт upgrade_rubackup_packages.sh:

```
$ sudo ./upgrade_rubackup_packages.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.

4. В результате автоматического обновления на клиентских узлах будут обновлены конфигурационные файлы с сохранением значений параметров следующих модулей:

- communigate_pro;
- communigate_pro_mail;
- postgres_pro;
- pg_dump_database;
- pg_dump_table;
- freeipa;
- universal (postgresql);
- vmware;
- openstack.

Более подробно смотрите документацию на соответствующий модуль.

5. Для прочих модулей 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 xvzf 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

6. На всех узлах, на которых было установлено обновление, выполните команду:

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

Режим ручного обновления

Чтобы обновить RuBackup в ручном режиме, выполните следующие действия:

1. Закройте окно RBM на АРМ администратора СРК или ином узле, использующемся 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_<version>_amd64.deb
$ sudo dpkg -i rubackup-client_<version>_amd64.deb
$ sudo dpkg -i rubackup-server_<version>_amd64.deb
$ sudo dpkg -i rubackup-rbm_<version>_amd64.deb
```

Или для пакетного менеджера, оперирующего rpm-пакетами:

```
$ sudo rpm -U rubackup-common_<version>.el7.x86_64.rpm
$ sudo rpm -U rubackup-client_<version>.el7.x86_64.rpm
$ sudo rpm -U rubackup-server_<version>.el7.x86_64.rpm
$ sudo rpm -U rubackup-rbm_<version>.el7.x86_64.rpm
```

Внимание! При обновлении пакета common с версии 2.0 на 2.1 могут возникнуть предупреждения (уровень Warning). Их можно проигнорировать, т.к. данные предупреждения никак не влияют на процесс установки системы. Пример предупреждения:

```
ldconfig: /opt/rubackup/lib/libQt5QuickShapesRB.so.5 не является
```

символьной ссылкой

Пример пакета модуля:

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

или

```
$ sudo rpm -U rubackup-postgresql_<version>.el7.x86_64.rpm
```

5. На основном сервере RuBackup вызовите утилиту rb_update одним из двух способов:

- При вызове rb_init или rb_init_gui (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. В результате обновления на клиентских узлах будут обновлены конфигурационные файлы с сохранением значений параметров следующих модулей:

- communigate_pro;
- communigate_pro_mail;
- postgres_pro;
- pg_dump_database;
- pg_dump_table;
- freeipa;
- universal (postgresql);
- vmware;

- openstack.

Более подробно смотрите документацию на соответствующий модуль.

7. Для прочих модулей 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 xvzf 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

8. На всех узлах, на которых было установлено обновление, выполните команду:

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

9. 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 медиасервере с помощью команд:

```
$ sudo systemctl start rubackup_client.service
$ sudo systemctl start rubackup_server.service
```

Критерий успешности установки обновления

Убедитесь, что СРК RuBackup обновилась корректно:

1. Проверьте, запускается ли RBM на АРМ администратора СРК или ином узле, использующемся 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 медиасерверах 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, использующимся 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 медиасервере 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.

Удаление RuBackup

Для удаления RuBackup в формате «Все в одном» (см. Подробнее раздел «Установка все в одном») необходимо выполнить следующие шаги:

- Проверить наличие резервных копий;
- Остановить сервисы;
- Удалить группу пользователей RuBackup;
- Удалить кластер БД Postrgres;
- Удалить пакеты установки.

Проверка резервных копий

Перед началом процесса удаления СРК RuBackup убедитесь, что директория с резервными копиями пуста, или удалите все резервные копии. В процессе установки СРК и запуска команды rb_init, администратор указывал путь для пула по умолчанию,

при удалении СРК перейдите в ранее созданный пул для проверки наличия в нем резервных копий.

Остановка сервисов СРК

Остановите все активные сервисы СРК, для этого выполните следующие шаги с правами суперпользователя:

- Остановите RBM: закройте окно графического интерфейса,
- Остановите клиент RuBackup
  $ rubackup_client stop
- Остановите REST API:
  $ rubackup_api stop
- Остановите сервер RuBackup:
  $ rubackup_server stop
- Проверьте логи на наличие активных или зомби процессов:
  cat /opt/rubackup/log/RuBackup.log
  ps -ef | grep rubackup

  Если остались активные или зомби процессы, то остановите их.

Удаление групп пользователей

Для удаления группы rubackup, созданной ранее, выполните следующие шаги с правами суперпользователя:

- Удалите группу следующей командой:
  $ groupdel rubackup
- Проверьте, что группы удалена:
  $ cat /etc/group | grep rubackup

Удаление кластера БД

Для удаления базы данных, необходимо выполнить команды с правами суперпользователя:

- Зайдите под пользователем «postgres», выполнив команду:
  $ sudo -u postgres psql
- Для предотвращения возможности новых подключений выполните команду:
  UPDATE pg_database SET datallowconn = 'false' WHERE datname = 'rubackup'
- Закройте все текущие сессии:

SELECT pg_terminate_backend(pg_stat_activity.pid)
FROM pg_stat_activity
WHERE pg_stat_activity.datname = 'rubackup' AND pid <> pg_backend_pid();

- Удалите базу данных:
  DROP DATABASE rubackup
- Завершите работу и выйдите:
  exit

Удаление пакетов СРК

Удалите пакеты RuBackup:

deb-пакеты:

$ sudo apt remove --purge rubackup-rbc
$ sudo apt remove --purge rubackup-rbm
$ sudo apt remove --purge rubackup-client
$ sudo apt remove --purge rubackup-common-gui
$ sudo apt remove --purge rubackup-server
$ sudo apt remove --purge rubackup-common

rpm-пакеты:

$ sudo rpm -e rubackup-rbc
$ sudo rpm -e rubackup-rmb
$ sudo rpm -e rubackup-client
$ sudo rpm -e rubackup-common-gui
$ sudo rpm -e rubackup-server
$ sudo rpm -e rubackup-common

Внимание!

Соблюдайте порядок удаления пакетов,

- Не удаляйте пакет rubackup-server до удаления пакета rubackup-client.
- Удаляйте пакет rubackup-common в последнюю очередь

Проверьте, что все пакеты удалились корректно:

deb-пакеты:

dpkg -l | grep rubackup

rpm-пакеты:

rpm -qa | grep rubackup

Проверьте содержимое директории RuBackup и удалите ее:

rm -rf /opt/rubackup

СРК RuBackup удален.

Приложение А

(справочное)

Настройка публичного репозитория RuBackup

Пакеты СРК RuBackup также можно установить из публичных репозиториев.

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/gc-astra-official-repo-key/public | sudo apt-key add -
```

Примечание: при возникновении ошибок с добавлением ключа убедитесь, что в файле /etc/resolv.conf указаны корректные серверы доменных имен.

2. В /etc/apt/sources.list добавляем строку:

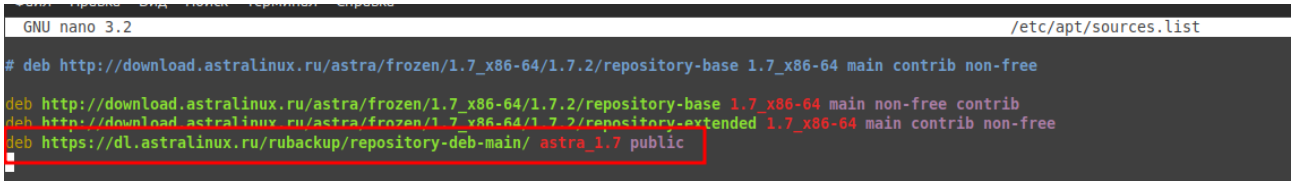
```
$ deb https://dl.astralinux.ru/rubackup/repository-deb-main/[OS-VERSION] public
```

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:



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

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
rubackup-client	Astra 1.6 Astra 1.7 Astra 1.8	deb	openssl, parsec-base, parsec-cap, parsec-mac wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-common
	Debian 10		openssl wget gnupg2 xauth (для запуска RBM через SSH)	
	Debian 12		openssl wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 18.04		openssl wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 20.04		openssl wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 22.04		openssl	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			wget gnupg2 xauth (для запуска RBM через SSH)	
	Альт 10	rpm	qt5-qtbase-gui xauth (для запуска RBM через SSH)	
	CentOS 7		qt5-qtbase-gui	
	CentOS 8		qt5-qtbase-gui	
	RedOS 7.3		qt5-qtbase-gui	
	RedOS 8		qt5-qtbase-gui	
	RHEL 9		qt5-qtbase-gui	
	Rosa Cobalt 7.3		qt5-qtbase-gui cups-libs fontconfig fontpackages-filesystem glx-utils libICE libSM libX11 libX11-common libXau libXdamage libXext libXfixes libXi libXrender libXxf86vm libicu libpng libxcb libxshmfence mesa-libEGL mesa-libGL mesa-libgbm	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			mesa-libglapi qt5-qtbase qt5-qtbase-common qt5-qtbase-gui xcb-util xcb-util-image xcb-util-keysyms xcb-util-renderutil xcb-util-wm	
	Rosa Cobalt 7.9		qt5-qtbase-gui libicu libxkbcommon-x11	
	Rosa Chrome 12		lib64qt5gui5 qt5-qtbase-gui	
rubackup-common	Astra 1.6	deb	wget gnupg2 xauth (для запуска RBM через SSH)	
	Astra 1.7		wget gnupg2 xauth (для запуска RBM через SSH)	
	Astra 1.8		wget gnupg2 xauth (для запуска RBM через SSH)	
	Debian 10		wget gnupg2 xauth (для запуска RBM через SSH)	
	Debian 12		wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 18.04		wget gnupg2 xauth (для запуска RBM через SSH)	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
	Ubuntu 20.04		wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 22.04		wget gnupg2 xauth (для запуска RBM через SSH)	
	Альт 10	rpm	xauth (для запуска RBM через SSH)	
	CentOS 7		-	
	CentOS 8		-	
	RedOS 7.3		-	
	RedOS 8		-	
	RHEL 9		-	
	Rosa Chrome 12		qt5-qtbase-gui	
	Rosa Cobalt 7.3		-	
	Rosa Cobalt 7.9		qt5-qtbase-gui libicu libxkbcommon-x11	
rubackup-common-gui	Astra 1.6 Astra 1.7 Astra 1.8	deb	wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-common
	Debian 10		wget gnupg2 xauth (для запуска RBM через SSH)	
	Debian 12		wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 18.04		wget gnupg2 xauth (для запуска RBM через SSH)	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
	Ubuntu 20.04		wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 22.04		wget gnupg2 xauth (для запуска RBM через SSH)	
	Альт 10	rpm	xauth (для запуска RBM через SSH)	rubackup-common
	CentOS 7		-	
	CentOS 8		-	
	RedOS 7.3		-	
	RedOS 8		-	
	RHEL 9		-	
	Rosa Chrome 12		qt5-qtbase-gui	
	Rosa Cobalt 7.9		qt5-qtbase-gui libicu libxkbcommon-x11	
rubackup-init-gui	Альт 10 CentOS 7 CentOS 8 RHEL 9 RedOS 7.3 RedOS 8 Rosa Chrome 12 Rosa Cobalt 7.9	rpm	-	rubackup-common-gui
	Astra 1.6 Astra 1.7 Astra 1.8 Debin 10 Debin 12 Ubuntu 18.04 Ubuntu 20.04	deb	-	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
	Ubuntu 22.04			
rubackup-communigate-pro	Astra 1.6 Astra 1.7	deb	openssl, psmisc wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-client
	Ubuntu 18.04		openssl, psmisc wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 20.04		openssl, psmisc wget gnupg2 xauth (для запуска RBM через SSH)	
	Альт 10	rpm	psmisc xauth (для запуска RBM через SSH)	
	CentOS 7		psmisc	
	CentOS 8		psmisc	
	RedOS 7.3		psmisc libxcrypt-compat	
rubackup-freeipa	Astra 1.6 Astra 1.7	deb	openssl wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-client
	Ubuntu 18.04		openssl wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 20.04		Openssl	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			wget gnupg2 xauth (для запуска RBM через SSH)	
	Альт 10	rpm	xauth (для запуска RBM через SSH)	
	CentOS 7 CentOS 8 RedOS 7.3		-	
rubackup-greenplum	Альт 10	rpm	xauth (для запуска RBM через SSH)	rubackup-client
	CentOS 7 CentOS 8 RedOS 7.3		-	
rubackup-isp-vmmanager	Astra 1.6 Astra 1.7	deb	openssl wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-client
	Альт 10	rpm	xauth (для запуска RBM через SSH)	
	CentOS 7		-	
	CentOS 8		-	
	RedOS 7.3		-	
	Ubuntu 20.04		wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 18.04		wget gnupg2 xauth (для запуска RBM через SSH)	
rubackup-pg-dump	Astra 1.6 Astra 1.7 Astra 1.8	deb	openssl wget gnupg2 xauth (для запуска RBM через SSH) postgresql	rubackup-client

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
	Debian 10		openssl wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 18.04 Ubuntu 20.04		openssl wget gnupg2 xauth (для запуска RBM через SSH) postgresql	
	Альт 10	rpm	xauth (для запуска RBM через SSH) libpq.so.5 python3 (psycopg2)	
	CentOS 7		glibc-0:2.17-317.el7.i686 bash-0:4.2.46-35.el7_9.x86_64 gdbm-0:1.10-8.el7.x86_64 zlib-0:1.2.7-21.el7_9.x86_64 postgresql-libs-0:9.2.24-8.el7_9.x86_64 libuuid-0:2.23.2-65.el7.x86_64 glibc-0:2.17-326.el7_9.x86_64 libffi-0:3.0.13-19.el7.x86_64 platform-python platform-python-libs	
	CentOS 8		libpq	
	RedOS 7.3		platform-python platform-python-libs	
rubackup-postgres-pro	Astra 1.6 Astra 1.7	deb	openssl, psmisc wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-client
	Ubuntu 18.04		openssl, psmisc wget	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 20.04		openssl, psmisc wget gnupg2 xauth (для запуска RBM через SSH)	
	Альт 10	rpm	xauth (для запуска RBM через SSH) libpq.so.5	
	CentOS 7		glibc bash gdbm zlib postgresql-libs libuuid glibc libffi	
	CentOS 8		-	
	RedOS 7.3		-	
rubackup-postgresql	Astra 1.6 Astra 1.8	deb	openssl, sudo, lsof, grep, gawk	rubackup-client
	Astra 1.7		openssl, sudo, lsof, grep, gawk perl	
	Debian 10		openssl, sudo,	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			lsof, grep, wget gnupg2 xauth (для запуска RBM через SSH) gawk libmpfr6 libsigsegv2	
	Ubuntu 18.04		openssl, sudo, lsof, grep, gawk libmpfr6 libsigsegv2 wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 20.04		openssl, sudo, lsof, grep, gawk wget gnupg2 xauth (для запуска RBM через SSH)	
	Альт 10	rpm	xauth (для запуска RBM через SSH) libcurl.so.4 sudo, lsof, grep, gawk	
	CentOS 7		sudo, lsof, grep,	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			gawk glibc bash libxml2 libcurl libgcc libstdc++	
	CentOS 8		sudo, lsof, grep, gawk lsof - - nobest	
	RedOS 7.3		sudo, lsof, grep, gawk	
	RHEL 9		sudo, lsof, grep, gawk lsof - - nobest	
rubackup-rbc	Astra 1.6	deb	libicu57 wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-common-gui, rubackup-client
	Astra 1.7		libicu63 wget gnupg2 xauth (для запуска RBM через SSH)	
	Astra 1.8		libicu72 wget gnupg2 xauth (для запуска RBM через SSH)	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
	Debian 10		libicu63 wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 18.04		libicu60 wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 20.04		libicu66 wget gnupg2 xauth (для запуска RBM через SSH)	
	CentOS 7	rpm	libicu50.2 libxkbcommon libxkbcommon-x11 xkeyboard-config	
	CentOS 8		libicu60.3	
	RedOS 7.3		libicu65.1	
	RHEL 9		libicu67.1	
	Альт 10	rpm	xauth (для запуска RBM через SSH) libicu69, libxkbcommon-x11	
	Альт Сервер 10		xauth (для запуска RBM через SSH) libicu69, libxkbcommon-x11	
	Альт Сервер 9		xauth (для запуска RBM через SSH) libicu65, libxkbcommon-x11	
	Rosa Cobalt 7.3		libicu50.2, libxkbcommon-x11	
	Rosa Cobalt 7.9		libicu50.1.2,	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			libxkbcommon-x11 qt5-qtbase-gui libicu libxkbcommon-x11	
	Rosa Chrome 12		lib64icudata71, libxkbcommon-x11 qt5-qtbase-gui	
rubackup-rbm	Astra 1.6	deb	libicu57 wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-common-gui
	Astra 1.7		libicu63 wget gnupg2 xauth (для запуска RBM через SSH)	
	Astra 1.8		libicu72 wget gnupg2 xauth (для запуска RBM через SSH)	
	Debian 10		libicu63 wget gnupg2 xauth (для запуска RBM через SSH)	
	Debian 12		libicu72 wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 18.04		libicu60 wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 20.04		libicu66	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 22.04		libicu70 wget gnupg2 xauth (для запуска RBM через SSH)	
	CentOS 7	rpm	libicu50.2	
	CentOS 8		libicu60.3	
	RedOS 7.3		libicu65.1	
	RedOS 8		libicu71.1	
	RHEL 9		libicu67.1	
	Альт 10	rpm	libicu69, libxkbcommon-x11 xauth (для запуска RBM через SSH)	
	Альт Сервер 10		libicu69, libxkbcommon-x11	
	Альт Сервер 9		libicu65, libxkbcommon-x11	
	Rosa Cobalt 7.3		libicu50.2, libxkbcommon-x11	
	Rosa Cobalt 7.9		libicu50.1.2, libxkbcommon-x11 qt5-qtbase-gui libicu libxkbcommon-x11	
	Rosa Chrome 12		lib64icudata71, libxkbcommon-x11 qt5-qtbase-gui	
rubackup-server	Astra 1.7	deb	openssl	rubackup-common,

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			libcurl3 или libcurl4, mailutils или bsd-mailx, libsasl2-2, libldap-2.4-2, libpugixml1v5 wget gnupg2 xauth (для запуска RBM через SSH) exim4-base exim4-config exim4-daemon-light guile-2.2-libs libevent-2.1-6 libfribidi0 libgc1c2 libgnutls-dane0 libgsasl7 libkyotocabinet16v5 libltdl7 liblzo2-2 libmailutils5 libmariadb3 libntlm0 libunbound8 mailutils-common mariadb-common mysql-common psmisc	rubackup-client
	Debian 10		openssl libcurl3 или libcurl4, mailutils или bsd-mailx, libsasl2-2, libldap-2.4-2, libpugixml1v5 wget gnupg2	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			xauth (для запуска RBM через SSH) exim4-base exim4-config exim4-daemon-light guile-2.2-libs libcurl4 libevent-2.1-6 libfribidi0 libgc1c2 libgnutls-dane0 libgsasl7 libkyotocabinet16v5 libltdl7 liblzo2-2 libmailutils5 libmariadb3 libntlm0 libpython2.7 libunbound8 mailutils-common mariadb-common mysql-common psmisc	
	Ubuntu 20.04		openssl libcurl3 или libcurl4, mailutils или bsd-mailx, libsasl2-2, libldap-2.4-2, libpugixml1v5 wget gnupg2 xauth (для запуска RBM через SSH) guile-2.2-libs libgc1c2 libgsasl7 libidn11 libkyotocabinet16v5	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			libmailutils6 libmysqlclient21 libntlm0 mailutils-common mysql-common postfix ssl-cert	
	Astra 1.8		openssl, libcurl3 или libcurl4, mailutils или bsd-mailx, libsasl2-2, libldap-2.5-0, libpugixml1v5 wget gnupg2 xauth (для запуска RBM через SSH) exim4-base exim4-config exim4-daemon-light gsasl-common guile-3.0-libs libevent-2.1-7 libgc1 libgnutls-dane0 libgnutls30 libgsasl18 libgssglue1 libidn12 libltdl7 libmailutils9 ibmariadb3 libncurses6 libncursesw6 libntlm0 libpq5 libtinfo6 libunbound8	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			mailutils-common mariadb-common mysql-common ncurses-base ncurses-bin ncurses-term psmisc	
	Debian 12		openssl, libcurl3 или libcurl4, mailutils или bsd-mailx, libsasl2-2, libldap-2.5-0, libpugixml1v5 wget gnupg2 xauth (для запуска RBM через SSH) exim4-base exim4-config exim4-daemon-light gsasl-common guile-3.0-libs libevent-2.1-7 libfribidi0 libgc1 libgnutls-dane0 libgnutls30 libgsasl18 libgssglue1 libidn12 libltdl7 libmailutils9 libmariadb3 libncurses6 libntlm0 libpq5 libpython3.11	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			libpython3.11-minimal libpython3.11-stdlib libunbound8 mailutils-common mariadb-common mysql-common psmisc python3.11 python3.11-minimal	
	Ubuntu 22.04		openssl, libcurl3 или libcurl4, mailutils или bsd-mailx, libsasl2-2, libldap-2.5-0, libpugixml1v5 wget gnupg2 xauth (для запуска RBM через SSH) gsasl-common guile-3.0-libs libfribidi0 libgc1 libgsasl7 libidn12 libltdl7 libmailutils8 libmysqlclient21 libntlm0 libpq5 mailutils-common mysql-common postfix ssl-cert	
	Astra 1.6		openssl, libcurl3 или libcurl4, mailutils или bsd-mailx,	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			libsasl2-2, libldap-2.4-2 wget gnupg2 xauth (для запуска RBM через SSH) exim4-base exim4-config exim4-daemon-light libldap-2.4-2 libldap-common liblockfile-bin liblockfile1 libnghttp2-14 librtmp1 libssh2-1 psmisc	
	Ubuntu 18.04		openssl, libcurl3 или libcurl4, mailutils или bsd-mailx, libsasl2-2, libldap-2.4-2 wget gnupg2 xauth (для запуска RBM через SSH) guile-2.0-libs libgc1c2 libgsasl7 libkyotocabinet16v5 libltdl7 liblzo2-2 libmailutils5 libmysqlclient20 libnghttp2-14 libntlm0 libpython2.7 libpython2.7-minimal libpython2.7-stdlib	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			librtmp1 mailutils-common mysql-common postfix ssl-cert	
	Альт 10	rpm	mailutils, libsasl2-3, libldap, pugixml xauth (для запуска RBM через SSH)	
	CentOS 7		mailx, cyrus-sasl, openldap, pugixml	
	CentOS 8		mailx, cyrus-sasl, openldap, pugixml	
	RedOS 7.3		mailx, cyrus-sasl, openldap, pugixml	
	RedOS 8		mailx, cyrus-sasl, openldap, pugixml	
	RHEL 9		s-nail, cyrus-sasl, openldap, pugixml mailx	
	Rosa Chrome 12		mailutils, lib64sasl2,	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			lib64ldap2.4_2, lib64pugixml1, qt5-qtbase-gui cyrus-sasl lib64db5.2 lib64ltdl7 lib64mailutils9 lib64mu_auth9 lib64mu_dbm9 lib64mu_dotmail9 lib64mu_imap9 lib64mu_maildir9 lib64mu_mailer9 lib64mu_mbox9 lib64mu_pop9 lib64mu_sieve9 lib64muaux9 mailutils-locales	
	Rosa Cobalt 7.3		mailx, cyrus-sasl, openldap	
	Rosa Cobalt 7.9		mailx, cyrus-sasl, openldap qt5-qtbase-gui libicu libxkbcommon-x11	
rubackup-rest-api	Astra 1.6 Astra 1.7 Astra 1.8	deb	wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-server
	CentOS 7		postgresql-libs	
	CentOS 8		-	
	Ubuntu 22.04		wget	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			gnupg2 xauth (для запуска RBM через SSH)	
	Альт 10	rpm	libpq.so.5	rubackup-server
	RedOS 7.3		-	
	Rosa Cobalt 7.3		postgresql-libs-0:9.2.18-1.res7.x86_64 bash-0:4.2.46-21.res7.x86_64 readline-0:6.2-9.res7.x86_64 gdbm-0:1.10-8.res7.x86_64 libstdc++-0:4.8.5-28.res7c.1.x86_64 libgcc-0:4.8.5-28.res7c.1.x86_64 sqlite-0:3.7.17-8.res7.x86_64 ncurses-libs-0:5.9-13.20130511.res7.x86_64 libuuid-0:2.23.2-33.res7.x86_64 zlib-0:1.2.7-17.res7.x86_64 glibc-0:2.17-222.res7.x86_64 libffi-0:3.0.13-18.res7.x86_64 glibc-0:2.17-222.res7.i686 openssl-libs-1:1.0.1e-60.res7c.1.x86_64	
	Rosa Cobalt 7.9		postgresql-libs-0:9.2.18-1.res7.x86_64 bash-0:4.2.46-21.res7.x86_64 readline-0:6.2-9.res7.x86_64 gdbm-0:1.10-8.res7.x86_64 libstdc++-0:4.8.5-28.res7c.1.x86_64 libgcc-0:4.8.5-28.res7c.1.x86_64 sqlite-0:3.7.17-8.res7.x86_64 ncurses-libs-0:5.9-13.20130511.res7.x86_64 libuuid-0:2.23.2-33.res7.x86_64 zlib-0:1.2.7-17.res7.x86_64 glibc-0:2.17-222.res7.x86_64 libffi-0:3.0.13-18.res7.x86_64 glibc-0:2.17-222.res7.i686 openssl-libs-1:1.0.1e-60.res7c.1.x86_64	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
rubackup-tionix	Альт 10	rpm	-	rubackup-client
	Rosa Cobalt 7.9		qt5-qtbase-gui libicu libxkbcommon-x11	
rubackup-ovirt-client	CentOS 8	rpm	-	rubackup-ovirt-common
rubackup-ovirt-common	CentOS 8	rpm	-	ВНИМАНИЕ! НЕ СОВМЕСТИМ с пакетом rubackup-common
rubackup-mailion	Astra 1.7	deb	wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-client
rubackup-rustack	Astra 1.6	deb	openssl libcurl3 libcurl4 wget gnupg2 xauth (для запуска RBM через SSH) libldap-2.4-2 libldap-common libnghttp2-14 librtmp1 libssh2-1	rubackup-client
	Astra 1.7		openssl libcurl3 libcurl4 wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 18.04		wget gnupg2 xauth (для запуска RBM через SSH) libcurl3	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			libnghttp2-14 librtmp1	
rubackup-aerodisk	Astra 1.6	deb	openssl, libcurl3 или libcurl4 libldap-2.4-2 libldap-common libnghttp2-14 librtmp1 libssh2-1 wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-client
	Astra 1.7		openssl, libcurl3 или libcurl4 wget gnupg2 xauth (для запуска RBM через SSH)	
	Debian 10		openssl, libcurl3 или libcurl4 wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 18.04		openssl, libcurl3 или libcurl4 libnghttp2-14 librtmp1 wget gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 20.04		openssl, libcurl3 или libcurl4 wget gnupg2 xauth (для запуска RBM через SSH)	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
rubackup-brest	Astra 1.6	deb	openssl, libcurl3 или libcurl4 wget gnupg2 xauth (для запуска RBM через SSH) libldap-2.4-2 libldap-common libnghttp2-14 librtmp1 libssh2-1	rubackup-client
	Astra 1.7		openssl, libcurl3 или libcurl4 wget gnupg2 xauth (для запуска RBM через SSH)	
rubackup-brest-template	Astra 1.6	deb	openssl, libcurl3 или libcurl4 wget gnupg2 xauth (для запуска RBM через SSH) libldap-2.4-2 libldap-common libnghttp2-14 librtmp1 libssh2-1	rubackup-client
	Astra 1.7		openssl, libcurl3 или libcurl4 wget gnupg2 xauth (для запуска RBM через SSH)	
rubackup-openstack	Astra 1.6 Astra 1.7	deb	openssl wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-client
	Ubuntu 18.04		openssl wget	

Устанавливаемый пакет	Операционная система	Формат пакетов ОС	Зависимости пакетов Linux	Требуемый пакет rubackup
			gnupg2 xauth (для запуска RBM через SSH)	
	Ubuntu 20.04		openssl wget gnupg2 xauth (для запуска RBM через SSH)	
	RedOS 7.3	rpm	-	
rubackup-vmware	Ubuntu 20.04	deb	openssl, libcurl3 или libcurl4 wget gnupg2 xauth (для запуска RBM через SSH)	rubackup-client

Приложение В

Структура установленных пакетов Системы резервного копирования и восстановления данных RuBackup

При установке инсталляционный rpm/deb-пакет будет автоматически распакован в директорию /opt/rubackup.

Структура установленных пакетов основного сервера приведена в таблице 13.

Таблица 13 — *Структура установленных пакетов основного сервера*

Структурный элемент	Назначение элемента
/opt/rubackup	Директория, в которой распакован установочный комплект компонента RuBackup, а также используемые дополнительные инструменты
Пакет rubackup-common	
/opt/rubackup/keys/client/	Папка содержит сертификат и закрытый ключ клиента для внутреннего взаимодействия компонентов СРК по протоколу SSL
/opt/rubackup/keys/server/	Папка содержит сертификат и закрытый ключ сервера для внутреннего взаимодействия компонентов СРК по протоколу SSL
/opt/rubackup/keys/rootCA/	Папка содержит самоподписанный сертификат и закрытый ключ центра сертификации для внутреннего взаимодействия компонентов СРК по протоколу SSL
/opt/rubackup/etc/	Папка содержит конфигурационные файлы СРК RuBackup
/opt/rubackup/etc/ld.so.conf.d/ rubackup.conf	Вспомогательный конфигурационный файл, указывающий ОС путь к дополнительным библиотекам, используемых СРК RuBackup
/opt/rubackup/copyrights/	Папка содержит файлы лицензионных соглашений
/opt/rubackup/rc/icons/	Папка содержит иконки интерфейса
Пакет rubackup-client	

Структурный элемент	Назначение элемента
/opt/rubackup/etc/systemd/system/	Папка содержит сервисы СРК RuBackup
/opt/rubackup/etc/rubackup.lsf	Файл локального расписания Клиента системы резервного копирования
/opt/rubackup/etc/systemd/system/ rubackup_client.service	Сервис Клиентской части СРК RuBackup
/opt/rubackup/scripts/	Папка содержит скрипты управления СРК RuBackup
/opt/rubackup/scripts/test-script.sh	Пример скрипта для выполнения при резервном копировании
/opt/rubackup/log/	Папка содержит журналы событий и задач
/opt/rubackup/man/	Папка содержит инструкции по использованию утилит
/opt/rubackup/modules/	Папка содержит исполнительные модули, поддерживающие резервное копирование и восстановление целевого ресурса (поддерживаемого клиентом СРК)
/opt/rubackup/modules/ rb_module_lvm	Исполняемый модуль для резервного копирования и восстановления логических томов lvm
/opt/rubackup/modules/ rb_module_filesystem	Исполняемый модуль резервного копирования файловой системы
/opt/rubackup/bin/	Папка содержит консольные утилиты, поддерживаемые на клиенте для управления резервным копированием и восстановлением данных
/opt/rubackup/bin/rb_schedule	Утилита клиента RuBackup для просмотра правил глобального расписания клиента в системе резервного копирования
/opt/rubackup/bin/rb_replicas	Утилита клиента RuBackup для управления правилами репликации на клиенте. Вы можете просмотреть список всех правил репликации, а также запустить выбранное правило
/opt/rubackup/bin/rb_health_check	Утилита клиента RuBackup для проверки

Структурный элемент	Назначение элемента
	конфигурации клиента и его окружения. Выполняется проверка переменных окружения, версии медиасервера. Проверяется подключение клиента к базе данных, серверу, медиасерверу и толстому клиенту
/opt/rubackup/bin/rubackup_client	Клиент резервного копирования RuBackup представляет собой фоновое приложение (сервис, демон), запущенное на хосте клиента и взаимодействующее с сервером RuBackup
/opt/rubackup/bin/rb_init	Утилита администратора RuBackup для первоначального конфигурирования клиента сразу после развёртывания пакета исполняемых файлов. Неинтерактивный режим необходим для сценариев массового развертывания
/opt/rubackup/bin/rb_archives	Утилита клиента RuBackup предназначена для просмотра списка резервных копий клиента в системе резервного копирования, создания срочных резервных копий, их удаления, проверки и восстановления. Работает только в том случае, если на клиенте работает служба (сервис, демон) клиента rubackup_client
/opt/rubackup/bin/rbfd	Утилита администратора RuBackup для создания и восстановления полных и инкрементальных резервных копий ресурсов в любых файловых системах. Ресурсом может быть файл, каталог или блочное устройство
/opt/rubackup/bin/rb_tasks	Утилита клиента RuBackup для просмотра списка задач клиента в системе резервного копирования RuBackup
/opt/rubackup/bin/ rb_client_defined_storages	Утилита администратора RuBackup для управления клиентскими хранилищами. Вы можете просматривать, добавлять и удалять клиентские хранилища в конфигурации
/opt/rubackup/rc/	Папка содержит конфигурационные скрипты СРК RuBackup
/opt/rubackup/mnt/	Предоставляется как временная точка монтирования для файловых систем
Пакет rubackup-server	

Структурный элемент	Назначение элемента
/opt/rubackup/etc/systemd/system/	Папка одержит сервисы СРК RuBackup
/opt/rubackup/etc/systemd/system/ rubackup_server.service	Сервис Серверной части СРК RuBackup
/opt/rubackup/man/	Папка содержит файлы описаний утилит
/opt/rubackup/log/	Папка содержит файлы журнала событий
/opt/rubackup/log/Rubackup.log	Системный журнал событий, также содержит информацию о лицензии
/opt/rubackup/log/task.log	Журналы событий, содержащие события задач СРК
/opt/rubackup/log/module_.log	Журналы событий исполняемых модулей
/opt/rubackup/log/rbfd	Информация о процессе выполнения создания РК для каждой задачи, которая использует rbfd
/opt/rubackup/lib/	Папка содержит библиотеки, необходимые для работы СРК RuBackup
/opt/rubackup/bin/	Папка содержит исполняемые файлы для запуска утилит
/opt/rubackup/bin/rb_modules	Утилита администратора RuBackup для управления Модулями
/opt/rubackup/bin/rb_tape_libraries	Утилита администратора RuBackup для управления ленточными библиотеками в системе резервного копирования RuBackup. Вы можете просматривать информацию о ленточных библиотеках в серверной группировке RuBackup, синхронизировать ленточную библиотеку с информацией о ней в базе данных,импортировать, экспортировать и перемещать картриджи в ленточной библиотеке, а также производить LTFS форматирование картриджей, находя-щихся в слотах ленточной библиотеки.
/opt/rubackup/bin/ rb_media_servers	Утилита администратора RuBackup для управления медиасерверами RuBackup. Вы можете просматривать список медиасерверов, добавлять их, удалять или изменять их описания.

Структурный элемент	Назначение элемента
	медиасервер предназначен для взаимодействия с клиентами при создании, восстановлении и передаче резервных копий
/opt/rubackup/bin/rb_user_groups	Утилита администратора RuBackup для управления группами пользователей. Вы можете просматривать группы пользователей, добавлять и удалять их, а также изменять их название и описание
/opt/rubackup/bin/rubackup_server	Сервер резервного копирования RuBackup представляет собой системное фоновое приложение (служба, демон), внутри которого одновременно выполняются множество потоков, отвечающих за разные функции системы резервного копирования
/opt/rubackup/bin/ rb_local_filesystems	Утилита администратора RuBackup для управления хранилищами резервных копий типа Файловая система. Хранилища такого типа должны быть ассоциированы с пулом того же типа
/opt/rubackup/bin/rb_security	Утилита RuBackup для работы с журналом событий информационной безопасности
/opt/rubackup/bin/rb_clients	Утилита администратора RuBackup для управления клиентами RuBackup. Вы можете просматривать список клиентов, а также добавлять, удалять или изменять их.
/opt/rubackup/bin/rb_update	Утилита администратора RuBackup для управления обновлениями баз данных. Создает sql инструкции, позволяющие сделать обновление базы данных
/opt/rubackup/bin/rb_block_devices	Утилита администратора RuBackup для управления блочными устройствами
/opt/rubackup/bin/rb_global_config	Утилита администратора RuBackup для управления параметрами глобальной конфигурации серверной группировки RuBackup. Параметры глобальной конфигурации действительны для всех серверов, входящих в кластер серверов RuBackup
/opt/rubackup/bin/ rb_global_schedule	Утилита администратора RuBackup для управления глобальным расписанием RuBackup.

Структурный элемент	Назначение элемента
	Глобальное расписание состоит из отдельных правил, которые могут выполняться по определённым условиям для определённого ресурса на клиенте системы резервного копирования. Можно просматривать список правил глобального расписания, экспортировать настройки правила в файл и импортировать правило из файла в глобальное расписание, удалять правила из глобального расписания, останавливать функционирование правила или запускать его в работу, а также немедленно создавать задачу на основе правила глобального расписания
/opt/rubackup/bin/rb_repository	Утилита администратора RuBackup для доступа к записям репозитория. Позволяет просматривать список резервных копий, удалять и перемещать резервные копии, проверять их целостность и выполнять их репликацию (копирование) в другие пулы. Для выполнения этих действий утилита создаёт соответствующую задачу в главной очереди задач и заканчивает своё выполнение до того момента, как задача будет выполнена
/opt/rubackup/bin/rb_users	Утилита администратора RuBackup для управления пользователями. Вы можете просматривать список пользователей, добавлять, удалять и изменять их
/opt/rubackup/bin/rb_tape_cartriges	Утилита администратора RuBackup для управления картриджами ленточных библиотек. Вы можете просматривать список картриджей, добавлять, удалять или изменять их. Каждый картридж принадлежит какому-либо пулу типа ленточная библиотека
/opt/rubackup/bin/rb_inventory	Утилита администратора RuBackup для внесения в базу данных RuBackup информации о резервных копиях, которые были сделаны вне текущей конфигурации RuBackup, например, в другой серверной группировке RuBackup
/opt/rubackup/bin/rb_interoperation	Утилита администратора RuBackup для управления задачами импорта или экспорта резервных копий между независимыми системами резервного копирования. Вы можете управлять списком систем, для которых существует возможность импорта или экспорта.

Структурный элемент	Назначение элемента
	Добавлять, просматривать, редактировать, удалять, останавливать и запускать правила экспорта или импорта. Также вы сможете проверять очередь задач и удалять выполненные задачи или завершившиеся с ошибкой. У вас будет возможность создать задачу на экспорт резервной копии из репозитория
/opt/rubackup/bin/rb_clouds	Утилита администратора RuBackup для просмотра конфигурации, добавления или удаления облаков S3 в системе резервного копирования
/opt/rubackup/bin/rb_copy2pool	Утилита администратора RuBackup для управления репликацией. Предоставляет возможность создавать точные копии (реплики) резервных копий для правил резервного копирования и для стратегий резервного копирования
/opt/rubackup/bin/rb_notifications	Утилита администратора RuBackup для управления очередью уведомлений. В очереди уведомлений содержатся все актуальные уведомления групп пользователей RuBackup о происходящих в системе событиях. Уведомления могут быть настроены в правилах глобального расписания и в стратегиях
/opt/rubackup/bin/ rb_remote_replication	Утилита администратора RuBackup для управления непрерывной удалённой репликацией. Непрерывная удалённая репликация состоит из отдельных правил, которые могут выполняться по определённым условиям для определённого ресурса. Можно просматривать список правил непрерывной удалённой репликации, экспортировать настройки правила в файл и импортировать правило из файла, удалять правила, останавливать функционирование правила или запускать его в работу
/opt/rubackup/bin/rb_pools	Утилита администратора RuBackup для управления пулами. Вы можете просматривать список пулов, добавлять, удалять и изменять их. Каждый пул принадлежит какому-либо медиасерверу. Пулы используются для группирования устройств хранения резервных копий

Структурный элемент	Назначение элемента
/opt/rubackup/bin/rb_tl_task_queue	Утилита администратора RuBackup для управления Очередью задач ленточных библиотек
/opt/rubackup/bin/ rb_block_device_check	Утилита администратора RuBackup для проверки целостности резервных копий на блочном устройстве
/opt/rubackup/bin/rb_client_group	Утилита администратора RuBackup для управления группами клиентов. Вы можете просматривать группы клиентов, добавлять их, удалять или изменять их название и описание. Группировка клиентов может потребоваться в случае необходимости выполнения групповых операций резервного копирования в стратегиях
/opt/rubackup/bin/rb_bandwidth	Утилита администратора RuBackup для управления ограничениями пропускной способности при выполнении операций резервного копирования для клиентов или правил глобального расписания. Вы можете установить одно или несколько ограничений пропускной способности для определённого клиента СРК или для какого-либо правила глобального расписания
/opt/rubackup/bin/rb_task_queue	Утилита администратора RuBackup для управления главной очередью задач. В очереди задач содержатся все актуальные задачи на создание, восстановление, удаление, перемещение и проверку резервных копий
/opt/rubackup/bin/ rb_cloud_task_queue	Утилита администратора RuBackup для просмотра задач, которые связаны с облачными операциями. При хранении резервных копий в облаке S3 вам может потребоваться загрузить резервную копию в облако или выгрузить какой-либо из файлов резервной копии из облака
/opt/rubackup/bin/rb_strategies	Утилита администратора RuBackup для управления стратегиями
/opt/rubackup/bin/rb_log_viewer	Утилита администратора RuBackup для просмотра и управления журналами сообщений
/opt/rubackup/rc/init/	Содержит конфигурационные скрипты СРК RuBackup
/opt/rubackup/mnt/	Предоставляется как временная точка

Структурный элемент	Назначение элемента
	монтирования для файловых систем
Пакет rubackup-rest-api	
/opt/rubackup/bin/rubackup_api	Bash-скрипт, ссылающийся на исполняемый файл по пути /opt/rubackup/lib/rubackup_rest_api_lib/rubackup_api.bin
/opt/rubackup/etc/systemd/system/ rubackup_api.service	Файл сервиса приложения rubackup_api, необходимый для взаимодействия с приложением через утилиты systemctl и system
/opt/rubackup/etc/rubackup_api.env	Файл настроек окружения, подгружаемый сервисом приложения. Нужен для настройки приложения в части подключений и отладки
/opt/rubackup/etc/ rubackup_api_logger.conf./opt/	Конфигурационный файл тонкой настройки журналирования в приложении СРК RuBackup REST API. Не предполагается внесение изменений со стороны пользователя
/opt/rubackup/lib/ rubackup_rest_api_lib/ rubackup_api.bin	Исполняемый файл приложений СРК RuBackup REST API и Tucana
/opt/rubackup/rc/rubackup_api/ui/	Папка содержит файлы графики приложения
Пакет rubackup-common-gui	
/opt/rubackup/keys/rbm/	Папка содержит сертификат и закрытый ключ приложения RBM для внутреннего взаимодействия компонентов СРК по протоколу SSL
/opt/rubackup/gui/plugins/	Папка содержит плагины
/opt/rubackup/gui/lib/	Папка содержит библиотеки, используе-мые графическим приложением RBM
/opt/rubackup/gui/qml/	Папка содержит QML-библиотеки, используемые графическим приложением RBM
/opt/rubackup/gui/rc/	Папка содержит настройки графического

Структурный элемент	Назначение элемента
	отображения, в т.ч. темы, переводы приложения RBM
/opt/rubackup/gui/rc/themes/	Файлы тем приложения RBM
Пакет rubackup-rbm	
~/.rbm2/.logs	Журнал событий, содержащий события в соответствии с установленным уровнем логирования, для служебного использования
~/.rbm2/.rb_gui_column_settings	Файл настройки колонок таблиц в окне RBM для запоминания настроек пользователя (true — показать колонку, false — скрыть колонку)
~/.rbm2/.rb_gui_main_settings	Конфигурационный файл, содержащий информацию о параметрах и настройках RBM
/opt/rubackup/gui/rc/langs/	Файлы с текстами переводов интерфейса приложения RBM
/opt/rubackup/gui/rc/info/	Информационные подсказки приложения RBM
/opt/rubackup/bin/rbm	Исполняемый файл приложения RBM
Пакет rubackup-init-gui	
/opt/rubackup/bin/rb_init_gui	Исполняемый файл мастера настройки RuBackup
/opt/rubackup/gui/rc/langs/rb_init2_ru.qm	Файл с текстами переводов интерфейса приложения «Мастер настройки RuBackup»
Пакет rubackup-rbc	
/opt/rubackup/gui/rc/langs/	Файлы с текстами переводов интерфейса приложения RBC
/opt/rubackup/bin/rbc	Исполняемый файл приложения RBC
Пакет rubackup-rustack	

Структурный элемент	Назначение элемента
/opt/rubackup/etc/ rb_module_rustack.conf	Конфигурационный файл модуля резервного копирования и восстановления данных виртуальных машин платформы виртуализации RUSTACK
/opt/rubackup/modules/ rb_module_rustack	Утилита резервного копирования и восстановления данных виртуальных машин платформы виртуализации RUSTACK
Пакет rubackup-postgres-pro	
/opt/rubackup/etc/ rb_module_postgres_pro.conf	Конфигурационный файл модуля резервного копирования и восстановления данных кластеров СУБД Postgres Pro
/opt/rubackup/modules/ rb_module_postgres_pro_lib/ rb_module_postgres_pro.bin	Исполняемый файл модуля, поддерживающий резервное копирование и восстановление данных кластеров СУБД Postgres Pro
/opt/rubackup/modules/ rb_module_postgres_pro	Bash-скрипт, ссылающийся на исполняемый файл по пути /opt/rubackup/modules/rb_module_postgres_pro_lib /rb_module_postgres_pro.bin
Пакет rubackup-postgresql	
/opt/rubackup/etc/ rb_module_postgresql.conf	Конфигурационный файл модуля резервного копирования и восстановления данных кластеров СУБД PostgreSQL
/opt/rubackup/modules/ rb_module_postgresql	Исполняемый модуль, поддерживающий резервное копирование и восстановление данных кластеров СУБД PostgreSQL
/opt/rubackup/mnt/ postgresql_archives/	Папка для временного хранения архив-ных WAL файлов
Пакет rubackup-greenplum	
/opt/rubackup/etc/ rb_module_greenplum.conf	Конфигурационный файл модуля резервного копирования и восстановления данных кластеров СУБД GreenPlum

Структурный элемент	Назначение элемента
/opt/rubackup/modules/ rb_module_greenplum	Исполняемый модуль, поддерживающий резервное копирование и восстановление данных кластеров СУБД GreenPlum
Пакет rubackup-vmware	
/opt/rubackup/etc/ rb_module_vmware.conf	Конфигурационный файл модуля резервного копирования и восстановления данных виртуальных машин среды виртуализации VMware vSphere
/opt/rubackup/modules/ rb_module_vmware	Исполняемый модуль, поддерживающий копирование и восстановление данных виртуальных машин среды виртуализации VMware vSphere
Пакет rubackup-openstack	
/opt/rubackup/etc/ rb_module_openstack.conf	Конфигурационный файл модуля резервного копирования и восстановления данных виртуальных машин и томов платформы виртуализации OpenStack
/opt/rubackup/modules/ rb_module_openstack	Исполняемый модуль, поддерживающий резервное копирование и восстановление данных виртуальных машин и томов платформы виртуализации OpenStack
Пакет rubackup-aerodisk	
/opt/rubackup/etc/ rb_module_aerodisk-vm.conf	Конфигурационный файл модуля резервного копирования и восстановления данных виртуальных машин среды виртуализации Aerodisk VAIR
/opt/rubackup/modules/ rb_module_aerodisk_vm	Исполняемый модуль, поддерживающий резервное копирование и восстановление виртуальных машин среды виртуализации Aerodisk VAIR.
Пакет rubackup-mailion	
/opt/rubackup/etc/ rb_module_mailion.conf	Конфигурационный файл модуля резервного копирования и восстановления данных корпоративной почты Mailion

Структурный элемент	Назначение элемента
/opt/rubackup/modules/ rb_module_mailion	Исполняемый модуль, поддерживающий резервное копирование и восстановлениее данных корпоративной почты Mailion
Пакет rubackup-brest	
/opt/rubackup/etc/ rb_module_brest_vm.conf	Конфигурационный файл модуля резервного копирования виртуальных машин ПК СВ «БРЕСТ»
/opt/rubackup/modules/ rb_module_brest_vm	Исполняемый модуль, поддерживающий резервное копирование и восстановление виртуальных машин ПК СВ «БРЕСТ»
Пакет rubackup-brest-template	
/opt/rubackup/etc/ rb_module_brest_template	Исполняемый модуль, поддерживающий резервное копирование и восстановление шаблонов виртуальных машин ПК СВ «БРЕСТ»
Пакет rubackup-tionix	
/opt/rubackup/etc/ rb_module_tionix.conf	Конфигурационный файл модуля резервного копирования и восстановления виртуальных машин платформы виртуализации TIONIX
/opt/rubackup/modules/ rb_module_tionix	Исполняемый модуль, поддерживающий резервное копирование и восстановление виртуальных машин платформы виртуализации TIONIX
Пакет rubackup-postgres-pro	
/opt/rubackup/etc/ rb_module_postgres-pro.conf	Конфигурационный файл модуля резервного копирования и восстановления кластеров СУБД Postgres Pro
/opt/rubackup/modules/ rb_module_postgres_pro_lib/ rb_module_postgres_pro.bin	Исполняемый файл модуля, поддерживающий резервное копирование и восстановление данных кластеров СУБД Postgres Pro
/opt/rubackup/modules/ rb_module_postgres-pro	Bash-скрипт, ссылающийся на исполняемый файл по пути /opt/rubackup/modules/rb_module_postgres_pro_lib /rb_module_postgres_pro.bin

Структурный элемент	Назначение элемента
Пакет rubackup-pg-dump	
/opt/rubackup/etc/ rb_module_pg_dump_database.conf	Конфигурационный файл модуля резервного копирования и восстановления баз данных СУБД PostgreSQL
/opt/rubackup/modules/ rb_module_pg_dump_table.conf	Конфигурационный файл модуля резервного копирования и восстановления таблиц данных СУБД PostgreSQL
/opt/rubackup/scripts/ rb_pg_dump_script.sql	Скрипт автоматического управления правами роли резервного копирования в ручном режиме
/opt/rubackup/modules/ rb_module_pg_dump_database	Bash-скрипт, ссылающийся на исполняемый файл по пути /opt/rubackup/modules/rb_module_pg_dump_database_lib/rb_module_pg-dump_database.bin
/opt/rubackup/modules/ rb_module_pg_dump_database_lib/ rb_module_pg_dump_database.bin	Исполняемый файл модуля, поддерживающий резервное копирование и восстановление баз данных СУБД PostgreSQL
/opt/rubackup/modules/ rb_module_pg_dump_table_lib/ rb_module_pg_dump_table.bin	Исполняемый файл модуля, поддерживающий резервное копирование и восстановление таблиц данных СУБД PostgreSQL
/opt/rubackup/modules/ rb_module_pg_dump_table	Bash-скрипт, ссылающийся на исполняемый файл по пути /opt/rubackup/modules/rb_module_pg_dump_table_lib/rb_module_pg_dump_table.bin
/opt/rubackup/bin/ rb_pg_dump_script	Консольная утилита для управления правами роли резервного копирования в ручном режиме
Пакет rubackup-isp-vmmanager	
/opt/rubackup/etc/rb_module_isp-vmmanager.conf	Конфигурационный файл модуля резервного копирования и восстановления виртуальных машин среды виртуализации ISP VMmanager
/opt/rubackup/modules/ rb_module_isp-vmmanager	Исполняемый модуль, поддерживающий резервное копирование и восстановление виртуальных машин среды виртуализации ISP

Структурный элемент	Назначение элемента
	VMmanager
Пакет rubackup-freeipa	
/opt/rubackup/etc/ rb_module_freeipa.conf	Конфигурационный файл модуля резервного копирования и восстановления данных службы каталогов FreeIPA
/opt/rubackup/modules/ rb_module_freeipa_lib/ rb_module_freeipa.bin	Исполняемый файл, поддерживающий резервное копирование и восстановление данных службы каталогов FreeIPA
/opt/rubackup/modules/ rb_module_freeipa	Bash-скрипт, ссылающийся на исполняемый файл по пути /opt/rubackup/modules/rb_module_freeipa_lib/rb_module_freeipa.bin
Пакет rubackup_communigate-pro	
/opt/rubackup/etc/ rb_module_communigate-pro.conf	Конфигурационный файл модуля резервного копирования и восстановления конфигурации CommuniGate Pro
/opt/rubackup/etc/ rb_module_communigate_pro_mail.conf	Конфигурационный файл модуля резервного копирования и восстановления писем CommuniGate Pro
/opt/rubackup/modules/ rb_module_communigate_pro_lib/ rb_modules_communigate_pro.bin	Исполняемый файл модуля, поддерживающий резервное копирование и восстановление конфигурации CommuniGate Pro
opt/rubackup/modules/ rb_module_communigate_pro_mail_lib/ rb_modules_communigate_pro_mail.bin	Исполняемый файл модуля, поддерживающий резервное копирование и восстановление данных писем CommuniGate Pro
/opt/rubackup/modules/ rb_module_communigate_pro_mail	Bash-скрипт, ссылающийся на исполняемый скрипт по пути /opt/rubackup/modules/rb_module_communigate_pro_lib/rb_modules_communigate_pro_mail.bin
/opt/rubackup/modules/ rb_module_communigate_pro	Bash-скрипт, ссылающийся на исполняемый скрипт по пути

Структурный элемент	Назначение элемента
	/opt/rubackup/modules/rb_module_communigate_pro_lib/rb_modules_communigate_pro.bin
Пакет rubackup-ovirt-common	
/opt/rubackup/keys/	Папка содержит сертификат и закрытый ключ для внутреннего взаимодействия компонентов СРК RuBackup по протоколу SSL
/opt/rubackup/rc/icons/	Папка содержит иконки интерфейса
/opt/rubackup/copyrights/	Папка содержит файлы лицензионных соглашений
Пакет rubackup-ovirt-client	
/opt/rubackup/log/rbfd	Информация о процессе выполнения создания РК для каждой задачи, которая использует rbfd
/opt/rubackup/var/tasks	Каталог для хранения секретов задач на клиенте
/opt/rubackup/etc/systemd/system/ rubackup_ovirt_client.service	Файл сервиса модуля, необходимый для взаимодействия с приложением через утилиты systemctl и system
/opt/rubackup/etc/ rb_module_ovirt.conf	Конфигурационный файл модуля резервного копирования и восстановления виртуальных машин сред виртуализации oVirt/zVirt/REDVirt
/opt/rubackup/etc/rubackup.lsf	Файл локального расписания Клиента системы резервного копирования
/opt/rubackup/bin/rb_archives	Утилита клиента RuBackup предназначена для просмотра списка резервных копий клиента в системе резервного копирования, создания срочных резервных копий, их удаления, проверки и восстановления. Работает только в том случае, если на клиенте работает служба (сервис, демон) клиента rubackup_client
/opt/rubackup/bin/rbfd	Утилита администратора RuBackup для создания и восстановления полных и инкрементальных резервных копий ресурсов в любых файловых системах. Ресурсом может быть файл, каталог или блочное устройство

Структурный элемент	Назначение элемента
/opt/rubackup/bin/rb_init	Утилита администратора RuBackup для первоначального конфигурирования клиента сразу после развёртывания пакета исполняемых файлов. Неинтерактивный режим необходим для сценариев массового развертывания
/opt/rubackup/bin/rb_schedule	Утилита клиента RuBackup для просмотра правил глобального расписания клиента в системе резервного копирования
/opt/rubackup/bin/rb_tasks	Утилита клиента RuBackup для просмотра списка задач клиента в системе резервного копирования RuBackup
/opt/rubackup/bin/rubackup_client	Клиент резервного копирования RuBackup представляет собой фоновое приложение (сервис, демон), запущенное на хосте клиента и взаимодействующее с сервером RuBackup
/opt/rubackup/modules/ rb_module_ovirt	Утилита резервного копирования и восстановления виртуальных машин сред виртуализации oVirt/zVirt/REDVirt
/opt/rubackup/scripts/test-script.sh	Пример скрипта
/opt/rubackup/log	Папка содержит файлы журнала событий и задач
/opt/rubackup/man/	Папка содержит файлы описаний утилит