



RuBackup

**Система резервного копирования
и восстановления данных**

РАЗВЁРТЫВАНИЕ

ВЕРСИЯ 3.0.0.0.0

Содержание

1. Назначение руководства	6
2. Преимущества распределённой установки	7
3. Этапы распределённой установки	8
4. Порядок развёртывания	9
5. Дистрибутивы	12
5.1. Компакт-диск	12
5.2. Публичный репозиторий	12
5.2.1. Подключение публичного репозитория DEB-систем	12
5.2.2. Подключение публичного репозитория RPM-систем	13
5.3. Облачный диск Астры	14
6. Сетевые порты	16
6.1. Порты управляющего сервера	16
6.2. Порты клиента	17
6.3. Порты медиасервера	18
6.4. Порт служебной базы данных	19
6.5. RuBackup Manager	20
6.6. Порт REST API	20
6.7. Порты экспорта-импорта резервных копий	21
7. Взаимодействие с Kaspersky Endpoint Security	22
7.1. Производительность и потребление ресурсов	22
7.2. Контроль сетевых соединений	22
7.3. Управление доверенными процессами	23
8. Служебная база данных	24
8.1. Системные требования	24
8.2. Установка СУБД	24
8.3. Настройка СУБД	25
8.3.1. Настройка <code>postgresql.conf</code>	25
8.3.2. Настройка <code>pg_hba.conf</code>	26
8.3.3. Настройка файла <code>mswitch.conf</code>	27
8.3.4. Применение изменений	28
8.3.5. Установка пароля пользователя <code>postgres</code>	28
8.4. Настройка SSL соединений	28
8.4.1. Выпуск сертификатов	29
8.4.2. Настройка SSL соединения на сервере PostgreSQL	31
8.4.3. Настройка SSL соединения на машинах компонентов RuBackup	33

Настройка SSL соединения на машинах компонентов RuBackup	33
8.5. Настройка балансировщика нагрузки	35
9. Серверная часть	37
9.1. Системные требования	37
9.1.1. Аппаратные требования	38
Основной/резервный сервер	38
Медиасервер	38
9.1.2. Программные требования	39
9.2. Предварительные настройки	48
9.2.1. Сетевые настройки	48
9.2.2. Настройка переменных среды для пользователя <code>root</code>	48
9.2.3. Настройка SSL соединения с базой данных	48
9.3. Установка на локальной машине	49
9.3.1. Подготовка	50
Подключение публичного репозитория	50
Подключение публичного репозитория DEB-систем	50
Подключение публичного репозитория RPM-систем	50
Установка зависимостей пакетов	52
9.3.2. Установка	54
Последовательность установки	54
Способы установки	54
Обновление конфигурации	54
9.3.3. Настройка	55
С помощью утилиты <code>rb_init</code>	55
С помощью утилиты <code>rb_init_gui</code>	55
Выбор настраиваемой машины	55
Выбор настраиваемого компонента	56
Основные настройки сервера CPK RuBackup	56
9.3.4. Запуск	60
9.4. Установка на удаленной машине	60
9.4.1. Подготовка	61
Управляющая машина	61
Удаленная машина	61
9.4.2. Установка, настройка и запуск	61
Выбор настраиваемой машины	61
Подключение к удаленной машине	62
Установка пакетов на удаленную машину	62

Выбор настраиваемого компонента	63
Основные настройки сервера CPK RuBackup	63
9.5. Автозапуск	67
9.6. Дополнительные настройки	67
9.6.1. Настройка пользователей	68
9.6.2. Настройка переменных среды	68
Добавление в группу	68
9.6.3. Настройка доступа к клиентским сертификатам	69
10. Клиентская часть	70
10.1. Linux	70
10.1.1. Системные требования	70
Аппаратные требования	70
Программные требования	74
10.1.2. Предварительные настройки	78
Сетевые настройки	78
Настройка переменных среды для пользователя <code>root</code>	78
Настройка SSL соединения с базой данных	79
10.1.3. Установка на локальной машине	79
Подготовка	80
Подключение публичного репозитория	80
Установка зависимостей пакетов	82
Установка	84
Последовательность установки	84
Способы установки	84
Обновление конфигурации	84
Настройка	85
С помощью утилиты <code>rb_init</code>	85
С помощью утилиты <code>rb_init_gui</code>	85
Запуск	89
10.1.4. Установка на удаленной машине	90
Подготовка	90
Управляющая машина	90
Удаленная машина	90
Установка, настройка и запуск	90
Выбор настраиваемой машины	90
Подключение к удаленной машине	91
Установка пакетов на удаленную машину	91

10.1.5. Автозапуск	95
10.1.6. Дополнительные настройки	96
Настройка пользователей	96
Настройка переменных среды	96
Добавление в группу	96
Настройка доступа к клиентским сертификатам	97
10.2. Windows	97
10.2.1. Системные требования	97
Аппаратные требования	97
Требования к аппаратным средствам клиента РК	97
Программные требования	99
10.2.2. Предварительные настройки	99
Служебная БД	99
Клиент СРК RuBackup	99
Сетевые настройки	99
Установка Nrcap	99
Установка пакета OpenSSL	99
Установка пакета Microsoft Visual C++	100
10.2.3. Установка на локальной машине	100
Установка	100
Настройка	100
Запуск	100
10.2.4. Установка на удаленной машине	101
Подготовка	101
Управляющая машина	101
Удаленная машина	101
Установка, настройка и запуск	101
Выбор настраиваемой машины	102
Подключение к удаленной машине	102
Установка пакетов на удаленную машину	102
Выбор режима настройки клиента	103
Основные настройки клиента СРК RuBackup (клиент-серверный режим)	103
10.2.5. Автозапуск	105
10.2.6. Настройка Windows Defender	105
11. Работа с сертификатами и ключами SSL	107
11.1. Размещение сертификатов и ключей	107

11.2. Использование цепочки сертификатов	107
11.3. Устранение уязвимостей SSL-соединений	108
11.4. Серверная часть	109
11.4.1. Создание сертификата	109
11.4.2. Подготовка сертификатов для сервера	110
11.4.3. Проверка созданных ключей и сертификатов	111
11.5. Клиентская часть	111
11.5.1. Создание сертификата	111
11.5.2. Подготовка сертификатов для клиента	112
11.5.3. Проверка созданных ключей и сертификатов	112
11.6. Менеджер администратора RuBackup	113
11.6.1. Создание сертификата	113
11.6.2. Проверка созданных ключей и сертификатов	113
12. Результаты установки	114
12.1. Каталог установки	114
12.2. Сетевые сервисы	121
12.3. Конфигурационный файл	121
12.3.1. Параметры конфигурационного файла клиента	121
12.3.2. Параметры конфигурационного файла сервера	126
13. Настройка ограничения на количество открытых файловых дескрипторов на машине сервера RuBackup	139
13.1. Зависимость количества файловых дескрипторов	139
13.2. Расчёт необходимого количества файловых дескрипторов	139
13.3. Способы настройки ограничения количества открытых файловых дескрипторов	141
13.3.1. Настройка ограничения количества открытых файловых дескрипторов при ручном запуске сервера	141
13.3.2. Настройка ограничения количества открытых файловых дескрипторов при запуске сервисов сервера	142

Глава 1. Назначение руководства

Настоящее руководство предназначено для **распределённой установки** основных компонентов СРК:

- *Служебная база данных;*
- *Серверная часть;*
 - основной сервер;
 - резервный сервер;
 - медиасервер;
- *Клиентская часть.*

Глава 2. Преимущества распределённой установки

Распределённая установка:

- рекомендована для средних и больших сред;
- делает систему эффективной;
- обеспечивает централизованное хранение и защиту данных;
- оптимально использует ресурсы организации: вычислительные мощности, сеть и хранилище;
- обеспечивает отказоустойчивость;
- обеспечивает масштабируемость по мере роста инфраструктуры.

Глава 3. Этапы распределённой установки

Очерёдность действий по развёртыванию СРК приведена в разделе [Порядок развёртывания](#)

Глава 4. Порядок развёртывания

Подготовка инфраструктуры СРК

1. Обеспечьте сетевое взаимодействие между машинами с компонентами СРК.
2. Обеспечьте взаимодействие компонентов СРК путем открытия соответствующих [сетевых портов](#) для входящего и исходящего трафика между машинами, на которых будут установлены компоненты СРК.

Установка и настройка служебной базы данных

Разверните поддерживаемую СУБД и настройте подключения к БД серверной группировки СРК (основной сервер, резервный сервер, медиасервер) и АРМ администратора:

- [Установка СУБД](#);
- [Настройка СУБД](#).

[Служебная база данных](#) может быть установлена на машине основного сервера или любом другом доступном по сети машине, удовлетворяющем системным требованиям.

Подготовка к установке

1. Получите [Дистрибутивы](#) компонентов СРК и скопируйте их на машины, на которых будут развёрнуты компоненты СРК.
2. Подготовьте машины к установке компонентов СРК:
 - [Подготовка](#) серверной части;
 - [Подготовка](#) клиентской части под управлением Linux-систем;
 - [Предварительные настройки](#) клиентской части под управлением Windows-систем.

Установка

Установите пакеты компонентов RuBackup на подготовленных машинах:

- [Установка](#) серверной части;
- [Установка](#) клиентской части под управлением Linux-систем;
- [Установка](#) клиентской части под управлением Windows-систем.

Настройка

1. Выполните настройку установленных компонентов СРК в строго определенном порядке:
 - а. [Настройка](#) на серверах (основном, резервном, медиасerverах);
 - б. на всех клиентах ПК:
 - [Настройка](#) под управлением Linux-систем;
 - [Настройка](#) под управлением Windows-систем.
2. Выполните настройки:
 - а. для пользователей, которые будут взаимодействовать с компонентами СРК:
 - [Дополнительные настройки](#) для серверной части;
 - [Дополнительные настройки](#) для клиентской части под управлением Linux-системы;
 - б. для машины под управлением Windows-систем — [Настройка Windows Defender](#).
3. Добавьте сервисы СРК в автозагрузку:
 - [Автозапуск](#) серверной части;
 - [Автозапуск](#) клиентской части под управлением Linux-системы;
 - [Автозапуск](#) клиентской части под управлением Windows-систем.
4. Произведите запуск развёрнутых компонентов СРК:
 - [Запуск](#) серверной части;
 - [Запуск](#) клиентской части под управлением Linux-системы;
 - [Запуск](#) клиентской части под управлением Windows-систем.

Лицензирование

[Получите](#) лицензионные файлы основного сервера (при необходимости, резервного сервера и медиасerverов) у поставщика и [установите](#) лицензию.

Управление СРК

Для управления СРК используйте одно из приложений:

1. для централизованного управления:
 - [Tucana](#);
 - [Утилиты командной строки](#);

2. для локального управления:

- Менеджер клиента RuBackup (RBC);
- Утилиты командной строки.

Глава 5. Дистрибутивы

Для развёртывания компонентов СРК RuBackup получите актуальные установочные пакеты:

- на компакт-диске, полученном от поставщика;
- из дополнительно подключаемого, публичного репозитория;
- с облачного диска по ссылке с официального сайта <https://www.rubackup.ru/go>.

5.1. Компакт-диск

Пакеты для развёртывания СРК RuBackup могут быть получены от поставщика на компакт-диске (CD).

5.2. Публичный репозиторий

Пакеты для развёртывания СРК RuBackup могут быть установлены из дополнительного публичного репозитория.

Публичные репозитории доступны для операционных систем:

- Astra Linux 1.8;
- Astra Linux 1.7;
- Debian 12;
- Ubuntu 22.04;
- Ubuntu 20.04;
- Ubuntu 18.04;
- CentOS 7;
- CentOS 8;
- РЕД ОС 7.3;
- РЕД ОС 8;
- Red Hat Enterprise Linux 9;
- ROSA Fresh Desktop 12;
- ROSA Enterprise Linux Server 7.3;
- ROSA Enterprise Linux Server 7.9.

5.2.1. Подключение публичного репозитория DEB-систем

1. Создайте файл с информацией о репозиториях.

```
cat <<EOF | sudo tee /etc/apt/sources.list.d/rubackup_deb.list
deb https://dl.astralinux.ru/rubackup/repository-deb-main/ <OS-VERSION>
public ①
deb https://dl.astralinux.ru/rubackup/repository-deb-main/ <OS-VERSION>
public-testing ①
EOF
```

① <OS-VERSION> — версия ОС (astra_1.7 | astra_1.8 | debian_12 | ubuntu_18.04 | ubuntu_20.04 | ubuntu_22.04).

2. Добавьте ключ репозитория:

```
sudo wget -qO-
https://dl.astralinux.ru/artifactory/api/security/keypair/gc-astra-
official-repo-key/public | sudo gpg --no-default-keyring --keyring gnupg-
ring:/etc/apt/trusted.gpg.d/rubackup-deb.gpg --import - && sudo chmod 644
/etc/apt/trusted.gpg.d/rubackup-deb.gpg
```

3. Обновите список пакетов:

```
sudo apt update
```

5.2.2. Подключение публичного репозитория RPM-систем

1. Создайте файл с информацией о репозиториях.

Подключение репозитория для ОС CentOS 7, CentOS 8, РЕД ОС 7.3, РЕД ОС 8, Red Hat Enterprise Linux 9, ROSA Fresh Desktop 12, ROSA Enterprise Linux Server 7.9 (менеджер пакетов yum)

```
cat <<EOF | sudo tee /etc/yum.repos.d/rubackup_rpm.repo
[rubackup-rpm-public-repository]
name=rubackup rpm public repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public/ ①
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public/repodata/repomd.xml.key ①
gpgcheck=0

[rubackup-rpm-public-testing-repository]
name=rubackup rpm public testing repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
```

```

VERSION>/public-testing/ ❶
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public-testing/repodata/repomd.xml.key ❶
gpgcheck=0
EOF

```

❶ <OS-VERSION> — версия ОС (centos_7 | centos_8 | redos_7.3 | redos_8 | rhel_9 | rosa_12 | rosa_7.9).

Подключение репозитория для ОС ROSA Enterprise Linux Server 7.3

```

cat <<EOF | sudo tee /etc/yum.repos.d/rubackup_rpm.repo
[rubackup-rpm-public-repository]
name=rubackup rpm public repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public/
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public/repodata/repomd.xml.key
gpgcheck=0
sslverify=0

[rubackup-rpm-public-testing-repository]
name=rubackup rpm public testing repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public-testing/
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public-testing/repodata/repomd.xml.key
gpgcheck=0
sslverify=0
EOF

```

2. Обновите список пакетов:

```
sudo yum update
```

5.3. Облачный диск Астры

Пакеты СРК RuBackup для распространенных операционных систем доступны по

ссылке с официального сайта <https://www.rubackup.ru/go>.

По ссылке вы также найдёте:

```
/Experimental/<OS> ❶  
/Experimental/Scripts/script_block_device_metadata.sh ❷  
/Experimental/Scripts/upgrade_rubackup_packages.sh ❸  
/Rubackup Key ❹
```

- ❶ Экспериментальные установочные пакеты CPK RuBackup, прошедшие только дизайн-тестирование.
- ❷ Экспериментальный скрипт резервного копирования метаданных дедуплицированного пула.
- ❸ Экспериментальный скрипт автоматического обновления.
- ❹ Специализированный загрузочный образ RuBackup.

Глава 6. Сетевые порты

Обмен информацией между компонентами СРК требует технической возможности коммуникации по сети. Перед установкой СРК обеспечьте сетевую связность компонентов СРК, открыв необходимые порты для трафика между узлами, на которых установлены компоненты СРК.

На одном узле может быть установлено более одного компонента СРК.

Подключения выполняются на транспортном уровне по TCP, на прикладном уровне — с использованием TLS.

Компоненты СРК по умолчанию

- обмениваются между собой данными на портах с 9991 по 9995 включительно,
- подключаются к служебной БД на порту, заданному настройками СУБД.

Таблицы и диаграммы ниже призваны помочь сетевому администратору в настройке правил фаерволов между узлами СРК.

6.1. Порты управляющего сервера

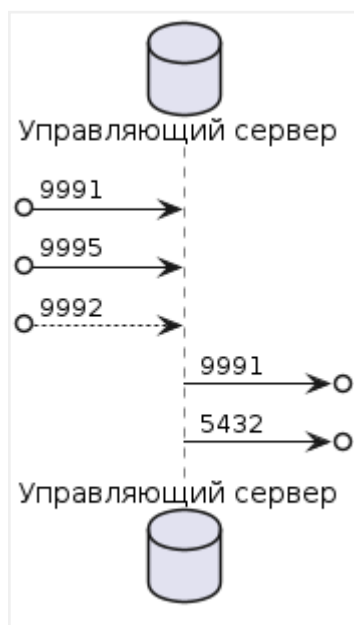


Рисунок 1. Подключения управляющего сервера СРК

По умолчанию управляющий сервер СРК (основной или подменивший его резервный) принимает данные на порту 9995 (сервис `rubackup-rbm`). На этот порт обращаются другие компоненты СРК ([Менеджер администратора RuBackup \(RBM\)](#), [REST API](#), [Утилиты командной строки](#)) для коммуникации с управляющим сервером (например, `rb_health_check`).

На резервном сервере, не принявшем на себя функции основного, порт 9995 не

принимает подключений.

Для проверки доступности основного сервера со стороны резервного используется порт 9991 (сервис `rubackup-cmd`).

Если на узле отсутствует [локальный файл лицензии](#), узел обращается к управляющему серверу на порт 9992.

Управляющий сервер всегда выполняет функции медиасервера с пулом файловой системы по умолчанию. Порты узла с управляющим сервером, выполняющим функции медиасервера, см. в разделе [Порты медиасервера](#).

Порт по умолчанию можно изменить в `/etc/services`. Эти изменения необходимо распространить на все узлы инсталляции СРК.

Таблица 1. Входящие подключения управляющего сервера

Отправитель	Сервис	Порт	Описание
Резервный сервер ^[1]			
Клиент резервного копирования	<code>rubackup-cmd</code>	9991	Обеспечение отказоустойчивости, управление операциями
Менеджер администратора RuBackup (RBM)			
REST API	<code>rubackup-rbm</code>	9995	Управление операциями СРК
Утилиты командной строки			
Клиент резервного копирования	<code>rubackup-lic</code>	9992	Проверка лицензий

Таблица 2. Исходящие подключения управляющего сервера

Адресат	Сервис	Порт	Описание
Медиасервер	<code>rubackup-cmd</code>	9991	Управление операциями на медиасервере
Служебная база данных	<code>postgresql</code>	5432 ^[2]	Сохранение конфигурационной и оперативной информации

6.2. Порты клиента

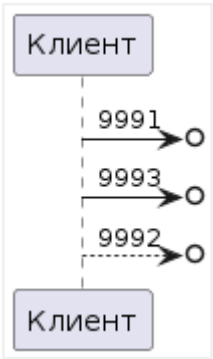


Рисунок 2. Подключения клиента СРК

Клиент СРК не ожидает входящих подключений.

Порт по умолчанию можно изменить в `/etc/services`. Эти изменения необходимо распространить на все узлы инсталляции СРК.

Таблица 3. Исходящие подключения узла клиента

Адресат	Сервис	Порт	Описание
Управляющий сервер	rubackup-cmd	9991	Управление операциями
Управляющий сервер	rubackup-lic	9992	Проверка лицензий
Медиасервер	rubackup-media	9993	Передача данных между медиасервером и клиентом

6.3. Порты медиасервера

Функции медиасервера по умолчанию выполняются в том числе управляющим сервером — с пулом файловой системы; порты медиасервера в этом случае используются и на управляющем сервере ([Порты управляющего сервера](#)).

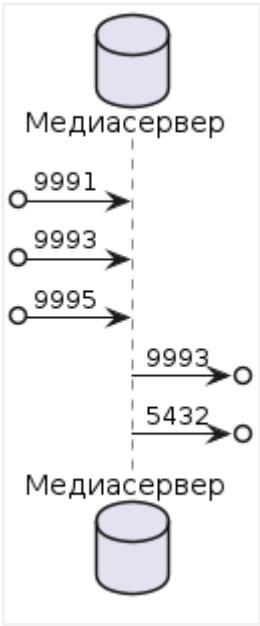


Рисунок 3. Подключения медиа сервера СРК

Медиа сервер или управляющий сервер, выполняющий функции медиа сервера, принимают данные от клиентов и других медиа серверов на порту 9993 .

Порт по умолчанию можно изменить в /etc/services . Эти изменения необходимо распространить на все узлы инсталляции СРК.

Таблица 4. Входящие подключения медиа сервера

Отправитель	Сервис	Порт	Описание
Управляющий сервер	rubackup-cmd	9991	Управление операциями на медиа сервере
Медиа сервер	rubackup-media	9993	Передача данных на медиа сервер клиентом или другим медиа сервером
Клиент			
Утилиты командной строки	rubackup-rbm	9995	Управление операциями СРК

Таблица 5. Исходящие подключения медиа сервера

Адресат	Сервис	Порт	Описание
Медиа сервер	rubackup-media	9993	Передача данных между медиа серверами
Служебная база данных	postgresql	5432 ^[2]	Сохранение конфигурационной и оперативной информации

6.4. Порт служебной базы данных

Компоненты СРК обращаются к служебной базе данных напрямую на порту, ука-

занному в настройках СУБД и [конфигурационном файле](#) (параметр `port`).

При изменении порта доступа к служебной БД необходимо внести изменения в конфигурационные файлы CRK на всех узлах инсталляции.

6.5. RuBackup Manager

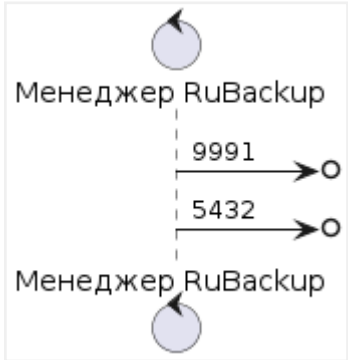


Рисунок 4. Подключения RuBackup Manager

[Менеджер администратора RuBackup \(RBM\)](#) обращается к управляющему серверу и к служебной базе данных.

Таблица 6. Исходящие подключения узла RuBackup Manager

Адресат	Сервис	Порт	Описание
Управляющий сервер	rubackup-cmd	9991	Управление операциями
Служебная база данных	postgresql	5432 ^[2]	Сохранение конфигурационной и оперативной информации

6.6. Порт REST API

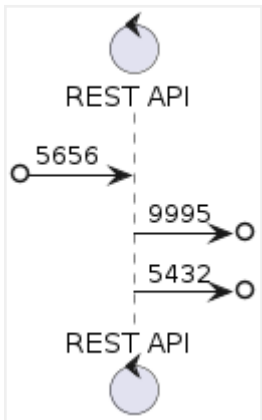


Рисунок 5. Подключения REST API (Tucana)

Сервис [REST API](#), предоставляющий веб-интерфейс [Tucana](#), по умолчанию доступен на порту `5656`.

Порт по умолчанию настраивается через переменные окружения пользователя, от имени которого запущен сервис, или в [конфигурационном файле](#) сервиса.

Таблица 7. Исходящие подключения REST API

Адресат	Сервис	Порт	Описание
Служебная база данных	postgresql	5432 ^[2]	Сохранение конфигурационной и оперативной информации
Управляющий сервер СРК	rubackup-rbm	9995	

6.7. Порты экспорта-импорта резервных копий

Экспортирующий сервер всегда подключается к импортирующему на порту 9993.

Для экспорта резервных копий на экспортирующей стороне используются порты в соответствии с настройкой [Список портов для задач экспорта РК](#).

Если настройка не задана, устанавливается сессия, которая используется на протяжении всей задачи экспорта-импорта. Порт на экспортирующей стороне выбирается случайным образом.

Если в глобальной конфигурации экспортирующей инсталляции СРК указан один или более портов для подключения ([Список портов для задач экспорта РК](#)), то экспортирующий сервер будет подключаться с исходящими портами из этого списка. На каждую задачу экспорта необходим один свободный порт из списка. Если порты из списка израсходованы (заняты выполняемыми задачами экспорта), задача экспорта завершится с ошибкой.

При [ускоренном экспорте-импорте](#) в зависимости от размера экспортируемой резервной копии может использоваться до четырёх подключений к импортирующей стороне (порт 9993). Если диапазон исходящих портов экспорта-импорта ограничен настройкой глобальной конфигурации, исчерпание списка доступных портов будет происходить быстрее.

Таблица 8. Исходящие подключения при экспорте-импорте резервных копий

Адресат	Порт	Описание
Другая инсталляция СРК	В соответствии с настройкой Список портов для задач экспорта РК	Порты экспорта-импорта резервных копий

[1] При наличии резервного сервера.

[2] Если база данных настроена с использованием нестандартного порта, то порт подключения может быть изменен в конфигурационном файле `/opt/rubackup/etc/config.file`.

Глава 7. Взаимодействие с Kaspersky Endpoint Security

Для совместной работы CPK RuBackup и Kaspersky Endpoint Security (KES) требуется выполнить ряд настроек, направленных на предотвращение конфликтов и сохранение производительности.

7.1. Производительность и потребление ресурсов

После развёртывания рекомендуем проверить влияние KES на производительность CPK в следующих сценариях:

1. Нагрузка при работе с резервными копиями

Процессы создания снимков и работа с РК связаны с интенсивной обработкой больших объемов данных. Активное сканирование операций антивирусом KES создает двойную нагрузку на ресурсы системы, что может приводить к значительному замедлению работы с ВМ, повышению общей нагрузки и возникновению непредвиденных сбоев.

Рекомендуется проверить скорость выполнения операций, стабильность работы CPK и KES, а также отсутствие ошибок в журналах событий.

2. Потребление оперативной памяти.

На серверах с объёмом оперативной памяти более 5 ГБ следует установить ограничение потребления памяти для процесса KES в соответствии с рекомендациями производителя.

Рекомендуем проверять потребление памяти KES и отсутствие ошибок нехватки памяти.

Чтобы указать ограничение на использование памяти при проверке файлов откройте файл `/var/opt/kaspersky/kes1/common/kes1.ini` и укажите нужное количество оперативной памяти в МБ для параметра `ScanMemoryLimit`

7.2. Контроль сетевых соединений



Не изменяйте настройки портов CPK по умолчанию (см. [Сетевые порты](#)).

При стандартных настройках порты CPK и KES не пересекаются.

В случае конфликтов, добавьте соответствующие адреса в исключения KES.

KES может блокировать устаревшие протоколы (TLS 1.0, SSL 2.0/3.0), что нару-

шает работу приложений, включая трафик внутри кластера Patroni.

Решение проблемы с блокировкой устаревших протоколов:

1. Обновите KES до версии, которая поддерживает актуальные протоколы.
2. Добавьте все процессы СРК как доверенные.
3. Добавить в исключения KES путь, который используется для монтирования моментальных снимков дисков.

Пример 1. Добавление директорий в исключения KES (Linux)

```
kesl-control --set-settings 1 --add-exclusion /tmp/logs ①
```

```
kesl-control --set-settings 1 --add-exclusion /tmp/logs/*.log ②
```

```
kesl-control --set-settings 1 --add-exclusion /tmp/**/*.log ③
```

- ① Исключение директории с поддиректориями
- ② Исключение файлов по маске
- ③ Рекурсивное исключение файлов по маске

При настройке исключений в Windows необходимо активировать опцию **Применять к дочерним процессам** для корректной работы всех компонентов СРК.

4. Добавьте в исключения KES сертификаты защищённых подключений.
5. Отключите в KES проверку защищённых соединений или блокировку устаревших TLS/SSL-протоколов.
6. Отключите сетевой экран KES.

7.3. Управление доверенными процессами

KES отслеживает действия доверенных процессов. Если процесс пытается получить доступ к директории вне своего доверенного пути, KES может заблокировать этот доступ.

СРК его модули для работы используют как собственные процессы, так и сторонние утилиты (например, для архивации).

Рекомендуем:

1. Добавьте все собственные и сторонние процессы СРК как доверенные.
2. Добавить в исключения KES путь, который используется для монтирования моментальных снимков дисков.

Глава 8. Служебная база данных

В служебной базе данных СРК хранятся:

- метаданные резервных копий;
- настройки СРК.

Служебная база данных может быть расположена на машине основного сервера или любом другом доступном по сети машине, удовлетворяющем системным требованиям.

Служебная БД может находиться под управлением следующих СУБД:

- PostgreSQL версий 11-18;
- Tantor версий 17 и 18.

Место установки

Служебная база данных может быть установлена на машине основного сервера или любом другом доступном по сети машине, удовлетворяющем системным требованиям.

8.1. Системные требования

Таблица 9. Аппаратные требования к серверу БД RuBackup

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



Для обеспечения максимального уровня отказоустойчивости и быстрого действия при промышленной эксплуатации, рекомендуем использовать в качестве служебной базу данных в отказоустойчивой конфигурации с использованием Patroni, развернутой на отдельно стоящих машинах, с совокупным объемом дискового пространства 3.84 ТБ, построенного с использованием твердотельных накопителей, подключенных через шину PCI Express (NVMe SSD).

8.2. Установка СУБД

СРК RuBackup поддерживает СУБД PostgreSQL и СУБД Tantor в качестве [служебных БД](#).

1. Установите ^[1] СУБД, допустимую [в качестве служебной](#).

Установка PostgreSQL

```
apt install postgresql postgresql-contrib
```

Установка Tantor

Установите Tantor в соответствии с официальными инструкциями по установке.

2. Инициализируйте БД, если это необходимо.

При установке PostgreSQL из репозитория в Astra Linux инициализация БД не требуется.

3. Запустите СУБД.

```
systemctl start <service>
```

4. Добавьте запуск СУБД в автозагрузку.

```
systemctl enable <service>
```

8.3. Настройка СУБД

Эта инструкция предполагает, что к [служебной БД](#) будут подключаться машины с адресами от `192.168.0.50` до `192.168.0.53`, а сервер СУБД получает адрес `192.168.0.9`.

8.3.1. Настройка `postgresql.conf`

Разрешите подключение к БД для всех серверов, которые будут входить в серверную группировку RuBackup (основной сервер, резервный сервер, медиасервер).

1. Отредактируйте `postgresql.conf`.

```
# CONNECTIONS AND AUTHENTICATION
#-----
-
# - Connection Settings -

#listen_addresses = 'localhost'
```

```
# what IP address(es) to listen on;

listen_addresses = '192.168.0.9,localhost' ❶
# comma-separated list of addresses;
# defaults to 'localhost'; use '*' for all
# (change requires restart)

port = 5432 ❷
# (change requires restart)

max_connections = 500 ❸
# (change requires restart)
```

- ❶ Установите `listen_addresses = '*'` для подключения с любых интерфейсов (с учётом настроек `pg_hba.conf`).
- ❷ Если установите другой порт подключения к БД по умолчанию, измените порт и в настройках RuBackup.
- ❸ При расчёте максимального количества подключений `max_connections` к БД следует учитывать: при подключении каждого медиасервера добавляется `parallelizm_media × 2 + 9` соединений.

2. Установите параметру `shared_buffers` значение в размере половины от объема оперативной памяти.
3. Установите параметру `max_parallel_workers` значение не менее половины от количества ядер процессора, если сервер СУБД совмещен с сервером RuBackup, и полного количества ядер процессора, если сервер СУБД является выделенным.
4. Сохраните изменения.

8.3.2. Настройка `pg_hba.conf`

Настройте возможность аутентификации с адресов, которые будут входить в серверную группировку RuBackup (основной, резервный, медиасервера), и АРМ администратора RuBackup.

1. Отредактируйте `pg_hba.conf`, указав IP-адреса и маску сети всех подключаемых серверов и АРМ администратора RuBackup к БД по протоколу IPv4.

Пример 2. Пример `pg_hba.conf`

```
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 192.168.0.50/32 md5
host all all 192.168.0.51/32 md5
host all all 192.168.0.52/32 md5
host all all 192.168.0.53/32 md5
```

2. Сохраните изменения.



Подключение к служебной БД с любых адресов

Подключение к служебной БД с любых адресов небезопасно.

При необходимости подключения к БД с любых внешних адресов установите `listen_addresses = '*'` в `postgresql.conf` и `host all all 0.0.0.0/0 md5` в `pg_hba.conf`.

`postgresql.conf`

```
listen_addresses = '*'
```

`pg_hba.conf`

```
host all all 0.0.0.0/0 md5
```

8.3.3. Настройка файла `mswitch.conf`



Этот шаг выполняется только для служебной БД, работающей в ОС Astra Linux Special Edition с максимальным уровнем защищенности («Смоленск»).

Чтобы не возникала ошибка при получении мандатных атрибутов, отредактируйте конфигурационный файл `/etc/parsec/mswitch.conf` в ОС Astra Linux Special Edition с максимальным уровнем защищенности («Смоленск»).

1. Откройте для редактирования файл `/etc/parsec/mswitch.conf` и измените параметр для создания пользователя служебной БД, который не назначен в ОС Astra Linux Special Edition 1.7:

```
sudo nano /etc/parsec/mswitch.conf
```

- Отредактируйте значение указанного параметра, изменив его на `yes`:

```
zero_if_notfound: yes
```

- Сохраните изменения.

8.3.4. Применение изменений

Перезапустите служебную БД для применения изменений.

```
systemctl restart postgresql
```

8.3.5. Установка пароля пользователя `postgres`

- Подключитесь к СУБД от имени пользователя `postgres`.

```
sudo -u postgres psql
```

- Задайте пароль для пользователя `postgres`.

```
alter user postgres password '12345'; ①
```

① `'12345'` — задаваемый пароль пользователя.

- Завершите работу под пользователем `postgres`:

```
\q
```

8.4. Настройка SSL соединений

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

Для настройки SSL соединений:

- Создайте сертификаты для сервера PostgreSQL и его клиентов (postgres-клиентов) (см. [Выпуск сертификатов](#)).
- Выполните настройку конфигурационных файлов на сервере PostgreSQL (см.

[Настройка SSL соединения на сервере PostgreSQL](#)).

3. После установки пакетов компонентов ЦРК скопируйте полученные сертификаты и выполните настройку SSL соединений для postgres-клиентов на машинах (см. [Настройка SSL соединения на машинах компонентов RuBackup](#)):
 - развёрнутой серверной части ЦРК;
 - использующих приложение «Менеджер администратора RuBackup» или «Веб-интерфейс Tuscana».

8.4.1. Выпуск сертификатов

Аутентификация клиента по сертификату позволяет серверу проверить личность подключающегося, подтверждая, что сертификат X.509, представленный postgres-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 которые будут сгенерированы сертификаты Центра сертификации, сервера PostgreSQL и для всех postgres-клиентов (в зависимости от архитектуры вашей ЦРК):

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

где:

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

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:

```
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:

```
cd ./pg-server
```

- Сгенерируйте закрытый ключ для сервера PostgreSQL /pg-server/server.key:

```
openssl genrsa -out server.key 2048
```

- Сгенерируйте запрос на сертификат сервера PostgreSQL /pg-server/server.csr:

```
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:

```
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 (postgres1.key) и выпустив сертификат (postgres1.crt) для всех postgres-клиентов, указав в сертификате соответствующее FQDN хоста, на котором развёрнут компонент CPK.

8.4.2. 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 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:

```
chmod 600 server.crt server.key ca.crt
```

Сделайте владельцем файлов пользователя и группу пользователя `postgres`:

```
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 строки `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] ①
```

① 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` укажите CN сертификата postgres-клиента;
- в столбце `PG-USERNAME` укажите имя пользователя, с которым нужно сопоставить.

6. Для применения изменений перезапустите сервер:

```
sudo systemctl restart postgresql
```

8.4.3. Настройка SSL соединения на машинах компонентов RuBackup

Для подключения серверных компонентов RuBackup и АРМ администратора СРК (использующего приложение «Менеджер администратора RuBackup») к [служебной базе данных PostgreSQL](#) с использованием защищённого соединения выполните приведённые ниже настройки на соответствующих машинах (postgres-клиентах):

- развёрнутой серверной части СРК;
- использующих приложение «Менеджер администратора RuBackup».

Настройка SSL соединения на машинах компонентов RuBackup

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

- сертификат Центра сертификации (`ca.crt`), чтобы postgres-клиент мог проверить, что конечный сертификат сервера PostgreSQL был подписан его доверенным корневым сертификатом;
- сертификат postgres-клиента (машины компонента СРК) (`postgresql.crt`);

- сгенерированный закрытый ключ сервера/клиента CPK (postgresql.key).

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

```
chmod 600 server.crt server.key ca.crt
```

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

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

4. Настройка SSL соединения на машине компонента RuBackup выполняется **после установки пакетов CPK** одним из способов:

- при настройке компонента CPK серверной или клиентской части;
- при внесении правки в файл настроек сервера (полученный после конфигурирования компонента CPK).

5. Для настройки SSL-соединения с БД предварительно необходимо выполнить настройку служебной базы данных в соответствии с разделом [Настройка СУБД](#) и подготовить сертификаты.

- a. Enter sslmode (allow, disable, prefer, require, verify-ca, verify-full) [require]

Enter path for sslrootcert file:

Enter path for sslcert file:

Enter path for sslkey file:

- выберите и введите название выбранного режима SSL в соответствии с [таблицей](#).

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.

sslmode	Защита от прослушивания	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 центра сертификации;
 - в поле `sslcert` укажите расположение сертификата настраиваемого хоста;
 - в поле `sslkey` укажите расположение закрытого ключа настраиваемого хоста.

8.5. Настройка балансировщика нагрузки

При наличии прокси-сервера HAProxy, принимающего запросы к [служебной базе данных](#) СРК, рекомендуем настроить HAProxy и СУБД, обслуживающую служебную базу данных, по этой инструкции.

- В файле `haproxy.cfg` задайте одинаковое значение для параметров `timeout client` и `timeout server`. Рекомендуемое значение `48h` или более.

Согласно официальной документации ^[2] значения параметров `timeout client` и `timeout server` должны быть идентичны.

- Убедитесь, что в настройках служебной БД отсутствуют таймауты, а если присутствуют, установите для них те же значения, что и в настройках HAProxy (см. [пункт 1](#)).
- Добавьте в файл `haproxy.cfg` в строку с проверкой машины СУБД параметр `shutdown-sessions`.

```
server primary 192.168.122.60:3306 check on-marked-down shutdown-sessions
```

4. Завершите все активные задачи в СРК.
5. Остановите сервис сервера СРК.

```
systemctl stop rubackup_server.service
```

6. Перезапустите СУБД.

```
systemctl restart <service>
```

7. Запустите сервис сервера СРК.

```
systemctl start rubackup_server.service
```

[1] Для некоторых ОС может потребоваться подключить дополнительный репозиторий.

[2] <https://docs.haproxy.org/2.6/configuration.html>

Глава 9. Серверная часть

Серверная часть СРК RuBackup может состоять из:

- обязательного компонента — основного сервера;
- одного или нескольких необязательных компонентов — резервного сервера и медиасервера.

Основной сервер

Основной сервер — это главный управляющий сервер, обеспечивающий взаимодействие компонентов СРК.

Основной сервер выполняет функцию медиасервера в случае установки способом «Всё в одном» (все компоненты СРК RuBackup развёрнуты на одной машине).

Резервный сервер

Резервный сервер, в случае отказа основного сервера, поддерживает функционал основного сервера RuBackup, а клиенты системы резервного копирования автоматически подключатся к резервному серверу. После восстановления функционирования основного сервера клиенты подключатся обратно к основному серверу.

Медиасервер

Медиасервер (это машина, на котором подключено устройство хранения) – ёмкое дисковое устройство или библиотека магнитных лент.

Медиасервер наполняет устройство хранения поступающими резервными копиями данных и управляет ими по требованию основного сервера.

Каждый медиасервер ассоциирован с пулом, который содержит логические устройства одного типа — хранилища.

9.1. Системные требования

В данном подразделе приведены системные требования для каждого серверного компонента СРК RuBackup, предъявляемые к техническим средствам, необходимым для нормального функционирования СРК RuBackup.



В случае установки на один хост нескольких компонентов СРК RuBackup (например, при способе установки «Всё в одном») следует консолидировать соответствующие аппаратные требования, предъявляемые к техническому средству, на которое производится установка.

9.1.1. Аппаратные требования

Основной/резервный сервер

Минимальные аппаратные требования, необходимые для стабильного функционирования сервера CPK RuBackup приведены в [таблице](#).

Таблица 11. Аппаратные требования, предъявляемые к серверу RuBackup

Аппаратный компонент	Объем хранимых данных			Примечание
	48 ТБ	96 ТБ	144 ТБ	
Процессор	10 ядер, 20 потоков (2 потока на 1 ядро или более)			Рекомендуемые модели: Intel Xeon 4210, AMD EPYC 7000 или более современные
Оперативная память	128 ГБ	256 ГБ	256 ГБ	—
Твердотельный накопитель (SSD)	RAID 1, 2 диска по 480 ГБ каждый			Объем дискового пространства для установки операционной системы и компонентов RuBackup, за исключением конфигурационной базы данных RuBackup.
Твердотельный накопитель, подключенный через шину PCI Express (NVMe SSD)	3.84 ТБ			Рекомендуется в случае развёртывания инстанса служебной БД 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 рекомендуется использовать в качестве служебной БД в отказоустойчивой конфигурации, например, с использованием решения Patroni, развернутом на отдельных 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	2 сетевых адаптера с пропускной способностью 10 Гб каждый, с 2 портами (dual port)			—

Медиасервер



Начиная с версии CPK 2.6.0.0.0 используется новый механизм сжатия

небольших блоков данных на блочном пуле.

При обновлении СРК с версии 2.5.7.0.0 необходимо увеличить объем оперативной памяти медиасервера на 60%.



В процессе резервного копирования для инкрементальной РК в хранилище на медиасервере резервируется столько же свободного места, сколько и для полной РК, так как объем измененных данных заранее неизвестен.

Рекомендуемая конфигурация медиасервера зависит от совокупного объема хранимых данных и схожа с конфигурацией сервера RuBackup. Для расчета конфигурации медиасервера воспользуйтесь [таблицей](#).

Таблица 12. Аппаратные требования, предъявляемые к медиасерверу

Аппаратный компонент	Объем хранимых данных			Примечание
	48 ТБ	96 ТБ	144 ТБ	
Процессор	10 ядер, 20 потоков (2 потока на 1 ядро или более)			Рекомендуемые модели: Intel Xeon 4210, AMD EPYC 7000 или более современные
Оперативная память	128 ГБ	256 ГБ	256 ГБ	—
Твердотельный накопитель (SSD)	RAID 1, 2 диска по 480 ГБ каждый			Объем дискового пространства для установки операционной системы и компонентов RuBackup, за исключением конфигурационной базы данных RuBackup.
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	2 сетевых адаптера с пропускной способностью 10 Гб каждый, с 2 портами (dual port)			—

9.1.2. Программные требования

Программные требования, необходимые для стабильного функционирования сервера СРК RuBackup:

- операционная система из совместимых с компонентами СРК RuBackup:
 - Astra 1.7;
 - Astra 1.8;
 - CentOS 7;

- CentOS 8;
 - Debian 12;
 - RHEL 9;
 - RedOS 7.3;
 - RedOS 8;
 - Rosa Chrome 12;
 - Rosa Cobalt 7.3;
 - Rosa Cobalt 7.9;
 - Ubuntu 18.04;
 - Ubuntu 20.04;
 - Ubuntu 22.04;
 - Альт 10;
- открытые порты в соответствии с таблицей [Сетевые порты](#);
 - зависимости пакетов для каждой совместимой ОС.



Зависимости приводятся только для подготовки установки СРК из локальных пакетов в закрытых контурах. В случае установки СРК [из публичного репозитория](#) зависимости устанавливаются автоматически из репозитория текущей ОС.

Таблица 13. Зависимости `rubackup-client`, `rubackup-server`, `rubackup-common`

Операционная система	Пакеты
Astra 1.7	exim4-base exim4-config exim4-daemon-light gnupg2 guile-2.2-libs libcurl3 или libcurl4 libevent-2.1-6 libfribidi0 libgc1c2 libgnutls-dane0 libgsasl7 libkyotocabinet16v5 libldap-2.4-2 libltdl7 liblzo2-2 libmailutils5 libmariadb3 libntlm0 libpugixml1v5 libsasl2-2 libunbound8 mailutils или bsd-mailx mailutils-common mariadb-common mysql-common openssl parsec-base parsec-cap parsec-mac psmisc wget xauth

Операционная система	Пакеты
Astra 1.8	exim4-base exim4-config exim4-daemon-light gnupg2 gssasl-common guile-3.0-libs libcurl3 или libcurl4 libevent-2.1-7 libgc1 libgnutls-dane0 libgnutls30 libgssasl18 libgssglue1 libidn12 libldap-2.5-0 libltdl7 libmailutils9 libmariadb3 libncurses6 libncursesw6 libntlm0 libpq5 libpugixml1v5 libsasl2-2 libtinfo6 libunbound8 mailutils или bsd-mailx mailutils-common mariadb-common mysql-common ncurses-base ncurses-bin ncurses-term openssl parsec-base parsec-cap parsec-mac psmisc wget xauth
CentOS 7	cyrus-sasl mailx openldap pugixml qt5-qtbase-gui
CentOS 8	cyrus-sasl mailx openldap pugixml qt5-qtbase-gui

Операционная система	Пакеты
Debian 12	exim4-base exim4-config exim4-daemon-light gnupg2 gsasl-common guile-3.0-libs libcurl3 или libcurl4 libevent-2.1-7 libfribidi0 libgc1 libgnutls-dane0 libgnutls30 libgsasl18 libgssglue1 libidn12 libldap-2.5-0 libltdl7 libmailutils9 libmariadb3 libncurses6 libntlm0 libpq5 libpugixml1v5 libpython3.11 libpython3.11-minimal libpython3.11-stdlib libsasl2-2 libunbound8 mailutils или bsd-mailx mailutils-common mariadb-common mysql-common openssl psmisc python3.11 python3.11-minimal wget xauth
RHEL 9	cyrus-sasl mailx openldap pugixml qt5-qtbase-gui s-nail
RedOS 7.3	cyrus-sasl mailx openldap pugixml qt5-qtbase-gui

Операционная система	Пакеты
RedOS 8	cyrus-sasl libstdc++ mailx openldap pugixml qt5-qtbase-gui
Rosa Chrome 12	cyrus-sasl lib64db5.2 lib64ldap2.4_2 lib64ltdl7 lib64mailutils9 lib64mu_auth9 lib64mu_dbm9 lib64mu_dotmail9 lib64mu_imap9 lib64mu_maildir9 lib64mu_mailer9 lib64mu_mbox9 lib64mu_pop9 lib64mu_sieve9 lib64muaux9 lib64pugixml1 lib64qt5gui5 lib64sasldb2 mailutils mailutils-locales qt5-qtbase-gui

Операционная система	Пакеты
Rosa Cobalt 7.3	cups-libs cyrus-sasl fontconfig fontpackages-filesystem glx-utils libICE libSM libX11 libX11-common libXau libXdamage libXext libXfixes libXi libXrender libXxf86vm libicu libpng libxcb libxshmfence mailx mesa-libEGL mesa-libGL mesa-libgbm mesa-libglapi openldap 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	cyrus-sasl libicu libxkbcommon-x11 mailx openldap qt5-qtbase-gui

Операционная система	Пакеты
Ubuntu 18.04	gnupg2 guile-2.0-libs libcurl3 или libcurl4 libgc1c2 libgsasl7 libkyotocabinet16v5 libldap-2.4-2 libltdl7 liblzo2-2 libmailutils5 libmysqlclient20 libnghttp2-14 libntlm0 libpython2.7 libpython2.7-minimal libpython2.7-stdlib librtmp1 libsasl2-2 mailutils или bsd-mailx mailutils-common mysql-common openssl postfix ssl-cert wget xauth
Ubuntu 20.04	gnupg2 guile-2.2-libs libcurl3 или libcurl4 libgc1c2 libgsasl7 libidn11 libkyotocabinet16v5 libldap-2.4-2 libmailutils6 libmysqlclient21 libntlm0 libpugixml1v5 libsasl2-2 mailutils или bsd-mailx mailutils-common mysql-common openssl postfix ssl-cert wget xauth

Операционная система	Пакеты
Ubuntu 22.04	gnupg2 gsasl-common guile-3.0-libs libcurl3 или libcurl4 libfribidi0 libgc1 libgsasl7 libidn12 libldap-2.5-0 libltdl7 libmailutils8 libmysqlclient21 libntlm0 libpq5 libpugixml1v5 libsasl2-2 mailutils или bsd-mailx mailutils-common mysql-common openssl postfix ssl-cert wget xauth
Альт 10	libldap libsasl2-3 mailutils pugixml qt5-qtbase-gui xauth

Таблица 14. Зависимости rubackup-common-gui

Операционная система	Пакеты
Astra 1.7	gnupg2 wget xauth
Astra 1.8	gnupg2 wget xauth
Debian 12	gnupg2 wget xauth
Rosa Chrome 12	qt5-qtbase-gui
Rosa Cobalt 7.9	libicu libxkbcommon-x11 qt5-qtbase-gui

Операционная система	Пакеты
Ubuntu 18.04	gnupg2 wget xauth
Ubuntu 20.04	gnupg2 wget xauth
Ubuntu 22.04	gnupg2 wget xauth
Альт 10	xauth

9.2. Предварительные настройки

Выполните предварительные настройки на машине, где будет установлен сервер CPK RuBackup.

9.2.1. Сетевые настройки

Если у вас не задействован DNS-сервер, то в файле `/etc/hosts` укажите IP-адрес и соответствующий ему FQDN (или имя хоста) клиента и дополнительных серверов (если они существуют) CPK RuBackup для настройки сетевого взаимодействия.

9.2.2. Настройка переменных среды для пользователя `root`

Настройте переменные среды для пользователя `root`:

1. Авторизуйтесь под пользователем `root`:

```
sudo -i
```

2. Добавьте в файл `/root/.bashrc` строки:

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

3. Перезагрузите переменные окружения:

```
source ~/.bashrc
```

9.2.3. Настройка SSL соединения с базой данных

Пропустите этот шаг, если не требуется защищённое подключение компонентов

RuBackup к [служебной базе данных](#).

Если необходимо использовать для подключения к служебной базе данных защищённое соединение, то выполните приведённые ниже настройки на хостах, на которых развёрнуты компоненты CPK (postgres-клиенты):

1. Перенесите из соответствующей postgres-клиенту папки на машине Центра сертификации подготовленные:
 - сертификат Центра сертификации (`ca.crt`), чтобы клиент CPK мог проверить, что конечный сертификат сервера СУБД был подписан его доверенным корневым сертификатом;
 - сертификат сервера/клиента CPK (`postgresql.crt`);
 - сгенерированный закрытый ключ сервера/клиента CPK (`postgresql.key`).
2. Для файлов сертификата и закрытого ключа установите полный доступ на чтение и запись только для владельцев:

```
chmod 600 server.crt server.key ca.crt
```

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

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

9.3. Установка на локальной машине

На локальной машине установка сервера CPK RuBackup происходит через интерфейс командной строки, настройка — через интерфейс командной строки с помощью утилиты `rb_init` (в интерактивном и неинтерактивном режиме) или через графический интерфейс с помощью утилиты `rb_init_gui`.

Установите и настройте сервер CPK RuBackup на локальной машине.

1. [Ознакомьтесь](#) с системными требованиями.
2. [Выполните](#) предварительные настройки.
3. [Выполните](#) подготовку к установке сервера CPK RuBackup.
4. [Установите](#) сервер CPK RuBackup.
5. [Настройте](#) сервер CPK RuBackup.
6. [Запустите](#) сервисы сервера CPK RuBackup
7. [Выполните](#) дополнительные настройки.

8. [Добавьте](#) сервисы сервера CPK RuBackup в автозапуск.
9. [Получите](#) и [установите](#) лицензию.

9.3.1. Подготовка

Выполните на локальной машине подготовку к установке пакетов сервера CPK RuBackup.

Подключение публичного репозитория



Данный шаг предназначен для установки из публичного репозитория. Если вы устанавливаете локальные пакеты, то пропустите этот шаг.

Подключение публичного репозитория DEB-систем

1. Создайте файл с информацией о репозиториях.

```
cat <<EOF | sudo tee /etc/apt/sources.list.d/rubackup-deb.list
deb https://dl.astralinux.ru/rubackup/repository-deb-main/    <OS-VERSION>
public ❶
deb https://dl.astralinux.ru/rubackup/repository-deb-main/    <OS-VERSION>
public-testing ❶
EOF
```

- ❶ <OS-VERSION> — версия ОС (astra_1.7 | astra_1.8 | debian_12 | ubuntu_18.04 | ubuntu_20.04 | ubuntu_22.04).

2. Добавьте ключ репозитория:

```
sudo wget -qO-
https://dl.astralinux.ru/artifactory/api/security/keypair/gc-astra-
official-repo-key/public | sudo gpg --no-default-keyring --keyring gnupg-
ring:/etc/apt/trusted.gpg.d/rubackup-deb.gpg --import - && sudo chmod 644
/etc/apt/trusted.gpg.d/rubackup-deb.gpg
```

3. Обновите список пакетов:

```
sudo apt update
```

Подключение публичного репозитория RPM-систем

1. Создайте файл с информацией о репозиториях.

Подключение репозитория для ОС CentOS 7, CentOS 8, РЕД ОС 7.3, РЕД ОС 8, Red Hat Enterprise Linux 9, ROSA Fresh Desktop 12, ROSA Enterprise Linux Server 7.9 (менеджер пакетов `yum`)

```
cat <<EOF | sudo tee /etc/yum.repos.d/rubackup_rpm.repo
[rubackup-rpm-public-repository]
name=rubackup rpm public repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public/ ❶
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public/repodata/repomd.xml.key ❶
gpgcheck=0

[rubackup-rpm-public-testing-repository]
name=rubackup rpm public testing repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public-testing/ ❶
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public-testing/repodata/repomd.xml.key ❶
gpgcheck=0
EOF
```

❶ `<OS-VERSION>` — версия ОС (`centos_7` | `centos_8` | `redos_7.3` | `redos_8` | `rhel_9` | `rosa_12` | `rosa_7.9`).

Подключение репозитория для ОС ROSA Enterprise Linux Server 7.3

```
cat <<EOF | sudo tee /etc/yum.repos.d/rubackup_rpm.repo
[rubackup-rpm-public-repository]
name=rubackup rpm public repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public/
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public/repodata/repomd.xml.key
gpgcheck=0
sslverify=0

[rubackup-rpm-public-testing-repository]
name=rubackup rpm public testing repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public-testing/
```

```
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public-testing/repodata/repomd.xml.key
gpgcheck=0
sslverify=0
EOF
```

2. Обновите список пакетов:

```
sudo yum update
```

Установка зависимостей пакетов



Данный шаг предназначен для установки локальных пакетов. Если вы устанавливаете пакеты из репозитория, то пропустите этот шаг.

Для успешной установки компонента СРК RuBackup из локальных пакетов предварительно на машине установите зависимости пакетов, указанные в системных требованиях.

1. Проверьте наличие установленных пакетов зависимостей в ОС, например:

Astra Linux, Debian, Ubuntu

```
dpkg-query -l
```

Альт

```
apt list --installed
```

Rosa Cobalt, RHEL

```
yum list с опцией installed
```

RedOS, CentOS, Rosa Chrome

```
dnf list installed
```

2. Если вы используете операционную систему CentOS 7, CentOS 8 или RHEL 9, то добавьте репозиторий EPEL ^[1], поддерживаемый в рамках проекта Fedora и содержащий некоторые пакеты, которые не вошли в стандартный набор RHEL (CentOS):

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-
latest-8.noarch.rpm
```

Файл репозитория будет автоматически загружен в каталог `/etc/yum.repos.d/epel.repo` и активирован.

3. Если вы используете операционную систему CentOS 7 или CentOS 8, то также рекомендуется включить репозиторий `PowerTools`, поскольку пакеты `EPEL` могут зависеть от пакетов из него:

```
sudo dnf config-manager --set-enabled powertools
```

4. Если вы используете операционную систему RHEL 9, то также рекомендуется включить репозиторий `codeready-builder-for-rhel-8-*` репозиторий `rpm`, поскольку пакеты `EPEL` могут зависеть от пакетов из него:

```
ARCH=$( /bin/arch )

sudo subscription-manager repos --enable "codeready-builder-for-rhel-8-
${ARCH}-rpms"
```

5. Обновите репозитории пакетов в системе:

Astra Linux, Debian, Ubuntu

```
sudo apt update
```

Альт

```
sudo apt-get update
```

Rosa Cobalt, RHEL

```
sudo yum update
```

RedOS, CentOS, Rosa Chrome

```
sudo dnf update
```

6. Установите недостающие зависимости пакетов:

Astra Linux, Debian, Ubuntu

```
sudo apt install <namepackage>
```

Альт

```
sudo apt-get install <namepackage>
```

Rosa Cobalt, RHEL

```
sudo yum install <namepackage>
```

RedOS, CentOS, Rosa Chrome

```
sudo dnf install <namepackage>
```

9.3.2. Установка

Выполните установку пакетов сервера CPK RuBackup на подготовленной локальной машине.

Последовательность установки

1. Обязательные пакеты:

- `rubackup-common`;
- `rubackup-client`;
- `rubackup-server`;

2. Дополнительные пакеты:

- `rubackup-init-gui` и `rubackup-common-gui` (для настройки сервера через графический интерфейс с помощью утилиты `rb_init_gui`);
- `rubackup-rest-api` (для управления CPK через приложение [Tucana](#));
- `rubackup-rbm` (для управления CPK через приложение [Менеджер администратора RuBackup \(RBM\)](#)).

Способы установки

1. Из репозитория.

Пример 3. Установка на ОС Astra Linux, Debian, Ubuntu

```
sudo apt install <namepackage> ①
```

① Имя устанавливаемого пакета.

2. Из локальной папки со скачанными пакетами.

Пример 4. Установка на ОС Astra Linux, Debian, Ubuntu

```
sudo apt install ./<namepackage>.deb
```

Обновление конфигурации

Выполните обновление конфигурации.



Данный шаг выполняется только для ОС Astra Linux Special Edition 1.7 с активированным режимом защитной программной среды!

1. Обновите конфигурацию:

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

2. Примените изменения:

```
sudo reboot
```

9.3.3. Настройка

На локальной машине с установленными пакетами выполните настройку сервера CPK RuBackup.

С помощью утилиты `rb_init`

Настройте сервер CPK RuBackup через интерфейс командной строки с помощью утилиты `rb_init` в [интерактивном](#) или в [неинтерактивном](#) режиме.

Пример 5. Запуск утилиты для настройки в интерактивном режиме

```
rb_init
```

С помощью утилиты `rb_init_gui`

Настройте сервер CPK RuBackup через графический интерфейс с помощью утилиты `rb_init_gui`.

Выбор настраиваемой машины

1. Запустите графическую утилиту `rb_init_gui` на локальной машине.

```
rb_init_gui&
```

2. Выберите язык интерфейса: русский или английский.
3. Примите лицензионное соглашение RuBackup и нажмите **[Далее]**.
4. Выберите для настройки [Локальную машину](#). Откроется страница выбора настраиваемого компонента CPK RuBackup.

Выбор настраиваемого компонента

Выберите настраиваемый компонент:

- основной сервер;
- резервный сервер;
- медиасервер.

Откроется страница с основными настройками сервера CPK RuBackup.

Основные настройки сервера CPK RuBackup

1. Настройте параметры компонента CPK RuBackup.

а. Блок **Общие параметры**.

основной, резервный, медиасервер:

- В поле **Количество сетевых потоков** укажите количество потоков для одновременной обработки задач резервного копирования на сервере (каждый поток имеет отдельное соединение со служебной базой данных CPK).

основной, резервный, медиасервер:

- Из списка **Версия IP для DNS запросов** выберите, какую версию IP использовать для запросов к DNS-серверу: IPv4, IPv6, Обе версии.

основной, резервный, медиасервер:

- Включите **Перезапись мастер-ключа** ☒, чтобы сформировать новый мастер-ключ или перезаписать текущий (при наличии). Мастер-ключ необходим для создания пары ключей электронно-цифровой подписи резервных копий и защитного преобразования резервных копий.

б. Блок **Параметры сервера**.

резервный, медиасервер:

- В поле **Имя основного сервера** укажите IP-адрес или полное доменное имя основного сервера RuBackup.

основной, резервный, медиасервер:

- В поле **Адрес сервера PostgreSQL** ^[2] укажите адрес, на котором развёрнута служебная БД:
- если служебная БД развёрнута на отдельной от основного сервера машине, то укажите адрес соответствующей машины;

- если служебная БД и основной сервер развёрнуты на одной машине, то оставьте значение по умолчанию `localhost`.

основной сервер:

- В поле **Пароль PostgreSQL** ^[2] укажите пароль пользователя базы данных `postgres`.

основной сервер:

- В поле **Имя суперпользователя RuBackup** укажите имя суперпользователя базы данных RuBackup. По умолчанию `rubackup`.

Суперпользователь будет создан в процессе настройки основного сервера.

основной, резервный, медиасервер:

- В поле **Пароль пользователя RuBackup** ^[2] укажите пароль для суперпользователя базы данных RuBackup. По умолчанию `rubackup`.

основной сервер:

- В поле **Имя базы RuBackup** введите имя базы данных, которая будет использоваться в качестве служебной БД. По умолчанию `rubackup`.



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

основной сервер:

- Из списка **Если база уже существует** выберите, обновить (`upgrade`) или удалить (`drop`) служебную БД. При значении `keep` БД будет сохранена в текущем состоянии.
- Включите **Отключить дамп**, чтобы не выполнять резервное копирование для текущей служебной базы данных перед удалением или обновлением. Если **Отключить дамп** выключен (по умолчанию), то перед удалением или обновлением существующей служебной базы данных будет создана резервная копия данных.
- Если **Отключить дамп** выключен, то из списка **Формат дампа** выберите тип резервной копии базы данных:
 - `custom archives`. Данные будут сохраняться в специальном архивном формате. Резервная копия в формате `custom` занимает меньше места на диске, по сравнению с форматом `plain`;
 - `plain`. Данные будут сохраняться в текстовом SQL-скрипте.
- Если выбран **Формат дампа** `custom archives`, то в поле **Уровень сжатия дампа**

укажите 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** ^[2] укажите путь для сохранения резервной копии. По умолчанию указана директория, из которой была вызвана утилита.

основной, резервный, медиасервер:

- Из списка **Сетевой интерфейс** выберите сетевой интерфейс для взаимодействия с системой резервного копирования.

основной сервер:

- В поле **Путь файловой системы для добавления в 'Default'** ^[2] укажите для пула Default каталог для хранения резервных копий.

основной, резервный, медиасервер:

- В поле **Локальный каталог резервного копирования** укажите локальный каталог для временных операций с файлами резервных копий (по умолчанию `/tmp`). Если указанная директория не существует, то она будет создана.

основной, медиасервер:

- В поле **Имя резервного сервера** укажите IP-адрес или полное доменное имя резервного сервера RuBackup.

основной, резервный, 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, резервный, медиасервер:

- В поле **Объём памяти дедупликации, байт** укажите объём для ограничения потребления оперативной памяти сервером при дедупликации резервных копий.

основной, резервный, медиасервер:

- Включите **Непрерывная удалённая репликация** ☒ для выполнения непрерывной удаленной репликации на клиенте. Непрерывная репликация данных обеспечивает актуальность информации на удалённом хосте и осуществляется только в хранилище блочного типа.

основной, резервный, медиасервер:

- Включите **Разрешить централизованное восстановление для клиента** ☐ для восстановления данных из резервной копии с помощью приложения [Менеджер администратора RuBackup \(RBM\)](#).

Если **Разрешить централизованное восстановление для клиента** выключен ☐, то восстановление из резервной копии будет возможно только с помощью утилиты командной строки [Алгоритмы защитного преобразования](#) или приложения [Менеджер клиента RuBackup \(RBC\)](#) на машине с клиентом СРК.

основной, резервный, медиасервер:

- Включите **Создать ключи ЭЦП** ☐, если хотите создать ключи электронно-цифровой подписи. Резервная копия может быть подписана ЭЦП для последующего контроля и предупреждения угрозы её подмены.

основной, резервный, медиасервер:

- Включите **Перезаписать ключи цифровой подписи** ☐ для создания новой связки ключей, используемых для электронно-цифровой подписи.

основной сервер:

- Включите **Аудит безопасности** ☐ для журналирования всех значимых таблиц, кроме очередей задач и временных таблиц.

Дополнительно данной опцией можно управлять с помощью утилиты для работы с журналом событий информационной безопасности [rb_security](#).

- Включите **Аудит задач** ☐ для журналирования всех значимых таблиц и задач в очередях.

Дополнительно данной опцией можно управлять с помощью утилиты для работы с журналом событий информационной безопасности [rb_security](#).

а. Блок **Настройка SSL**.

основной, резервный, медиасервер:

- Включите **Использовать SSL соединение с базой данных** ☐ для настройки безопасного соединения со служебной базой данных RuBackup, а затем настройте параметры:
 - из списка **SSL режим работы с Postgres** — выберите режим работы в зависимости от настроек машины, на котором установлена БД. Если в конфигурации PostgreSQL SSL выключен, то по умолчанию SSL режим будет `disable`. Подробное описание режимов см. в разделе [Настройка SSL соединений](#);
 - в поле **Корневой сертификат** ^[2] укажите полный путь к сертификату

доверенного Центра сертификации. Предварительно сертификат должен быть размещен в `opt/rubackup/keys`;

- в поле **Сертификат клиента** ^[2] укажите полный путь к сертификату (открытому ключу) настраиваемой машины, выданный доверенным Центром сертификации. Предварительно ключ должен быть размещен в `opt/rubackup/keys`;
- в поле **Ключ клиента** ^[2] укажите полный путь к закрытому ключу сертификата настраиваемой машины, выданный доверенным Центром сертификации. Предварительно ключ должен быть размещен в `opt/rubackup/keys`.

b. Блок **Командная строка rb_init**.

В блоке отображается командная строка с итоговыми параметрами компонента.

1. Чтобы выполнить настройку выбранной машины, нажмите **[Далее]**.
2. Нажмите **Да**, чтобы подтвердить выбранные параметры и приступить к настройке. При необходимости подтвердите создание указанных директорий.
3. После успешной настройки нажмите **Запустить** для запуска сервисов компонента.
4. Нажмите **Завершить**, чтобы закончить настройку компонента.

Чтобы настроить другую машину, нажмите **Настроить другую машину**.

9.3.4. Запуск

Запустите сервисы сервера CPK RuBackup на настроенной локальной машине.

1. Запустите сервисы.

```
sudo systemctl start rubackup_client rubackup_server rubackup_api
```

2. Проверьте статус сервисов.

```
sudo systemctl status rubackup_client rubackup_server rubackup_api
```

9.4. Установка на удаленной машине

На удаленной машине установка и настройка сервера CPK RuBackup происходит по SSH через графический интерфейс с помощью утилиты `rb_init_gui`.

При установке и настройке сервера CPK RuBackup по SSH машина с `rb_init_gui` называется управляющей, а машина, на которой устанавливается сервер CPK RuBackup — удаленной.

Установите и настройте сервер CPK RuBackup на удаленной машине.

1. [Ознакомьтесь](#) с системными требованиями.
2. [Выполните](#) предварительные настройки.
3. [Выполните](#) подготовку к установке сервера CPK RuBackup.
4. [Установите, настройте и запустите](#) сервер CPK RuBackup.
5. [Выполните](#) дополнительные настройки.
6. [Добавьте](#) сервисы сервера CPK RuBackup в автозапуск.
7. [Получите](#) и [установите](#) лицензию.

9.4.1. Подготовка

Выполните подготовку к установке пакетов сервера CPK RuBackup на удаленной машине по SSH.

Управляющая машина

1. Скачайте пакеты `rubackup-init-gui` и `rubackup-common-gui` для установки графической утилиты `rb_init_gui` на управляющей машине, `rubackup-common`, `rubackup-client` и `rubackup-server` — для установки сервера CPK RuBackup на удаленной машине.
2. Установите пакеты `rubackup-init-gui` и `rubackup-common-gui`.

```
sudo apt install ./<namepackage>.deb
```

Удаленная машина

Установите и запустите сервис `sshd`.

9.4.2. Установка, настройка и запуск

Выполните установку, настройку и запуск сервера CPK RuBackup на удаленной машине по SSH через графический интерфейс на управляющей машине.

Выбор настраиваемой машины

1. Запустите графическую утилиту `rb_init_gui` на управляющей машине.

```
rb_init_gui&
```

2. Выберите язык интерфейса: русский или английский.
3. Примите лицензионное соглашение RuBackup и нажмите **[Далее]**.
4. Выберите Удаленную машину по SSH. Откроется окно настроек для подключения к удаленной машине.

Подключение к удаленной машине

1. В поле **Имя пользователя** укажите имя пользователя удаленной машины.
2. В поле **Имя хоста** укажите FQDN, имя хоста или IP-адрес удаленной машины для подключения по SSH.
3. В поле **Порт** укажите порт для подключения к удаленной машине по SSH.
4. В поле **Пароль** укажите пароль пользователя.
5. В поле **Пароль sudo** укажите пароль для sudo.
6. Нажмите **[...]** и укажите **Путь до SSH** на управляющей машине. По умолчанию `/usr/bin/ssh`.

Установка пакетов на удаленную машину

1. Для выбора устанавливаемых пакетов включите **Загрузить и установить пакеты**.
2. Включите **Обновить источники перед установкой**, если необходимо обновить информацию о пакетах в репозитории.
3. Нажмите **[...]** и укажите **Путь до SCP** на управляющей машине для передачи пакетов по SCP. По умолчанию `/usr/bin/scp`.
4. Нажмите **[...]** и выберите пакеты для установки. В поле **Выбранные пакеты** отобразится итоговый список пакетов для установки.
5. Нажмите **Продолжить**. Запустится процесс загрузки и установки выбранных пакетов.

Если загрузку и установку пакетов на удаленную машину удалось выполнить успешно, то отобразится информационное сообщение.



Если загрузку и установку пакетов на удаленную машину выполнить не удалось, то отобразится сообщение об ошибке. Чтобы закрыть сообщение, нажмите **Отмена**. Для просмотра информации о загрузке и установке пакетов в журнале событий нажмите **Открыть лог загрузки и установки**. Чтобы закрыть журнал событий нажмите **Назад**. Если нужно перезапустить загрузку и установку пакетов, нажмите **Перезапустить**, в окне настроек повторите пароль пользователя для подключения к удаленному

серверу и выберите пакеты для установки, затем нажмите **Продолжить**.

После успешной установки пакетов перейдите к выбору настраиваемого компонента CPK RuBackup.

Выбор настраиваемого компонента

Выберите настраиваемый компонент:

- основной сервер;
- резервный сервер;
- медиасервер.

Откроется страница с основными настройками сервера CPK RuBackup.

Основные настройки сервера CPK RuBackup

1. Настройте параметры компонента CPK RuBackup.

а. Блок **Общие параметры**.

основной, резервный, медиасервер:

- В поле **Количество сетевых потоков** укажите количество потоков для одновременной обработки задач резервного копирования на сервере (каждый поток имеет отдельное соединение со служебной базой данных CPK).

основной, резервный, медиасервер:

- Из списка **Версия IP для DNS запросов** выберите, какую версию IP использовать для запросов к DNS-серверу: IPv4, IPv6, Обе версии.

основной, резервный, медиасервер:

- Включите **Перезапись мастер-ключа** ☒, чтобы сформировать новый мастер-ключ или перезаписать текущий (при наличии). Мастер-ключ необходим для создания пары ключей электронно-цифровой подписи резервных копий и защитного преобразования резервных копий.

б. Блок **Параметры сервера**.

резервный, медиасервер:

- В поле **Имя основного сервера** укажите IP-адрес или полное доменное имя основного сервера RuBackup.

основной, резервный, медиасервер:

- В поле **Адрес сервера PostgreSQL** ^[2] укажите адрес, на котором развёрнута служебная БД:
- если служебная БД развёрнута на отдельной от основного сервера машине, то укажите адрес соответствующей машины;
- если служебная БД и основной сервер развёрнуты на одной машине, то оставьте значение по умолчанию `localhost`.

основной сервер:

- В поле **Пароль PostgreSQL** ^[2] укажите пароль пользователя базы данных `postgres`.

основной сервер:

- В поле **Имя суперпользователя RuBackup** укажите имя суперпользователя базы данных RuBackup. По умолчанию `rubackup`.

Суперпользователь будет создан в процессе настройки основного сервера.

основной, резервный, медиасервер:

- В поле **Пароль пользователя RuBackup** ^[2] укажите пароль для суперпользователя базы данных RuBackup. По умолчанию `rubackup`.

основной сервер:

- В поле **Имя базы RuBackup** введите имя базы данных, которая будет использоваться в качестве служебной БД. По умолчанию `rubackup`.



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

основной сервер:

- Из списка **Если база уже существует** выберите, обновить (`upgrade`) или удалить (`drop`) служебную БД. При значении `keep` БД будет сохранена в текущем состоянии.
- Включите **Отключить дампы**, чтобы не выполнять резервное копирование для текущей служебной базы данных перед удалением или обновлением. Если **Отключить дампы** выключен (по умолчанию), то перед удалением или обновлением существующей служебной базы данных будет создана резервная копия данных.
- Если **Отключить дампы** выключен, то из списка **Формат дампа** выберите тип резервной копии базы данных:
 - `custom archives`. Данные будут сохраняться в специальном архивном фор-

мате. Резервная копия в формате `custom` занимает меньше места на диске, по сравнению с форматом `plain`;

- `plain`. Данные будут сохраняться в текстовом SQL-скрипте.
- Если выбран **Формат дампа** `custom archives`, то в поле **Уровень сжатия дампа** укажите 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** ^[2] укажите путь для сохранения резервной копии. По умолчанию указана директория, из которой была вызвана утилита.

основной, резервный, медиасервер:

- Из списка **Сетевой интерфейс** выберите сетевой интерфейс для взаимодействия с системой резервного копирования.

основной сервер:

- В поле **Путь файловой системы для добавления в 'Default'** ^[2] укажите для пула Default каталог для хранения резервных копий.

основной, резервный, медиасервер:

- В поле **Локальный каталог резервного копирования** укажите локальный каталог для временных операций с файлами резервных копий (по умолчанию `/tmp`). Если указанная директория не существует, то она будет создана.

основной, медиасервер:

- В поле **Имя резервного сервера** укажите IP-адрес или полное доменное имя резервного сервера RuBackup.

основной, резервный, 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 CPK).

основной, резервный, медиасервер:

- В поле **Объём памяти дедупликации, байт** укажите объём для ограничения потребления оперативной памяти сервером при дедупликации резервных копий.

основной, резервный, медиасервер:

- Включите **Непрерывная удалённая репликация** ☒ для выполнения непрерывной удаленной репликации на клиенте. Непрерывная репликация данных обес-

печивает актуальность информации на удалённом хосте и осуществляется только в хранилище блочного типа.

основной, резервный, медиасервер:

- Включите **Разрешить централизованное восстановление для клиента** ☐ для восстановления данных из резервной копии с помощью приложения [Менеджер администратора RuBackup \(RBM\)](#).

Если **Разрешить централизованное восстановление для клиента** выключен ☐, то восстановление из резервной копии будет возможно только с помощью утилиты командной строки [Алгоритмы защитного преобразования](#) или приложения [Менеджер клиента RuBackup \(RBC\)](#) на машине с клиентом СРК.

основной, резервный, медиасервер:

- Включите **Создать ключи ЭЦП** ☐, если хотите создать ключи электронно-цифровой подписи. Резервная копия может быть подписана ЭЦП для последующего контроля и предупреждения угрозы её подмены.

основной, резервный, медиасервер:

- Включите **Перезаписать ключи цифровой подписи** ☐ для создания новой связки ключей, используемых для электронно-цифровой подписи.

основной сервер:

- Включите **Аудит безопасности** ☐ для журналирования всех значимых таблиц, кроме очередей задач и временных таблиц.

Дополнительно данной опцией можно управлять с помощью утилиты для работы с журналом событий информационной безопасности [rb_security](#).

- Включите **Аудит задач** ☐ для журналирования всех значимых таблиц и задач в очередях.

Дополнительно данной опцией можно управлять с помощью утилиты для работы с журналом событий информационной безопасности [rb_security](#).

а. Блок **Настройка SSL**.

основной, резервный, медиасервер:

- Включите **Использовать SSL соединение с базой данных** ☐ для настройки безопасного соединения со служебной базой данных RuBackup, а затем настройте параметры:
 - из списка **SSL режим работы с Postgres** — выберите режим работы в зависимости от настроек машины, на котором установлена БД. Если в

конфигурации PostgreSQL SSL выключен, то по умолчанию SSL режим будет `disable`. Подробное описание режимов см. в разделе [Настройка SSL соединений](#);

- в поле **Корневой сертификат** ^[2] укажите полный путь к сертификату доверенного Центра сертификации. Предварительно сертификат должен быть размещен в `opt/rubackup/keys`;
- в поле **Сертификат клиента** ^[2] укажите полный путь к сертификату (открытому ключу) настраиваемой машины, выданный доверенным Центром сертификации. Предварительно ключ должен быть размещен в `opt/rubackup/keys`;
- в поле **Ключ клиента** ^[2] укажите полный путь к закрытому ключу сертификата настраиваемой машины, выданный доверенным Центром сертификации. Предварительно ключ должен быть размещен в `opt/rubackup/keys`.

б. Блок **Командная строка rb_init**.

В блоке отображается командная строка с итоговыми параметрами компонента.

1. Чтобы выполнить настройку выбранной машины, нажмите **[Далее]**.
2. Нажмите **Да**, чтобы подтвердить выбранные параметры и приступить к настройке. При необходимости подтвердите создание указанных директорий.
3. После успешной настройки нажмите **Запустить** для запуска сервисов компонента.
4. Нажмите **Завершить**, чтобы закончить настройку компонента.

Чтобы настроить другую машину, нажмите **Настроить другую машину**.

9.5. Автозапуск

Добавьте предварительно запущенные сервисы сервера CPK RuBackup в автозапуск при загрузке ОС.

```
sudo systemctl enable rubackup_client rubackup_server rubackup_api
```

9.6. Дополнительные настройки

Выполните дополнительные настройки на машине, где установлен и настроен сервер CPK RuBackup.

9.6.1. Настройка пользователей

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup или приложения для управления СПК RuBackup (RBM, RBC) должны:

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



Выполните приведённые ниже настройки для пользователей на всех машинах с развёрнутыми компонентами СПК RuBackup.

9.6.2. Настройка переменных среды

Настройте переменные среды для всех пользователей, которые будут работать с СПК RuBackup.

1. Добавьте в файл `/home/<имя пользователя>/.bashrc` строки:

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

2. Перезагрузите переменные окружения:

```
source ~/.bashrc
```

Добавление в группу

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

1. Добавьте пользователя в группу `rubackup`, выполнив команду:

```
sudo usermod -a -G rubackup <имя пользователя>
```

2. Если требуется запуск утилит командной строки RuBackup, RBM или RBC в текущем сеансе пользователя (без перезагрузки ОС) выполните:

```
newgrp rubackup
```

9.6.3. Настройка доступа к клиентским сертификатам

Настройте доступ пользователя, входящего в группу `rubackup`, к каталогам с сертификатами для запуска некоторых утилит командной строки, например, `rb_clients`.

По умолчанию доступ к каталогам есть только у пользователя `root`, для доступа другого пользователя:

1. Измените владельца и группу для каталогов, содержащих сертификаты:

```
sudo chown -R suser:rubackup /opt/rubackup/keys/client/  
sudo chown -R suser:rubackup /opt/rubackup/keys/rootCA/
```

2. Перезапустите сервисы для применения изменений:

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

[1] Выполните установку актуальной версии репозитория EPEL, для примера приведена установка репозитория EPEL 8

[2] Обязательно для заполнения.

Глава 10. Клиентская часть

Клиентская часть СРК RuBackup может состоять из одного или нескольких клиентов резервного копирования, которые могут быть объединены в группы клиентов.

Клиент резервного копирования — это отдельный сервер, компьютер или виртуальная машина, которая содержит данные для резервирования (ресурс) или имеет доступ к ним и на которой установлено клиентское ПО RuBackup для выполнения резервного копирования.

10.1. Linux

10.1.1. Системные требования

В данном подразделе приведены системные требования для каждого клиентского компонента СРК RuBackup, предъявляемые к техническим средствам, необходимым для нормального функционирования СРК RuBackup.

В случае установки на один хост нескольких компонентов СРК RuBackup (например, при способе установки «Всё в одном») следует консолидировать соответствующие аппаратные требования, предъявляемые к техническому средству, на которое производится установка.

Аппаратные требования

Минимальные аппаратные требования, необходимые для стабильного функционирования клиента системы резервного копирования приведены в [таблице](#).

Таблица 15. Аппаратные требования, предъявляемые к Клиенту системы резервного копирования

Аппаратный компонент	Значение
Процессор	1 ядро

Аппаратный компонент	Значение
Оперативная память (RAM) ^[1]	Пример 6. Расчёт RAM при однопоточном режиме резервирования $RAM_1 = 1 + 0,04 \times V_{resource} \quad (Gb)$ Пример 7. Расчёт RAM при многопоточном режиме резервирования $RAM = RAM_1 + RAM_2 + \dots + RAM_N$ • RAM_1 — объём оперативной памяти необходимый для резервирования одного ресурса; • $0,04 \times V_{resource}$ — 4% от размера резервируемого ресурса; • N — количество одновременно резервируемых ресурсов

Аппаратный компонент	Значение
Дисковое пространство (HDD) ^[2]	Пример 8. Формула расчёта дискового пространства $V = \frac{Vol_{resource}}{Size_{block}} \times (Size_{hash} + 20) \times (K + 1) + Size_{metadata}$ где: • $K = 1$ при однопоточном режиме; • $K = worker_parallelism$, если заданы многопоточный режим (<code>enable_multithreading</code>) и слабая дедупликация (<code>enable_flexible_dedup</code>); ◦ <code>worker_parallelism</code> — количество рабочих потоков, используемых для выполнения РК; ◦ <code>enable_multithreading</code> — флаг, указывающий на использование многопоточности; ◦ <code>enable_flexible_dedup</code> — флаг, указывающий на использование гибкой дедупликации; • $Vol_{resource}$ — общий объём данных, подлежащих РК; • $Size_{block}$ — размер блока данных, используемого для обработки данных во время РК (для пулов типов <i>File system</i>, <i>Tape library</i>, <i>Cloud</i> размер блока является фиксированным и равен 16384 Б); • $Size_{hash}$ — размер хеша (длина хеш-функции ÷ 8), используемого для идентификации данных; • 20 — максимальный размер сериализованной позиции в файле; • 1 — временная база для вычисления сигнатуры или отправки хешей на сервер; • $Size_{metadata}$ — это $0,02 \times$ объём ресурса

Таблица 16. Примеры расчётов оперативной памяти и дискового пространства

Ресурс (Б)	Хеш (Б)	Блок (Б)	К	Размер метаданных (Б)	Дисковое пространство (ГБ)
536 870 912 000	64	8192	8	10 737 418 240	56

Ресурс (Б)	Хеш (Б)	Блок (Б)	К	Размер мета- данных (Б)	Дисковое про- странство (ГБ)
536 870 912 000	64	8192	32	10 737 418 2 40	179
536 870 912 000	64	8192	64	10 737 418 2 40	343
536 870 912 000	64	8192	128	10 737 418 2 40	671
536 870 912 000	64	16 384	8	10 737 418 2 40	33
536 870 912 000	64	16 384	32	10 737 418 2 40	94
536 870 912 000	64	16 384	64	10 737 418 2 40	176
536 870 912 000	64	16 384	128	10 737 418 2 40	340
536 870 912 000	64	1 048 576	8	10 737 418 2 40	10
536 870 912 000	64	1 048 576	32	10 737 418 2 40	11
536 870 912 000	64	1 048 576	64	10 737 418 2 40	12
536 870 912 000	64	1 048 576	128	10 737 418 2 40	15
1 099 511 627 776	64	8192	8	21 990 232 5 55	114
1 099 511 627 776	64	8192	32	21 990 232 5 55	366
1 099 511 627 776	64	8192	64	21 990 232 5 55	702
1 099 511 627 776	64	8192	128	21 990 232 5 55	1374
1 099 511 627 776	64	16 384	8	21 990 232 5 55	67
1 099 511 627 776	64	16 384	32	21 990 232 5 55	193
1 099 511 627 776	64	16 384	64	21 990 232 5 55	361
1 099 511 627 776	64	16 384	128	21 990 232 5 55	697
1 099 511 627 776	64	1 048 576	8	21 990 232 5 55	21

Ресурс (Б)	Хеш (Б)	Блок (Б)	К	Размер мета- данных (Б)	Дисковое про- странство (ГБ)
1 099 511 627 776	64	1 048 576	32	21 990 232 5 55	23
1 099 511 627 776	64	1 048 576	64	21 990 232 5 55	25
1 099 511 627 776	64	1 048 576	128	21 990 232 5 55	31

Программные требования

Программные требования, необходимые для стабильного функционирования клиентской части СРК RuBackup:

- операционная система из совместимых с компонентами СРК RuBackup:
 - Astra 1.7;
 - Astra 1.8;
 - CentOS 7;
 - CentOS 8;
 - Debian 12;
 - RHEL 9;
 - RedOS 7.3;
 - RedOS 8;
 - Rosa Chrome 12;
 - Rosa Cobalt 7.3;
 - Rosa Cobalt 7.9;
 - Ubuntu 18.04;
 - Ubuntu 20.04;
 - Ubuntu 22.04;
 - Альт 10;
- открытые порты в соответствии с таблицей [Сетевые порты](#);
- зависимости пакетов для каждой совместимой ОС:

Таблица 17. Зависимости `rubackup-client`

Операционная система	Пакеты
Astra 1.7	gnupg2 openssl parsec-base parsec-cap parsec-mac wget xauth
Astra 1.8	gnupg2 openssl parsec-base parsec-cap parsec-mac wget xauth
CentOS 7	qt5-qtbase-gui
CentOS 8	qt5-qtbase-gui
Debian 12	gnupg2 openssl wget xauth libcurl3 или libcurl4
RHEL 9	qt5-qtbase-gui
RedOS 7.3	qt5-qtbase-gui
RedOS 8	libstdc++ qt5-qtbase-gui
Rosa Chrome 12	lib64qt5gui5 qt5-qtbase-gui

Операционная система	Пакеты
Rosa Cobalt 7.3	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 mesa-libglapi qt5-qtbase qt5-qtbase-common qt5-qtbase-gui qt5-qtbase-gui xcb-util xcb-util-image xcb-util-keysyms xcb-util-renderutil xcb-util-wm
Rosa Cobalt 7.9	libicu libxkbcommon-x11 qt5-qtbase-gui
Ubuntu 18.04	gnupg2 openssl wget xauth
Ubuntu 20.04	gnupg2 openssl wget xauth
Ubuntu 22.04	gnupg2 openssl wget xauth

Операционная система	Пакеты
Альт 10	qt5-qtbase-gui xauth

Таблица 18. Зависимости rubackup-common-gui

Операционная система	Пакеты
Astra 1.7	gnupg2 wget xauth
Astra 1.8	gnupg2 wget xauth
Debian 12	gnupg2 wget xauth
Rosa Chrome 12	qt5-qtbase-gui
Rosa Cobalt 7.9	libicu libxkbcommon-x11 qt5-qtbase-gui
Ubuntu 18.04	gnupg2 wget xauth
Ubuntu 20.04	gnupg2 wget xauth
Ubuntu 22.04	gnupg2 wget xauth
Альт 10	xauth

Таблица 19. Зависимости rubackup-common

Операционная система	Пакеты
Astra 1.7	gnupg2 wget xauth
Astra 1.8	gnupg2 wget xauth
Debian 12	gnupg2 wget xauth
Rosa Chrome 12	qt5-qtbase-gui

Операционная система	Пакеты
Rosa Cobalt 7.9	libicu libxkbcommon-x11 qt5-qtbase-gui
Ubuntu 18.04	gnupg2 wget xauth
Ubuntu 20.04	gnupg2 wget xauth
Ubuntu 22.04	gnupg2 wget xauth
Альт 10	xauth

10.1.2. Предварительные настройки

Выполните предварительные настройки на машине, где будет установлен клиент CPK RuBackup.

Сетевые настройки

Если у вас не задействован DNS-сервер, то в файле `/etc/hosts` укажите IP-адрес и соответствующий ему FQDN (или имя хоста) сервера CPK RuBackup для настройки сетевого взаимодействия.

Настройка переменных среды для пользователя `root`

Настройте переменные среды для пользователя `root`:

1. Авторизуйтесь под пользователем `root`:

```
sudo -i
```

2. Добавьте в файл `/root/.bashrc` строки:

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

3. Перезагрузите переменные окружения:

```
source ~/.bashrc
```

Настройка SSL соединения с базой данных

Пропустите этот шаг, если не требуется защищённое подключение компонентов RuBackup к [служебной базе данных](#).

Если необходимо использовать для подключения к служебной базе данных защищённое соединение, то выполните приведённые ниже настройки на машинах, на которых развёрнуты компоненты CPK (postgres-клиенты):

1. Перенесите из соответствующей postgres-клиенту папки на машине Центра сертификации подготовленные:
 - сертификат Центра сертификации (`ca.crt`), чтобы клиент CPK мог проверить, что конечный сертификат сервера был подписан его доверенным корневым сертификатом;
 - сертификат клиента резервного копирования (`postgresql.crt`);
 - сгенерированный закрытый ключ клиента резервного копирования (`postgresql.key`).
2. Для файлов сертификата и закрытого ключа установите полный доступ на чтение и запись только для владельцев:

```
chmod 600 server.crt server.key ca.crt
```

3. Сделайте владельцем файлов пользователя, от имени которого будет запущен клиент резервного копирования (postgres-клиент):

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

10.1.3. Установка на локальной машине

На локальной машине установка клиента CPK RuBackup происходит через интерфейс командной строки, настройка — через интерфейс командной строки с помощью утилиты `rb_init` (в интерактивном и неинтерактивном режиме) или через графический интерфейс с помощью утилиты `rb_init_gui`.

Установите и настройте клиент CPK RuBackup на локальной машине.

1. [Ознакомьтесь](#) с системными требованиями.
2. [Выполните](#) предварительные настройки.
3. [Выполните](#) подготовку к установке клиента CPK RuBackup.
4. [Установите](#) клиент CPK RuBackup.
5. [Настройте](#) клиент CPK RuBackup.

6. **Запустите** сервис клиента CPK RuBackup
7. **Выполните** дополнительные настройки.
8. **Добавьте** сервис клиента CPK RuBackup в автозапуск.

Подготовка

Выполните на локальной машине подготовку к установке пакетов клиента CPK RuBackup.

Подключение публичного репозитория



Данный шаг предназначен для установки из публичного репозитория. Если вы устанавливаете локальные пакеты, то пропустите этот шаг.

Подключение публичного репозитория DEB-систем

1. Создайте файл с информацией о репозиториях.

```
cat <<EOF | sudo tee /etc/apt/sources.list.d/rubackup_deb.list
deb https://dl.astralinux.ru/rubackup/repository-deb-main/    <OS-VERSION>
public ①
deb https://dl.astralinux.ru/rubackup/repository-deb-main/    <OS-VERSION>
public-testing ①
EOF
```

- ① <OS-VERSION> — версия ОС (astra_1.7 | astra_1.8 | debian_12 | ubuntu_18.04 | ubuntu_20.04 | ubuntu_22.04).

2. Добавьте ключ репозитория:

```
sudo wget -qO-
https://dl.astralinux.ru/artifactory/api/security/keypair/gc-astra-
official-repo-key/public | sudo gpg --no-default-keyring --keyring gnupg-
ring:/etc/apt/trusted.gpg.d/rubackup-deb.gpg --import - && sudo chmod 644
/etc/apt/trusted.gpg.d/rubackup-deb.gpg
```

3. Обновите список пакетов:

```
sudo apt update
```

Подключение публичного репозитория RPM-систем

1. Создайте файл с информацией о репозиториях.

Подключение репозитория для ОС CentOS 7, CentOS 8, РЕД ОС 7.3, РЕД ОС 8, Red Hat Enterprise Linux 9, ROSA Fresh Desktop 12, ROSA Enterprise Linux Server 7.9 (менеджер пакетов `yum`)

```
cat <<EOF | sudo tee /etc/yum.repos.d/rubackup_rpm.repo
[rubackup-rpm-public-repository]
name=rubackup rpm public repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public/ ❶
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public/repodata/repomd.xml.key ❶
gpgcheck=0

[rubackup-rpm-public-testing-repository]
name=rubackup rpm public testing repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public-testing/ ❶
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-main/<OS-
VERSION>/public-testing/repodata/repomd.xml.key ❶
gpgcheck=0
EOF
```

❶ `<OS-VERSION>` — версия ОС (`centos_7` | `centos_8` | `redos_7.3` | `redos_8` | `rhel_9` | `rosa_12` | `rosa_7.9`).

Подключение репозитория для ОС ROSA Enterprise Linux Server 7.3

```
cat <<EOF | sudo tee /etc/yum.repos.d/rubackup_rpm.repo
[rubackup-rpm-public-repository]
name=rubackup rpm public repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public/
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public/repodata/repomd.xml.key
gpgcheck=0
sslverify=0

[rubackup-rpm-public-testing-repository]
name=rubackup rpm public testing repository
baseurl=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public-testing/
```

```
enabled=1
repo_gpgcheck=1
gpgkey=https://dl.astralinux.ru/artifactory/rubackup-rpm-
main/rosa_7.3/public-testing/repodata/repomd.xml.key
gpgcheck=0
sslverify=0
EOF
```

2. Обновите список пакетов:

```
sudo yum update
```

Установка зависимостей пакетов



Данный шаг предназначен для установки локальных пакетов. Если вы устанавливаете пакеты из репозитория, то пропустите этот шаг.

Для успешной установки компонента СРК RuBackup из локальных пакетов предварительно на машине установите зависимости пакетов, указанные в системных требованиях.

1. Проверьте наличие установленных пакетов зависимостей в ОС, например:

Astra Linux, Debian, Ubuntu

```
dpkg-query -l
```

Альт

```
apt list --installed
```

Rosa Cobalt, RHEL

```
yum list с опцией installed
```

RedOS, CentOS, Rosa Chrome

```
dnf list installed
```

2. Если вы используете операционную систему CentOS 7, CentOS 8 или RHEL 9, то добавьте репозиторий EPEL ^[3], поддерживаемый в рамках проекта Fedora и содержащий некоторые пакеты, которые не вошли в стандартный набор RHEL (CentOS):

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-
latest-8.noarch.rpm
```

Файл репозитория будет автоматически загружен в каталог `/etc/yum.repos.d/epel.repo` и активирован.

3. Если вы используете операционную систему CentOS 7 или CentOS 8, то также рекомендуется включить репозиторий `PowerTools`, поскольку пакеты `EPEL` могут зависеть от пакетов из него:

```
sudo dnf config-manager --set-enabled powertools
```

4. Если вы используете операционную систему RHEL 9, то также рекомендуется включить репозиторий `codeready-builder-for-rhel-8-*` репозиторий `rpm`, поскольку пакеты `EPEL` могут зависеть от пакетов из него:

```
ARCH=$( /bin/arch )
```

```
sudo subscription-manager repos --enable "codeready-builder-for-rhel-8-
${ARCH}-rpms"
```

5. Обновите репозитории пакетов в системе:

Astra Linux, Debian, Ubuntu

```
sudo apt update
```

Альт

```
sudo apt-get update
```

Rosa Cobalt, RHEL

```
sudo yum update
```

RedOS, CentOS, Rosa Chrome

```
sudo dnf update
```

6. Установите недостающие зависимости пакетов:

Astra Linux, Debian, Ubuntu

```
sudo apt install <namepackage>
```

Альт

```
sudo apt-get install <namepackage>
```

Rosa Cobalt, RHEL

```
sudo yum install <namepackage>
```

RedOS, CentOS, Rosa Chrome

```
sudo dnf install <namepackage>
```

Установка

Выполните установку пакетов клиента CPK RuBackup на подготовленной машине.

Последовательность установки

1. Обязательные пакеты:

- `rubackup-common`;
- `rubackup-client`.

2. Дополнительные пакеты (для настройки сервера через графический интерфейс с помощью утилиты `rb_init_gui`):

- `rubackup-init-gui`;
- `rubackup-common-gui`.

Способы установки

1. Из репозитория.

Пример 9. Установка на ОС Astra Linux, Debian, Ubuntu

```
sudo apt install <namepackage> ①
```

① Имя устанавливаемого пакета.

2. Из локальной папки со скачанными пакетами.

Пример 10. Установка на ОС Astra Linux, Debian, Ubuntu

```
sudo apt install ./<namepackage>.deb
```

Обновление конфигурации

Выполните обновление конфигурации.



Данный шаг выполняется только для ОС Astra Linux Special Edition 1.7 с активированным режимом защитной программной среды!

1. Обновите конфигурацию:

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

2. Примените изменения:

```
sudo reboot
```

Настройка

На локальной машине с установленными пакетами выполните настройку клиента CPK RuBackup.

С помощью утилиты `rb_init`

Настройте клиент CPK RuBackup через интерфейс командной строки с помощью утилиты `rb_init` в **интерактивном** или в **неинтерактивном** режиме.

Пример 11. Запуск утилиты для настройки в интерактивном режиме

```
rb_init
```

С помощью утилиты `rb_init_gui`

Настройте клиент CPK RuBackup через графический интерфейс с помощью утилиты `rb_init_gui`.

Выбор настраиваемой машины

1. Запустите графическую утилиту `rb_init_gui` на локальной машине.

```
rb_init_gui&
```

2. Выберите язык интерфейса: русский или английский.
3. Примите лицензионное соглашение RuBackup и нажмите **[Далее]**.
4. Выберите для настройки **Локальную машину**. Откроется страница выбора режима настройки клиента CPK RuBackup.

Выбор режима настройки клиента

Выберите режим настройки клиента:

- автономный (предусматривает использование функций CPK без серверной части с сохранением возможности использования любых функциональных

модулей для создания резервных копий);

- клиент-серверный (предусматривает использование всех доступных функций СРК).

Откроется страница с основными настройками клиента СРК RuBackup.

Основные настройки клиента СРК RuBackup

Настройка автономного режима

1. Настройте параметры компонента СРК RuBackup.

а. Блок **Общие параметры**.

- В поле **Количество сетевых потоков** укажите количество потоков для одновременной обработки задач резервного копирования на сервере (каждый поток имеет отдельное соединение со служебной базой данных СРК).
- Из списка **Версия IP для DNS запросов** выберите, какую версию IP использовать для запросов к DNS-серверу: IPv4, IPv6, Обе версии.
- Включите **Перезапись мастер-ключа** ☒, чтобы сформировать новый мастер-ключ или перезаписать текущий (при наличии). Мастер-ключ необходим для создания пары ключей электронно-цифровой подписи резервных копий и защитного преобразования резервных копий.

б. Блок **Параметры автономного клиента**.

- В поле **Каталог архивирования** выберите каталог для временного хранения резервных копий. Если этот параметр не определен в файле конфигурации, то клиент будет запрашивать у медиасервера временное пространство для операций с резервными копиями (NFS папку).
- 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.
- В поле **Тип хранилища резервных копий** выберите тип каталога для хранения резервных копий:
 - локальный каталог — каталог расположен на текущей машине 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 файловой системы LTFS. По умолчанию точка монтирования — каталог `/opt/rubackup/mnt`.

с. Блок **Командная строка rb_init**.

В блоке отображается командная строка с итоговыми параметрами компонента.

2. Чтобы выполнить настройку выбранной машины, нажмите **[Далее]**.
3. Нажмите **Да**, чтобы подтвердить выбранные параметры и приступить к настройке. При необходимости подтвердите создание указанных директорий.
4. После успешной настройки нажмите **Запустить** для запуска сервисов компонента.
5. Нажмите **Завершить**, чтобы закончить настройку компонента.

Чтобы настроить другую машину, нажмите **Настроить другую машину**.

Настройка клиент-серверного режима

1. Настройте параметры компонента CPK RuBackup.

а. Блок **Общие параметры**.

- В поле **Количество сетевых потоков** укажите количество потоков для одновременной обработки задач резервного копирования на сервере (каждый поток имеет отдельное соединение со служебной базой данных CPK).
- Из списка **Версия IP для DNS запросов** выберите, какую версию IP использовать для запросов к DNS-серверу: `IPv4`, `IPv6`, `Обе версии`.
- Включите **Перезапись мастер-ключа** ☒, чтобы сформировать новый мастер-ключ или перезаписать текущий (при наличии). Мастер-ключ необходим для создания пары ключей электронно-цифровой подписи резервных копий и защитного преобразования резервных копий.

b. Блок **Параметры клиент-серверного режима.**

- В поле **Имя основного сервера** укажите IP-адрес или FQDN основного сервера RuBackup (в соответствии с настройками файла `hosts` машины основного сервера).
- В поле **Имя резервного сервера** укажите IP-адрес или FQDN резервного сервера RuBackup (в соответствии с настройками файла `hosts` машины основного сервера).
- В поле **Сетевой интерфейс** выберите сетевой интерфейс, посредством которого клиенту ПК разрешено взаимодействовать с системой резервного копирования.
- 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.
- В поле **Количество параллельных задач** укажите количество потоков для одновременной обработки задач резервного копирования на медиа-сервере (каждый поток имеет отдельное соединение со служебной базой данных CPK).
- В поле **Объём памяти дедупликации, байт** для ограничения потребления оперативной памяти сервером при дедупликации резервных копий.

При использовании дедупликации рекомендуется минимальный объём оперативной памяти сервера 64 GB `effective_cache_size` $\approx 70\%$ от размера оперативной памяти `work_mem` 32 MB.

- При необходимости включите **Непрерывная удалённая репликация** ☐. Непрерывная удалённая репликация осуществляется только в хранилище блочного типа.
- При необходимости включите **Разрешать централизованное восстановление для клиента** ☐ для восстановления данных из резервной копии в приложении «Менеджер администратора RuBackup» (RBM), с помощью консольной утилиты `rbfd` или приложения «Менеджер клиента RuBackup» (RBC).

В случае деактивированного переключателя ☐ восстановление из резервной копии будет возможно с помощью консольной утилиты `rbfd` или приложения «Менеджер клиента RuBackup» на машине клиента резервного копирования.

Централизованное восстановление данных из резервной копии с помощью приложения «Менеджер администратора RuBackup» (используемом

на любой машине) будет отключено.

- При необходимости включите **Создать ключи ЭЦП** **О**, если хотите создать ключи электронно-цифровой подписи.

Резервная копия может быть подписана цифровой подписью для последующего контроля и предупреждения угрозы её подмены.

- При необходимости включите **Системный мониторинг для клиента** **О**, если хотите включить системный мониторинг для данного клиента.

Файл мониторинга производительности системных компонентов будет размещён в папке `/opt/rubackup/monitoring/`.

- При необходимости включите **Перезаписать ключи цифровой подписи** **О**, для создания новой связки ключей, используемых для электронно-цифровой подписи.

с. Блок **Командная строка rb_init**.

В блоке отображается командная строка с итоговыми параметрами компонента.

2. Чтобы выполнить настройку выбранной машины, нажмите **[Далее]**.
3. Нажмите **Да**, чтобы подтвердить выбранные параметры и приступить к настройке. При необходимости подтвердите создание указанных директорий.
4. После успешной настройки нажмите **Запустить** для запуска сервисов компонента.
5. Нажмите **Завершить**, чтобы закончить настройку компонента.

Чтобы настроить другую машину, нажмите **Настроить другую машину**.

Запуск

Запустите сервис клиента СРК RuBackup на настроенной локальной машине.

1. Запустите сервис.

```
sudo systemctl start rubackup_client
```

2. Проверьте статус сервиса.

```
sudo systemctl status rubackup_client
```

10.1.4. Установка на удаленной машине

На удаленной машине установка и настройка клиента CPK RuBackup происходит по SSH через графический интерфейс с помощью утилиты `rb_init_gui`.

При установке и настройке клиента CPK RuBackup по SSH машина с `rb_init_gui` называется управляющей, а машина, на которой устанавливается клиент CPK RuBackup — удаленной.

Установите и настройте клиент CPK RuBackup на удаленной машине.

1. [Ознакомьтесь](#) с системными требованиями.
2. [Выполните](#) предварительные настройки.
3. [Выполните](#) подготовку к установке клиента CPK RuBackup.
4. [Установите, настройте и запустите](#) клиент CPK RuBackup.
5. [Выполните](#) дополнительные настройки.
6. [Добавьте](#) сервис клиента CPK RuBackup в автозапуск.

Подготовка

Выполните подготовку к установке пакетов клиента CPK RuBackup на удаленной машине по SSH.

Управляющая машина

1. Скачайте пакеты `rubackup-init-gui` и `rubackup-common-gui` для установки графической утилиты `rb_init_gui` на управляющей машине, `rubackup-common` и `rubackup-client` — для установки клиента CPK RuBackup на удаленной машине.
2. Установите пакеты `rubackup-init-gui` и `rubackup-common-gui`.

```
sudo apt install ./<namepackage>.deb
```

Удаленная машина

Установите и запустите сервис `sshd`.

Установка, настройка и запуск

Выполните установку, настройку и запуск клиента CPK RuBackup на удаленной машине по SSH через графический интерфейс на управляющей машине.

Выбор настраиваемой машины

1. Запустите графическую утилиту `rb_init_gui` на управляющей машине.

```
rb_init_gui&
```

2. Выберите язык интерфейса: русский или английский.
3. Примите лицензионное соглашение RuBackup и нажмите **[Далее]**.
4. Выберите Удаленную машину по SSH. Откроется окно настроек для подключения к удаленной машине.

Подключение к удаленной машине

1. В поле **Имя пользователя** укажите имя пользователя удаленной машины.
2. В поле **Имя хоста** укажите FQDN, имя хоста или IP-адрес удаленной машины для подключения по SSH.
3. В поле **Порт** укажите порт для подключения к удаленной машине по SSH.
4. В поле **Пароль** укажите пароль пользователя.
5. В поле **Пароль sudo** укажите пароль для sudo.
6. Нажмите **[...]** и укажите **Путь до SSH** на управляющей машине. По умолчанию `/usr/bin/ssh`.

Установка пакетов на удаленную машину

1. Для выбора устанавливаемых пакетов включите **Загрузить и установить пакеты**.
2. Включите **Обновить источники перед установкой**, если необходимо обновить информацию о пакетах в репозитории.
3. Нажмите **[...]** и укажите **Путь до SCP** на управляющей машине для передачи пакетов по SCP. По умолчанию `/usr/bin/scp`.
4. Нажмите **[...]** и выберите пакеты для установки. В поле **Выбранные пакеты** отобразится итоговый список пакетов для установки.
5. Нажмите **Продолжить**. Запустится процесс загрузки и установки выбранных пакетов.

Если загрузку и установку пакетов на удаленную машину удалось выполнить успешно, то отобразится информационное сообщение.



Если загрузку и установку пакетов на удаленную машину выполнить не удалось, то отобразится сообщение об ошибке. Чтобы закрыть сообщение, нажмите **Отмена**. Для просмотра информации о загрузке и установке пакетов в журнале событий нажмите **Открыть лог загрузки и установки**. Откроется журнал событий. Чтобы закрыть журнал событий нажмите **Назад**. Если нужно перезапустить загрузку и установку пакетов, нажмите **Перезапустить**. Откроется окно настроек для подключения к удаленной

машине. Повторите пароль пользователя, выберите пакеты для установки и нажмите **Продолжить**.

После успешной установки пакетов перейдите к выбору режима настройки клиента CPK RuBackup.

Выбор режима настройки клиента

Выберите режим настройки клиента:

- автономный (предусматривает использование функций CPK без серверной части с сохранением возможности использования любых функциональных модулей для создания резервных копий);
- клиент-серверный (предусматривает использование всех доступных функций CPK).

Откроется страница с основными настройками клиента CPK RuBackup.

Основные настройки клиента CPK RuBackup

Настройка автономного режима

1. Настройте параметры компонента CPK RuBackup.

а. Блок **Общие параметры**.

- В поле **Количество сетевых потоков** укажите количество потоков для одновременной обработки задач резервного копирования на сервере (каждый поток имеет отдельное соединение со служебной базой данных CPK).
- Из списка **Версия IP для DNS запросов** выберите, какую версию IP использовать для запросов к DNS-серверу: IPv4, IPv6, Обе версии.
- Включите **Перезапись мастер-ключа** ☒, чтобы сформировать новый мастер-ключ или перезаписать текущий (при наличии). Мастер-ключ необходим для создания пары ключей электронно-цифровой подписи резервных копий и защитного преобразования резервных копий.

б. Блок **Параметры автономного клиента**.

- В поле **Каталог архивирования** выберите каталог для временного хранения резервных копий. Если этот параметр не определен в файле конфигурации, то клиент будет запрашивать у медиасервера временное пространство для операций с резервными копиями (NFS папку).
- 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.
- В поле **Тип хранилища резервных копий** выберите тип каталога для хранения резервных копий:
 - локальный каталог — каталог расположен на текущей машине 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 файловой системы LTFS. По умолчанию точка монтирования — каталог `/opt/rubackup/mnt`.

с. Блок **Командная строка rb_init**.

В блоке отображается командная строка с итоговыми параметрами компонента.

2. Чтобы выполнить настройку выбранной машины, нажмите **[Далее]**.
3. Нажмите **Да**, чтобы подтвердить выбранные параметры и приступить к настройке. При необходимости подтвердите создание указанных директорий.
4. После успешной настройки нажмите **Запустить** для запуска сервисов компонента.
5. Нажмите **Завершить**, чтобы закончить настройку компонента.

Чтобы настроить другую машину, нажмите **Настроить другую машину**.

Настройка клиент-серверного режима

1. Настройте параметры компонента CPK RuBackup.

а. Блок **Общие параметры**.

- В поле **Количество сетевых потоков** укажите количество потоков для

одновременной обработки задач резервного копирования на сервере (каждый поток имеет отдельное соединение со служебной базой данных СРК).

- Из списка **Версия IP для DNS запросов** выберите, какую версию IP использовать для запросов к DNS-серверу: IPv4, IPv6, Обе версии.
- Включите **Перезапись мастер-ключа** ☐, чтобы сформировать новый мастер-ключ или перезаписать текущий (при наличии). Мастер-ключ необходим для создания пары ключей электронно-цифровой подписи резервных копий и защитного преобразования резервных копий.

б. Блок **Параметры клиент-серверного режима**.

- В поле **Имя основного сервера** укажите IP-адрес или FQDN основного сервера RuBackup (в соответствии с настройками файла hosts машины основного сервера).
- В поле **Имя резервного сервера** укажите IP-адрес или FQDN резервного сервера RuBackup (в соответствии с настройками файла hosts машины основного сервера).
- В поле **Сетевой интерфейс** выберите сетевой интерфейс, посредством которого клиенту ПК разрешено взаимодействовать с системой резервного копирования.
- 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.
- В поле **Количество параллельных задач** укажите количество потоков для одновременной обработки задач резервного копирования на медиа-сервере (каждый поток имеет отдельное соединение со служебной базой данных СРК).
- 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.

- При необходимости включите **Непрерывная удалённая репликация** ☐. Непрерывная удалённая репликация осуществляется только в хранилище блочного типа.
- При необходимости включите **Разрешать централизованное восстановление для клиента** ☐ для восстановления данных из резервной копии в

приложении «Менеджер администратора RuBackup» (RBM), с помощью консольной утилиты `rbfd` или приложения «Менеджер клиента RuBackup» (RBC).

В случае деактивированного переключателя **О** восстановление из резервной копии будет возможно с помощью консольной утилиты `rbfd` или приложения «Менеджер клиента RuBackup» на машине клиента резервного копирования.

Централизованное восстановление данных из резервной копии с помощью приложения «Менеджер администратора RuBackup» (используемом на любой машине) будет отключено.

- При необходимости включите **Создать ключи ЭЦП О**, если хотите создать ключи электронно-цифровой подписи.

Резервная копия может быть подписана цифровой подписью для последующего контроля и предупреждения угрозы её подмены.

- При необходимости включите **Системный мониторинг для клиента О**, если хотите включить системный мониторинг для данного клиента.

Файл мониторинга производительности системных компонентов будет размещён в папке `/opt/rubackup/monitoring/`.

- При необходимости включите **Перезаписать ключи цифровой подписи О**, для создания новой связки ключей, используемых для электронно-цифровой подписи.

с. Блок **Командная строка rb_init**.

В блоке отображается командная строка с итоговыми параметрами компонента.

2. Чтобы выполнить настройку выбранной машины, нажмите **[Далее]**.
3. Нажмите **Да**, чтобы подтвердить выбранные параметры и приступить к настройке. При необходимости подтвердите создание указанных директорий.
4. После успешной настройки нажмите **Запустить** для запуска сервисов компонента.
5. Нажмите **Завершить**, чтобы закончить настройку компонента.

Чтобы настроить другую машину, нажмите **Настроить другую машину**.

10.1.5. Автозапуск

Добавьте предварительно запущенный сервис клиента СРК RuBackup в автозапуск при загрузке ОС.

```
sudo systemctl enable rubackup_client
```

10.1.6. Дополнительные настройки

Выполните дополнительные настройки на машине, где установлен и настроен клиент СРК RuBackup.

Настройка пользователей

Пользователи, от имени которых будет осуществляться запуск утилит командной строки RuBackup или приложения для управления СРК RuBackup (RBM, RBC) должны:

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



Выполните приведённые ниже настройки для пользователей на всех машинах с развёрнутыми компонентами СРК RuBackup.

Настройка переменных среды

Настройте переменные среды для всех пользователей, которые будут работать с СРК RuBackup.

1. Добавьте в файл `/home/<имя пользователя>/.bashrc` строки:

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

2. Перезагрузите переменные окружения:

```
source ~/.bashrc
```

Добавление в группу

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

1. Добавьте пользователя в группу `rubackup`, выполнив команду:

```
sudo usermod -a -G rubackup <имя пользователя>
```

2. Если требуется запуск утилит командной строки RuBackup, RBM или RBC в

текущем сеансе пользователя (без перезагрузки ОС) выполните:

```
newgrp rubackup
```

Настройка доступа к клиентским сертификатам

Настройте доступ пользователя, входящего в группу `rubackup`, к каталогам с сертификатами для запуска некоторых утилит командной строки, например, `rb_clients`.

По умолчанию доступ к каталогам есть только у пользователя `root`, для доступа другого пользователя:

1. Измените владельца и группу для каталогов, содержащих сертификаты:

```
sudo chown -R suser:rubackup /opt/rubackup/keys/client/
sudo chown -R suser:rubackup /opt/rubackup/keys/rootCA/
```

2. Перезапустите сервисы для применения изменений:

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

10.2. Windows

10.2.1. Системные требования

В данном подразделе приведены системные требования для каждого клиентского компонента СРК RuBackup, предъявляемые к техническим средствам, необходимым для нормального функционирования СРК RuBackup.

Аппаратные требования

Требования к аппаратным средствам клиента РК

Узел, выполняющий функции клиента РК, на котором предполагается развёртывание, должен обладать характеристиками, приведёнными в таблице [Требования к аппаратным средствам клиента РК](#).

Таблица 20. Требования к аппаратным средствам клиента РК

Аппаратное требование	Значение	Примечание
-----------------------	----------	------------

Процессор	Однопоточный режим 1 ядро	Многopotочный режим Количество ядер = количеству потоков	—
Твердотельный накопитель	Значение требуемого дискового пространства может быть рассчитано по формуле	Не менее 400 ГБ	
Оперативная память	Сумма значений оперативной памяти для всех задач резервного копирования	Где оперативная память одного ресурса равна 1ГБ + 4% от размера целевого ресурса	
Интерфейсное устройство	Сетевой адаптер	—	

Пример 12. Формула расчёта дискового пространства

$$V = \frac{Vol_{resource}}{Size_{block}} \times (Size_{hash} + 20) \times (K + 1) + Size_{metadata}$$

где:

- $K = 1$ при однопоточном режиме;
- $K = worker_parallelism$, если заданы многопоточный режим (`enable_multithreading`) и слабая дедупликация (`enable_flexible_dedup`);
 - `worker_parallelism` — количество рабочих потоков, используемых для выполнения РК;
 - `enable_multithreading` — флаг, указывающий на использование многопоточности;
 - `enable_flexible_dedup` — флаг, указывающий на использование гибкой дедупликации;
- $Vol_{resource}$ — общий объём данных, подлежащих РК;
- $Size_{block}$ — размер блока данных, используемого для обработки данных во время РК (для пулов типов *File system*, *Tape library*, *Cloud* размер блока является фиксированным и равен 16384 Б);
- $Size_{hash}$ — размер хеша (длина хеш-функции ÷ 8), используемого для идентификации данных;
- 20 — максимальный размер сериализованной позиции в файле;
- 1 — временная база для вычисления сигнатуры или отправки хешей на сервер;
- $Size_{metadata}$ — это $0,02 \times$ объём ресурса

Программные требования

Программные требования к среде функционирования клиентской части СРК RuBackup:

- одна из 64-битных операционных систем:
 - Windows Server 2012;
 - Windows Server 2016;
 - Windows Server 2019;
 - Windows Server 2022;
- библиотека Npcap;
- библиотека OpenSSL версия 3.3.0, установленная в директорию C:\OpenSSL-Win64;
- пакет Microsoft Visual C++ версия 2015.

10.2.2. Предварительные настройки

Служебная БД

Разрешите использование символа \ в конфигурационном файле postgresql.conf на машине [служебной базы данных](#) (standard_conforming_strings равен on).

Клиент СРК RuBackup

Выполните предварительные настройки на машине, где будет установлен клиент СРК RuBackup.

Сетевые настройки

Если у вас не задействован DNS-сервер, то в файле C:\Windows\System32\drivers\etc\hosts укажите IP-адрес и соответствующий ему FQDN (или имя хоста) сервера СРК RuBackup для настройки сетевого взаимодействия.

Установка Npcap

Скачайте с официального сайта пакет npcap-`<version>`.exe и установите Npcap.

Установка пакета OpenSSL

Скачайте с официального сайта пакет Win64openssl-3.3.0.exe и установите OpenSSL в C:\OpenSSL-Win64. Добавьте в переменную окружения PATH пути C:\OpenSSL-Win64 и C:\OpenSSL-Win64\bin.

Установка пакета Microsoft Visual C++

Скачайте с официального сайта пакеты `vc_redist.x86.exe`, `vc_redist.x64.exe` версии 2015 и установите Microsoft Visual C++.

10.2.3. Установка на локальной машине

На локальной машине установка клиента CPK RuBackup происходит через интерфейс командной строки, настройка — через интерфейс командной строки с помощью утилиты `rb_init` в интерактивном режиме.

Установите и настройте клиент CPK RuBackup на локальной машине.

1. [Ознакомьтесь](#) с системными требованиями.
2. [Выполните](#) предварительные настройки.
3. [Установите](#) клиент CPK RuBackup.
4. [Настройте](#) клиент CPK RuBackup.
5. [Запустите](#) сервис клиента CPK RuBackup.
6. [Добавьте](#) сервис клиента CPK RuBackup в автозапуск.
7. [Настройте](#) Windows Defender.

Установка

[Скачайте](#) на локальную машину пакет клиента CPK RuBackup `RuBackup_client_installer<version>.exe`, где `<version>` — актуальная версия пакета, запустите его с правами администратора и следуйте инструкциям.

Для ОС Windows Server версий 2012 и 2016 перезагрузите ОС для применения настроек.

Настройка

На локальной машине с установленным пакетом запустите утилиту `rb_init` и настройте клиент CPK RuBackup в [интерактивном режиме](#).

Пример 13. Запуск утилиты для настройки в интерактивном режиме

```
start C:\RuBackup-win-client\bin\rb_init.exe
```

Запуск

Запустите сервис клиента CPK RuBackup на настроенной локальной машине.

1. Откройте **Диспетчер серверов** → **Средства** → **Службы**.

2. Выберите **RuBackup Service** и запустите его.

10.2.4. Установка на удаленной машине

Графический интерфейс `rb_init_gui` доступен только на ОС семейства Linux, но позволяет установить и настроить клиент CPK RuBackup по SSH на удаленной машине под управлением ОС Windows.

При установке и настройке клиента CPK RuBackup по SSH машина с `rb_init_gui` называется управляющей, а машина, на которой устанавливается клиент CPK RuBackup — удаленной.

Установите и настройте клиент CPK RuBackup на удаленной машине.

1. [Ознакомьтесь](#) с системными требованиями.
2. [Выполните](#) предварительные настройки.
3. [Выполните](#) подготовку к установке клиента CPK RuBackup.
4. [Установите, настройте и запустите](#) клиент CPK RuBackup.
5. [Добавьте](#) сервис клиента CPK RuBackup в автозапуск.
6. [Настройте](#) Windows Defender.

Подготовка

Выполните подготовку к установке пакетов клиента CPK RuBackup на удаленной машине по SSH.

Управляющая машина

1. Скачайте пакеты `rubackup-init-gui` и `rubackup-common-gui` для установки графической утилиты `rb_init_gui` на управляющей машине, `RuBackup_client_installer<version>.exe` — для установки клиента CPK RuBackup на удаленной машине.
2. Установите пакеты `rubackup-init-gui` и `rubackup-common-gui`.

```
sudo apt install ./<namepackage>.deb
```

Удаленная машина

Установите и запустите сервис OpenSSH Server.

Установка, настройка и запуск

Выполните установку, настройку и запуск клиента CPK RuBackup на удаленной машине по SSH через графический интерфейс на управляющей машине.

Выбор настраиваемой машины

1. Запустите графическую утилиту `rb_init_gui` на управляющей машине.

```
rb_init_gui&
```

2. Выберите язык интерфейса: русский или английский.
3. Примите лицензионное соглашение RuBackup и нажмите **[Далее]**.
4. Выберите Удаленную машину по SSH. Откроется окно настроек для подключения к удаленной машине.

Подключение к удаленной машине

1. В поле **Имя пользователя** укажите имя пользователя удаленной машины.
2. В поле **Имя хоста** укажите FQDN, имя хоста или IP-адрес удаленной машины для подключения по SSH.
3. В поле **Порт** укажите порт для подключения к удаленной машине по SSH.
4. В поле **Пароль** укажите пароль пользователя.
5. В поле **Пароль sudo** укажите пароль для sudo.
6. Нажмите **[...]** и укажите **Путь до SSH** на управляющей машине. По умолчанию `/usr/bin/ssh`.

Установка пакетов на удаленную машину

1. Для выбора устанавливаемых пакетов включите **Загрузить и установить пакеты**.
2. Включите **Обновить источники перед установкой**, если необходимо обновить информацию о пакетах в репозитории.
3. Нажмите **[...]** и укажите **Путь до SCP** на управляющей машине для передачи пакетов по SCP. По умолчанию `/usr/bin/scp`.
4. Нажмите **[...]** и выберите пакеты для установки. В поле **Выбранные пакеты** отобразится итоговый список пакетов для установки.
5. Нажмите **Продолжить**. Запустится процесс загрузки и установки выбранных пакетов.

Если загрузку и установку пакетов на удаленную машину удалось выполнить успешно, то отобразится информационное сообщение.



Если загрузку и установку пакетов на удаленную машину выполнить не удалось, то отобразится сообщение об ошибке. Чтобы закрыть сообщение, нажмите **Отмена**. Для просмотра информации о загрузке и установке пакетов в журнале событий нажмите **Открыть лог загрузки и установки**.

Откроется журнал событий. Чтобы закрыть журнал событий нажмите **Назад**. Если нужно перезапустить загрузку и установку пакетов, нажмите **Перезапустить**. Откроется окно настроек для подключения к удаленной машине. Повторите пароль пользователя, выберите пакеты для установки и нажмите **Продолжить**.

После успешной установки пакетов на удаленной машине перейдите к выбору режима настройки клиента CPK RuBackup.

Выбор режима настройки клиента

Выберите клиент-серверный режим настройки (предусматривает использование всех доступных функций CPK). Откроется страница с основными настройками клиента CPK RuBackup.

Основные настройки клиента CPK RuBackup (клиент-серверный режим)

1. Настройте параметры компонента CPK RuBackup.

а. Блок **Общие параметры**.

- В поле **Количество сетевых потоков** укажите количество потоков для одновременной обработки задач резервного копирования на сервере (каждый поток имеет отдельное соединение со служебной базой данных CPK).
- Из списка **Версия IP для DNS запросов** выберите, какую версию IP использовать для запросов к DNS-серверу: IPv4, IPv6, Обе версии.
- Включите **Перезапись мастер-ключа** ☒, чтобы сформировать новый мастер-ключ или перезаписать текущий (при наличии). Мастер-ключ необходим для создания пары ключей электронно-цифровой подписи резервных копий и защитного преобразования резервных копий.

б. Блок **Параметры клиент-серверного режима**.

- В поле **Имя основного сервера** укажите IP-адрес или FQDN основного сервера RuBackup (в соответствии с настройками файла hosts машины основного сервера).
- В поле **Имя резервного сервера** укажите IP-адрес или FQDN резервного сервера RuBackup (в соответствии с настройками файла hosts машины основного сервера).
- В поле **Сетевой интерфейс** выберите сетевой интерфейс, посредством которого клиенту РК разрешено взаимодействовать с системой резервного копирования.
- 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.

- В поле **Количество параллельных задач** укажите количество потоков для одновременной обработки задач резервного копирования на медиа-сервере (каждый поток имеет отдельное соединение со служебной базой данных CPK).
- В поле **Объём памяти дедупликации, байт** для ограничения потребления оперативной памяти сервером при дедупликации резервных копий.

При использовании дедупликации рекомендуется минимальный объём оперативной памяти сервера 64 GB `effective_cache_size` $\approx 70\%$ от размера оперативной памяти `work_mem` 32 MB.

- При необходимости включите **Непрерывная удалённая репликация** ☐. Непрерывная удалённая репликация осуществляется только в хранилище блочного типа.
- При необходимости включите **Разрешать централизованное восстановление для клиента** ☐ для восстановления данных из резервной копии в приложении «Менеджер администратора RuBackup» (RBM), с помощью консольной утилиты `rbfd` или приложения «Менеджер клиента RuBackup» (RBC).

В случае деактивированного переключателя ☐ восстановление из резервной копии будет возможно с помощью консольной утилиты `rbfd` или приложения «Менеджер клиента RuBackup» на машине клиента резервного копирования.

Централизованное восстановление данных из резервной копии с помощью приложения «Менеджер администратора RuBackup» (используемом на любой машине) будет отключено.

- При необходимости включите **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.

- При необходимости включите **Системный мониторинг для клиента** ☐, если хотите включить системный мониторинг для данного клиента.

Файл мониторинга производительности системных компонентов будет размещён в папке `/opt/rubackup/monitoring/`.

- При необходимости включите **Перезаписать ключи цифровой под-**

писи **О**, для создания новой связки ключей, используемых для электронно-цифровой подписи.

с. Блок **Командная строка rb_init**.

В блоке отображается командная строка с итоговыми параметрами компонента.

1. Чтобы выполнить настройку выбранной машины, нажмите **[Далее]**.
2. Нажмите **Да**, чтобы подтвердить выбранные параметры и приступить к настройке. При необходимости подтвердите создание указанных директорий.
3. После успешной настройки нажмите **Запустить** для запуска сервисов компонента.
4. Нажмите **Завершить**, чтобы закончить настройку компонента.

Чтобы настроить другую машину, нажмите **Настроить другую машину**.

10.2.5. Автозапуск

Добавьте предварительно запущенный сервис клиента СРК RuBackup в автозапуск при загрузке ОС.

1. Откройте **Диспетчер серверов** → **Средства** → **Службы**.
2. Выберите **RuBackup Service** → **Свойства** → вкладка **Общие**.
3. Для параметра **Тип запуска** установите значение **Автоматически**.
4. Нажмите **ОК** для сохранения изменений.

10.2.6. Настройка Windows Defender

При использовании антивируса Windows Defender на машине с клиентом СРК RuBackup с помощью PowerShell исключите папку `C:\RuBackup-win-client` из автоматической проверки.

```
Add-MpPreference -ExclusionPath 'C:\RuBackup-win-client'
```

[1] 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.

[2] Резервное копирование: объем свободного дискового пространства, составляющий не менее 3 % от совокупного объема данных, резервное копирование которых осуществляется одновременно. Восстановление данных: объем свободного дискового пространства должен быть не менее совокупного объема одновременно восстанавливаемых данных с использованием данного клиента. Многопоточное резервное копирование: объем свободного дискового пространства зависит от выбранных параметров: количества потоков, размера блока и длины хеша. Чем больше используется потоков, тем больше требуемый объем. Чем меньше выбранный размер блока, тем больше требуется доступного пространства на диске. Чем больше длина хеша, тем больше требуется памяти. Расчет требуемого объема: приблизительный расчет требуемого объема доступного пространства в многопоточном режиме можно оценить как `worker_parallelism` × процент от ресурса. Это

означает, что для каждого рабочего потока, который будет использоваться при многопоточной обработке данных, требуется определённый объём доступного пространства на диске.

[3] Выполните установку актуальной версии репозитория EPEL, для примера приведена установка репозитория EPEL 8

Глава 11. Работа с сертификатами и ключами SSL

В этом разделе описан процесс создания собственных ключей и сертификатов вместо тех, которые входят в стандартную поставку RuBackup. В комплекте поставки RuBackup есть необходимые для работы SSL-сертификаты клиента и сервера.

Сертификаты, необходимые для работы RuBackup, располагаются в каталоге `/opt/rubackup/keys` и предоставляются в составе пакета `rubackup-common`.

В процессе подключения к серверу клиент отправляет свой сертификат `/opt/rubackup/keys/client/clientCert.crt` для проверки подлинности клиента сервером. Также клиент принимает от сервера его сертификат `/opt/rubackup/keys/server/serverCert.crt` и проверяет его подлинность с использованием серверного корневого сертификата `/opt/rubackup/keys/rootCA/serverRootCACert.crt`. Сервер проверяет подлинность полученного клиентского сертификата с помощью клиентского корневого сертификата `/opt/rubackup/keys/rootCA/clientRootCACert.crt`.

При подключении к серверу оконный менеджер отправляет свой сертификат `/opt/rubackup/keys/rbm/rbmCert.crt` на проверку. Также он принимает от сервера его сертификат `/opt/rubackup/keys/server/serverCert.crt` и проверяет его подлинность с использованием серверного корневого сертификата `/opt/rubackup/keys/rootCA/serverRootCACert.crt`. Сервер проверяет подлинность полученного сертификата оконного менеджера с помощью клиентского корневого сертификата `/opt/rubackup/keys/rootCA/clientRootCACert.crt`.

Для взаимодействия с сервером лицензий и проверки его на подлинность используется корневой сертификат сервера лицензий `/opt/rubackup/keys/rootCA/licenseServerRootCACert.crt`.

11.1. Размещение сертификатов и ключей

Файлы приватных ключей следует хранить в надёжном месте, недоступном ни с сервера, ни с клиента RuBackup.

При замене сертификатов на собственные необходимо убедиться, что все сертификаты обновлены на всех узлах, где установлены компоненты RuBackup: клиент, сервер, медиасервер, резервный сервер, оконный менеджер, REST API сервис и другие.

11.2. Использование цепочки сертификатов

Иногда клиентский или серверный сертификат подписывается не корневым кли-

ентским или серверным сертификатом, а промежуточным сертификатом, который, в свою очередь, подписан корневым или следующим промежуточным сертификатом. Это называется цепочкой сертификатов.

Чтобы RuBackup мог работать с такой цепочкой сертификатов, необходимо объединить все промежуточные и корневой сертификаты в единый корневой клиентский или серверный сертификат.

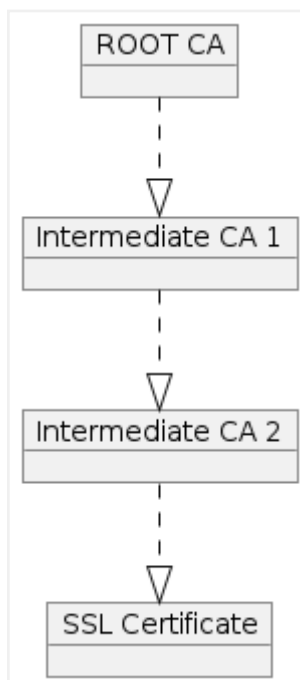


Рисунок 6. Цепочка сертификатов

11.3. Устранение уязвимостей SSL-соединений

SSL/TLS обладает уязвимостями, которые необходимо устранить, отключив небезопасные шифры и алгоритмы шифрования.

Выполните следующие действия на каждом узле (серверах, клиентах, медиасерверах) инсталляции СРК.

1. Откройте `/etc/ssl/openssl.cnf` на редактирование, и добавьте в секцию `[openssl_init]` строку `ssl_conf = ssl_conf_section`.

Добавление указателя на раздел `ssl_conf_section`

```

[openssl_init]
providers = provider_section
engines = engine_section
ssl_conf = ssl_conf_section ❶
  
```

- ❶ Строка-указатель на раздел `ssl_conf_section`.

2. Добавьте в конец файла разделы `[ssl_conf_section]` и

```
[system_default_section].
```

Добавление разделов `ssl_conf_section` и `system_default_section`

```
[ssl_conf_section]
system_default = system_default_section

[system_default_section]
CipherString = ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:!aNULL:!eNULL:!LOW:!MEDIUM:!EXP:!CBC:!SHA1:!MD5:!PSK:!SRP
CipherSuites =
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
MinProtocol = TLSv1.2
MaxProtocol = TLSv1.3
```

Перезагрузите каждый из узлов инсталляции CPK, на котором вносились изменения.

11.4. Серверная часть

11.4.1. Создание сертификата

Чтобы создать серверный сертификат, выполните следующие шаги:

1. Создайте приватный ключ для серверного корневого сертификата командой:

```
openssl genrsa -out serverRootCAKey.key 2048
```



Храните этот ключ в надежном месте!

2. Создайте серверный корневой сертификат. В представленном примере сертификат действует 20000 дней:

```
openssl req -x509 -new -nodes -key serverRootCAKey.key -days 20000 -out /opt/rubakup/keys/rootCA/serverRootCACert.crt
```

3. В интерактивном меню введите двухбуквенный код страны, провинцию, город, организацию, подразделение, Common Name и e-mail адрес.
4. Создайте приватный ключ сервера:

```
openssl genrsa -out /opt/rubackup/keys/server/serverKey.key 2048
```

5. Создайте запрос на подпись:

```
openssl req -new -key /opt/rubackup/keys/server/serverKey.key -out  
/opt/rubackup/keys/server/serverCert.csr
```

6. В интерактивном меню впишите ответ на те же вопросы, что и при создании корневого сертификата. Введенный **Common Name** должен отличаться от **Common Name** у корневого сертификата.

7. Создайте серверный сертификат и подпишите его серверным корневым сертификатом. В представленном примере сертификат действует 20000 дней:

```
openssl x509 -req -in /opt/rubackup/keys/server/serverCert.csr -CA  
/opt/rubackup/keys/rootCA/serverRootCACert.crt -CAkey serverRootCAKey.key  
-CAcreateserial -out /opt/rubackup/keys/server/serverCert.crt -days 20000
```

8. При необходимости пересоздайте файл, используемый в алгоритме Диффи-Хеллмана, для обмена сессионными ключами с клиентом:

```
openssl dhparam -out /opt/rubackup/keys/server/dh_2048.pem 2048
```

11.4.2. Подготовка сертификатов для сервера

Чтобы подготовить сертификаты для сервера, выполните следующие шаги:

1. Разместите в отдельной папке промежуточные сертификаты и корневой сертификат.
2. Если некоторые из промежуточных или корневой сертификат имеют расширение **.cer** или **.pem**, конвертируйте их в формат **.crt** с помощью одной из следующих команд:

```
openssl x509 -in '<имя сертификата>.pem' -out '<имя сертификата>.crt'  
-outform DER  
  
openssl x509 -inform PEM -in '<имя сертификата>.cer' -out '<имя  
сертификата>.crt'
```

3. Объедините промежуточные сертификаты и корневой сертификаты в единый корневой серверный сертификат:

```
cat <путь к промежуточному сертификату 1> <путь к промежуточному
сертификату 2> <путь к корневому сертификату>
/opt/rubackup/keys/rootCA/serverRootCACert.crt
```

11.4.3. Проверка созданных ключей и сертификатов

```
openssl verify -no-CApath -CAfile
/opt/rubackup/keys/rootCA/serverRootCACert.crt
/opt/rubackup/keys/server/serverCert.crt
```

Вывод команды должен содержать: **OK**.

11.5. Клиентская часть

11.5.1. Создание сертификата

Чтобы создать клиентский сертификат, выполните следующие шаги:

1. Создайте приватный ключ для клиентского корневого сертификата командой:

```
openssl genrsa -out clientRootCAKey.key 2048
```



Храните этот ключ в надежном месте!

2. Создайте клиентский корневой сертификат. В представленном примере сертификат действует 20000 дней:

```
openssl req -x509 -new -nodes -key serverRootCAKey.key -days 20000 -out
/opt/rubackup/keys/rootCA/clientRootCACert.crt
```

3. В интерактивном меню введите двухбуквенный код страны, провинцию, город, организацию, подразделение, Common Name и e-mail адрес.
4. Создайте приватный ключ клиента:

```
openssl genrsa -out /opt/rubackup/keys/client/clientKey.key 2048
```

5. Создайте запрос на подпись:

```
openssl req -new -key /opt/rubackup/keys/client/clientKey.key -out
```

```
/opt/rubackup/keys/client/clientCert.csr
```

- В интерактивном меню впишите ответ на те же вопросы, что и при создании корневого сертификата. Введенный `Common Name` должен отличаться от `Common Name` у корневого сертификата.
- Создайте клиентский сертификат и подписать его клиентским корневым сертификатом. В представленном примере сертификат действует 20000 дней:

```
openssl x509 -req -in /opt/rubackup/keys/client/clientCert.csr -CA
/opt/rubackup/keys/rootCA/clientRootCACert.crt -CAkey clientRootCAKey.key
-CAcreateserial -out /opt/rubackup/keys/client/clientCert.crt -days 20000
```

11.5.2. Подготовка сертификатов для клиента

Чтобы подготовить сертификаты для клиента, выполните следующие шаги:

- Разместите в отдельной папке промежуточные сертификаты и корневой сертификат.
- Если некоторые из промежуточных или корневой сертификат имеют расширение `.cer` или `.pem`, конвертируйте их в формат `.crt` с помощью одной из следующих команд:

```
openssl x509 -in '<имя сертификата>.pem' -out '<имя сертификата>.crt'
-outform DER

openssl x509 -inform PEM -in '<имя сертификата>.cer' -out '<имя
сертификата>.crt'
```

- Объедините промежуточные сертификаты и корневой сертификат в единый корневой клиентский сертификат:

```
cat <путь к промежуточному сертификату 1> <путь к промежуточному
сертификату 2> <путь к корневому сертификату>
/opt/rubackup/keys/rootCA/clientRootCACert.crt
```

11.5.3. Проверка созданных ключей и сертификатов

```
openssl verify -no-CApath -CAfile
/opt/rubackup/keys/rootCA/clientRootCACert.crt
```

```
/opt/rubackup/keys/client/clientCert.crt
```

Вывод команды должен содержать: **ОК**.

11.6. Менеджер администратора RuBackup

11.6.1. Создание сертификата

Чтобы создать сертификат, выполните следующие шаги:

1. Создайте приватный ключ оконного менеджера:

```
openssl genrsa -out /opt/rubackup/keys/rbm/rbmKey.key 2048
```

2. Создайте запрос на подпись:

```
openssl req -new -key /opt/rubackup/keys/rbm/rbmKey.key -out  
/opt/rubackup/keys/rbm/rbmCert.csr
```

3. В интерактивном меню впишите ответ на те же вопросы, что и при создании корневого сертификата. Введенный **Common Name** должен отличаться от **Common Name** у корневого сертификата.
4. Создайте сертификат оконного менеджера и подпишите его клиентским корневым сертификатом. В представленном примере сертификат действует 20000 дней:

```
openssl x509 -req -in /opt/rubackup/keys/rbm/rbmCert.csr -CA  
/opt/rubackup/keys/rootCA/clientRootCACert.crt -CAkey clientRootCAKey.key  
-CAcreateserial -out /opt/rubackup/keys/rbm/rbmCert.crt -days 20000
```

11.6.2. Проверка созданных ключей и сертификатов

```
openssl verify -no-CApath -CAfile /opt/rubackup/keys/rootCA/clientRootCACert  
.crt /opt/rubackup/keys/rbm/rbmCert.crt
```

Вывод команды должен содержать: **ОК**.

Глава 12. Результаты установки

12.1. Каталог установки

При установке инсталляционный rpm/deb-пакет будет автоматически распакован в директорию:

- для *Linux-систем*: `/opt/rubackup`.
- для *Windows-систем*: `C:\RuBackup-win-client\`

Структура установленных пакетов CPK RuBackup приведена в [таблице](#).

Таблица 21. Структура установленных пакетов CPK RuBackup

Структурный элемент	Назначение элемента
<code>/opt/rubackup</code>	Директория, в которой распакован установочный комплект компонента RuBackup, а также используемые дополнительные инструменты
Пакет rubackup-common	
<code>/opt/rubackup/keys/client/</code>	Папка содержит сертификат и закрытый ключ клиента для внутреннего взаимодействия компонентов CPK по протоколу SSL
<code>/opt/rubackup/keys/server/</code>	Папка содержит сертификат и закрытый ключ сервера для внутреннего взаимодействия компонентов CPK по протоколу SSL
<code>/opt/rubackup/keys/rootCA/</code>	Папка содержит самоподписанный сертификат и закрытый ключ центра сертификации для внутреннего взаимодействия компонентов CPK по протоколу SSL
<code>/opt/rubackup/etc/</code>	Папка содержит конфигурационные файлы CPK RuBackup
<code>/opt/rubackup/etc/ld.so.conf.d/rubackup.conf</code>	Вспомогательный конфигурационный файл, указывающий ОС путь к дополнительным библиотекам, используемых CPK RuBackup
<code>/opt/rubackup/copyrights/</code>	Папка содержит файлы лицензионных соглашений
<code>/opt/rubackup/rc/icons/</code>	Папка содержит иконки интерфейса
Пакет rubackup-client	
<code>/opt/rubackup/etc/systemd/system/</code>	Папка содержит сервисы CPK RuBackup
<code>/opt/rubackup/etc/rubackup.lsf</code>	Файл локального расписания Клиента системы резервного копирования
<code>/opt/rubackup/etc/systemd/system/rubackup_client.service</code>	Сервис клиентской части CPK RuBackup

Структурный элемент	Назначение элемента
<code>/opt/rubackup/scripts/</code>	Папка содержит скрипты управления СРК RuBackup
<code>/opt/rubackup/scripts/test-script.sh</code>	Пример скрипта для выполнения при резервном копировании
<code>/opt/rubackup/log/</code>	Папка содержит журналы событий и задач
<code>/opt/rubackup/man/</code>	Папка содержит инструкции по использованию утилит
<code>/opt/rubackup/modules/</code>	Папка содержит исполнительные модули, поддерживающие резервное копирование и восстановление целевого ресурса (поддерживаемого клиентом СРК)
<code>/opt/rubackup/modules/rb_module_lvm</code>	Исполняемый модуль для резервного копирования и восстановления логических томов LVM
<code>/opt/rubackup/modules/rb_module_filesystem</code>	Исполняемый модуль резервного копирования файловой системы
<code>/opt/rubackup/bin/</code>	Папка содержит консольные утилиты, поддерживаемые на клиенте для управления резервным копированием и восстановлением данных
<code>/opt/rubackup/bin/rb_schedule</code>	Утилита клиента RuBackup для просмотра правил глобального расписания клиента в системе резервного копирования
<code>/opt/rubackup/bin/rb_replicas</code>	Утилита клиента RuBackup для управления правилами репликации на клиенте. Вы можете просмотреть список всех правил репликации, а также запустить выбранное правило
<code>/opt/rubackup/bin/rb_health_check</code>	Утилита клиента RuBackup для проверки конфигурации клиента и его окружения. Выполняется проверка переменных окружения, версии медиасервера. Проверяется подключение клиента к базе данных, серверу, медиасерверу и толстому клиенту
<code>/opt/rubackup/bin/rubackup_client</code>	Клиент резервного копирования RuBackup представляет собой фоновое приложение (сервис, демон), запущенное на хосте клиента и взаимодействующее с сервером RuBackup
<code>/opt/rubackup/bin/rb_init</code>	Утилита администратора RuBackup для первоначальной настройки клиента сразу после развёртывания пакета исполняемых файлов. Неинтерактивный режим необходим для сценариев массового развёртывания

Структурный элемент	Назначение элемента
<code>/opt/rubackup/bin/rb_archives</code>	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 <code>rubackup_client</code>
<code>/opt/rubackup/bin/rbfd</code>	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
<code>/opt/rubackup/bin/rb_tasks</code>	Утилита клиента RuBackup для просмотра списка задач клиента в системе резервного копирования RuBackup
<code>/opt/rubackup/bin/rb_client_defined_storages</code>	Утилита администратора RuBackup для управления клиентскими хранилищами. Вы можете просматривать, добавлять и удалять клиентские хранилища в конфигурации
<code>/opt/rubackup/rc/</code>	Папка содержит конфигурационные скрипты программы
<code>/opt/rubackup/mnt/</code>	Предоставляется как временная точка монтирования для файловых систем
Пакет <code>rubackup-server</code>	
<code>/opt/rubackup/etc/systemd/system/</code>	Папка одержит сервисы СРК RuBackup
<code>/opt/rubackup/etc/systemd/system/rubackup_server.service</code>	Сервис Серверной части СРК RuBackup
<code>/opt/rubackup/man/</code>	Папка содержит файлы описаний утилит
<code>/opt/rubackup/log/</code>	Папка содержит файлы журнала событий
<code>/opt/rubackup/log/RuBackup.log</code>	Системный журнал событий, также содержит информацию о лицензии
<code>/opt/rubackup/log/task.log</code>	Журналы событий, содержащие события задач СРК
<code>/opt/rubackup/log/module_.log</code>	Журналы событий исполняемых модулей
<code>/opt/rubackup/log/rbfd</code>	Информация о процессе выполнения создания РК для каждой задачи, которая использует <code>rbfd</code>
<code>/opt/rubackup/lib/</code>	Папка содержит библиотеки, необходимые для работы СРК RuBackup
<code>/opt/rubackup/bin/</code>	Папка содержит исполняемые файлы для запуска утилит
<code>/opt/rubackup/bin/rb_modules</code>	Утилита администратора RuBackup для управления Модулями

Структурный элемент	Назначение элемента
<code>/opt/rubackup/bin/rb_tape_libraries</code>	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, находящихся в слотах ленточной библиотеки.
<code>/opt/rubackup/bin/rb_media_servers</code>	Утилита администратора RuBackup для управления медиасерверами RuBackup. Вы можете просматривать список медиасерверов, добавлять их, удалять или изменять их описания. медиасервер предназначен для взаимодействия с клиентами при создании, восстановлении и передаче резервных копий
<code>/opt/rubackup/bin/rb_user_groups</code>	Утилита администратора RuBackup для управления группами пользователей. Вы можете просматривать группы пользователей, добавлять и удалять их, а также изменять их название и описание
<code>/opt/rubackup/bin/rubackup_server</code>	Сервер резервного копирования RuBackup представляет собой системное фоновое приложение (служба, демон), внутри которого одновременно выполняются множество потоков, отвечающих за разные функции системы резервного копирования
<code>/opt/rubackup/bin/rb_local_filesystems</code>	Утилита администратора RuBackup для управления хранилищами резервных копий типа Файловая система. Хранилища такого типа должны быть ассоциированы с пулом того же типа
<code>/opt/rubackup/bin/rb_security</code>	Утилита RuBackup для работы с журналом событий информационной безопасности
<code>/opt/rubackup/bin/rb_clients</code>	Утилита администратора RuBackup для управления клиентами RuBackup. Вы можете просматривать список клиентов, а также добавлять, удалять или изменять их.
<code>/opt/rubackup/bin/rb_update</code>	Утилита администратора RuBackup для управления обновлениями баз данных. Создает sql инструкции, позволяющие сделать обновление базы данных
<code>/opt/rubackup/bin/rb_block_devices</code>	Утилита администратора RuBackup для управления блочными устройствами

Структурный элемент	Назначение элемента
<code>/opt/rubackup/bin/rb_global_config</code>	Утилита администратора RuBackup для управления параметрами глобальной конфигурации серверной группировки RuBackup. Параметры глобальной конфигурации действительны для всех серверов, входящих в кластер серверов RuBackup
<code>/opt/rubackup/bin/rb_global_schedule</code>	Утилита администратора RuBackup для управления глобальным расписанием RuBackup. 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
<code>/opt/rubackup/bin/rb_repository</code>	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
<code>/opt/rubackup/bin/rb_users</code>	Утилита администратора RuBackup для управления пользователями. Вы можете просматривать список пользователей, добавлять, удалять и изменять их
<code>/opt/rubackup/bin/rb_tape_cartridges</code>	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
<code>/opt/rubackup/bin/rb_inventory</code>	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

Структурный элемент	Назначение элемента
<code>/opt/rubackup/bin/rb_interoperation</code>	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. 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
<code>/opt/rubackup/bin/rb_clouds</code>	Утилита администратора RuBackup для просмотра конфигурации, добавления или удаления облаков S3 в системе резервного копирования
<code>/opt/rubackup/bin/rb_copy2pool</code>	Утилита администратора RuBackup для управления репликацией. Предоставляет возможность создавать точные копии (реплики) резервных копий для правил резервного копирования и для стратегий резервного копирования
<code>/opt/rubackup/bin/rb_notifications</code>	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
<code>/opt/rubackup/bin/rb_remote_replication</code>	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
<code>/opt/rubackup/bin/rb_pools</code>	Утилита администратора RuBackup для управления пулами. Вы можете просматривать список пулов, добавлять, удалять и изменять их. Каждый пул принадлежит какому-либо медиа-серверу. Пулы используются для группирования устройств хранения резервных копий

Структурный элемент	Назначение элемента
<code>/opt/rubackup/bin/rb_tl_task_queue</code>	Утилита администратора RuBackup для управления Очередью задач ленточных библиотек
<code>/opt/rubackup/bin/rb_block_device_check</code>	Утилита администратора RuBackup для проверки целостности резервных копий на блочном устройстве
<code>/opt/rubackup/bin/rb_client_group</code>	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
<code>/opt/rubackup/bin/rb_bandwidth</code>	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
<code>/opt/rubackup/bin/rb_task_queue</code>	Утилита администратора RuBackup для управления главной очередью задач. В очереди задач содержатся все актуальные задачи на создание, восстановление, удаление, перемещение и проверку резервных копий
<code>/opt/rubackup/bin/rb_cloud_task_queue</code>	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
<code>/opt/rubackup/bin/rb_strategies</code>	Утилита администратора RuBackup для управления стратегиями
<code>/opt/rubackup/bin/rb_log_viewer</code>	Утилита администратора RuBackup для просмотра и управления журналами сообщений
<code>/opt/rubackup/rc/init/</code>	Содержит конфигурационные скрипты программы
<code>/opt/rubackup/mnt/</code>	Предоставляется как временная точка монтирования для файловых систем
Пакет rubackup-common-gui	
<code>/opt/rubackup/keys/rbm/</code>	Папка содержит сертификат и закрытый ключ приложения RBM для внутреннего взаимодействия компонентов СРК по протоколу SSL
<code>/opt/rubackup/gui/plugins/</code>	Папка содержит плагины

Структурный элемент	Назначение элемента
<code>/opt/rubackup/gui/lib/</code>	Папка содержит библиотеки, используемые графическим приложением RBM
<code>/opt/rubackup/gui/qml/</code>	Папка содержит QML-библиотеки, используемые графическим приложением RBM
<code>/opt/rubackup/gui/rc/</code>	Папка содержит настройки графического отображения, в т.ч. темы, переводы приложения RBM
<code>/opt/rubackup/gui/rc/themes/</code>	Файлы тем приложения RBM

12.2. Сетевые сервисы

В результате настройки компонентов CPK RuBackup будут добавлены необходимые сетевые сервисы в файл `/etc/services`:

- `rubackup-cmd` — сервис обеспечивает командное взаимодействие серверов и клиентов CPK RuBackup;
- `rubackup-lic` — сервис лицензирования;
- `rubackup-media` — сервис обеспечивает взаимодействие между медиасерверами и передачу файлов.

12.3. Конфигурационный файл

Данные, полученные после настройки (с помощью утилиты `rb_init` или `rb_init_gui`), сохраняются в файле:

- для *Linux-систем*: `/opt/rubackup/etc/config.file`;
- для *Windows-систем*: `C:\RuBackup-win-client\etc\config.file.txt`.

12.3.1. Параметры конфигурационного файла клиента

Таблица 22. Описание параметров конфигурационного файла клиента

Параметр	Назначение
<code>node</code>	Тип машины RuBackup
	Возможные значения <code>primaryserver</code> , <code>secondaryserver</code> , <code>mediaserver</code> , <code>client</code> .
<code>who-is-primary-server</code>	Имя машины основного сервера RuBackup
<code>client-inet-interface</code>	Сетевой интерфейс клиента. Используется для отображения дополнительной информации о клиенте в CPK RuBackup.
	Медиасервер осуществляет связь с основным или резервным сервером, а также с утилитой <code>rbfd</code> через сетевой интерфейс

Параметр	Назначение
<code>use-local-backup-directory</code>	Каталог для временного хранения резервных копий. Для создания резервных копий и хранения временных файлов, которые создаются при их восстановлении, требуется определённое дисковое пространство. Рекомендуем выделить для этой цели отдельный локальный диск и примонтировать его к <code>/backup-tmp</code> либо к другой точке монтирования. Точку монтирования временного каталога нужно указать как значение параметра <code>use-local-backup-directory</code> и перезагрузить клиент RuBackup. Возможные значения <code><path></code>.
<code>parallel-tasks</code>	Максимальное количество одновременно выполняемых задач Возможные значения от 1 до 64. По умолчанию 2
<code>deduplication-task-memory</code>	Исключение дублирующих копий повторяющихся данных (в Б) Возможные значения от 0. По умолчанию 268435456
<code>remote-replication</code>	Включение удаленной репликации — копирование данных из хоста источника на удаленный хост приемника Возможные значения yes, no. По умолчанию yes
<code>centralized-recovery</code>	Централизованное восстановление данных из резервной копии yes Централизованное восстановление данных из резервной копии возможно с помощью приложения «Менеджер администратора RuBackup». no Централизованное восстановление данных возможно только на клиенте резервного копирования с помощью утилиты командной строки <code>rbfd</code> или «Менеджера клиента RuBackup». По умолчанию yes

Параметр	Назначение
<code>logfile</code>	Расположение системного файла журнала событий Возможные значения <code><path></code> .
<code>rbd_algorithm</code>	Выбор хеш функции при дедупликации Возможные значения <code>sha1</code> , <code>sha2</code> , <code>skein</code> , <code>streebog</code> , <code>blake2b</code> . По умолчанию <code>sha2</code>
<code>rbd_block_size</code>	Размер блока данных при дедупликации (в Б) Возможные значения <code>8192</code> , <code>16384</code> , <code>32768</code> , <code>65536</code> , <code>131072</code> , <code>262144</code> , <code>524288</code> , <code>1048576</code> . По умолчанию <code>16384</code>
<code>rbd_hash_length</code>	Допустимая длина хеша (в б) Возможные значения <code>256</code> , <code>512</code> . По умолчанию <code>256</code>
<code>client-shutdown_scenario</code>	Сценарий выключения клиента Возможные значения <code>immediately</code> , <code>after-all-tasks</code> , <code>cancel-if-tasks</code> . По умолчанию <code>cancel-if-tasks</code>

Параметр	Назначение
<code>use-product-uuid</code>	Для версии CPK RuBackup 2.1 и более поздней: Генерировать идентификатор <code>hardware id</code> машины лицензируемого сервера на основании: • для ОС <i>Linux</i>: идентификатора UUID материнской платы, установленного производителем платы, и закодированной информации в DMI BIOS; • для ОС <i>Windows</i>: имени машины <code>hostname</code>; Для версии CPK RuBackup 2.0 и ранее: параметра нет, <code>hardware id</code> генерируется на основании идентификатора <code>/etc/machine-id</code> и имени машины <code>/etc/hostname</code> Возможные значения <code>false</code>, <code>true</code>. По умолчанию <code>false</code>
<code>use-ip-instead-hostname</code>	Использовать IP-адреса вместо DNS-имён при взаимодействии компонентов CPK Возможные значения <code>false</code>, <code>true</code>. По умолчанию <code>false</code>
<code>reconnect-period-count</code>	Количество периодов переподключения в рамках выполнения задач между Клиентом и Медиасервером^[1] Возможные значения от 0. По умолчанию 3
<code>reconnect-period-timeout</code>	Таймаут (в сек.) между периодами переподключения в рамках выполнения задач между Клиентом и Медиасервером^[1] Возможные значения от 0. По умолчанию 20

Параметр	Назначение
<code>reconnect-count</code>	Количество попыток переподключения в рамках выполнения задач между Клиентом и Медиасервером^[1] Возможные значения от 0. По умолчанию 3
<code>reconnect-timeout</code>	Таймаут (в сек.) между попытками переподключения в рамках выполнения задач между Клиентом и Медиасервером^[1] Возможные значения от 0. По умолчанию 5
<code>digital-signature</code>	Использовать электронно-цифровую подпись для аутентификации и защиты данных в СРК Возможные значения yes, no. По умолчанию yes
<code>parallelizm</code>	Число потоков для обработки сообщений от клиентов на главном сервере Возможные значения от 1 до 4096. По умолчанию 8
<code>parallelizm_media</code>	Число потоков для обработки сообщений от клиентов на медиа сервере Возможные значения от 1 до 4096. По умолчанию 8

Параметр	Назначение
monitoring-client	Включить отправку метрик мониторинга хоста на главный сервер Возможные значения yes, no. По умолчанию yes
used-ip-version	Версия IP-протокола для сетевого взаимодействия компонентов СРК Возможные значения ipv4, ipv6, both.
digital-sign-hash	Хеш-функция для электронно-цифровой подписи. В соответствии с командами группы OpenSSL Message Digest Возможные значения blake2b512, blake2s256, gost, md4, md5, rmd160, sha1, sha224, sha256, sha384, sha512. По умолчанию sha1
memory-threshold	Верхняя граница использования оперативной памяти (в ГБ) при создании полной резервной копии. Для хранения уникальных хешей и обеспечения дедупликации нужно выделить на диске дополнительное место $\approx 0,3\%$ от размера ресурса.  <i>Ограничения</i> При использовании параметра в кластерной группе убедитесь, что все клиенты группы имеют одну версию СРК. Параметр используется только для создания полной резервной копии Минимальное значение параметра равно 4, при меньшем значении параметра в процессе резервного копирования будет выведено предупреждение и параметр не будет учтён По умолчанию 0

12.3.2. Параметры конфигурационного файла сервера

Таблица 23. Описание параметров конфигурационного файла сервера

Параметр	Назначение
node	Тип машины RuBackup
	Возможные значения primaryserver, secondaryserver, mediaserver, client.
server-inet-interfaces	Список сетевых интерфейсов сервера в одну строку через пробел, через которые серверу резервного копирования разрешено взаимодействовать с клиентами
dbname	Имя служебной базы данных
	По умолчанию rubackup
user	Пользователь служебной базы данных
	По умолчанию rubackup
password crypted	Закодированное значение пароля пользователя служебной базы данных
host	FQDN или IP адрес сервера, на котором расположена служебная база данных
port	Порт подключения служебной базы данных
	По умолчанию 5432
server-shutdown_scenario	Сценарий выключения сервера
	Возможные значения immediately, after-all-tasks, cancel-if-tasks.
	По умолчанию cancel-if-tasks
s3-fuse-type	Тип FUSE-драйвера для монтирования S3 бакетов
	По умолчанию geesefs
geesefs-memory-limit	Лимит оперативной памяти (в МБ) для кэширования данных и метаданных
	По умолчанию 4000

Параметр	Назначение
<code>geeseefs-max-flushers</code>	Максимальное количество фоновых потоков для асинхронной записи данных в S3 Возможные значения от 0. По умолчанию 16
<code>geeseefs-read-retry-attempts</code>	Максимальное количество повторных попыток при ошибках чтения из S3 Возможные значения от 0. По умолчанию 3
<code>geeseefs-no-specials</code>	Отключение поддержки специальных файлов Возможные значения yes, no. По умолчанию no
<code>geeseefs-list-type</code>	Тип листинга объектов Возможные значения 0, 1. По умолчанию 1
<code>geeseefs-http-timeout</code>	Таймаут (в сек.) HTTP-запросов к S3 Возможные значения от 0. По умолчанию 30
<code>geeseefs-fsync-on-close</code>	Ожидание подтверждения записи данных на сервер S3 при закрытии файла Возможные значения yes, no. По умолчанию no

Параметр	Назначение
<code>geeseefs-no-preload-dir</code>	Отключение предварительной загрузки листинга директории при открытии отдельных файлов. Возможные значения <code>yes</code>, <code>no</code>. По умолчанию <code>no</code>
<code>geeseefs-no-implicit-dir</code>	Предполагать существование всех объектов-директорий без проверки Возможные значения <code>yes</code>, <code>no</code>. По умолчанию <code>no</code>
<code>geeseefs-no-checksum</code>	Отключение проверки <code>MD5</code> и <code>SHA256</code> контрольных сумм для повышения производительности Возможные значения <code>yes</code>, <code>no</code>. По умолчанию <code>no</code>
<code>geeseefs-max-parallel-parts</code>	Количество параллельных запросов из общего числа, используемых для многокомпонентной загрузки крупных частей. Возможные значения от <code>0</code>. По умолчанию <code>0</code>
<code>geeseefs-part-sizes</code>	Размеры частей (в МБ) для многокомпонентной загрузки. Возможные значения от <code>0</code>. По умолчанию <code>0</code>
<code>geeseefs-no-verify-ssl</code>	Отключение проверки SSL-сертификатов Возможные значения <code>yes</code>, <code>no</code> По умолчанию <code>no</code>

Параметр	Назначение
<code>geeseefs-ignore-fsync</code>	Не ожидать сохранения изменений на сервере Возможные значения <code>yes</code>, <code>no</code> По умолчанию <code>no</code>
<code>geeseefs-read-ahead-large</code>	Размер буфера предварительного чтения (в КБ) для файлов По умолчанию <code>102400</code>
<code>geeseefs-read-ahead-parallel</code>	Размер (в КБ) для параллельных блоков предварительного чтения при работе с большими файлами По умолчанию <code>20480</code>
<code>s3-cloud-http-version</code>	Тип HTTP-протокола для S3 бакетов. Параметр является опциональным Возможные значения <code>best-possible</code>, <code>http_1_0</code>, <code>http_1_1</code>, <code>http2</code>, <code>http2_tls</code>, <code>http2_prior</code>, <code>http3</code>.
<code>s3-cloud-http-request-timeout</code>	Таймаут (в сек.) ожидания ответа от S3 сервера для каждого HTTP-запроса. При отсутствии ответа в течение указанного времени задача завершится с ошибкой. Таймаут отключен при значении по умолчанию Возможные значения от <code>0</code>. По умолчанию <code>0</code>
<code>s3-cloud-http-debug</code>	Параметр отвечает за печать лога HTTP трафика с S3 сервера в Debug лог медиасервера Возможные значения <code>true</code>, <code>false</code>.
<code>s3-cloud-http-request-retries</code>	Максимальное количество повторных запросов к S3 (только для пула типа <code>Cloud</code>, <code>gen.2</code>) <code>0</code> Параметр отключен. По умолчанию <code>3</code>

Параметр	Назначение
<code>s3_buckets_max_threads</code>	Максимальное количество параллельных потоков для обработки S3-бакетов Возможные значения от 0. По умолчанию 8
<code>dedup-gen2-server-dedup</code>	Параметр отвечает за включение дополнительной серверной дедупликации алгоритмом FastCDC (только для пула типа <code>Block device, gen.2</code>). Параметр не отключает глобальную дедупликацию на уровне блочного устройства Возможные значения <code>true, false</code>. По умолчанию <code>false</code>
<code>dedup-gen2-reset-db-period</code>	Плановый период (в сек.) автоматического сброса кеша внутренней БД блочного устройства (только для пула типа <code>Block device, gen.2</code>). Сброс применяется для периодической очистки занятого ОЗУ Возможные значения от 0. По умолчанию 3600
<code>dedup-gen2-reset-db-min-interval</code>	Минимальный интервал (в сек.) между двумя последовательными сбросами кеша внутренней БД блочного устройства (только для пула типа <code>Block device, gen.2</code>). Ограничивает число сбросов кеша в единицу времени Возможные значения от 0. По умолчанию 600
<code>keepalives</code>	Использование сообщений keepalive протокола TCP на стороне клиента Возможные значения 0, 1. По умолчанию 0

Параметр	Назначение
keepalives_idle	Управление длительностью периода отсутствия активности (в сек.) Возможные значения от 0. По умолчанию 7200
keepalives_interval	Интервал (в сек.) между отправкой повторных сообщений при отсутствии ответа Возможные значения от 0. По умолчанию 75
keepalives_count	Количество последовательно потерянных проверочных сообщений перед разрывом соединения Возможные значения от 0. По умолчанию 9
copy-move-parallelism	Число параллельных потоков для копирования или перемещения РК в другой пул  Параметр применяется только для копирования и перемещения РК в пулы типа Ленточная библиотека, LTFS и Ленточная библиотека, Native. Подробнее о сценариях копирования и перемещения РК см. Пулы. Возможные значения от 0. По умолчанию 8

Параметр	Назначение
<code>copy-move-sequential-read</code>	Включить последовательное чтение исходной РК при копировании или перемещении в другой пул  Параметр применяется только для копирования и перемещения РК в пулы типа <code>Ленточная библиотека</code>, <code>LTFS</code> и <code>Ленточная библиотека, Native</code>. Подробнее о сценариях копирования и перемещения РК см. Пулы. Возможные значения <code>yes</code>, <code>no</code>. По умолчанию <code>yes</code>
<code>who-is-primary-server</code>	Имя машины основного сервера RuBackup
<code>who-is-secondary-server</code>	Имя машины резервного сервера RuBackup
<code>client-inet-interface</code>	Сетевой интерфейс клиента. Используется для отображения дополнительной информации о клиенте в СРК RuBackup. Медиасервер осуществляет связь с основным или резервным сервером, а также с утилитой <code>rbfd</code> через сетевой интерфейс
<code>use-local-backup-directory</code>	Каталог для временного хранения резервных копий. Для создания резервных копий и хранения временных файлов, которые создаются при их восстановлении, требуется определённое дисковое пространство. Рекомендуем выделить для этой цели отдельный локальный диск и примонтировать его к <code>/backup-tmp</code> либо к другой точке монтирования. Точку монтирования временного каталога нужно указать как значение параметра <code>use-local-backup-directory</code> и перезагрузить клиент RuBackup. Возможные значения <code><path></code>.
<code>parallel-tasks</code>	Максимальное количество одновременно выполняемых задач Возможные значения от <code>1</code> до <code>64</code>. По умолчанию <code>2</code>
<code>deduplication-task-memory</code>	Исключение дублирующих копий повторяющихся данных (в Б) Возможные значения от <code>0</code>. По умолчанию <code>268435456</code>

Параметр	Назначение
<code>remote-replication</code>	Включение удаленной репликации — копирование данных из хоста источника на удаленный хост приемника Возможные значения <code>yes</code>, <code>no</code>. По умолчанию <code>yes</code>
<code>centralized-recovery</code>	Централизованное восстановление данных из резервной копии yes Централизованное восстановление данных из резервной копии возможно с помощью приложения «Менеджер администратора RuBackup». no Централизованное восстановление данных возможно только на клиенте резервного копирования с помощью утилиты командной строки <code>rbfd</code> или «Менеджера клиента RuBackup». По умолчанию <code>yes</code>
<code>logfile</code>	Расположение системного файла журнала событий Возможные значения <code><path></code>.
<code>rbd_algorithm</code>	Выбор хеш функции при дедупликации Возможные значения <code>sha1</code>, <code>sha2</code>, <code>skein</code>, <code>streebog</code>, <code>blake2b</code>. По умолчанию <code>sha2</code>
<code>rbd_block_size</code>	Размер блока данных при дедупликации (в Б) Возможные значения <code>8192</code>, <code>16384</code>, <code>32768</code>, <code>65536</code>, <code>131072</code>, <code>262144</code>, <code>524288</code>, <code>1048576</code>. По умолчанию <code>16384</code>

Параметр	Назначение
<code>rbd_hash_length</code>	Допустимая длина хеша (в б) Возможные значения <code>256</code>, <code>512</code>. По умолчанию <code>256</code>
<code>client-shutdown_scenario</code>	Сценарий выключения клиента Возможные значения <code>immediately</code>, <code>after-all-tasks</code>, <code>cancel-if-tasks</code>. По умолчанию <code>cancel-if-tasks</code>
<code>use-product-uuid</code>	Для версии CPK RuBackup 2.1 и более поздней: Генерировать идентификатор <i>hardware id</i> машины лицензируемого сервера на основании: • для ОС <i>Linux</i>: идентификатора UUID материнской платы, установленного производителем платы, и закодированной информации в DMI BIOS; • для ОС <i>Windows</i>: имени машины <code>hostname</code>; Для версии CPK RuBackup 2.0 и ранее: параметра нет, <code>hardware id</code> генерируется на основании идентификатора <code>/etc/machine-id</code> и имени машины <code>/etc/hostname</code> Возможные значения <code>false</code>, <code>true</code>. По умолчанию <code>false</code>
<code>use-ip-instead-hostname</code>	Использовать IP-адреса вместо DNS-имён при взаимодействии компонентов CPK Возможные значения <code>false</code>, <code>true</code>. По умолчанию <code>false</code>
<code>reconnect-period-count</code>	Количество периодов переподключения в рамках выполнения задач между Клиентом и Медиасервером^[1] Возможные значения от <code>0</code>. По умолчанию <code>3</code>

Параметр	Назначение
reconnect-period-timeout	Таймаут (в сек.) между периодами переподключения в рамках выполнения задач между Клиентом и Медиасервером^[1] Возможные значения от 0. По умолчанию 20
reconnect-count	Количество попыток переподключения в рамках выполнения задач между Клиентом и Медиасервером^[1] Возможные значения от 0. По умолчанию 3
reconnect-timeout	Таймаут (в сек.) между попытками переподключения в рамках выполнения задач между Клиентом и Медиасервером^[1] Возможные значения от 0. По умолчанию 5
digital-signature	Использовать электронно-цифровую подпись для аутентификации и защиты данных в СРК Возможные значения yes, no. По умолчанию yes
parallelizm	Число потоков для обработки сообщений от клиентов на главном сервере Возможные значения от 1 до 4096. По умолчанию 8

Параметр	Назначение
<code>parallelizm_media</code>	Число потоков для обработки сообщений от клиентов на медиа сервере Возможные значения от 1 до 4096. По умолчанию 8
<code>monitoring-client</code>	Включить отправку метрик мониторинга хоста на главный сервер Возможные значения yes, no. По умолчанию yes
<code>used-ip-version</code>	Версия IP-протокола для сетевого взаимодействия компонентов СРК Возможные значения ipv4, ipv6, both.
<code>client-hello-timeout</code>	Таймаут (в сек.) на получение ClientHello от клиента после установки TCP-соединения Возможные значения от 0. По умолчанию 60
<code>client-ping-timeout</code>	Время ожидания (в сек.) ответа на ping-запрос Возможные значения от 0. По умолчанию 1
<code>client-alive-timeout</code>	Общее время (в сек.) для поддержания неактивного соединения Возможные значения от 0. По умолчанию 60

Параметр	Назначение
client-request-timeout	Время (в сек.) на получение полного запроса после подключения
	Возможные значения
	от 0.
	По умолчанию
	60

[1] При отсутствии отдельно настроенных медиасерверов их функции выполняют основной и резервный сервера (при наличии).

Глава 13. Настройка ограничения на количество открытых файловых дескрипторов на машине сервера RuBackup

При увеличении количества входящих соединений (если число клиентов/медиа серверов в группировке растёт и/или на клиентах включена функция многопоточной передачи данных) сервер RuBackup может достичь предела выделенных лимитов на открытые файловые дескрипторы. Сетевые соединения также используют файловые дескрипторы.

Ограничения на количество открытых файловых дескрипторов устанавливает администратор машины, на котором запущен сервер RuBackup. Достижение этого ограничения приводит к ошибкам при выполнении резервного копирования/восстановления. Иногда сервер RuBackup может аварийно завершить работу.

13.1. Зависимость количества файловых дескрипторов

В зависимости от способа запуска сервера RuBackup, максимальное число (лимит) открытых дескрипторов будет разным.

Чтобы рассчитать необходимое количество файловых дескрипторов, учтите следующее:

- В режиме простоя сервер использует около 100 файловых дескрипторов.
- Каждый подключённый клиент РК или медиасервер добавляет по два открытых файловых дескриптора на сервере.
- Выполнение любой задачи на стороне клиента РК при выключенном параметре `network_parallelism`^[1] требует двух дополнительных файловых дескриптора на сервере.
- При активированном параметре `network_parallelism` клиент РК открывает N соединений к серверу, где N — значение, заданное для этого параметра. В рамках каждого сетевого соединения, как правило, на стороне сервера требуется запросить информацию из базы данных, поэтому требуемое число открытых файловых дескрипторов будет $N \times 2$.

13.2. Расчёт необходимого количества файловых дескрипторов

Проверьте, рассчитав по формулам, число нужных вам файловых дескрипторов и

убедитесь, что на машине сервера RuBackup их достаточно.

Пример 14. Общая формула для расчёта необходимого количества файловых дескрипторов

$$100 + MS \times 2 + CL \times 2 + CL \times N$$

где:

- MS — число медиасерверов;
- CL — число клиентов;
- N — значение, заданное для сетевого параллелизма, параметра `network_parallelism`. Если сетевой параллелизм выключен, то $N = 2$.

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:

100 (для основного сервера) + 2×2 (для медисерверов) + 50×2 (для клиентов в простое) + 50×2 (для клиентов с задачами одновременно) = 304

Таким образом, общее количество необходимых файловых дескрипторов составляет 304.

Стандартное значение лимита — 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:

100 (для основного сервера) + 2×2 (для медиасерверов) + 50×2 (для клиентов в простое) + 50×40 (для всех клиентов с задачами одновременно) = 2204

Таким образом, общее количество необходимых файловых дескрипторов состав-

ляет 2204.

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

13.3. Способы настройки ограничения количества открытых файловых дескрипторов

Настройка ограничения количества открытых файловых дескрипторов производится в зависимости от способа запуска сервера.

13.3.1. Настройка ограничения количества открытых файловых дескрипторов при ручном запуске сервера

Для настройки ограничения количества открытых файловых дескрипторов при ручном запуске сервера:

1. Остановите сервер (в случае ручного запуска сервера):

```
rubackup_server stop
```

2. Для проверки текущего лимита выполните:

```
sudo ulimit -n
```

По умолчанию ограничение количества открытых файловых дескрипторов установлено 1024 файла.

3. Изменение лимита открытых файловых дескрипторов возможно выполнить для текущей сессии пользователя `root` или установить постоянное значение.

- а. Для временного изменения лимита открытых файловых дескрипторов только в текущей сессии пользователя `root` необходимо выполнить команду:

```
ulimit -n N
```

где `N` — это желаемое значение лимита открытых файловых дескрипторов.

Внесённые изменения будут отменены после завершения текущей сессии.

- б. Для установки постоянного лимита открытых файловых дескрипторов:

- отредактируйте файл `/etc/security/limits.conf`:

```
sudo nano /etc/security/limits.conf
```

и добавив строки:

```
root hard nfile N
root soft nfile N
```

где `N` — это желаемое значение лимита открытых файловых дескрипторов;

- сохраните изменения;
- завершите сессию и откройте новую сессию;
- проверьте значение лимита открытых файловых дескрипторов:

```
ulimit -n
```

4. Перезапустите сервер:

```
rubackup_server start
```

13.3.2. Настройка ограничения количества открытых файловых дескрипторов при запуске сервисов сервера

Для настройки ограничения количества открытых файловых дескрипторов при запуске сервисов сервера:

1. Для проверки текущего лимита выполните:

```
sudo ulimit -n
```

По умолчанию ограничение количества открытых файловых дескрипторов задаётся в службе `systemd` и стандартное значение — 1024 файла.

2. Для изменения лимита открытых файловых дескрипторов:
 - откройте файл `/etc/systemd/system/rubackup_server.service`:

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

- отредактируйте секцию [Service], добавив строку:

```
LimitNOFILE=N
```

где **N** — это желаемое значение лимита открытых файловых дескрипторов;

- сохраните изменения.

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

```
systemctl daemon-reload
```

4. Перезапустите сервис сервера RuBackup:

```
systemctl stop rubackup_server  
  
systemctl start rubackup_server
```

[1] Параметр задает количество потоков, которые будут передавать блоки данных на медиасервер