



RuBackup

**Система резервного копирования
и восстановления данных**

СЦЕНАРИИ РАБОТЫ

ВЕРСИЯ 2.7.0.0.0, 14.10.2025

Содержание

Резервное копирование и восстановление метаданных дедуплицированного пула	5
Резервное копирование метаданных дедуплицированного пула	5
Резервное копирование метаданных дедуплицированного пула с помощью скрипта	5
Резервное копирование метаданных дедуплицированного пула вручную	6
Восстановление метаданных дедуплицированного пула	8
Листинг скрипта script_block_device_metadata.sh	9
Настройка хранилища резервных копий	12
Непрерывная удаленная репликация	13
Настройка сервера	13
Настройка клиента	18
Модуль ядра Linux <code>dattobd</code>	21
Установка	21
Дедупликация	23
Принципы дедупликации	23
Общий алгоритм дедупликации	24
Создание резервной копии	25
Восстановление резервной копии	26
Настройка	26
Блочное устройство	27
Пул хранения данных	27
Добавление блочного устройства в пул	29
Параметры системы	31
Особенности	33
Интеграция с ALD Pro	36
Подготовка данных для настройки соединения	36
Настройка соединения с контроллером домена	37
Определение прав группам доменных пользователей	39
Добавление ассоциации группы	40
Удаление ассоциации группы	41
Решение проблем	41
Проверка параметров соединения с контроллером домена	41
Интеграция с Microsoft Active Directory	44
Предварительные настройки	44

Первичная настройка СРК для работы с MS AD	45
Выбор типа аутентификации по умолчанию	54
Аутентификация пользователя СРК посредством MS AD	56
Аудит аутентификации пользователей	59
Решение проблем	60
Ограничения	62
Работа с сертификатами и ключами SSL	63
Размещение сертификатов и ключей	63
Использование цепочки сертификатов	63
Серверная часть	64
Создание сертификата	64
Подготовка сертификатов для сервера	65
Проверка созданных ключей и сертификатов	66
Клиентская часть	66
Создание сертификата	66
Подготовка сертификатов для клиента	67
Проверка созданных ключей и сертификатов	68
Менеджер администратора RuBackup	68
Создание сертификата	68
Проверка созданных ключей и сертификатов	68
Ленточные библиотеки	70
Подготовка к работе с ленточной библиотекой	70
Установка дополнительного ПО	70
Проверка наличия sg-драйвера	71
Установка sg-драйвера	71
Astra Linux 1.6 и 1.7	71
Ubuntu 18.04 и 20.04	72
CentOS 7 и 8	72
Alt Linux 10	72
РЕД ОС 7.3	72
Настройки автоматического запуска sg-драйвера	73
Сборка LTFS	74
Конфигурация ленточной библиотеки	79
Работа с ленточной библиотекой	87
Синхронизация ленточной библиотеки и RuBackup	87
Перемещение ленточного картриджа в другой слот	88
Перемещение ленточного картриджа в другой пул	89

Импорт и экспорт ленточных картриджей	90
Инвентаризация резервных копий	91
Удаление ленточной библиотеки	94
Коллекция картриджей ленточных библиотек	99
Дополнительные настройки	102
Статусы ленточных картриджей	105
Утилиты командной строки RuBackup для работы с ленточной библиотекой . .	107

В разделе описаны отдельные вопросы, с которыми встречаются пользователи СРК.

Резервное копирование и восстановление метаданных дедуплицированного пула

Резервное копирование метаданных дедуплицированного пула

Метаданные дедуплицированного пула хранятся в следующих таблицах СРК RuBackup:

- `pool_list`.
- `pool_block_device_extention`.
- `storage_block_devices`.
- `deduplicated_block_device_<signature>`.

Существует два способа резервного копирования метаданных дедуплицированного пула:

- с помощью скрипта `script_block_device_metadata.sh`.
- вручную, используя утилиту `pg_dump`.

Резервное копирование метаданных дедуплицированного пула с помощью скрипта

Для резервного копирования метаданных дедуплицированного пула с помощью скрипта необходимо:

1. Перевести СРК RuBackup в сервисный режим. Для этого необходимо перейти в меню *Настройки* → *Глобальная конфигурация* и включить переключатель *Сервисный режим* ([Рисунок 1](#)).

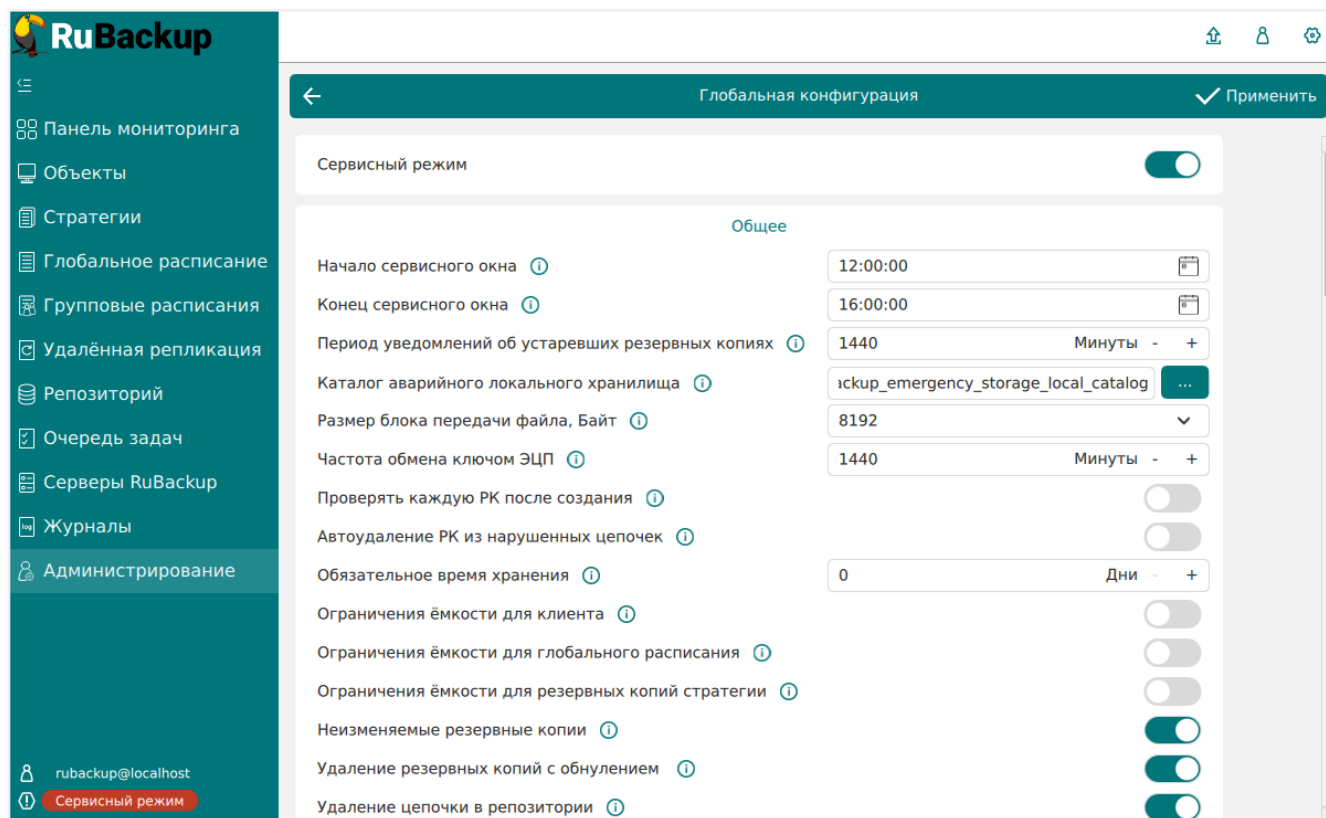


Рисунок 1. Окно "Глобальная конфигурация"

- Открыть на редактирование скрипт `script_block_device_metadata.sh` (Листинг скрипта `script_block_device_metadata.sh`) и задать обязательные параметры:
 - `HOST="localhost"` — адрес хоста с базой данных.
 - `DBNAME="rubackup"` — имя базы данных.
 - `USER="rubackup"` — имя пользователя базы данных.
 - `PASS="12345"` — пароль пользователя базы данных.
 - `BACKUP_FILENAME="rb_block_device_metadata_backup.sql"` — имя файла резервной копии выбранных таблиц.
- Запустить скрипт `script_block_device_metadata.sh` с параметром `dump`:

```
bash ./script_block_device_metadata.sh dump
```

В результате в текущем каталоге будет создана резервная копия выбранных таблиц в формате `.sql`.

Резервное копирование метаданных дедуплицированного пула вручную

Для резервного копирования метаданных дедуплицированного пула вручную необходимо:

- Перевести СРК RuBackup в сервисный режим. Для этого необходимо перейти в

меню *Настройки* → *Глобальная конфигурация* и включить переключатель *Сервисный режим* (Рисунок 1).

2. С помощью команды `pg_dump` выполнить резервное копирование следующих таблиц из базы данных RuBackup:

- `pool_list`.
- `pool_block_device_extention`.
- `storage_block_devices`.
- `deduplicated_block_device_<signature>`.

Пример команды для резервного копирования таблицы `pool_list` в файл `backup.sql`:

```
pg_dump -h localhost -d rubackup -U rubackup -t pool_list >backup.sql
```

Для таблицы `deduplicated_block_device_<signature>` необходимо получить параметр `signature`. `Signature` — это уникальная подпись для каждого блочного устройства. Значение `signature` можно получить следующими способами:

- с помощью утилиты `rb_block_devices`:

```
$ rb_block_devices -v
```

- В RBM в разделе «Блочные устройства» в колонке «Подпись» (Рисунок 2).

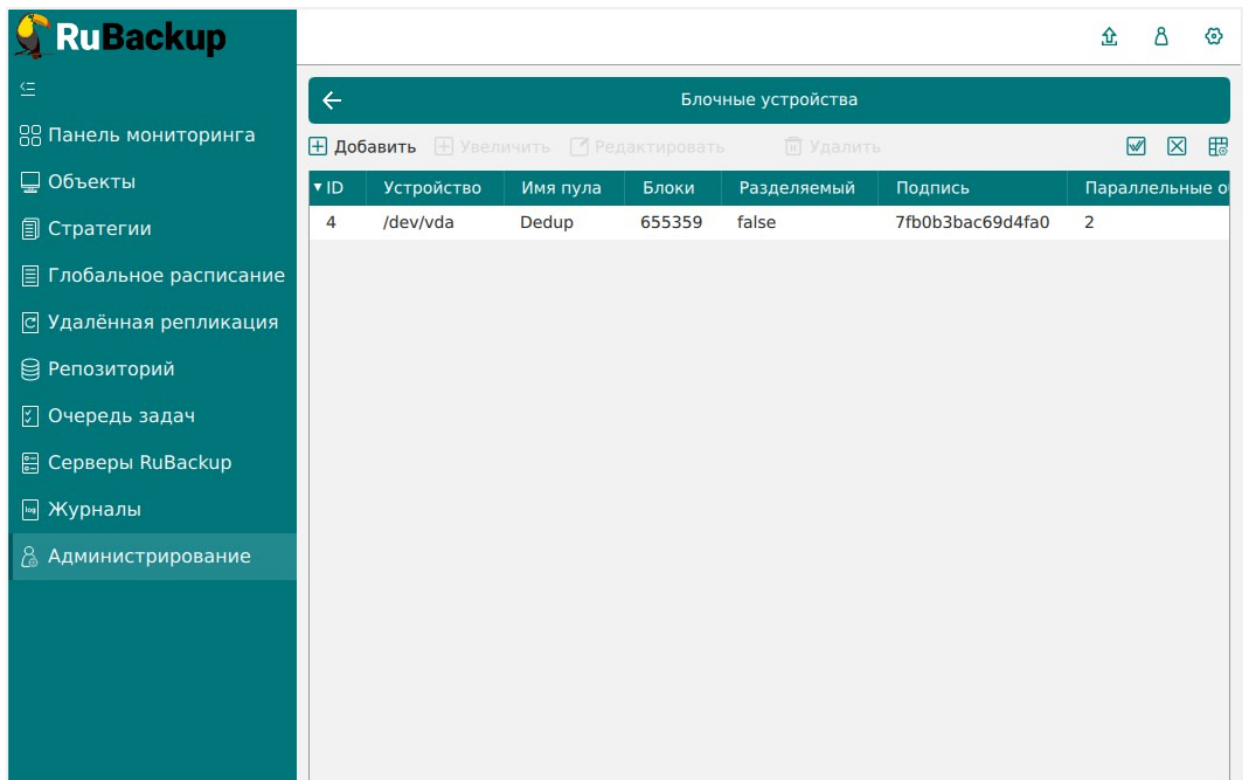


Рисунок 2. Окно "Блочные устройства"

Пример команды для резервного копирования таблицы `deduplicated_block_device_<signature>` в файл `backup.sql`:

```
$ pg_dump -h localhost -d rubakup -U rubakup -t
deduplicated_block_device_7fb0b3bac69d4fa0 >backup.sql
```

В результате в текущем каталоге будет создана резервная копия выбранных таблиц в формате `.sql`.

Восстановление метаданных дедуплицированного пула

Для восстановления метаданных дедуплицированного пула необходимо: . После сбоя СРК, заново настроить RuBackup согласно руководству по установке (см. [Развёртывание](#)).

1. Перевести СРК RuBackup в сервисный режим ([Рисунок 1](#)).
2. Восстановить резервные копии таблиц одним из двух способов:
 - С помощью скрипта `script_block_device_metadata.sh` ([Листинг скрипта script_block_device_metadata.sh](#)) с параметром `restore`:

```
bash ./script_block_device_metadata.sh restore
```

- С помощью команды `psql` восстановить заранее сохраненные таблицы (`pool_list`, `pool_block_device_extention`, `storage_block_devices`, `deduplicated_block_device_<signature>`):

```
psql -h localhost -d rubackup -U rubackup -f
rb_block_device_metadata_backup.sql
```

3. С помощью утилиты `rb_inventory` внести в базу данных RuBackup информацию о всех резервных копиях, которые были сделаны до сбоя:

```
rb_inventory -i /dir
```

4. С помощью утилиты `rb_block_devices` обновить имя устройства, если оно было изменено:

```
rb_block_devices -c ID -n block_device
```

Где:

- ID – уникальный номер блочного устройства, который можно узнать, запустив утилиту `rb_block_devices -v`:

```
rb_block_devices -v
```

- `block_device` – новое имя блочного устройства.

5. Перезапустить сервер RuBackup:

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

В результате будут восстановлены метаданные дедуплицированного пула.

Листинг скрипта script_block_device_metadata.sh

```
#!/bin/bash

# Параметры подключения к базе данных
HOST="localhost"
DBNAME="rubackup"
USER="rubackup"
```

```

PASS="12345"
BACKUP_FILENAME="rb_block_device_metadata_backup.sql"

TABLE_LIST="-t pool_list -t pool_block_device_extention -t
storage_block_devices" # Список таблиц для резервного копирования

if [ "$#" -eq 1 ]; then
    if [ "$1" = "dump" ]; then
        echo "RuBackup script handler saving Dedup pool metadata started"

        # Извлечение подписей из столбца "signature" в таблице
        "storage_block_devices"
        SIGNATURES=$(PGPASSWORD=$PASS psql -h $HOST -d $DBNAME -U $USER -qt
-c "SELECT DISTINCT signature FROM storage_block_devices")

        # Формирование строки с перечислением подписей
        for signature in $SIGNATURES; do
            table_name="deduplicated_block_device_${signature}" #
Формирование имени таблицы
            TABLE_LIST="$TABLE_LIST -t $table_name" # Добавляем
таблицу к списку
        done

        # Создание резервной копии всех таблиц в одном файле
        PGPASSWORD=$PASS pg_dump -h $HOST -d $DBNAME -U $USER $TABLE_LIST
>$BACKUP_FILENAME

        echo "A backup copy of the table is saved in a file $BACKUP_FILENAME"
        exit 0
    fi

    if [ "$1" = "restore" ]; then
        echo "RuBackup script handler restores Dedup pool metadata started"
        # Восстановление
        PGPASSWORD=$PASS psql -h localhost -d rubackup -U rubackup -f
$BACKUP_FILENAME
        echo "RuBackup script handler restores Dedup pool metadata finished"
        exit 0
    fi

    echo "Incorrect argument. Type 'dump' or 'restore'"
    exit 1
fi

echo "Argument required. Type 'dump' or 'restore'"

```

```
exit 1
```


Настройка хранилища резервных копий

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

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

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

1. Настройка хранилища с помощью `rb_local_filesystem`:

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

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

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

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



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

Порядок настройки хранилища изложен в документе [Менеджер администратора RuBackup \(RBM\)](#).

Непрерывная удаленная репликация

Система резервного копирования *RuBackup*, начиная с версии 1.7 поддерживает выполнение непрерывной репликации различных источников данных на удалённый хост. Эта возможность позволяет минимизировать время восстановления информационных систем, поскольку для восстановления функциональности потребуется только сделать доступной для работы реплику источника данных, например, включить виртуальную машину или изменить IP-адрес узла, на который происходила репликация данных.

Для выполнения непрерывной удалённой репликации необходимо использовать дедуплицированное хранилище резервных копий. При репликации от источника в место назначения передаются только изменённые блоки данных. Это позволяет выполнять репликацию настолько часто, насколько позволяет производительность систем. При этом минимальное время отставания реплики от источника данных составляет всего 1 минуту.

Непрерывная удалённая репликация с помощью *RuBackup* может выполняться для разных источников данных, включая файловые системы, виртуальные машины и т.д. Там, где это возможно, в ходе репликации задействуется возможность создания мгновенных снимков для источника данных (например, в случае файловой системы *BTRFS*).

Поддержка непрерывной удалённой репликации реализуется непосредственно в модуле резервного копирования, который отвечает за работу с источником данных.

Настройка сервера

Для осуществления непрерывной удалённой репликации на сервере резервного копирования должен быть настроен хотя бы один пул типа **Блочное устройство**, содержащий минимум одно блочное устройство для использования в качестве дедуплицированного хранилища резервных копий.

Управление правилами непрерывной удалённой репликации осуществляется при помощи *Менеджера администратора RuBackup (RBM)*. Вкладка **Удалённая репликация** в главном окне RBM содержит информацию обо всех правилах непрерывной удалённой репликации. Включенные правила имеют статус *run*, выключенные — *wait* ([Рисунок 3](#)).

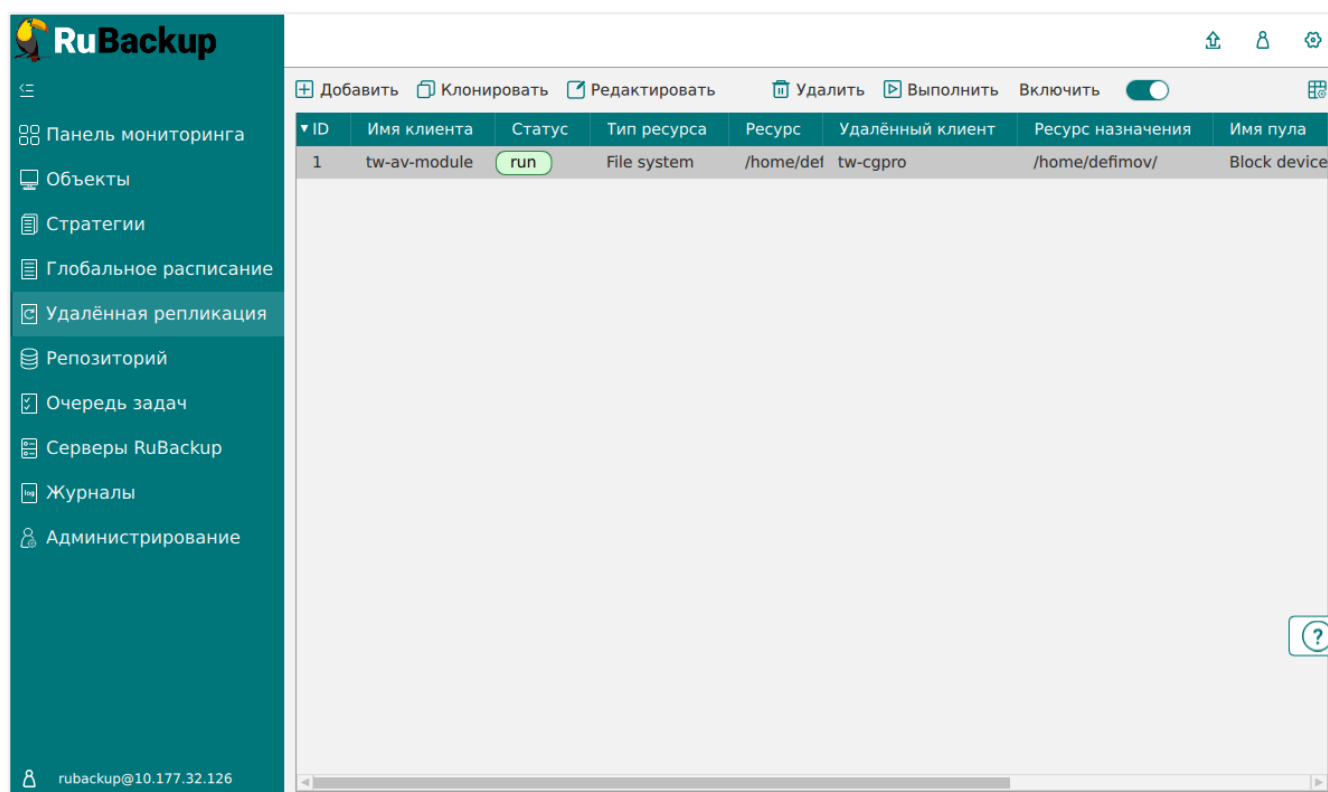


Рисунок 3. Раздел «Удаленная репликация»

Управление правилами удалённой репликации осуществляется в контекстном меню, вызываемом нажатием правой кнопки мыши (Рисунок 4).

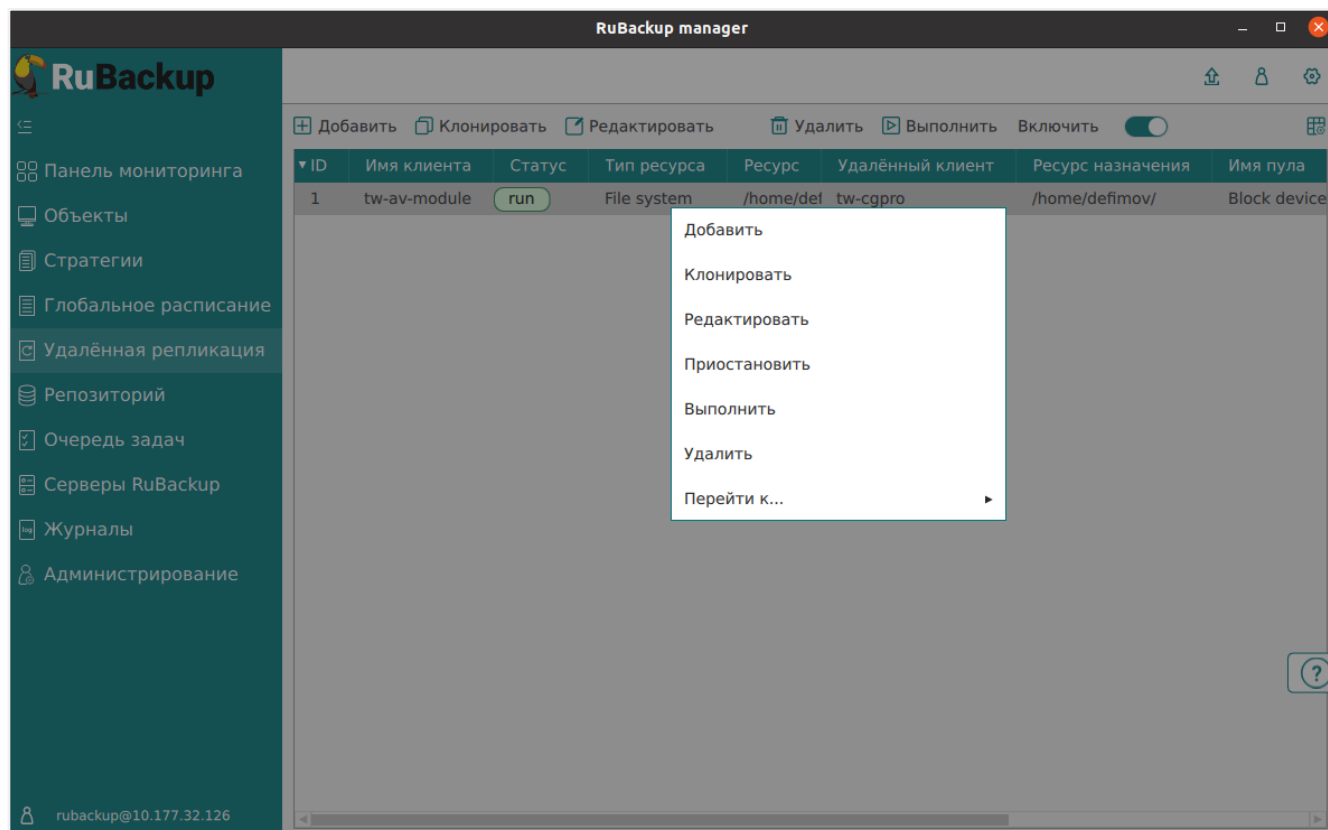

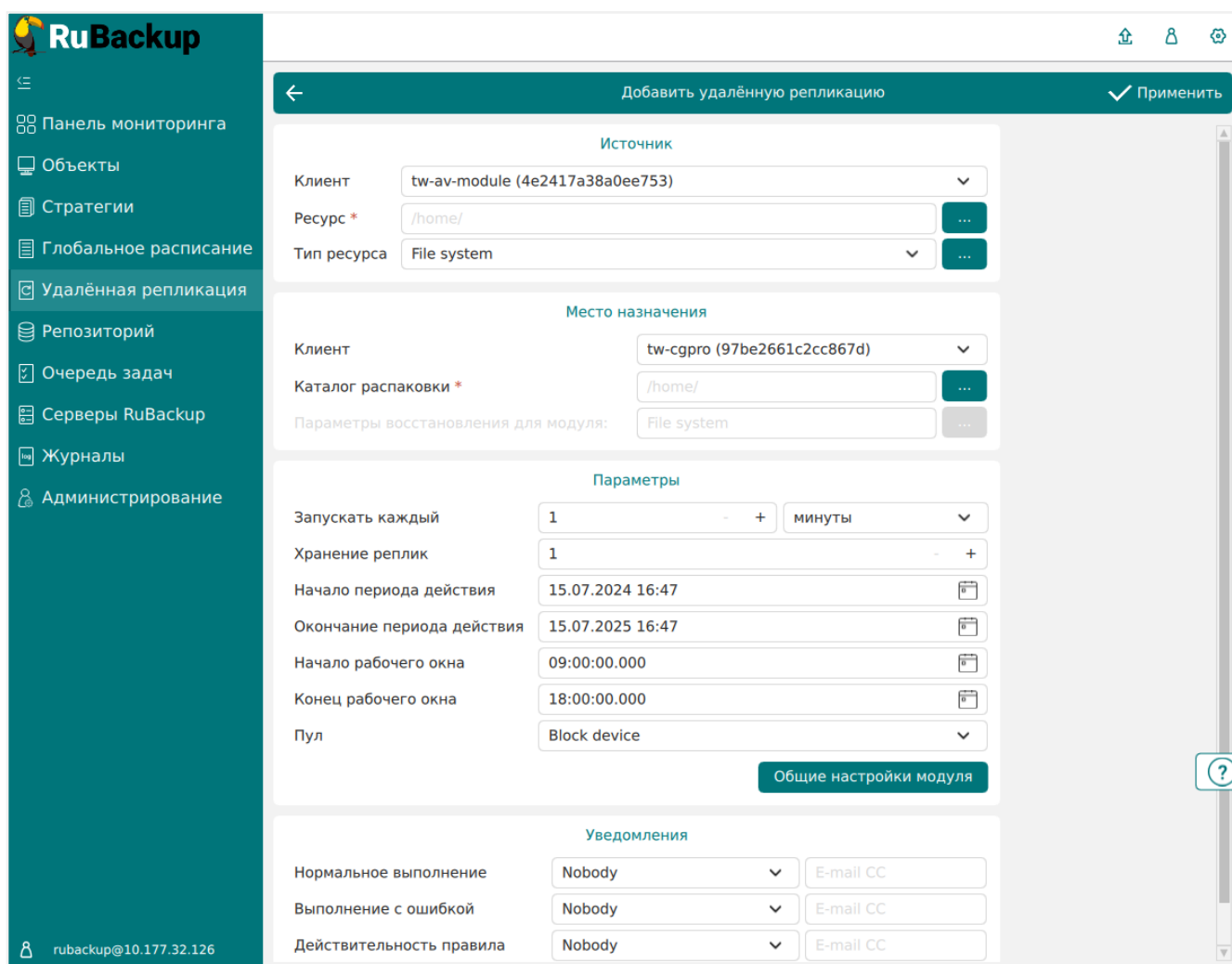


Рисунок 4. Меню для управления правилами удаленной репликации

В этом контекстном меню вы можете:

- Добавить новое правило удалённой репликации.
- Клонировать правило.
- Редактировать правило.
- Приостановить — перевести из *run* в *wait*.
- Выполнить правило немедленно.
- Удалить правило репликации.
- Перейти к... — перейти к разделам  **Репозиторий** или ☒ **Задачи**. В данных разделах будет отображаться информация только о правиле, из которого осуществляется переход.

При добавлении нового правила непрерывной удалённой репликации необходимо установить следующие параметры ([Рисунок 5](#)):



Добавить удалённую репликацию ✓ Применить

Источник

Клиент: tw-av-module (4e2417a38a0ee753)

Ресурс *: /home/

Тип ресурса: File system

Место назначения

Клиент: tw-cgpro (97be2661c2cc867d)

Каталог распаковки *: /home/

Параметры восстановления для модуля: File system

Параметры

Запускать каждый: 1 - + минуты

Хранение реплик: 1 - +

Начало периода действия: 15.07.2024 16:47

Окончание периода действия: 15.07.2025 16:47

Начало рабочего окна: 09:00:00.000

Конец рабочего окна: 18:00:00.000

Пул: Block device

Общие настройки модуля

Уведомления

Нормальное выполнение: Nobody E-mail CC

Выполнение с ошибкой: Nobody E-mail CC

Действительность правила: Nobody E-mail CC

Рисунок 5. Создание правила удаленной репликации

- Блок **Источник**:
 - Клиент (клиент системы резервного копирования, откуда будут передаваться данные).

- Ресурс (ресурс, откуда будут передаваться данные. Например, каталог, файловая система, идентификатор виртуальной машины и т. д.).
- Тип ресурса.

- Блок **Место назначения:**

- Клиент (клиент системы резервного копирования, на который будут передаваться данные).
- Каталог распаковки (директория, в которую будут переданы реплицированные данные).



Если в блоке **Источник** у поля **Ресурс** и в блоке **Место назначения** у поля **Каталог распаковки** указать одну и ту же директорию, например `/home/user`, то папка `user` из директории `/home` со всем содержимым будет реплицирована из источника в место назначения в директорию `/home/user`, то есть конечный путь реплицированной папки будет: `/home/user/user`. Чтобы этого избежать, необходимо в блоке **Место назначения** у поля **Каталог распаковки** установить путь на один каталог ниже, например, при пути источника `/home/user` путь назначения должен быть `/home`.

- Блок **Параметры:**

- Период репликации.
- Хранение реплик (количество хранимых реплик в репозитории).
- Дата начала и окончания действия правила.
- Пул для хранения резервных копий (можно использовать только пул типа **Блочное устройство**).

- Блок **Уведомления**. Настройки уведомлений о событиях правила.

В качестве места расположения реплики данных на целевом клиенте (месте назначения) вы можете выбрать иной ресурс, но он должен уже существовать, иначе задачи применения реплики на удалённом хосте будут завершаться с ошибкой. Настройки Перед настройкой непрерывной репликации необходимо оценить время, необходимое и достаточное для завершения операций по созданию и применению реплики, и в соответствии с этим настраивать период репликации правила. Вы также можете изменить настройки правила после его создания ([Рисунок 6](#)).

RuBackup

Панель мониторинга
Объекты
Стратегии
Глобальное расписание
Удалённая репликация
Репозиторий
Очередь задач
Серверы RuBackup
Журналы
Администрирование

rubackup@10.177.32.126

Добавить удалённую репликацию

✓ Применить

Источник

Клиент: tw-primary (24c336bcbf089bde)

Ресурс *: /home/

Тип ресурса: File system

Место назначения

Клиент: tw-av-module (4e2417a38a0ee753)

Каталог распаковки *: /home/

Параметры восстановления для модуля: File system

Параметры

Запускать каждый: 1 - + минуты

Хранение реплик: 1 - +

Начало периода действия: 15.07.2024 16:47

Окончание периода действия: 15.07.2025 16:47

Начало рабочего окна: 09:00:00.000

Конец рабочего окна: 18:00:00.000

Пул: Block device

Общие настройки модуля

Уведомления

Нормальное выполнение: Nobody E-mail CC

Выполнение с ошибкой: Nobody E-mail CC

Действительность правила: Nobody E-mail CC

Рисунок 6. Изменение настроек

Реплики располагаются в репозитории в виде записей с типом задачи **Create replica** (Рисунок 7).

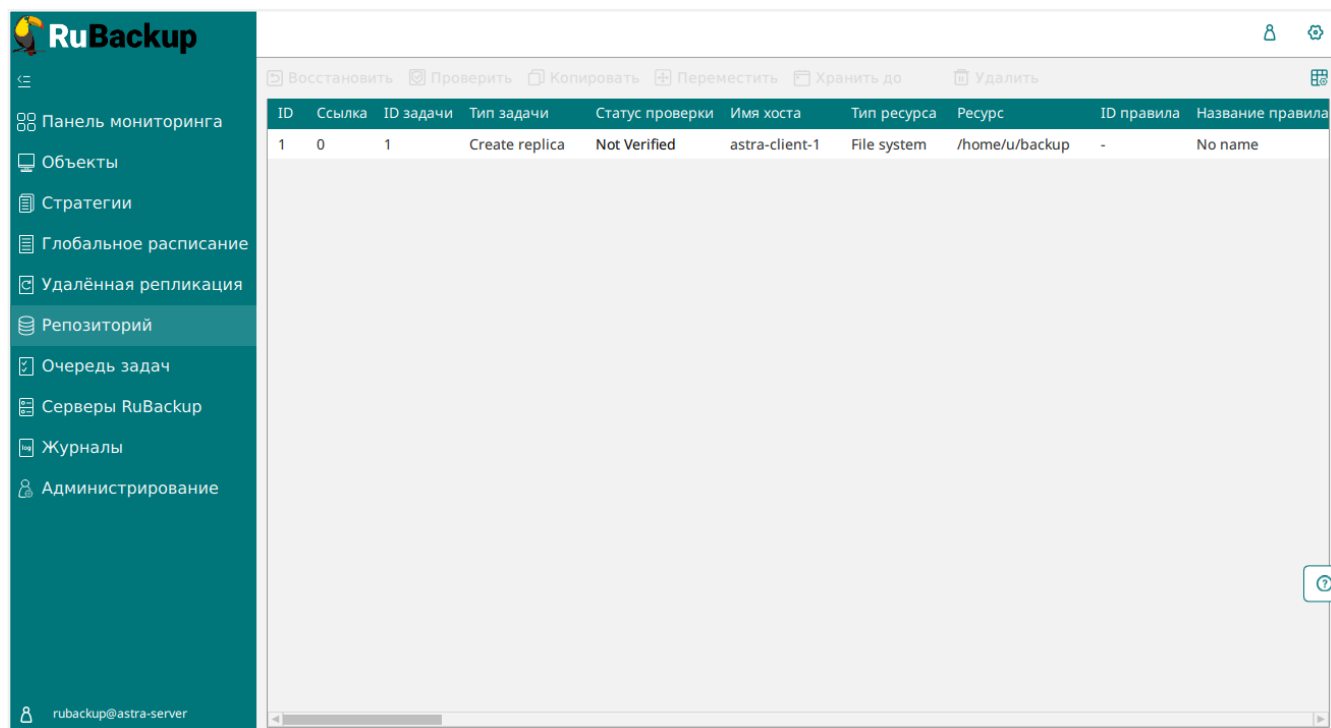
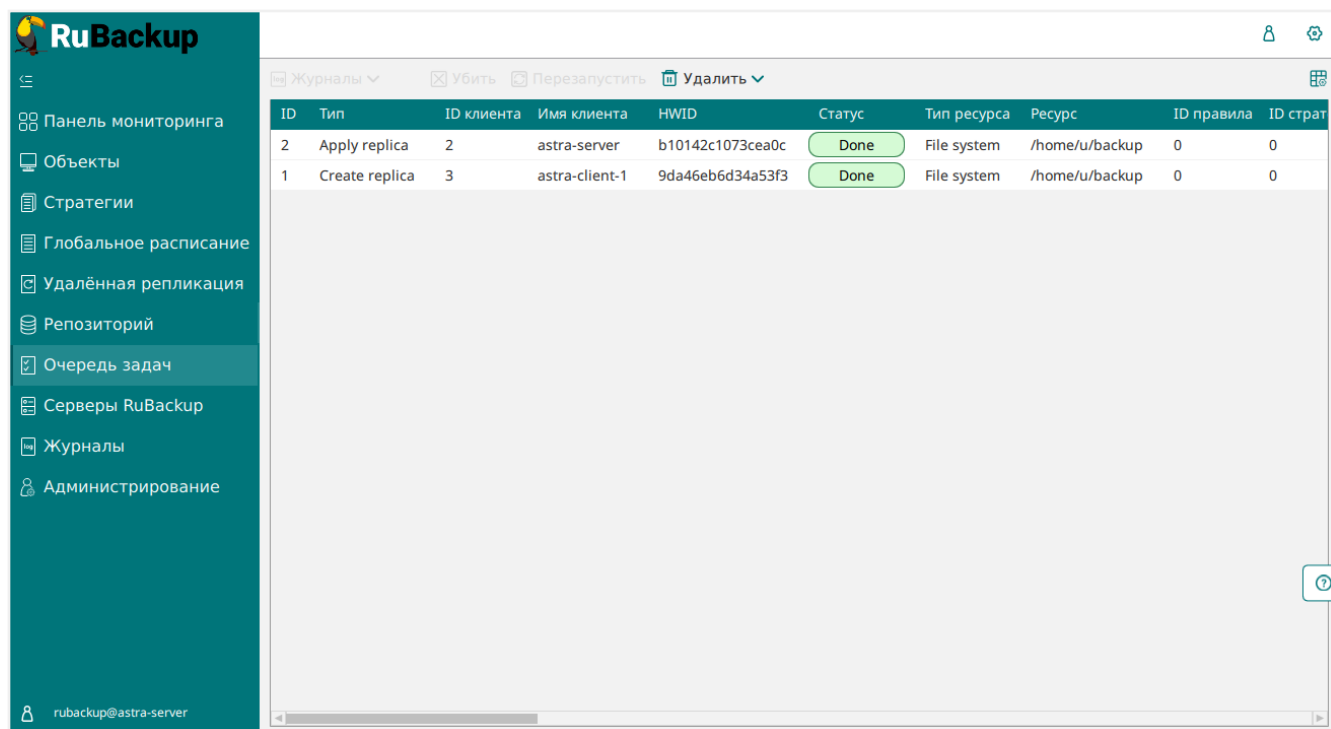



Рисунок 7. Раздел «Репозиторий»

В ходе работы старые реплики будут удаляться из репозитория, для чего в главной очереди задач будут создаваться соответствующие задачи (Рисунок 8).

Рисунок 8. Раздел  **Задачи**

Настройка клиента

В качестве источника для выполнения непрерывной удалённой репликации может быть использован любой клиент *RuBackup*, входящий в серверную группировку.

Чтобы целевой клиент (место назначения) мог применять реплику, следует в конфигурационный файл клиента `/opt/rubackup/etc/config.file` задать параметр:

```
remote-replication yes
```

Если в конфигурационном файле клиента отсутствует этот параметр, то все задачи на применение реплики (тип задачи `Apply replica`) на клиенте будут завершены с ошибкой. После изменения конфигурационного файла необходимо перезагрузить сервис (демон) клиента *RuBackup*, чтобы изменения вступили в силу.

Требования к клиенту

Для успешной репликации ресурса в месте назначения (целевом клиенте), необходимо выполнить два условия:

- Ресурс на целевом клиенте существует.
- Ресурс на целевом клиенте не используется.

При восстановлении из реплики файлового ресурса необходимо, чтобы он существовал в виде каталога с необходимыми правами и владельцем. Как минимум, необходимо создать целевой каталог.

При восстановлении из реплики виртуальной машины необходимо, чтобы она существовала в той же дисковой конфигурации, что и в оригинальной среде виртуализации. При этом её идентификатор с среде виртуализации может и, скорее всего, будет отличаться от оригинального. Чтобы создать такой источник данных на удалённом узле рекомендуется восстановить на нем резервную копию оригинального источника данных с развёртыванием, таким образом, чтобы он располагался на дисковой подсистеме реплики точно так же, как и на источнике. При этом диски виртуальной машины должны быть расположены с теми же путями, что и на источнике.

Ресурс в месте назначения (целевом клиенте), к которому применяется реплика, не должен использоваться. Если ресурсом является виртуальная машина, она должна быть выключена. Если ресурсом является какое-либо блочное устройство (например, том `LVM`), то оно должно быть отмонтировано, а приложения, которые его используют, должны быть выключены. Если ресурсом являются каталоги в какой-либо файловой системе, то в ходе репликации в них не должна осуществляться запись.

Для восстановления резервной копии на другом хосте вам необходимо включить оба клиента (источник и место назначения) в одну клиентскую группу и сделать её разделяемой (см. [Менеджер администратора RuBackup \(RBM\)](#)). После восста-

новления резервной копии клиентов можно вывести из разделяемой группы.

В некоторых случаях реплику можно использовать как обычную резервную копию. В ходе восстановления будут затребованы все блоки данных из дедуплицированного хранилища, которых нет в месте восстановления. Если в месте восстановления не хватает несколько блоков (например одного файла, если он был создан или изменён), то для восстановления будут переданы только недостающие блоки данных.

Стоит обратить внимание на то, что при восстановлении реплик как обычных резервных копий для нефайловых ресурсов (например, виртуальных машин), данные будут восстановлены в то же место, где они располагаются на источнике. При этом могут быть изменены файлы, которые находятся в месте восстановления. В любом случае, для восстановления рекомендуется использовать обычные резервные копии.

Модуль ядра Linux dattobd

`dattobd` — это модуль ядра Linux, который используется для создания снимков блочных устройств.

Модуль применяется в СРК RuBackup для резервного копирования некоторых ресурсов без остановки их работы.

В Linux существуют встроенные инструменты для создания мгновенных копий (снимков) файловой системы, из которых наиболее известен LVM. Однако, у них есть ограничения, которые делают их неудобными в работе с постоянно работающими серверами. Резервирование тома «на горячую» требует отмонтировать том, сделать его снимок, примонтировать снимок и отправить том на резервное хранение. Промышленный сервер редко может быть отключен на это время.

Блочный драйвер Datto (`dattobd`) предоставляет функциональность, похожую на теневое копирование тома (Volume Shadow Copy, VSS) в Windows и позволяет делать мгновенные снимки файловых систем. Драйвер `dattobd` может быть установлен без перезагрузки машины. `dattobd` создает снимок любого блочного устройства, после чего отслеживает инкрементальные изменения на блочном устройстве и обновляет его резервные копии, копируя только измененные блоки.



`dattobd` работает на уровне слоя блоков, и поддерживает большинство актуальных файловых систем (ext2, ext3, ext4 и xfs). Файловые системы с собственной реализацией управления блоками (ZFS, BTRFS) не поддерживаются.

Установка

Для установки модуля ядра Linux `dattobd` на ОС Astra Linux 1.7 или Astra Linux 1.8:

1. Добавьте GPG-ключ репозитория Datto:

```
sudo apt-key adv --fetch-keys https://cpkg.datto.com/DATTO-PKGS-GPG-KEY
```

2. Добавьте репозиторий Datto:

Пример 1. Добавление репозитория для ОС Astra Linux 1.7

```
echo "deb [arch=amd64] https://cpkg.datto.com/datto-deb/public/buster  
buster main" | sudo tee /etc/apt/sources.list.d/datto-linux-agent.list
```

Пример 2. Добавление репозитория для ОС Astra Linux 1.8

```
echo "deb [arch=amd64] https://cpkg.datto.com/datto-deb/public/bookworm
bookworm main" | sudo tee /etc/apt/sources.list.d/datto-linux-
agent.list
```

3. Обновите список пакетов:

```
sudo apt update
```

4. Установите пакеты:

```
sudo apt install dattobd-dkms dattobd-utils
```

5. Загрузите модуль в ядро:

```
sudo modprobe dattobd
```

Дедупликация

Система резервного копирования *RuBackup* позволяет использовать режим дедупликации при создании резервных копий данных.

В режиме дедупликации данные, которые должны попасть в резервную копию, разделяются на блоки равного размера, и для каждого блока вычисляется хеш-сумма по алгоритму `sha1`, `sha2`, `blake2b`, `skein` или `streebog`. Перед выполнением резервного копирования сервер передаёт клиенту хеш-таблицу блоков, уже расположенных в дедуплицированном хранилище и которые с высокой степенью вероятности могут содержаться в источнике данных, резервное копирование которых будет выполняться. Серверу передаются только уникальные блоки резервной копии, которые размещаются в дедуплицированном хранилище резервных копий, представляющее собой блочное устройство в операционной системе (это может быть одиночный диск, RAID массив или LUN система хранения данных).

Таким образом, при первом резервном копировании источника данных серверу резервного копирования будет передан полный уникальный набор блоков. При повторном резервном копировании будут переданы только изменившиеся блоки данных. Это позволяет уменьшить окно резервного копирования, снизить нагрузку на сеть передачи данных и сэкономить место в хранилище резервных копий.

При восстановлении сервер передаёт клиенту метафайл, содержащий всю необходимую информацию о резервной копии и целевом ресурсе, который требует восстановления. Если восстановление информации происходит непосредственно в источник данных, где были утеряны или изменены какие-либо блоки данных, и требуется восстановить целостность источника данных, то сервер передаст клиенту только те блоки данных, которые были изменены и требуют восстановления. Это позволяет значительно уменьшить время восстановления.

Система резервного копирования *RuBackup* позволяет объединять дедуплицированные блочные устройства в пулы типа **Блочное устройство**. Любой сервер в серверной группировке *RuBackup* может управлять несколькими пулами типа **Блочное устройство**. Это может быть полезно для использования пула только для определённых данных. Например, вы можете использовать один пул для хранения резервных копий виртуальных машин с гостевой операционной системой *MS Windows*, и другой пул для резервных копий ВМ с ОС *Astra Linux*. Параметры пула определяют размер блока дедупликации, алгоритм хеш-функции длину хеша.

Принципы дедупликации

При выполнении дедупликации происходит вычисление хеша для всех блоков данных, которые должны попасть в резервную копию. Хеш-алгоритмы, поддерживаемые *RuBackup*, приведены в [таблице](#).

Таблица 1. Алгоритмы хеш-функций, поддерживаемые RuBackup

Алгоритм	Длина хэш, бит	Ссылка на описание
sha1	160	https://en.wikipedia.org/wiki/SHA-1
sha2	256, 512	https://en.wikipedia.org/wiki/SHA-2
skein	256, 512	https://en.wikipedia.org/wiki/Skein_%28hash_function%29
blake2b	256, 512	https://en.wikipedia.org/wiki/BLAKE_%28hash_function%29#BLAKE2
streebog	256, 512	https://en.wikipedia.org/wiki/Streebog

Вы можете определить параметры дедупликации при создании пула типа **Блочное устройство**. К ним относятся:

- Размер блока дедупликации (от 16 КБ до 1 МБ),
- Хеш-алгоритм,
- Длина хеш (где поддерживается).

Следует учитывать, что чем больше длина хеш-функции и чем меньше размер блока дедупликации, тем больше процессорных ресурсов и времени будет затрачено на выполнение процесса дедупликации. Но чем меньше длина хеш-функции, тем больше вероятность возникновения коллизии. И чем меньше размер блока дедупликации, тем более эффективен процесс дедупликации, т.к. вероятность нахождения одинаковых блоков возрастает.

Использование дедупликации целесообразно для тех источников данных, которые могут содержать в себе повторяющиеся блоки данных. Это файловые системы, блочные устройства (например, тома LVM), виртуальные машины и т.п. Некоторые источники данных в ходе своего функционирования могут значительно изменить своё содержимое, например, СУБД после переиндексации таблиц. Использование дедупликации для таких ресурсов может быть значительно менее эффективно.

Общий алгоритм дедупликации

1. Определение блочного устройства, в которое будут переданы дедуплицированные блоки данных резервной копии после её создания.
2. Получение от сервера хеш-таблицы блоков данных, которые уже располагаются в дедуплицированном блочном устройстве и которые с наибольшей степенью вероятности могут располагаться в источнике данных, для которых выполняется резервное копирование.
3. Расчёт хеш-функций для всех блоков данных резервной копии. Если хеш находится в ранее переданной таблице, то этот блок помечается, как не требующий передачи на сервер, но учитывается в метаданных резервной копии. Блоки данных для резервной копии помещаются в дедупликационный буфер в опе-

ративной памяти клиента системы резервного копирования (параметр `deduplication-task-memory` в конфигурационном файле `/opt/rubackup/etc/config.file` определяет максимально возможный объём памяти, который разрешено использовать для этой задачи, по умолчанию равен 256 МБ). Когда буфер полностью заполнен, он передаётся на сервер резервного копирования вместе с сопроводительной хеш-таблицей.

4. Когда сервер резервного копирования принимает блоки данных от клиента, он должен проверить, что блочное устройство не содержит точно такие же блоки. Таблица всех блоков данных блочного устройства располагается в оперативной памяти сервера резервного копирования. Для быстрой проверки того, что переданные блоки точно не содержатся в блочном устройстве используется вероятностный фильтр Блума. Если блок данных точно не содержится в блочном устройстве, происходит его запись в первый свободный блок, а также происходит запись в хеш-таблицу оперативной памяти и в базу данных *RuBackup*. Если фильтр Блума указывает, что блок данных, вероятно, уже существует в блочном устройстве, происходит проверка наличия соответствующего дайджеста в общей хеш-таблице блочного устройства. Если блок найден, то происходит запись в соответствующую таблицу базы данных *RuBackup* о том, что резервная копия использует этот блок данных; если блок не найден - происходит его запись в блочное устройство в первый свободный блок, запись дайджеста в хеш-таблицу и записи в соответствующие таблицы базы данных *RuBackup*.
5. При восстановлении резервной копии происходит проверка наличия восстанавливаемых блоков непосредственно в месте восстановления. Если в месте восстановления присутствует информация, которую не нужно восстанавливать, то будут переданы только те блоки данных, которые отсутствуют в месте восстановления. Например, если в месте восстановления требуется восстановить структуру каталогов и отсутствует несколько файлов или каталогов, то сервер резервного копирования передаст только недостающие или изменённые блоки данных.

Создание резервной копии

Система осуществляет создание резервной копии с применением дедупликации следующим образом:

1. Сервер резервного копирования:

Запускает задачу резервного копирования, принимает от клиента дедуплицированные данные, размещает их в соответствующее хранилище и создаёт необходимые записи в базе данных

2. Клиент резервного копирования:

Запускает соответствующий модуль и ожидает передачу дедуплицированных

блоков от утилиты `rbfd`.

3. Модуль *RuBackup*:

Подготавливает источник данных к резервному копированию и запускает утилиту `rbfd`.

4. Утилита `rbfd`:

Выполняет дедупликацию источника данных и передаёт дедуплицированные блоки клиенту резервного копирования.

Восстановление резервной копии

Система осуществляет восстановление резервной копии, созданной с применением дедупликации, следующим образом:

1. Сервер резервного копирования:

Передаёт клиенту необходимые для восстановления блоки данных.

2. Клиент резервного копирования:

Запускает соответствующий модуль и принимает блоки данных от сервера

3. Модуль *RuBackup*:

Запускает утилиту `rbfd` и, после получения всех необходимых данных, при необходимости, развёртывает резервную копию в информационной системе.

4. Утилита `rbfd`:

Выполняет сборку данных резервной копии из дедуплицированных блоков.

Настройка

Настройка дедупликации включает в себя следующие действия:

1. На сервере *RuBackup* выделить блочное устройство для хранения.
2. На сервере *RuBackup* создать пул типа **Блочное устройство**.
3. Добавить выделенное блочное устройство в созданный пул.
4. Добавить созданный пул к правилу или стратегии резервного копирования.



Для использования дедупликации необходимо, чтобы модуль резервного копирования соответствующего типа ресурса поддерживал дедупликацию.

Блочное устройство

Чтобы использовать дедупликацию в системе резервного копирования *RuBackup*, необходимо на сервере резервного копирования выделить блочное устройство (одно или несколько) для хранения дедуплицированных резервных копий. Блочным устройством может быть обычный жёсткий диск, RAID массив или LUN система хранения данных.

В ОС *Linux* получить информацию о доступных блочных устройствах можно с помощью команды `lsblk`, например:

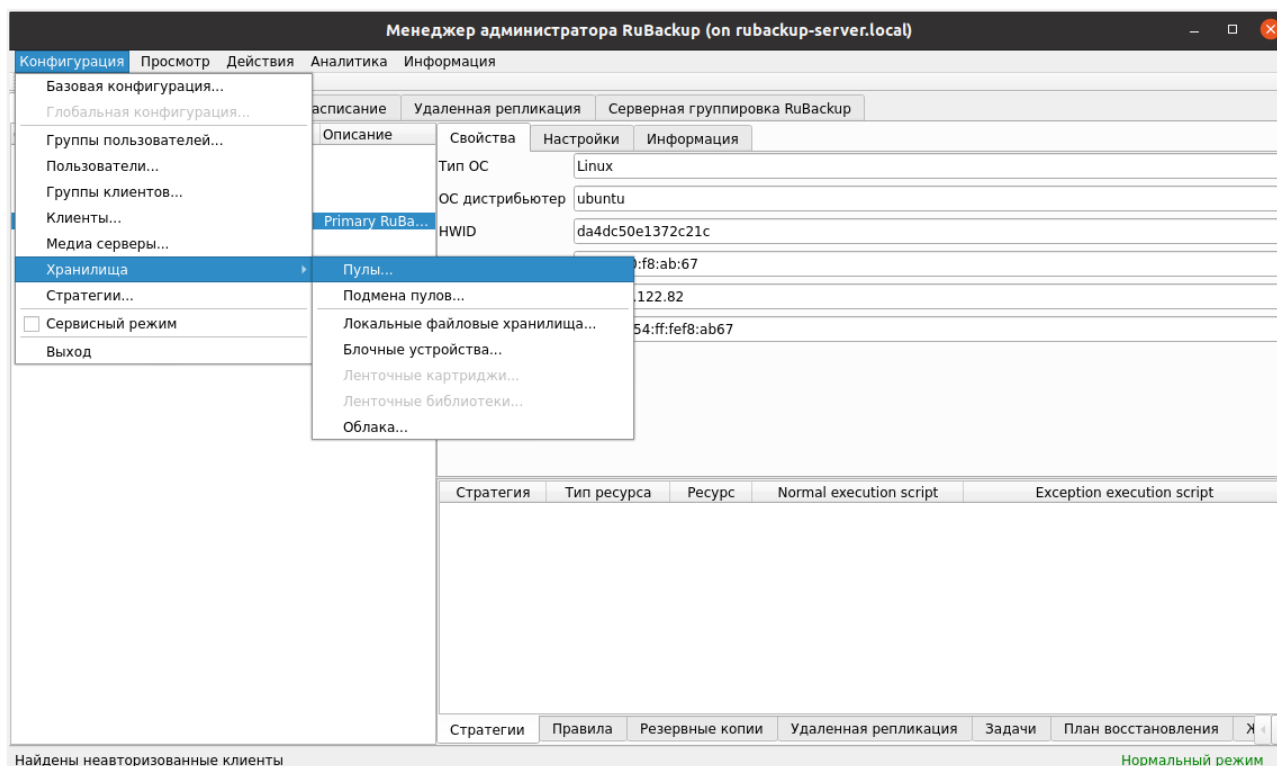
```
# lsblk
NAME                MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda                  8:0    0 931,5G  0 disk
  sda1               8:1    0 931,5G  0 part /rubackup1
sdb                  8:16    0 931,5G  0 disk
  sdb1               8:17    0 931,5G  0 part /rubackup2
sdc                  8:32    0   1,8T  0 disk
  sdc1               8:33    0   1,8T  0 part /rubackup3
sdd                  8:48    0   3,6T  0 disk
nvme0n1             259:0    0 953,9G  0 disk
  nvme0n1p1          259:1    0   512M  0 part /boot/efi
  nvme0n1p2          259:2    0 953,4G  0 part /
```

В этом примере на сервере резервного копирования в качестве устройства для хранения дедуплицированных резервных копий может быть использован диск `/dev/sdd`.

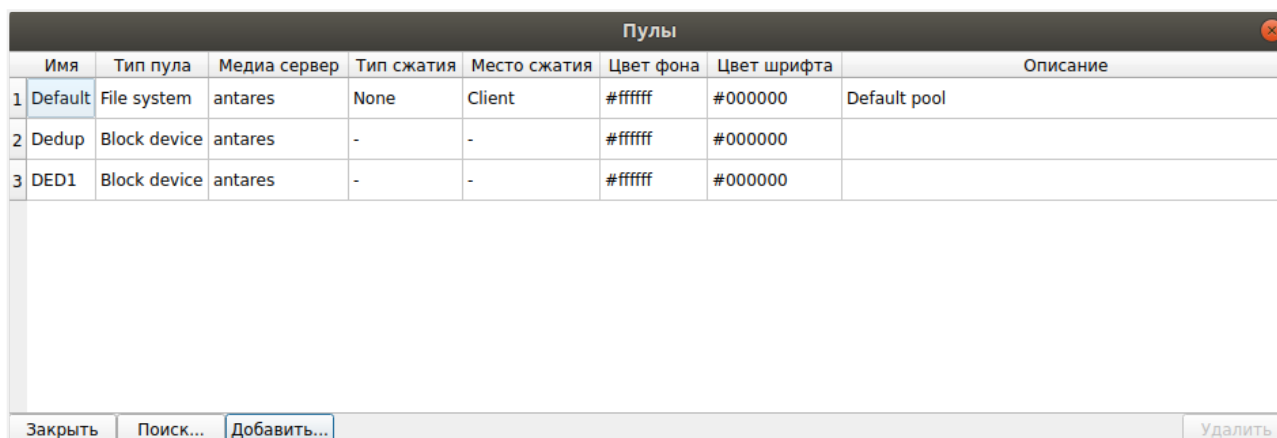
Пул хранения данных

Чтобы использовать дедупликацию на сервере резервного копирования необходимо создать пул типа **Блочное устройство**. Это можно сделать при помощи утилиты командной строки `rb_pools` или при помощи менеджера администратора *RBM*, следующим образом:

1. В главном меню *RBM* открыть пункт **Конфигурация** → **Хранилища** → **Пулы** (Рисунок 9).

Рисунок 9. Пункт **Пулы** главного меню RBM

- В окне **Пулы** нажать кнопку **Добавить** (Рисунок 10).

Рисунок 10. Окно **Пулы** в RBM

- Для нового пула указать имя и тип пула **Блочное устройство**, выбрать размер блока дедупликации, алгоритм хеш-функции и длину хеш-функции (если доступно), а также указать медиасервер, которому будет принадлежать создаваемый пул (если серверная группировка *RuBackup* содержит несколько серверов) (Рисунок 11).

Добавить новый пул (on rubackup-server.local)

Имя пула: Dedup

Тип пула: Block device

Размера блока данных: 131072

Хэш-функция: sha2

Длина хэш: 256

Медиа сервер: rubackup-server.local

Тип сжатия: None

Место сжатия: Client

Описание: Block device storage pool for deduplication

Цвет шрифта

Цвет фона

OK Cancel

Рисунок 11. Добавление нового пула в RBM

Добавление блочного устройства в пул

В созданный пул типа **Блочное устройство** необходимо добавить одно или несколько выделенных блочных устройств. Это можно сделать при помощи утилиты командной строки `rb_block_devices` или при помощи менеджера администратора *RBM*, следующим образом:

1. В главном меню *RBM* открыть пункт **Конфигурация** → **Хранилища** → **Блочные устройства** (Рисунок 12).

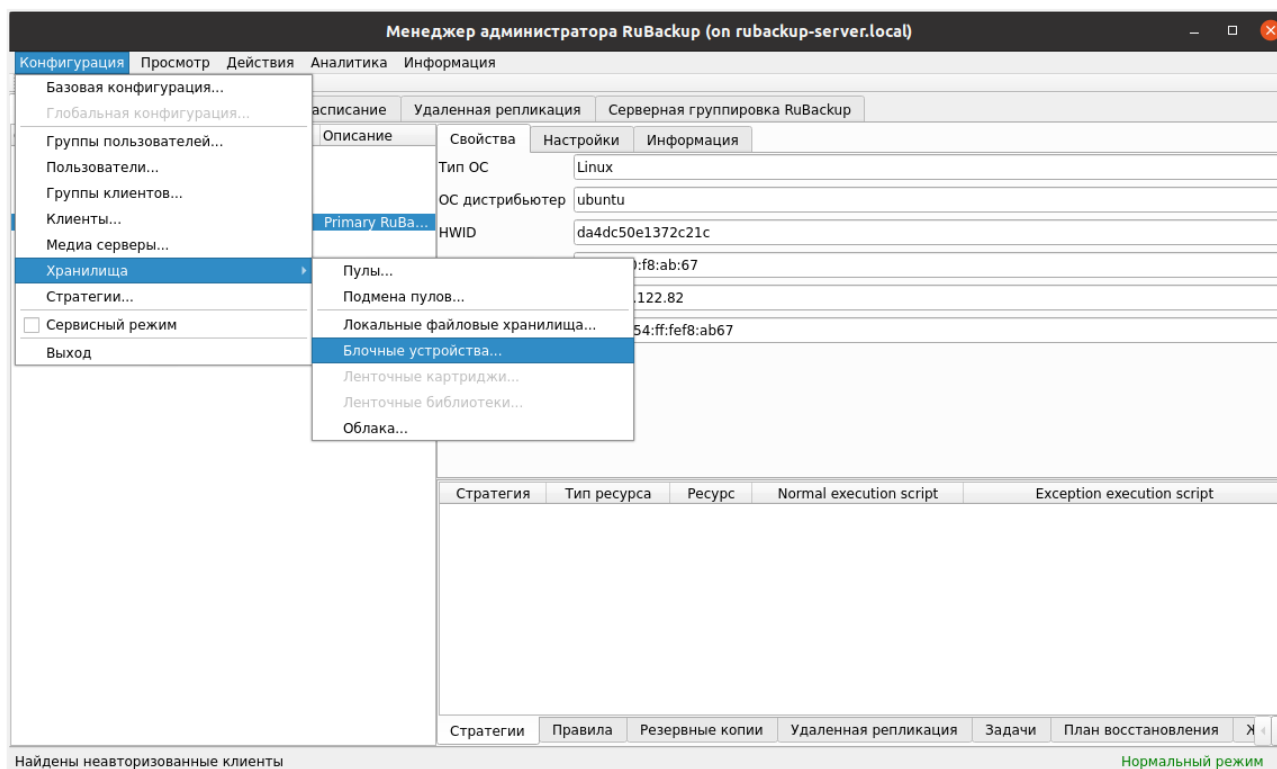


Рисунок 12. Пункт «Блочные устройства» главного меню RBM

- В окне **Пулы** нажать кнопку **Добавить** (Рисунок 13).



Рисунок 13. Окно «Блочные устройства» в RBM

- Выбрать созданный пул и выделенное блочное устройство хранения. Если на выбранном блочном устройстве уже существует файловая система, то, чтобы использовать его для хранения дедуплицированных резервных копий, следует перезаписать существующую файловую систему, включив переключатель **Перезаписать сущ. ФС** (Рисунок 14).

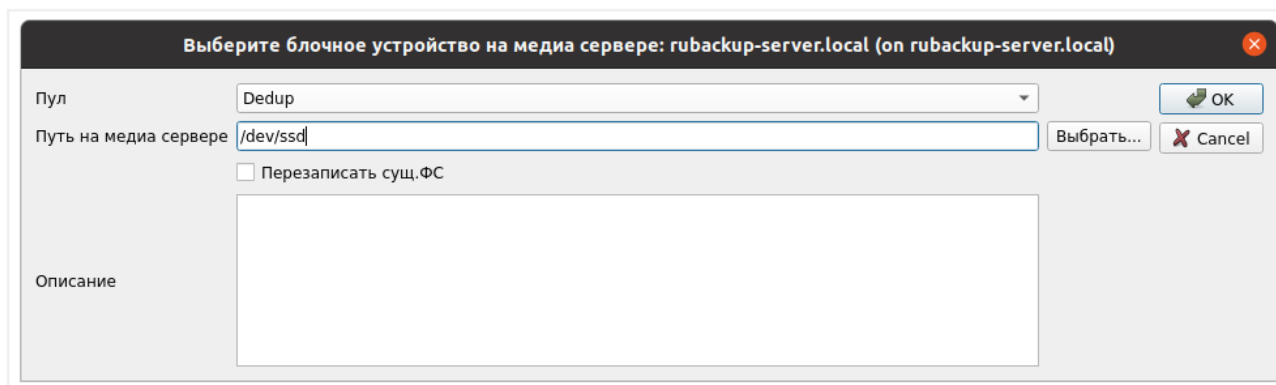


Рисунок 14. Добавление блочного устройства в пул хранения данных

После добавления блочного устройства в систему резервного копирования, оно появится в конфигурации RuBackup. При этом в системный журнальный файл на сервере резервного копирования будет записана информация о добавлении блочного устройства, например:

```
Request to add block device as storage: /dev/sda2 in pool: 'Dedup' media
server: antares
RuBackup block device signature not found on the device: /dev/sda2. Try to
create it: ffc64b63aeef891C
Block device size: 14268435456000
without signature: 14268435451904
Total usable blocks: 82047999
Create table name: deduplicated_block_device_ffc64b63aeef891C for local block
device: /dev/sdd
Local block device: /dev/sdd was included in the pool: Dedup
Load meta data of deduplicated block device: /dev/sdd in memory...
Hash table of: /dev/sda2 loaded
```

Чтобы выполнять резервное копирование с использованием дедупликации, для соответствующего правила или стратегии должен быть выбран пул типа **Блочное устройство** с назначенным в качестве хранилища резервных копий блочным устройством. Также необходимо, чтобы модуль резервного копирования соответствующего типа ресурса поддерживал дедупликацию. Если модуль не поддерживает дедупликацию, то резервное копирование будет завершено с ошибкой.

Для получения дополнительной информации об утилитах командной строки см. [Утилиты командной строки](#).

Параметры системы

Настройка глобальных параметров дедупликации осуществляется в окне настроек глобальной конфигурации системы.

Для получения доступа к меню **Глобальная конфигурация** нужно перевести

систему в сервисный режим. Для этого включите переключатель в меню **Конфигурация** → **Сервисный режим** (Рисунок 15).

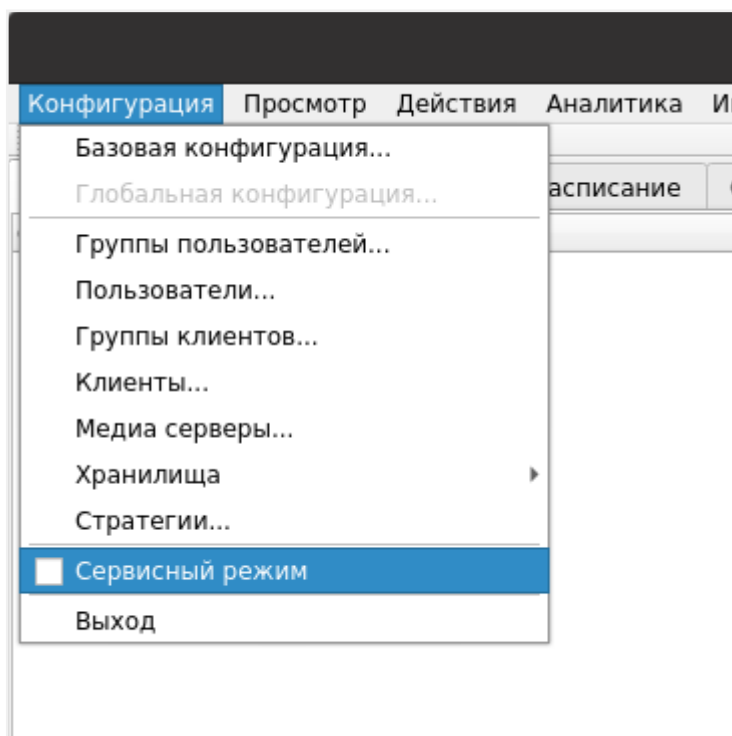


Рисунок 15. Переключение сервисного режима



По завершении работы с окном «Глобальная конфигурация» следует отключить сервисный режим.

Настройки глобальной конфигурации доступны в меню **Конфигурация** → **Глобальная конфигурация** на вкладке **Дедупликация** (Рисунок 16). Там вы можете настроить следующие параметры:

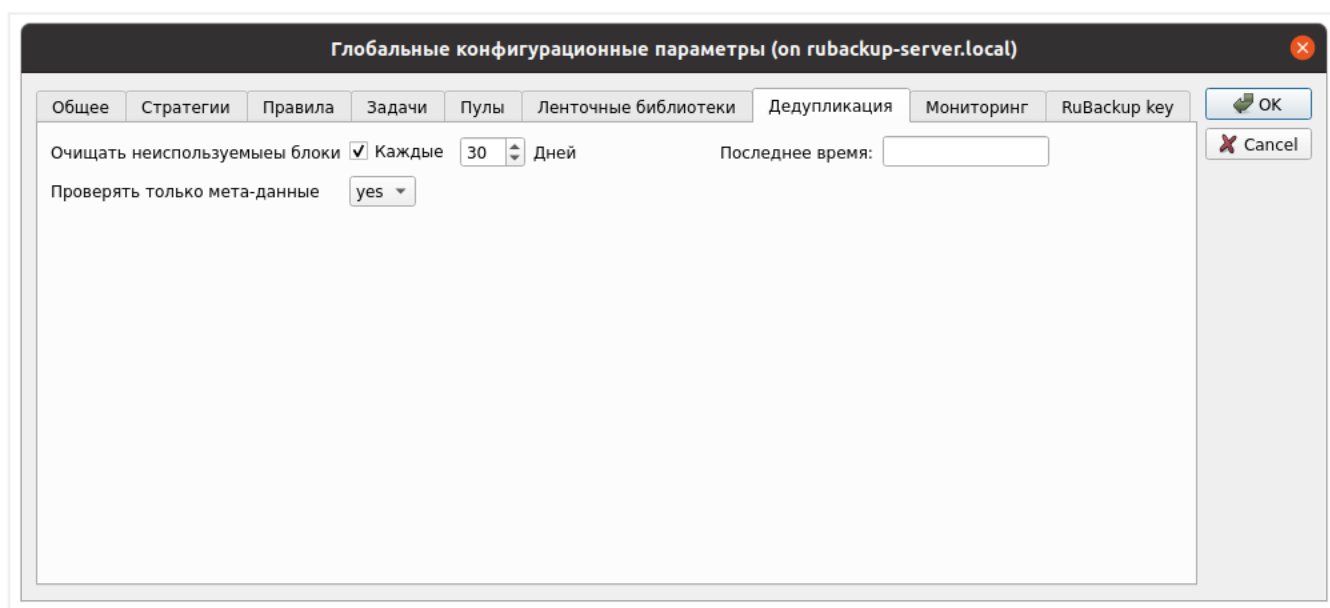


Рисунок 16. Параметры дедупликации в настройках глобальной конфигурации RuBackup

- Какие данные будут проверены на соответствие хеша при проверке резервной копии. Если параметр **Проверять только метаданные** имеет значение **yes** (по умолчанию), то будут проверены только метаданные. При значении этого параметра **no** будут проверены все используемые резервной копией блоки данных в блочном устройстве.
- Возможность периодической очистки блочных устройств. Очистка блочных устройств будет проводиться только в установленное сервисное окно, которое настраивается на вкладке **Общее** при помощи параметров **Начало сервисного окна** и **Окончание сервисного окна** (Рисунок 17).

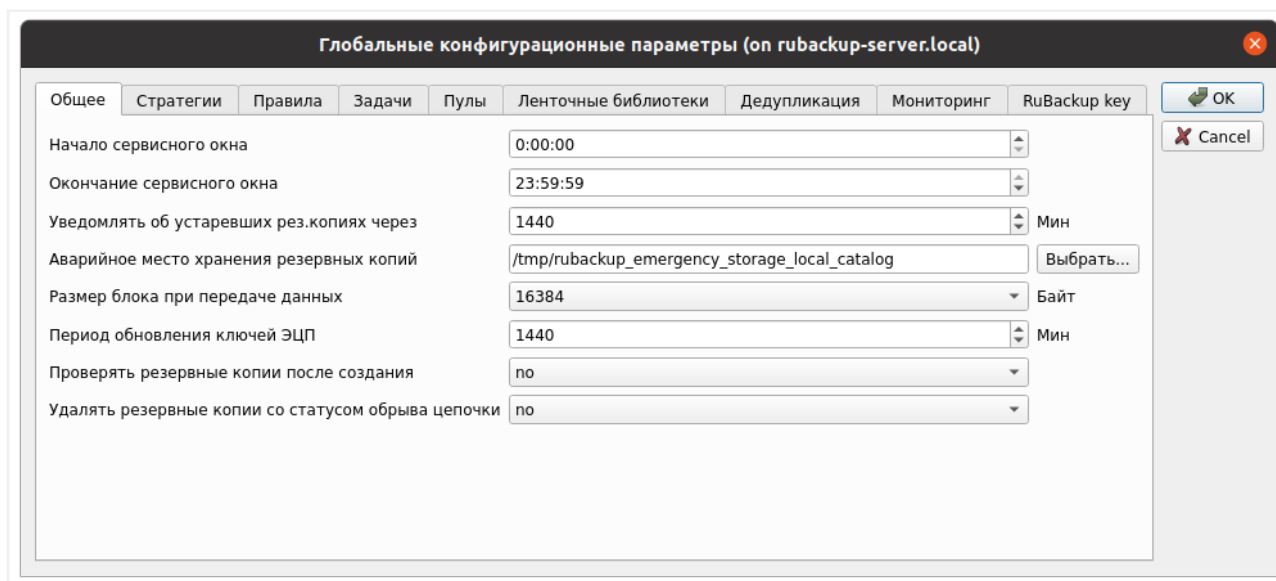


Рисунок 17. Общие параметры в настройках глобальной конфигурации RuBackup



По завершении работы с окном **Глобальная конфигурация** следует отключить сервисный режим.

Особенности

При использовании дедупликации следует учитывать следующие нюансы:

- Для использования дедупликации при выполнении резервного копирования каких-либо данных, необходимо убедиться, что модуль резервного копирования этих данных поддерживает дедупликацию. Показателем этого является поддержка модулем параметра вызова **-D**. При его вызове с этим параметром будет возвращён **0**, например:

```
/opt/rubackup/modules/rb_module_filesystem -D
echo $?
0
```

- Перемещение и копирование резервных копий, созданных с применением

дедупликации, возможно только в пулы типа **Блочное устройство**. При этом параметры пула назначения (размер блока дедупликации, алгоритм хеш-функции и длина хеш-функции) должны совпадать с параметрами пула хранения резервной копии.

- При создании дедуплицированной резервной копии создаётся метафайл, который размещается в пуле типа **Файловая система** сервера резервного копирования. В репозитории RuBackup этот файл указывается одновременно как `archive` и `snapshot` резервной копии. При этом сами данные резервной копии располагаются в блочном устройстве.
- При удалении резервной копии из репозитория происходит удаление только метафайла резервной копии и записи в базе данных RuBackup. Непосредственно блоки данных из хранилища не удаляются. Для освобождения хранилища от неиспользуемых блоков можно периодически выполнять операцию очистки. Настройка этой операции осуществляется в окне настроек глобальной конфигурации системы на вкладке **Дедупликация**.
- При выполнении операции электронной подписи резервной копии будет подписан только метафайл резервной копии, но не сами дедуплицированные блоки данных. При проверке резервной копии будет проверен метафайл. В окне настроек глобальной конфигурации системы на вкладке **Дедупликация** вы можете установить для параметра **Проверять только метаданные** значение `no`. В таком случае на соответствие хеша будут проверены все используемые резервной копией блоки данных в блочном устройстве.
- Если в пул добавлено несколько блочных устройств, то хеш-таблица уникальных блоков будет создана для каждого из устройств. Это означает, что дедупликация работает в рамках одного блочного устройства. Разные устройства могут содержать одинаковые блоки данных.
- Хеш-таблица блочного устройства загружается в оперативную память сервера резервного копирования. Это означает, что при большом объёме блочного устройства потребуется учесть необходимость в большем объёме оперативной памяти.
- Максимально возможный объём памяти для отдельной операции резервного копирования или восстановления определяется в конфигурационном файле `/opt/rubackup/etc/config.file` значением параметра `deduplication-task-memory`. Если на сервере резервного копирования предполагается выполнение большого количества одновременных операций с использованием дедупликации, необходимо учесть это в требованиях к объёму оперативной памяти сервера.
- В репозитории резервного копирования в качестве объёма дедуплицированной резервной копии указывается объём её метафайла.
- При выполнении дедуплицированного резервного копирования файловой системы с файлами разного размера, файл размером больше, чем размер дедуплицированного блока данных, займёт несколько блоков в блочном

устройстве (по возможности, последовательно). Файл размером меньше, чем размер дедуплицированного блока данных, займёт один блок.

- В случае выполнения полной резервной копии на сервер передаются только те блоки данных, которых нет в дедуплицированном хранилище. Это фактически означает, что исчезает практический смысл выполнения инкрементального и дифференциального резервного копирования, и вместо разностного резервного копирования можно всегда выполнять полное резервное копирование. Несмотря на это, модули резервного копирования могут поддерживать разностное резервное копирование и для дедупликационного режима работы.

Интеграция с ALD Pro

Выполните следующие действия для возможности авторизации доменных пользователей в *RBM* и управления СРК *RuBackup*:


- подготовка данных для настройки соединения ([Подготовка данных для настройки соединения](#));
- настройка соединения с использованием подготовленных данных ([Настройка соединения с контроллером домена](#));
- определение прав группам доменных пользователей ([Определение прав группам доменных пользователей](#)).

Подготовка данных для настройки соединения

Необходимо получить данные для последующей настройки соединения с контроллером домена *ALD Pro*, для этого:

- для установки безопасного соединения (**LDAPS**) подготовьте *сертификат контроллера домена* в формате **.pem**, обратившись к администратору *Центра Сертификации*.

Сертификат КД возможно получить из корневого хранилища сертификатов, для этого:

- откройте в браузере веб-интерфейс контроллера домена;
- нажмите на кнопку , расположенную слева от адресной строки браузера, которая отображает свойства соединения;
- просмотрите сведения о соединении и найдите информацию о сертификате;
- нажмите кнопку для просмотра сертификата;
- найдите секцию **Miscellaneous** и скачайте сертификат в формате **PEM (cert)**;
- данный шаг следует пропустить, если сертификат контроллера домена является самоподписанным, в ином случае необходимо скопировать сертификат корневого *Центра Сертификации*, выпустившего сертификат контроллера домена (или цепочки сертификатов, если сертификат контроллера домена выпущен подчинённым *Центром Сертификации*) в формате **.pem**;
- подготовьте учётные данные пользователя (**Bind User**), которому назначены права на просмотр общей информации о конфигурации службы каталогов: список существующих групп, список существующих пользователей, общая информация о пользователях;
- подготовьте названия групп пользователей, которым будут назначены права для управления СРК *RuBackup*.

Настройка соединения с контроллером домена

Необходимо сконфигурировать соединение сервера СРК *RuBackup* с контроллером домена, для этого:

- в Менеджере администратора RuBackup включите  **Сервисный режим**, активировав переключатель  в меню **Настройки**. Выполнение текущих задач будет продолжено, выполнение новых задач резервного копирования и восстановления данных будет приостановлено до момента деактивации сервисного режима. В случае, если СРК не переведена в сервисный режим, то переход в блок **Контроллеры домена** будет невозможен и пользователь будет уведомлён соответствующим предупреждением;
- перейдите в подраздел **Контроллеры домена** на вкладке **Администрирование** ([Рисунок 18](#));



Рисунок 18. Контроллеры домена

- в подразделе **Контроллеры домена** перейдите в блок **Настройки соединения** и заполните в текущем окне ([Рисунок 19](#)):

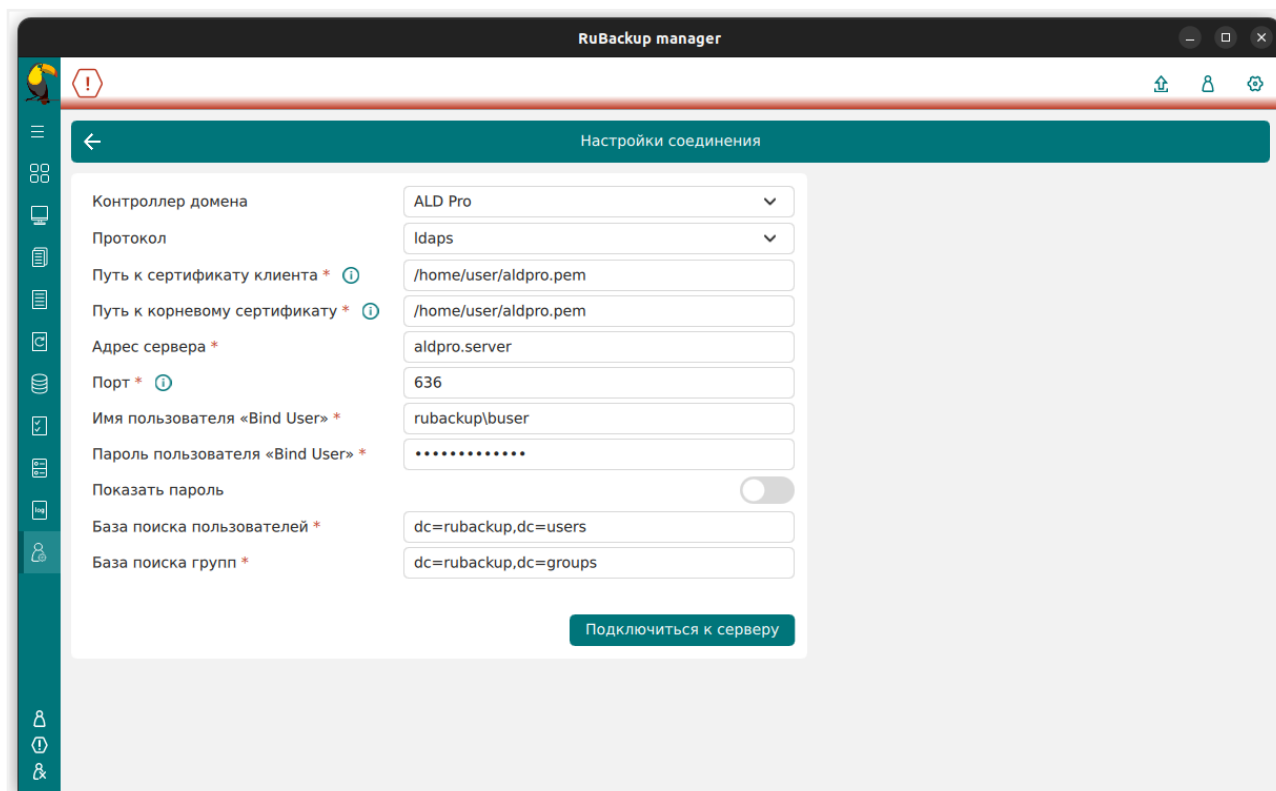


Рисунок 19. Настройки соединения

- поле **Контроллер домена** — из выпадающего списка выберите тип контроллера домена *ALD Pro*;
- поле **Протокол** — из выпадающего списка выберите тип протокола для доступа к службе каталогов: *LDAP* или *LDAPS* для защищённого соединения;
- поле **Путь к сертификату клиента** — при использовании протокола *LDAPS* необходимо указать расположение подготовленного сертификата контроллера домена в формате *.pem*;
- поле **Путь к корневому сертификату** — при использовании протокола *LDAPS* необходимо указать расположение подготовленного сертификата *Центра Сертификации*, выпустившего сертификат контроллера домена (или цепочки сертификатов, если сертификат контроллера домена выпущен подчинённым *Центром Сертификации*) в формате *.pem*;
- поле **Адрес сервера** — укажите hostname или ip-адрес контроллера домена для *LDAP*-протокола, для *LDAPS* — только hostname контроллера домена.



Имя хоста *hostname* должно совпадать с *Common Name* в сертификате контроллера домена, к которому происходит подключение;

- поле **Порт** — верификация данных учётных записей осуществляется при подключении к службе каталогов с использованием порта *389* при выборе протокола подключения *LDAP* и *636* при выборе протокола безопасного подключения *LDAPS*, доступных с основного сервера RuBackup;

- поле **Имя пользователя «Bind User»** — укажите имя учётной записи пользователя, используемой для подключения к службе каталогов, в формате `<домен>\<логин>`. Пользователь учётной записи `Bind User` должен обладать правами на получение данных о пользователях и группах из дерева LDAP, для последующей аутентификации;
- поле **Пароль пользователя «Bind User»** — укажите пароль учётной записи пользователя, используемой для подключения к службе каталогов.
- переключатель **Показать пароль** — активируйте ☐ для отображения знаков пароля, введённых в поле **«Bind User» password**;
- поле **База поиска пользователей** — укажите полный LDAP-путь к объекту, от которого в иерархии службы каталогов будет производиться поиск пользователей;
- поле **База поиска групп** — укажите полный LDAP-путь к объекту, от которого в иерархии службы каталогов будет производиться поиск групп пользователей;
- нажмите на кнопку **Подключиться к серверу**, чтобы произвести тестовый запрос и проверить подключение к указанной службе каталогов для возможности получения информации о пользователях и группах из дерева LDAP.



Если у вас не получается соединиться с контроллером домена рекомендуется ознакомиться [Решение проблем](#)



В случае некорректных учётных данных `Bind User` появится предупреждение об ошибке аутентификации LDAP/LDAPS.

В случае некорректно указанного адреса или имени сервера появится предупреждение о невозможности открытия сервера LDAP/LDAPS.

В случае успешного подключения к службе каталогов указанные настройки соединения будут сохранены в служебной базе данных *RuBackup*.

Пароль учётной записи пользователя `Bind User` сохраняется в базе данных *RuBackup* в зашифрованном средствами PostgreSQL виде.

Определение прав группам доменных пользователей

Необходимо определить роли (права) СРК *RuBackup* для групп доменных пользователей, для этого в подразделе **Контроллеры домена** перейдите в блок **Ассоциации групп и ролей** или после успешного завершения настройки соединения со службой каталогов автоматически откроется окно **Ассоциации групп и ролей** (Рисунок 20).

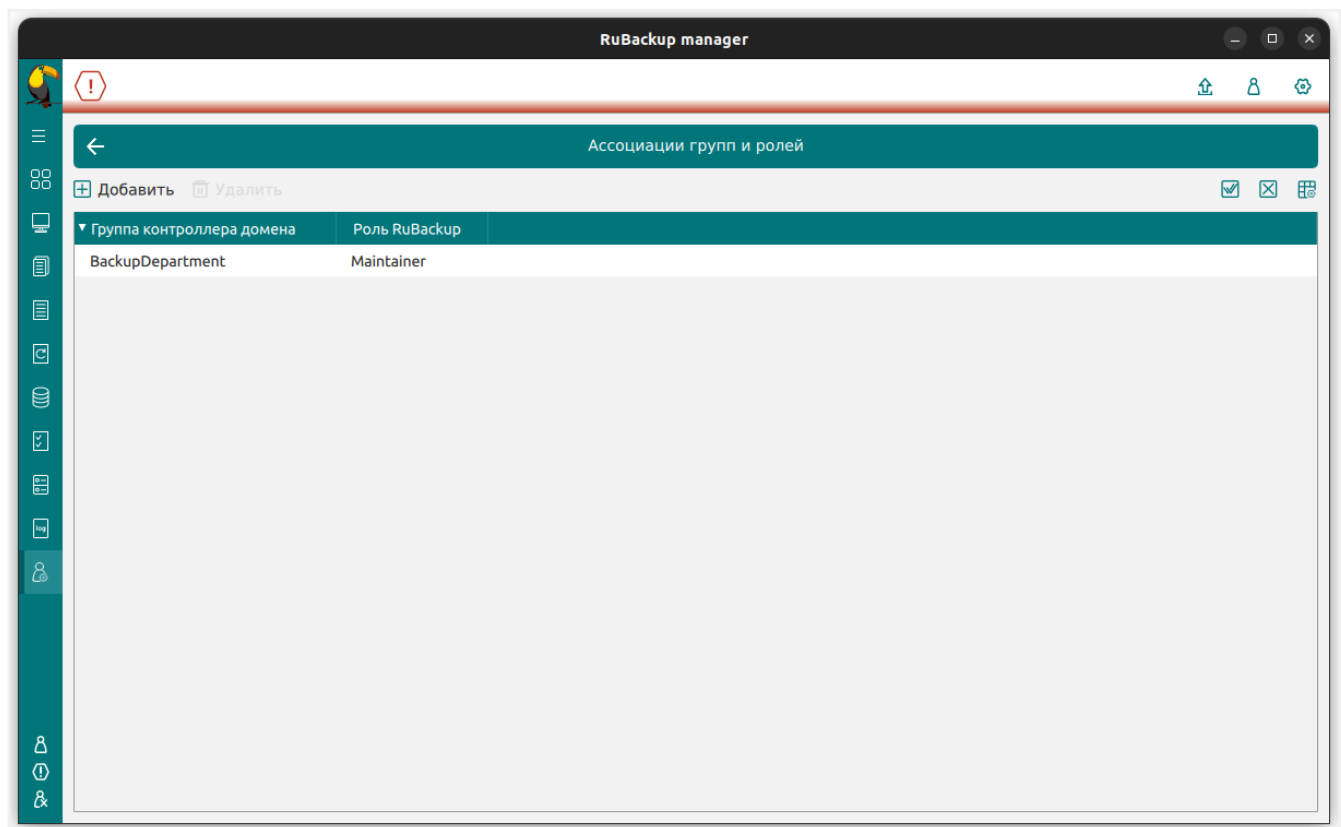


Рисунок 20. Ассоциации групп и ролей

Добавление ассоциации группы

Нажмите кнопку **+** **Добавить** и в открывшейся форме заполните поля ([Рисунок 21](#)):

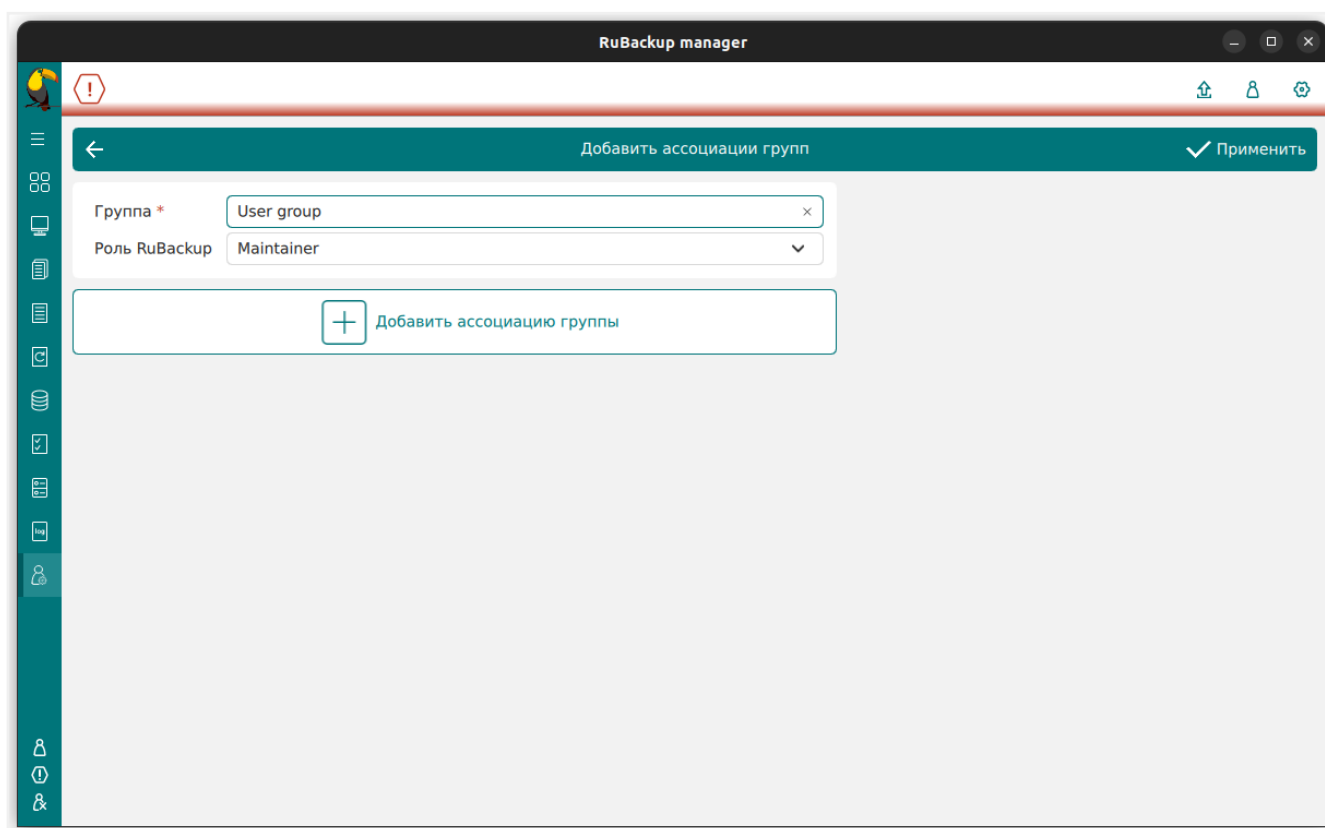


Рисунок 21. Добавить ассоциации групп

- в поле **Группа** введите название доменной группы пользователей, которым будет назначена роль. Для одной группы может быть назначена только одна роль в СРК *RuBackup*;
- в поле **Роль RuBackup** из выпадающего списка выберите роль СРК *RuBackup* (администратор, супервайзер, сопровождающий, аудитор) для указанной в поле **Группа** доменной группы пользователей;
- нажмите кнопку **Добавить ассоциацию группы** для назначения нескольким доменным группам прав доступа к СРК *RuBackup*;
- для применения назначения нажмите кнопку **Применить**.

Удаление ассоциации группы

Для удаления ассоциации группы выделите её левой кнопкой мыши в окне блока **Ассоциации групп и ролей** и нажмите активировавшуюся кнопку **Удалить**.

Решение проблем

Проверка параметров соединения с контроллером домена

Чтобы проверить параметры соединения, можно воспользоваться сторонней утилитой `ldapsearch`. Использовать утилиту необходимо на одном хосте с сервером *RuBackup*:



Установите утилиту ^[1], если она отсутствует:

```
sudo apt install ldap-utils
```

1. С помощью web интерфейса ALD PRO найдите необходимого пользователя и зафиксируйте данные полей (Рисунок 22):

- **Логин** - логин пользователя
- **Пароль** - пароль пользователя
- **Расположение подразделения в организационной структуре** (важна часть **dc**) - база поиска

The screenshot shows the 'Пользователи и компьютеры' (Users and Computers) section in the ALD PRO web interface. The breadcrumb path is 'Пользователи > Пользователь: bind bind'. The 'Основное' (Main) tab is selected, displaying the following fields:

- Идентификатор пользователя (UID)** (required): 1917600019
- Логин** (required): bind_user
- Пароль**: Введите пароль (password field with eye icon)
- Подтверждение пароля**: Введите пароль (password field with eye icon)
- Окончание действия пароля**: 11/13/2024, 5:23:26 PM
- Организационные данные** (expanded):
 - Подразделение** (required): rubakup.test
 - Расположение подразделения в организационной структуре**: ou=rubakup.test,cn=orgunits,cn=accounts,dc=rubakup,dc=test

Buttons on the right include 'Заблокировать УЗ' (Lock User) and 'Сбросить пароль' (Reset Password).

Рисунок 22. Web интерфейс ALD PRO

2. Убедитесь с помощью утилиты `ldapsearch`, что используются правильный логин, пароль и база поиска записанные в первом пункте (данные необходимо подставить свои):

```
ldapsearch -x -h 10.177.32.23 -D
'uid=bind_user,cn=users,cn=accounts,dc=rubackup,dc=test' -b
'dc=rubackup,dc=test' -w '1q2w3e4r'
```

где:

- `-x` - использование простой аутентификации вместо SASL;
- `-h` - адрес контроллера домена;
- `-D` - параметры для авторизации;
- `-b` - база поиска;
- `-w` - пароль пользователя;

В случае правильных данных для соединения утилита `ldapsearch` выведет информацию (для вышеописанного примера) ([Рисунок 23](#)):

```
isurovegin@isuroveginbackup:~$ ldapsearch -x -h 10.177.32.23 -D 'uid=bind_user,cn=users,cn=accounts,dc=rubackup,dc=test' -b 'dc=rubackup,dc=test' -w '1q2w3e4r'
# extended LDIF
#
# LDAPv3
# base <dc=rubackup,dc=test> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#
# users, compat, rubackup.test
dn: cn=users,cn=compat,dc=rubackup,dc=test
objectClass: extensibleObject
cn: users
# auditor, users, compat, rubackup.test
dn: uid=auditor,cn=users,cn=compat,dc=rubackup,dc=test
objectClass: posixAccount
objectClass: inetOrgPerson
```

Рисунок 23. Вывод утилиты `ldapsearch`

Если утилита выводит сообщение об ошибке, то проблема в настройке ALD PRO. Необходимо уточнить параметры для авторизации и повторить пункт 2 настоящей инструкции.

[1] Подробную информацию об утилите смотрите на официальном ресурсе

Интеграция с Microsoft Active Directory

Система резервного копирования и восстановления данных RuBackup (далее — СРК, Система) предоставляет возможность использовать имеющиеся учетные данные *MS AD* для аутентификации и работы в СРК *RuBackup*. Предварительно необходимо установить соединение с *MS AD* и настроить ассоциацию групп *MS AD* с ролями пользователей СРК *RuBackup*.

Предварительные настройки

СРК поддерживает интеграцию с *Microsoft Active Directory* версий *2012 R2* или *2016*, развернутой на *Microsoft Windows Server 2016*.

1. Установите и настройте *MS AD*. Для этого:

- Скачайте корневой сертификат в Службе сертификации и разместите его на основном сервере *RuBackup* в формате **PEM**. Для конвертации сертификата в формат **PEM** выполните команду:

```
openssl x509 -inform der -in <имя_сертификата>.cer -out  
<имя_сертификата>.pem
```



Имя хоста в сертификате должно совпадать с именем хоста, на котором запущен *Microsoft Windows Server 2016* с настроенным на нем сервисом *MS AD* и к которому будет осуществляться подключение по протоколу **LDAP/LDAPS**.

- Сконфигурируйте сервис *MS AD*;
- Создайте необходимые группы пользователей в *MS AD*;
- Создайте пользователя *MS AD*, который будет использоваться в качестве служебного (**Bind User**). Пользователь **Bind User** должен иметь права на просмотр общей информации о конфигурации: список существующих групп, список существующих пользователей, общая информация о пользователях;
- С помощью стандартных средств *Microsoft Windows* убедитесь, что *MS AD* доступна через **LDAP/LDAPS**-протоколы. Это можно сделать с помощью стандартной утилиты **ldp.exe**;
- Скачайте клиентский сертификат и разместите его на основном сервере *RuBackup* в формате **PEM**. Для конвертации сертификата в формат **PEM** выполните команду:

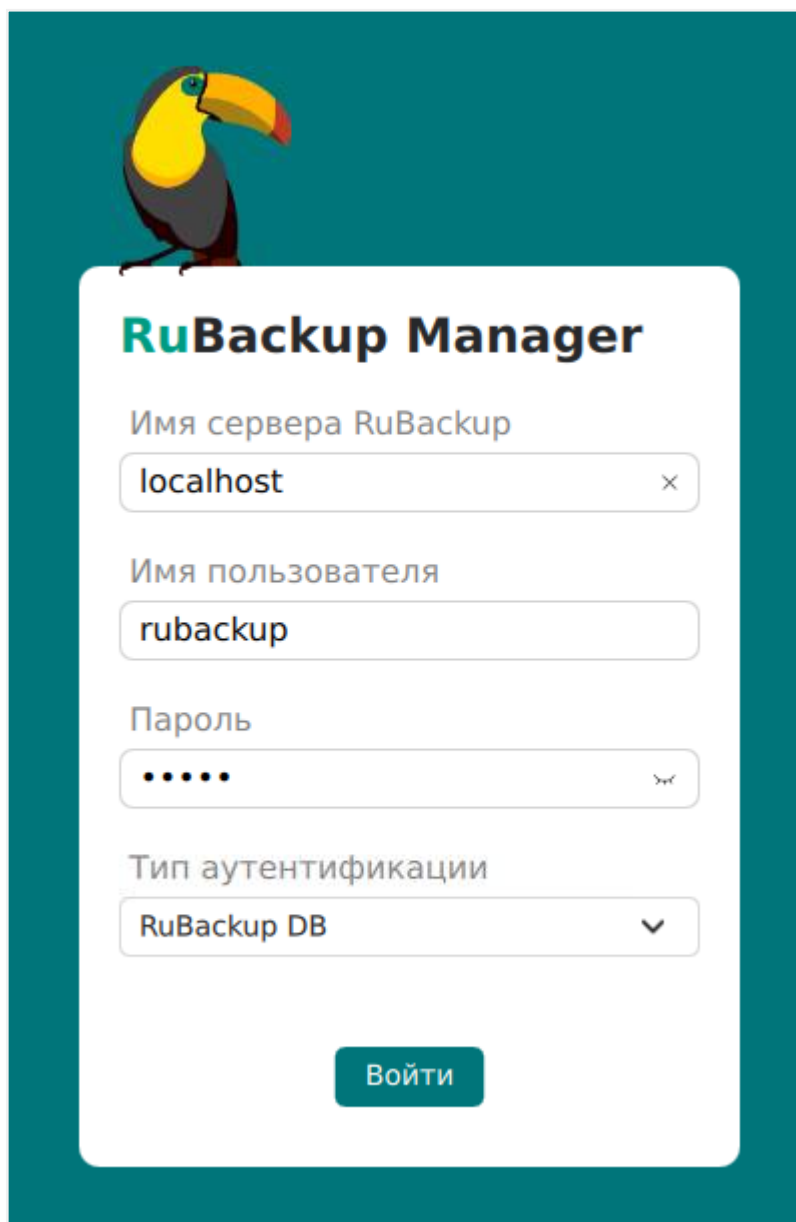
```
openssl x509 -inform der -in <имя_сертификата>.cer -out
```

<имя_сертификата>.pem

2. Обеспечьте возможность подключения *MS AD* по протоколам **LDAP/LDAPS** с хоста, на котором установлен сервер СРК (как основной, так и резервный). Для этого нужно, чтобы:
 - Хост, на котором запущен *Microsoft Windows Server 2016*, был доступен по имени с хоста, на котором установлен основной сервер *RuBackup*;
 - Были доступны порты **389** (**LDAP**) и **636** (**LDAPS**) с сервера *RuBackup*.

Первичная настройка СРК для работы с MS AD

1. Запросите у Администратора *MS AD* наименования созданных групп пользователей, которые будут ассоциированы с ролями СРК, а также аутентификационную информацию служебной учетной записи **Bind User**, обладающей правами на получение данных о пользователях и группах из дерева **LDAP**, для последующей аутентификации.
2. Войдите в *RBM* посредством существующего механизма аутентификации, основанного на СУБД *PostgreSQL* ([Рисунок 24](#)).



RuBackup Manager

Имя сервера RuBackup

localhost

Имя пользователя

rubackup

Пароль

.....

Тип аутентификации

RuBackup DB

Войти

Рисунок 24. Авторизация в RBM

3. Активируйте в *RBM* сервисный режим СРК в разделе настроек в правом верхнем углу экрана ([Рисунок 25](#)).

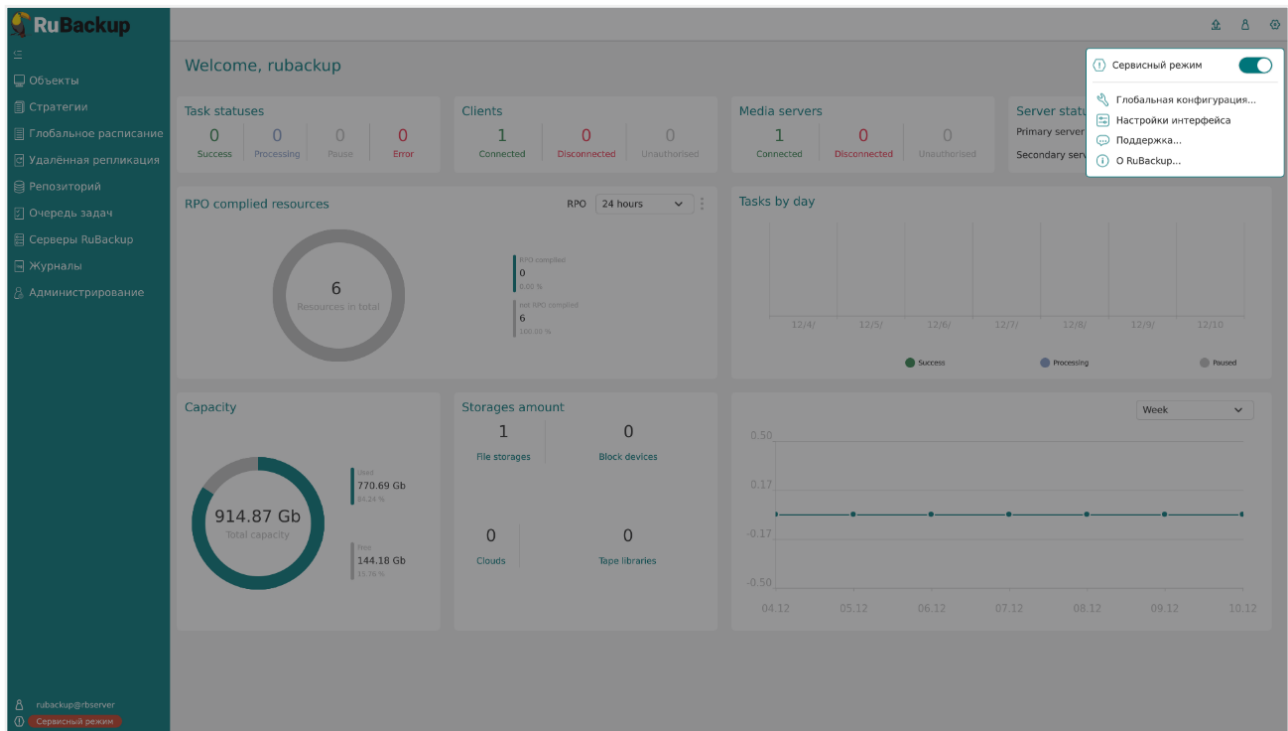
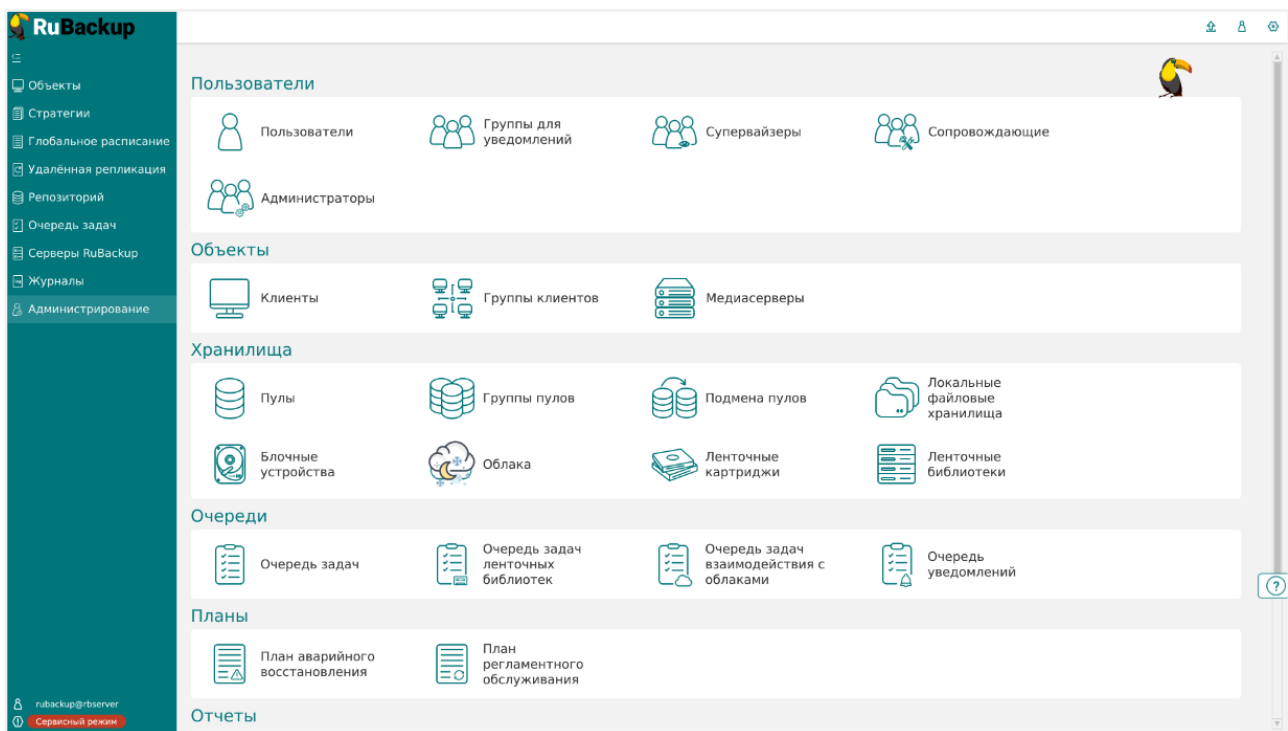


Рисунок 25. Активация Сервисного режима

4. Перейдите в раздел **Администрирование** (Рисунок 26).

Рисунок 26. Раздел **Администрирование**

5. Перейдите в подраздел **Настройки соединения с MS Active Directory** (Рисунок 27).

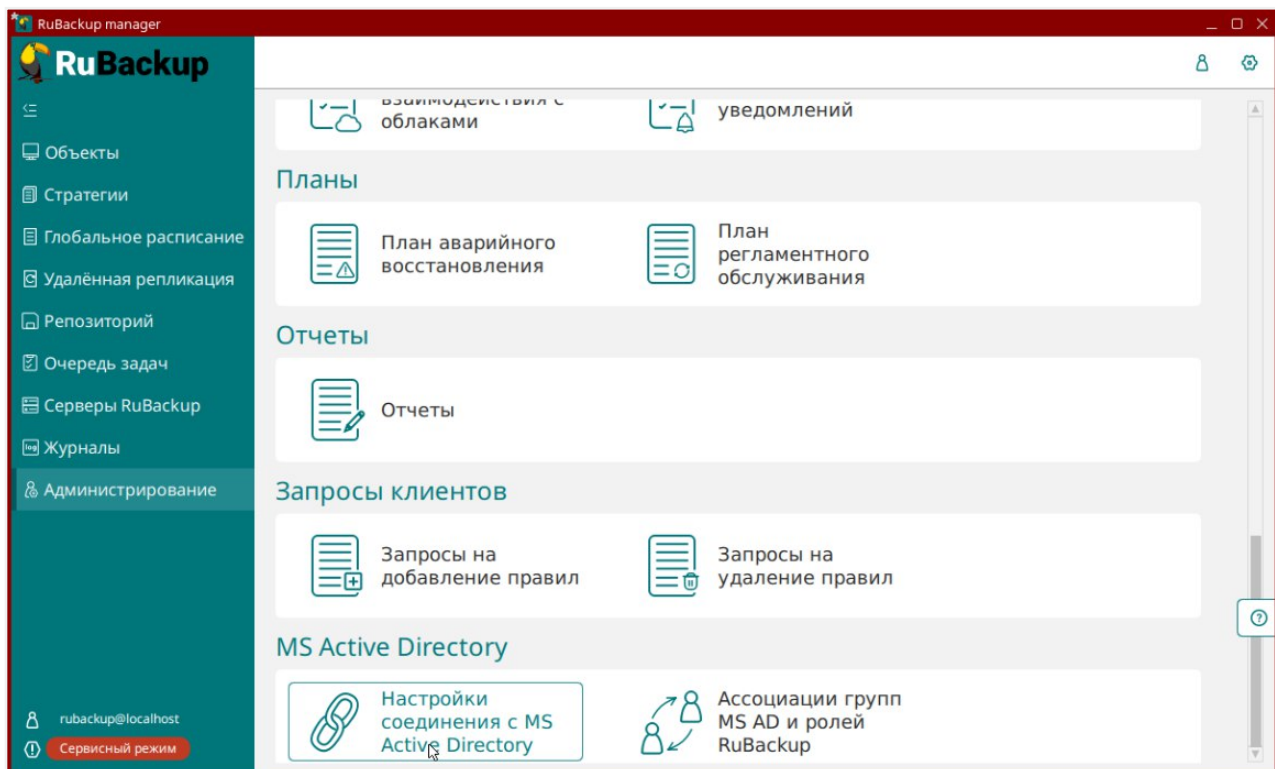


Рисунок 27. Переход в **Настройки соединения с MS Active Directory**

6. Укажите следующие настройки для подключения к MS AD (Рисунок 28):

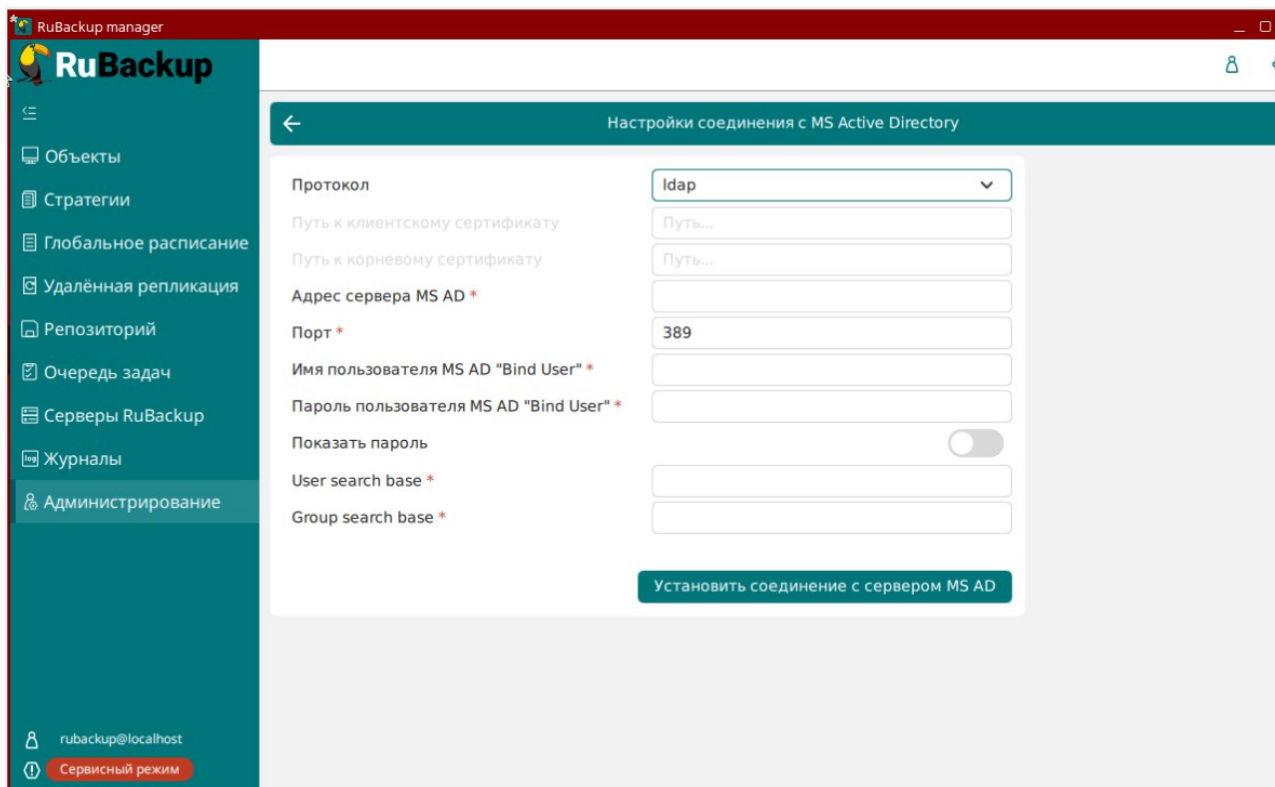


Рисунок 28. Настройки соединения с MS Active Directory

- **Протокол** (LDAP/LDAPS);

При выборе **LDAPS** указывается путь к клиентскому и корневому сертификатам.

там *Службы сертификации*, выдающей сертификаты контроллерам домена (Рисунок 29).

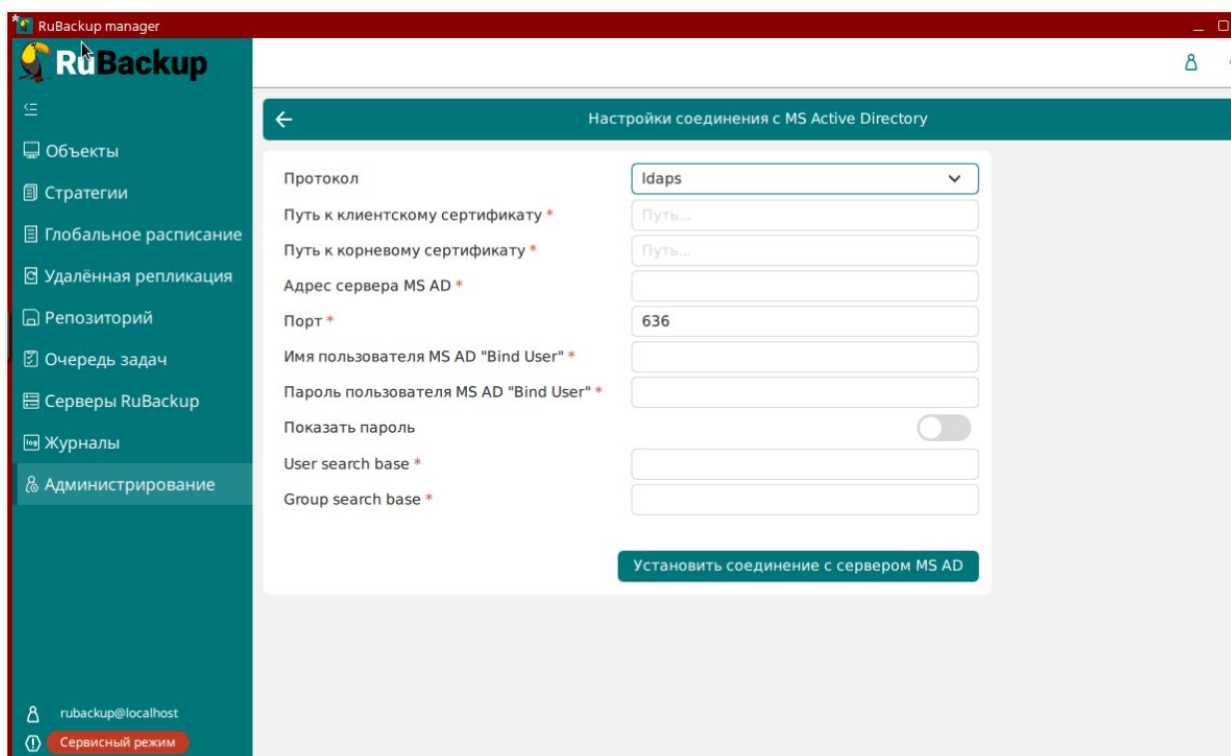


Рисунок 29. Выбор протокола

Сертификаты должны находиться на основном сервере СРК. Проверкой сертификатов будет служить первое подключение к серверу *MS AD*;

- **Адрес сервера *MS AD*** - `hostname` или `ip-адрес` для `LDAP`-протокола, для `LDAPS` — только `hostname`.

При установке соединения с неправильным адресом сервера появится предупреждение (Рисунок 30):

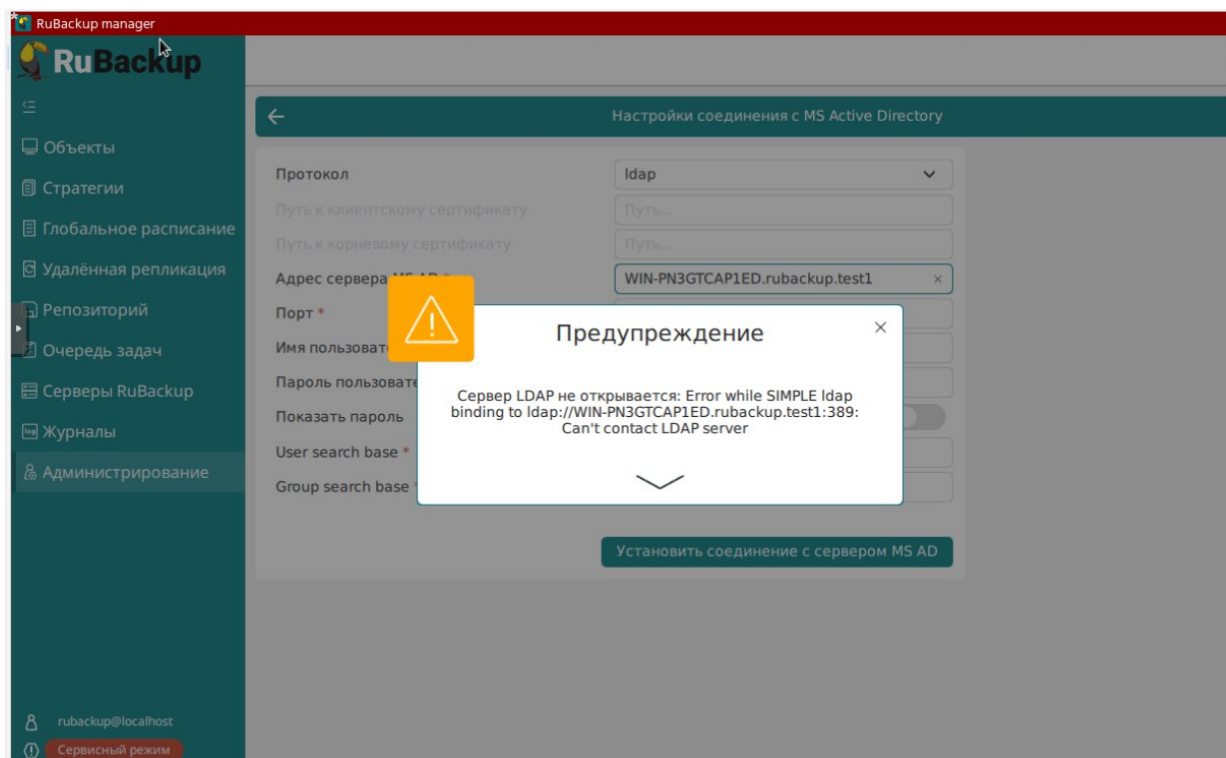


Рисунок 30. Предупреждение. Сервер LDAP не открывается

- **Порт.** Значениями по умолчанию являются — 389 для LDAP, для LDAPS — 636;
- Учетные данные для служебного пользователя Bind User: домен и логин в формате <домен>\<логин>, а также пароль;



Логин, пароль и базу поиска можно узнать в свойствах пользователя MS AD (Рисунок 31):

- Логин - второе поле секции **"Имя входа пользователя (пред- Windows 2000)"** (bind_user для данного случая);
- Пароль - устанавливается если нужен;
- Домен - первое поле секции **"Имя входа пользователя (пред- Windows 2000)"** (RUBACKUP для данного случая);
- База поиска - второе поле секции **"Имя входа пользователя"**. Для каждой части устанавливается тег dc. (rubackup.msad образует базу поиска "dc=rubackup, dc=msad" для данного случая)

Свойства: bind user

Профиль служб удаленных рабочих столов COM+

Член групп Входящие звонки Среда Сеансы Удаленное управление

Общие Адрес Учетная запись Профиль Телефоны Организация

Имя входа пользователя:
bind_user @rubackup.msad

Имя входа пользователя (пред-Windows 2000):
RUBACKUP\ bind_user

Время входа... Вход на...

☐ Разблокировать учетную запись

Параметры учетной записи:

- ☐ Требовать смены пароля при следующем входе в систему
- ☒ Запретить смену пароля пользователем
- ☒ Срок действия пароля не ограничен
- ☐ Хранить пароль, используя обратимое шифрование

Срок действия учетной записи

☒ Никогда

☐ Истекает: 7 марта 2025 г.

OK Отмена Применить Справка

Рисунок 31. Свойства пользователя

При установке соединения с неправильным логином и паролем появится предупреждение (Рисунок 32):

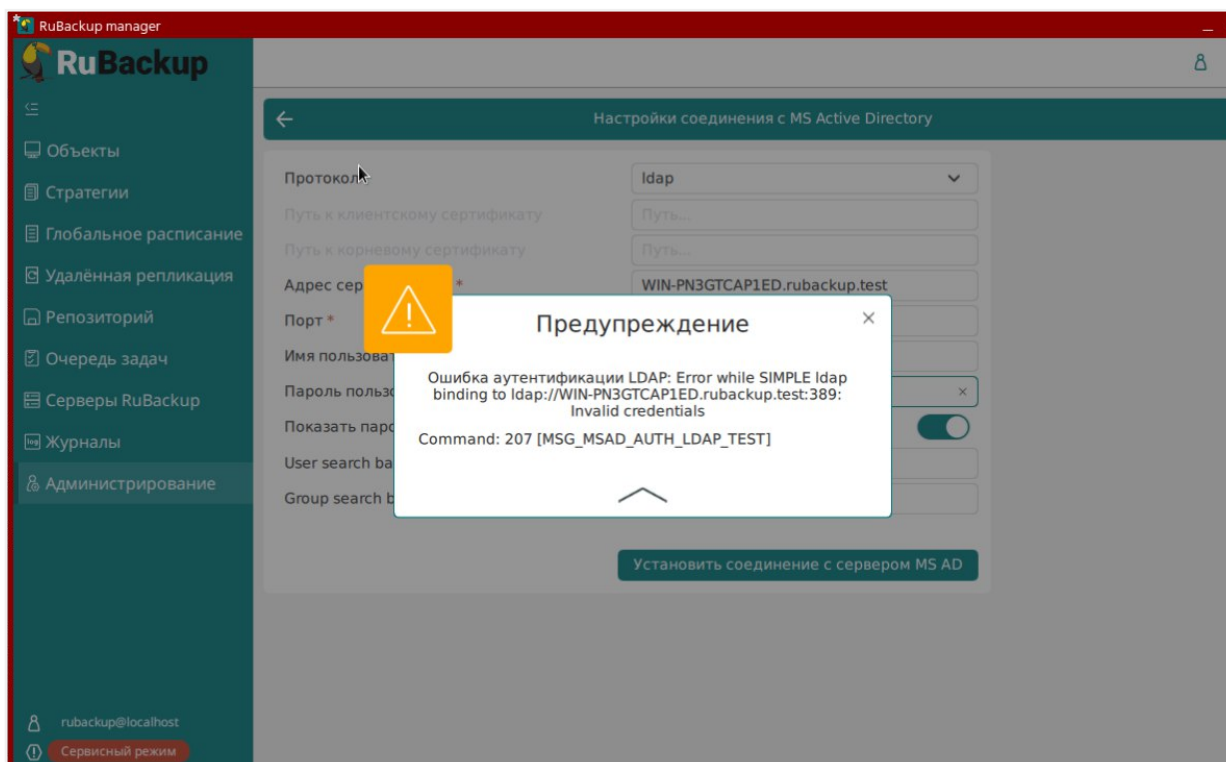


Рисунок 32. Предупреждение. Ошибка аутентификации LDAP

- **User search base** — указывает, от какого объекта в иерархии *Active Directory* начинать поиск пользователей;
 - **Group search base** — указывает, от какого объекта в иерархии *Active Directory* начинать поиск групп.
7. Нажмите на кнопку **Установить соединение с сервером MS AD**, чтобы произвести тестовый запрос и проверить:
- Возможность подключения к указанному серверу *MS AD*, используя предоставленные параметры для подключения;
 - Возможность получения списка информации о пользователях и группах из дерева *LDAP*.
8. Если вы успешно прошли шаги с п. 6, предварительная настройка *СРК* для работы с *MS AD* успешно завершена — открывается окно **Ассоциация групп MS AD и ролей RuBackup** (Рисунок 33):

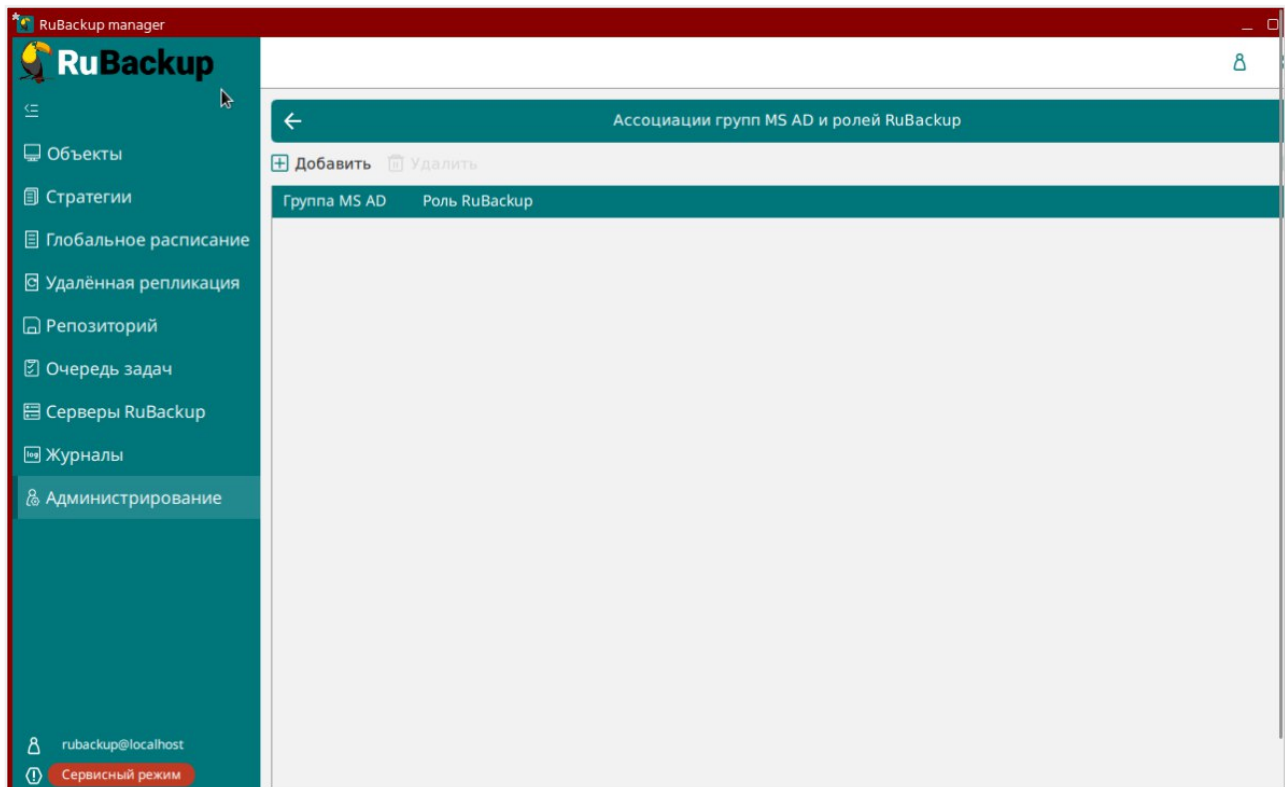


Рисунок 33. Ассоциация групп MS AD и ролей RuBackup

9. Если Вам не удалось успешно пройти шаги с п. 6, *RBM* отображает сообщение о невозможности подключения к серверу *MS AD*.

Выполните шаги из раздела [Решение проблем](#) для устранения сложностей, а затем повторите шаги раздела [Первичная настройка CPK для работы с MS AD](#), начиная с п. 4.

10. CPK сохраняет указанную конфигурационную информацию в БД *RuBackup*. Пароль от пользователя `Bind User` сохраняется в БД *RuBackup* в зашифрованном средствами *PostgreSQL* виде.
11. Находясь в подразделе **Ассоциация групп MS AD и ролей RuBackup**, добавьте ассоциации групп *MS AD* с ролями CPK ([Рисунок 34](#)):

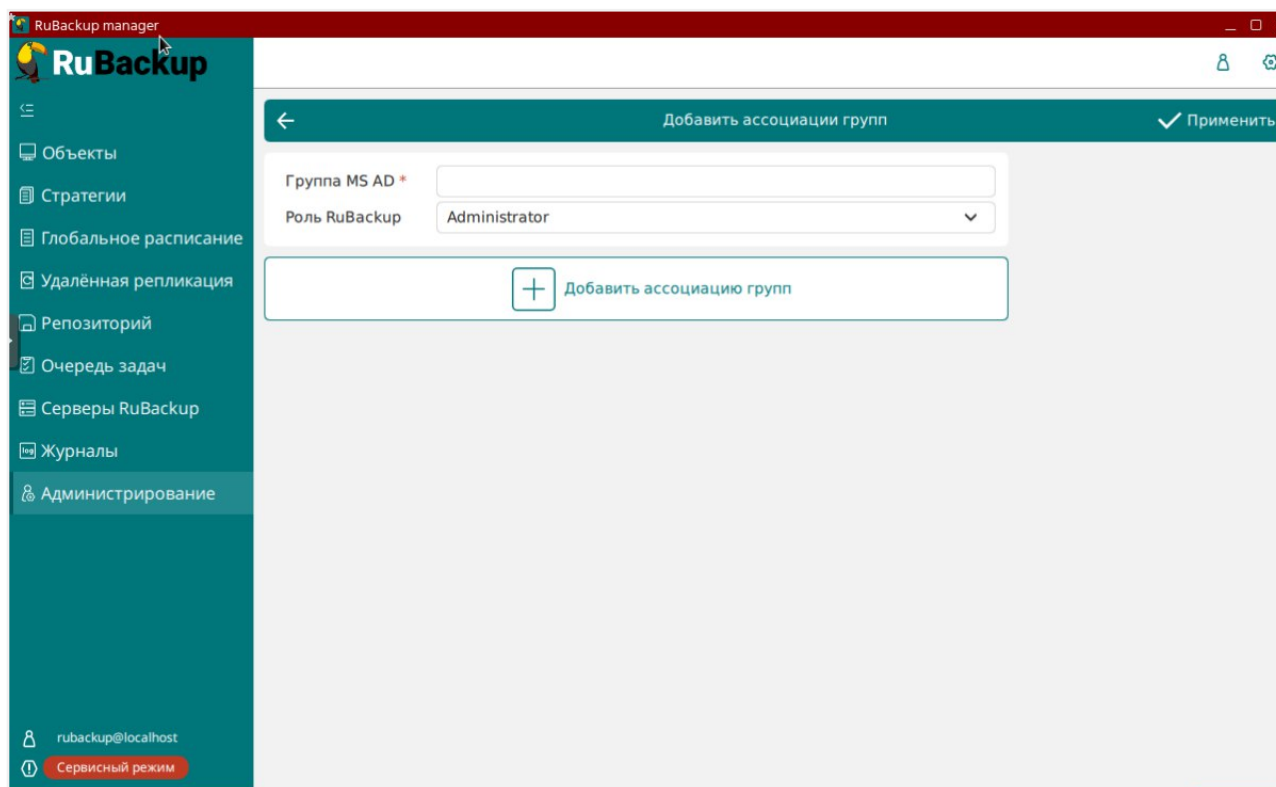


Рисунок 34. Добавление ассоциации групп

Одну роль доступа *RuBackup* вы можете связать с одной или несколькими группами *MS AD*. Связать одну группу *MS AD* с несколькими ролями *СРК* нельзя: учетная запись *MS AD* не может принадлежать нескольким ролям *RuBackup*.



Информация о пользователях, входящих в группу *MS AD*, есть только у администратора *MS AD* и не отображается в *СРК RuBackup*.

12. Сохраните информацию в *RBM*, нажав на кнопку **Применить**.
13. Деактивируйте сервисный режим.

Настройка *СРК* для работы с *MS AD* успешно завершена.

Выбор типа аутентификации по умолчанию

1. Активируйте в *RBM* сервисный режим *СРК* (Рисунок 35).

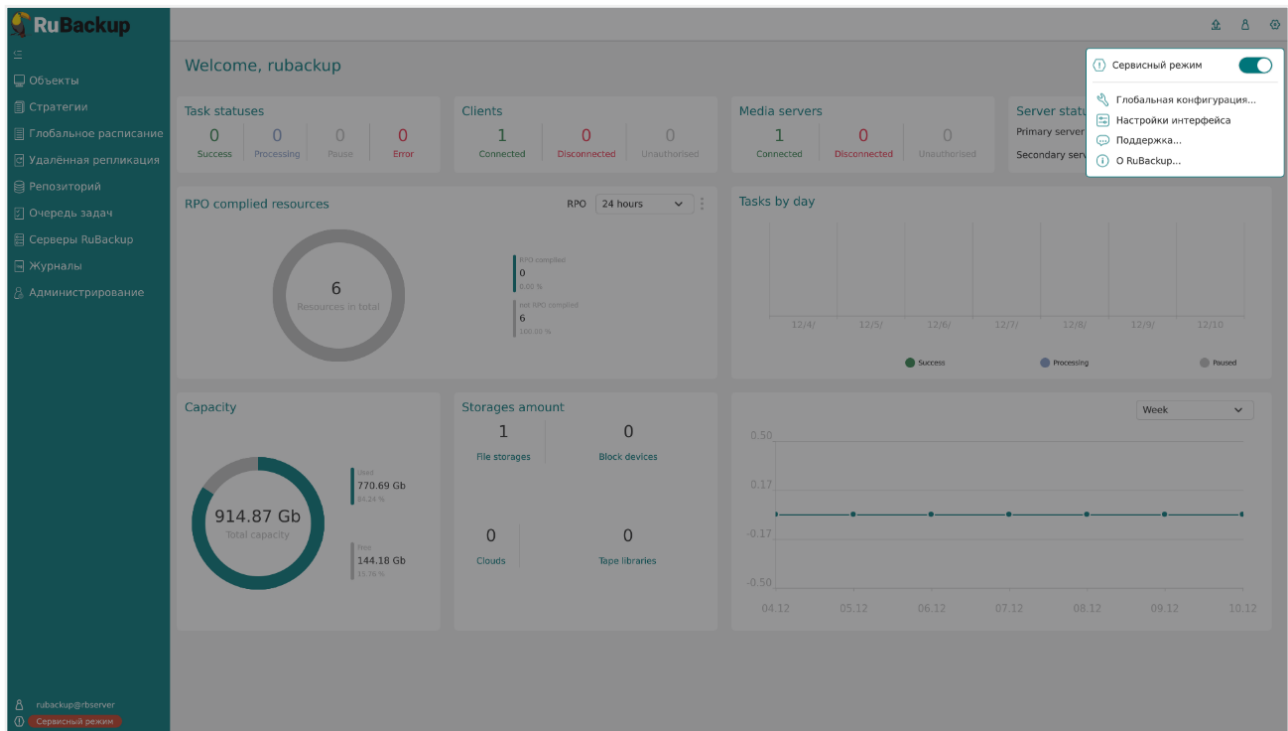


Рисунок 35. Активация Сервисного режима

2. Перейдите во вкладку **Глобальная конфигурация** (Рисунок 36).

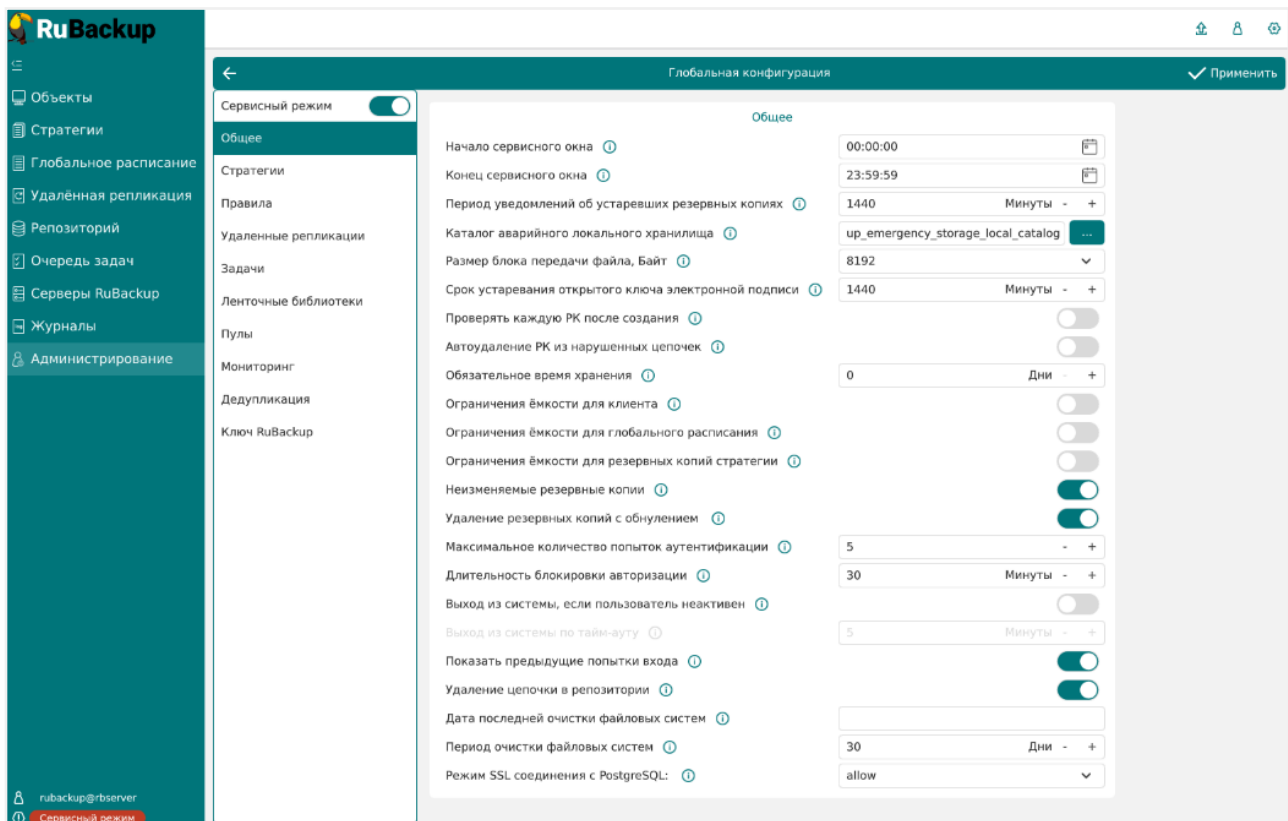


Рисунок 36. Глобальная конфигурация

3. Перейдите в раздел с настройками аутентификации (Рисунок 37).

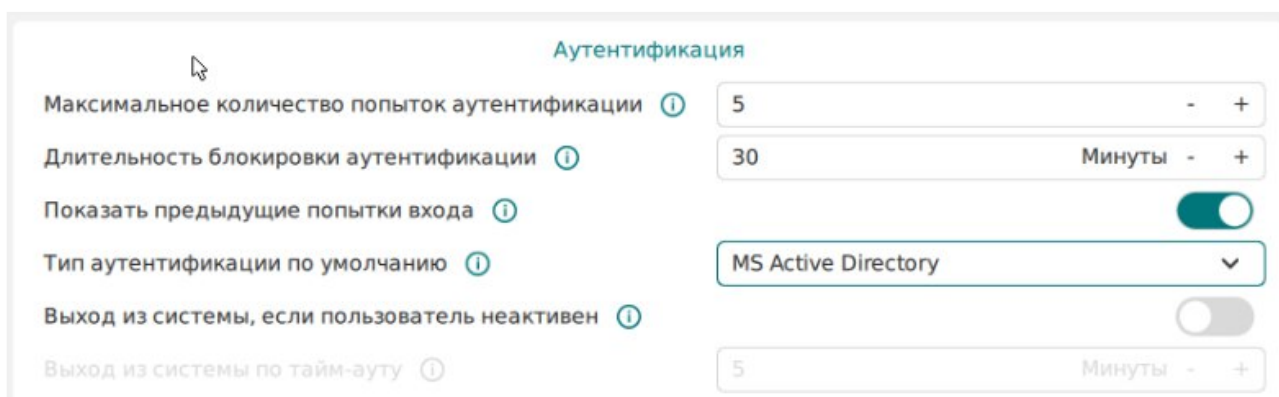


Рисунок 37. Настройки аутентификации

4. Выберите тип аутентификации по умолчанию - *MS Active Directory*.
5. Сохраните настройки в *RBM* нажатием кнопки **Применить**.
6. Деактивируйте сервисный режим (Рисунок 38).

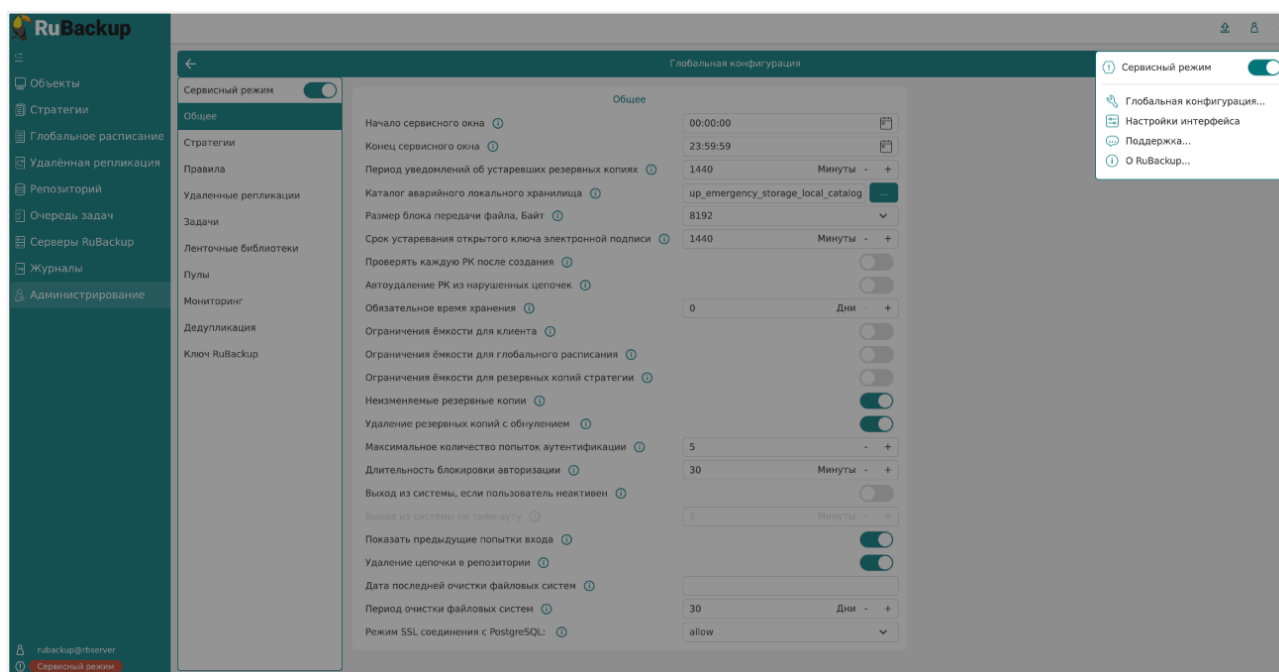
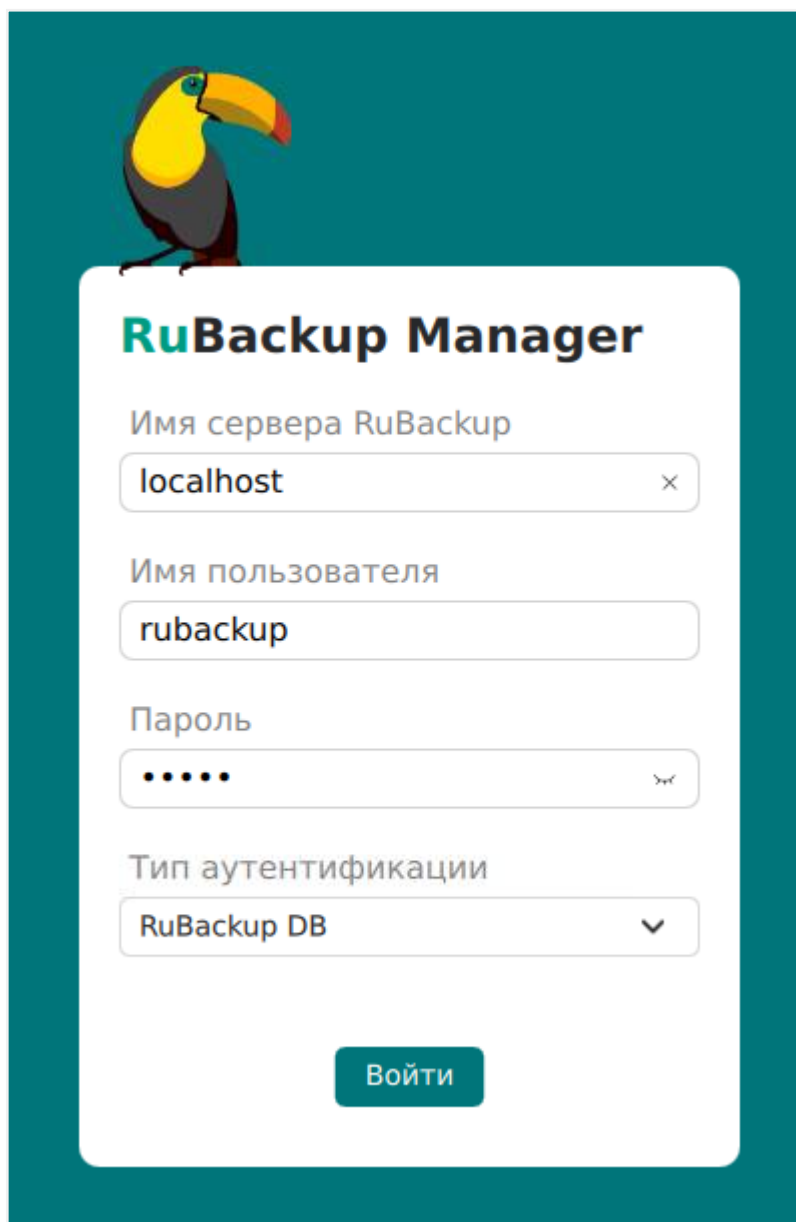


Рисунок 38. Деактивация Сервисного режима

Аутентификация пользователя СРК посредством MS AD

1. Запустите *RBM*.
2. Появится окно для ввода логина и пароля с выпадающим списком, в котором вы можете выбрать тип аутентификации (Рисунок 39). Выберите в выпадающем списке *MS Active Directory*.



RuBackup Manager

Имя сервера RuBackup
localhost

Имя пользователя
rubackup

Пароль
•••••

Тип аутентификации
RuBackup DB

Войти

Рисунок 39. Авторизация в RBM

При этом по умолчанию выбран тип аутентификации, установленный в глобальной конфигурации CPK (см. [Выбор типа аутентификации по умолчанию](#)).

3. Введите в *RBM*:
 - Домен и логин от учетной записи *MS AD* в формате `<домен>\<пароль>`.
 - Пароль от учетной записи *MS AD*.
4. Войдите в CPK нажатием на кнопку **Войти**.
5. Если аутентификационные данные введены неверно, *RBM* выводит сообщение об ошибке с текстом: «Неверно введены логин или пароль» ([Рисунок 40](#)):

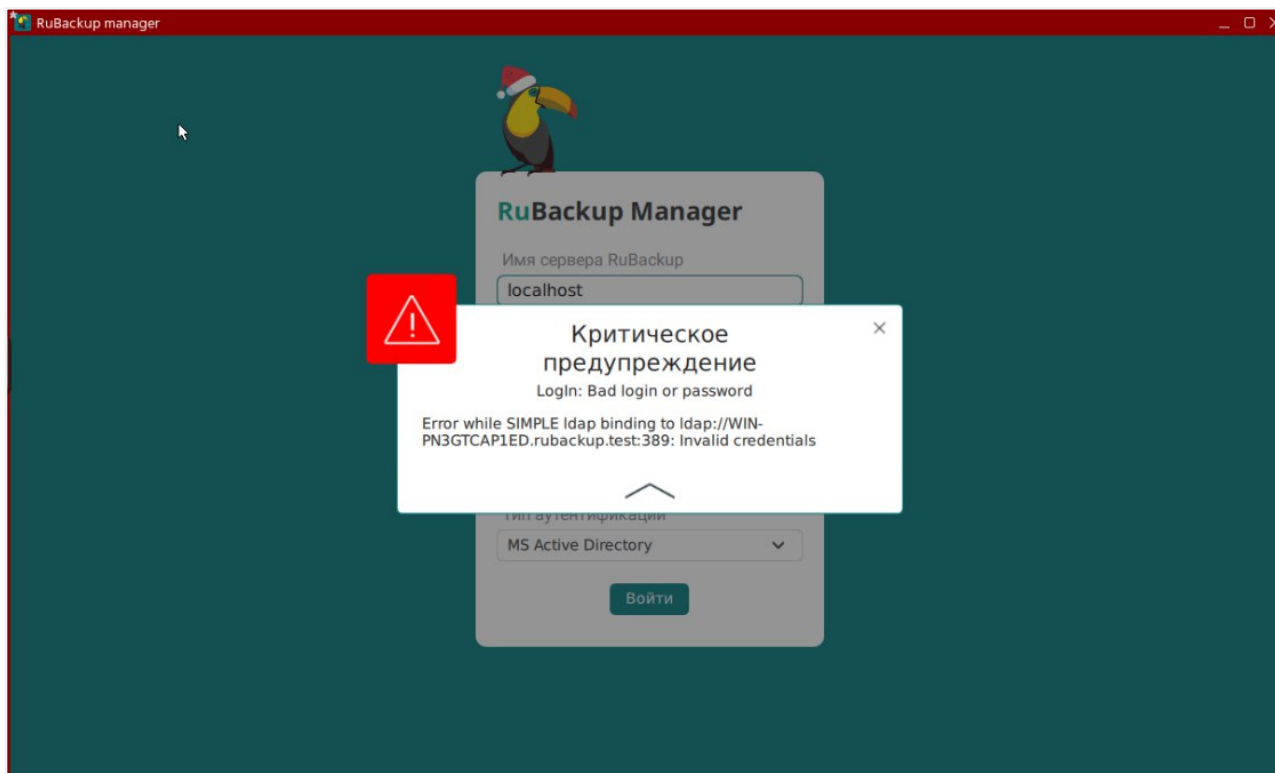


Рисунок 40. Критическое предупреждение. Неверно введены логин или пароль

В этом случае:

- Введите корректные логин и пароль.
 - В случае возникновения проблем обратитесь к Администратору СРК. Администратор СРК выполняет шаги из раздела [Решение проблем](#).
6. Если пользователь СРК находится в одной или нескольких группах *MS AD*, которым соответствует одна роль СРК, то он видит главное меню *RBM*.

Если пользователь не находится ни в одной группе, соответствующей роли СРК, *RBM* выводит сообщение об ошибке: «Данному пользователю не назначена роль СРК. Обратитесь к Администратору СРК» ([Рисунок 41](#)).

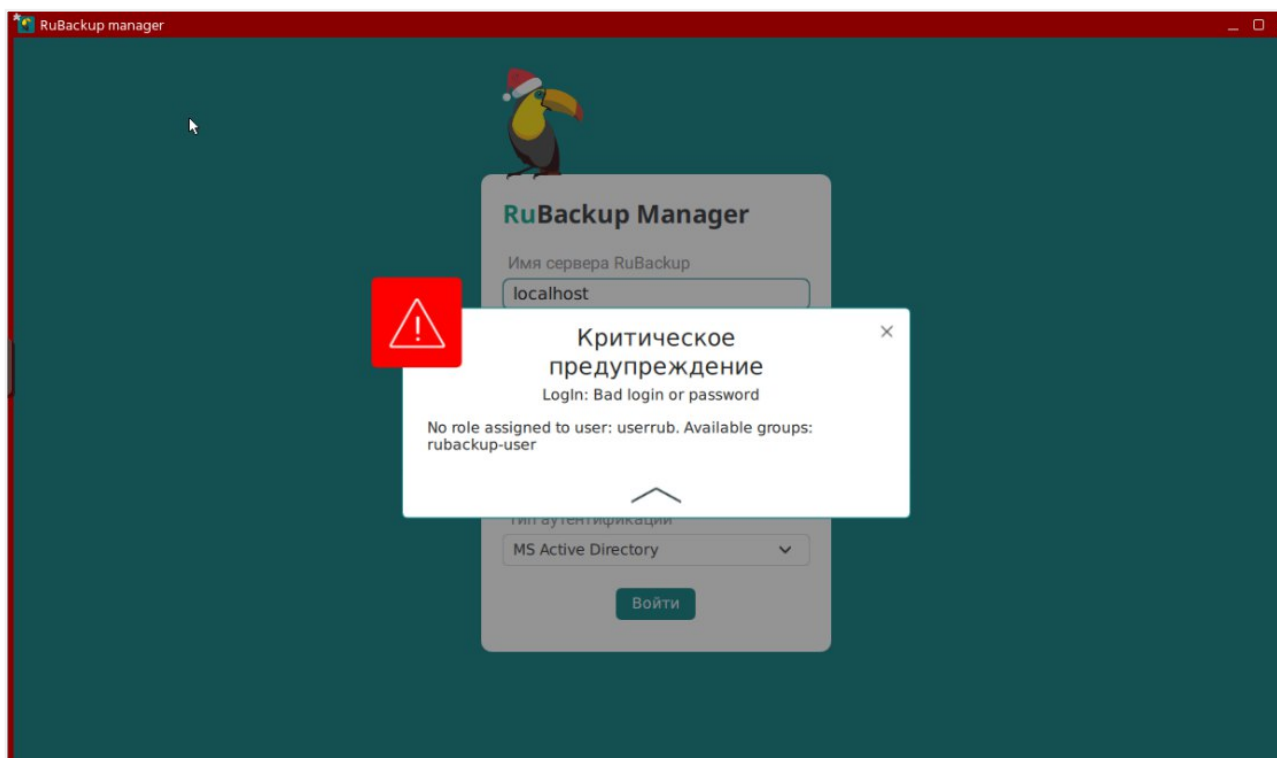


Рисунок 41. Критическое предупреждение. Данному пользователю не назначена роль СРК

- Обратитесь к администратору *MS AD* для добавления данного пользователя средствами *MS AD* в необходимую группу *MS AD*, соответствующую его роли доступа в СРК.
- Выполните шаги из данного раздела с начала.

Аудит аутентификации пользователей

СРК *RuBackup* предоставляет возможность просмотра операций аутентификации пользователей. Для этого:

1. Перейдите в пункт меню **Журналы**, выберите **Журнал операций аутентификации** (Рисунок 42).

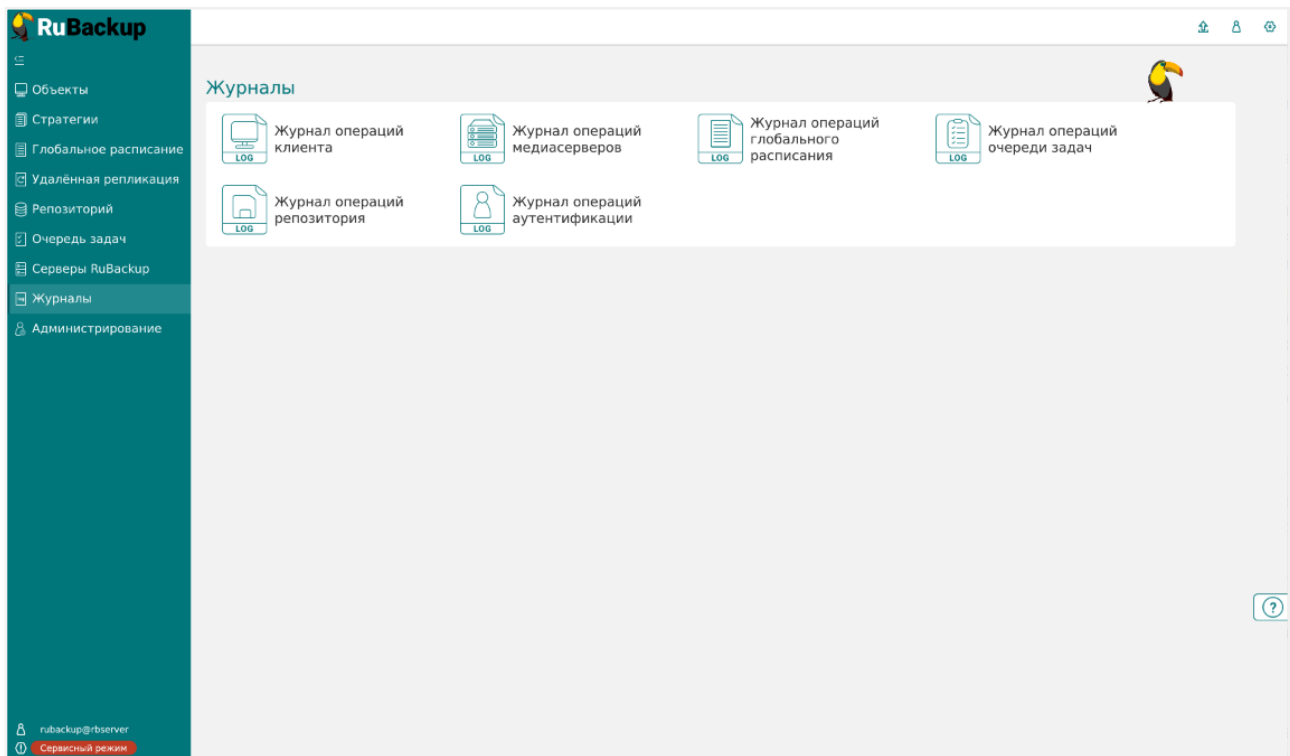


Рисунок 42. Журналы

2. В данном разделе вы можете проанализировать успешные и неудачные попытки аутентификации, а также их количество (Рисунок 43).

Строка	Имя пользователя	Действие	Успешно	Удалённый IP	Дата/Время
94	rubackup	Connected	true	172.18.0.1	2023.12.11 11:
93	rubackup	Connected	true	172.18.0.1	2023.12.10 22:
92	rubackup	Disconnect	true	172.18.0.1	2023.12.10 22:
91	rubackup	Connected	true	172.18.0.1	2023.12.10 22:
90	rubackup	Connected	true	172.18.0.1	2023.12.09 11:
89	rubackup	Connected	true	172.18.0.1	2023.12.08 15:
88	rubackup	Connected	false	172.18.0.1	2023.12.08 14:
87	rubackup	Connected	false	172.18.0.1	2023.12.08 14:
86	rubackup	Disconnect	true	172.18.0.1	2023.12.08 14:
85	rubackup	Connected	true	172.18.0.1	2023.12.08 14:
84	rubackup	Disconnect	true	172.18.0.1	2023.12.08 14:
83	rubackup	Connected	true	172.18.0.1	2023.12.08 13:
82	rubackup	Connected	true	172.18.0.1	2023.12.06 21:
81	rubackup	Connected	true	172.18.0.1	2023.12.06 09:
80	rubackup	Connected	true	172.18.0.1	2023.12.05 17:
79	rubackup	Connected	true	172.18.0.1	2023.12.05 15:
78	rubackup	Connected	true	172.18.0.1	2023.12.05 14:
77	rubackup	Disconnect	true	172.18.0.1	2023.12.05 09:
76	rubackup	Connected	true	172.18.0.1	2023.12.05 09:
75	rubackup	Connected	true	172.18.0.1	2023.12.04 18:
74	rubackup	Connected	true	172.18.0.1	2023.12.04 17:
73	rubackup	Connected	true	172.18.0.1	2023.12.01 17:
72	rubackup	Connected	false	172.18.0.1	2023.12.01 17:
71	rubackup	Connected	true	172.18.0.1	2023.12.01 10:
70	rubackup	Connected	true	172.18.0.1	2023.12.01 10:
69	rubackup	Connected	true	172.18.0.1	2023.11.29 15:
68	rubackup	Connected	true	172.18.0.1	2023.11.27 17:
67	rubackup	Connected	true	172.18.0.1	2023.11.15 10:

Рисунок 43. Журнал операций аутентификации

Решение проблем

1. Подключитесь к хосту сервера *RuBackup*, перейдите в директорию

`/opt/rubackup/log/`, откройте файл `RuBackup.log`, проверьте журнал на наличие ошибок, касающихся взаимодействия CPK с сервером *MS AD*.

2. Проанализируйте ошибки в файле `RuBackup.log`:

- Если найденная ошибка заключается в отсутствии связи с сервером *MS AD*, то проверьте корректность данных для подключения к серверу *MS AD*. Проверьте сетевую доступность сервера *MS AD* с хоста, где в данный момент запущен основной сервер CPK, с помощью сторонней утилиты `ldapsearch`:

Данные для подключения можно посмотреть в [свойствах пользователя MS AD](#)

```
ldapsearch -x -h 10.177.32.128 -D 'bind_user@RUBACKUP' -b
'dc=rubackup,dc=msad' -w 'As!q2w3e4r'
```

где:

- `-x` - использование простой аутентификации вместо SASL;
- `-h` - адрес контроллера домена;
- `-D` - параметры для авторизации:
 - `bind_user` - логин
 - `RUBACKUP` - домен
- `-b` - база поиска;
- `-w` - пароль;

В случае правильных данных для подключения утилита `ldapsearch` выведет соответствующую информацию.

Если утилита выводит сообщение об ошибке, то проблема в настройке *MS AD*. Необходимо уточнить параметры для авторизации и повторить проверку.

- Если найденная ошибка связана с неверными логином или паролем, проверьте корректность учетных данных для пользователя *MS AD Bind User* в настройках. Если данные учетной записи корректны, то, используя их, подключитесь к серверу *MS AD* с использованием сторонних инструментов.
- Если вы нашли несоответствие в правах, проверьте принадлежность пользователя CPK к группам *MS AD*, используемым для аутентификации в CPK *RuBackup*.
- Если найденная ошибка связана с внутренней ошибкой CPK, обратитесь в службу технической поддержки продукта CPK, предоставив информацию о выполненных шагах и журнал логов.

3. Проверьте доступность сервера *MS AD*, валидность наименований групп доступа и учетных записей, устраните проблемы.

В случае отсутствия явных ошибок на стороне сервера *MS AD*, откройте запрос в личном кабинете ГК Астра.

Ограничения

- Аутентификация с использованием *MS AD* не распространяется на клиенты РК. Аутентификация клиентов РК остается без изменений и осуществляется посредством **HWID** (см. [Администрирование](#)).
- Опцию аутентификации посредством *PostgreSQL* нельзя отключить, т.к. в случае утери доменного контроллера *MS AD* вы должны иметь возможность аутентифицироваться в СРК для изменения настроек аутентификации, а также для решения других внештатных ситуаций.
- Аутентификация с использованием *MS AD* не распространяется на утилиты командной строки ([Утилиты командной строки](#)).

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`.

Размещение сертификатов и ключей

Файлы частных 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.

Использование цепочки сертификатов

Иногда клиентский или серверный сертификат подписывается не корневым клиентским или серверным сертификатом, а промежуточным сертификатом, кото-

рый, 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.

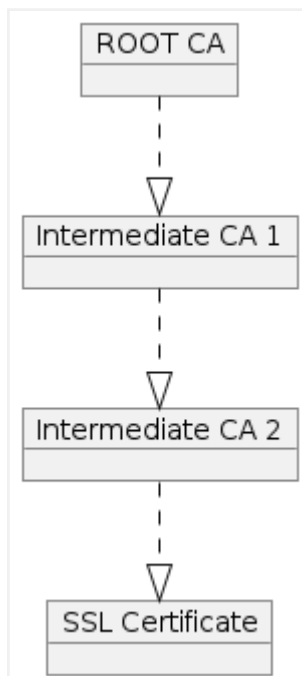


Рисунок 44. Цепочка сертификатов

Серверная часть

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/rubackup/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
```

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
```

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**.

Клиентская часть

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/rubakup/keys/client/clientKey.key 2048
```

5. Создайте запрос на подпись:

```
openssl req -new -key /opt/rubakup/keys/client/clientKey.key -out  
/opt/rubakup/keys/client/clientCert.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 клиентский сертификат и подписать его клиентским корневым сертификатом. В представленном примере сертификат действует 20000 дней:

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

Подготовка сертификатов для клиента

Чтобы подготовить сертификаты для клиента, 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 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
```

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: **OK**.

Менеджер администратора RuBackup

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
```

Проверка созданных ключей и сертификатов

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

```
.cert /opt/rubackup/keys/rbm/rbmCert.cert
```

Вывод команды должен содержать: **OK**.

Ленточные библиотеки

Система резервного копирования RuBackup позволяет работать с ленточными библиотеками. Ленточная библиотека должна быть подключена к хосту, на котором функционирует сервер RuBackup (основной, резервный или медиасервер).

Ленточные картриджи должны относиться к пулу типа «Tape library, LTFS» либо «Tape library, Native». По умолчанию в конфигурации RuBackup создаётся пул типа с названием «TL pool», ассоциированный с основным сервером RuBackup. Картриджи могут находиться в ленточной библиотеке или быть выгружены из неё.

Если картридж выгружен и создана задача, которой необходим доступ к этому картриджу (находящемуся вне ленточной библиотеки), эта задача перейдёт в статус «Suspended» до того, пока необходимый картридж не будет загружен в один из слотов ленточной библиотеки.

Для хранения резервных копий на ленточных картриджах может быть создана файловая система LTFS или использоваться нативное хранение. LTFS позволяет получить доступ к резервным копиям вне системы резервного копирования RuBackup. Нативное хранение позволяет сохранять резервную копию объемом больше, чем объем одного картриджа.

Подготовка к работе с ленточной библиотекой

Установка дополнительного ПО

Для корректной работы с ленточной библиотекой установите драйвер st. Для этого введите команду:

```
uname -r
```

Команда выведет версию ядра (например, 5.15.0-91-generic).

Установите дополнительные модули для вашей версии ядра:

```
sudo apt install linux-modules-extra-5.15.0-91-generic
```

Подгрузите модуль st:

```
sudo modprobe st
```

Также необходимо установить пакеты:

- `mtx` — управляет устройствами смены носителей SCSI с одним или несколькими приводами, такими как устройства смены лент, автозагрузчики, ленточные библиотеки. Пакет должен находиться в `/usr/sbin/mtx`, если он находится в другом месте, то необходимо создать символическую ссылку на исполняемый путь.
- `open-iscsi` — нужен для работы с устройствами подключёнными по iSCSI.
- `lsscsi` — выводит список SCSI-устройств (или хостов), выводит список NVMe-устройств.
- `sg3-utils` — содержит утилиту `sg_reset`, которая отправляет сброс SCSI-устройства, целевого объекта, шины или хоста; или проверяет состояние сброса.

Проверка наличия sg-драйвера

Для проверки наличия sg-драйвера выполните команду:

```
lsscsi -g
```

Команда должна показать подключённые устройства, в их числе привод (приводы) ленточной библиотеки и робота ленточной библиотеки, например:

```
[root@rubackup-media-vtl ltfs]# lsscsi -g
[1:0:0:0]    cd/dvd  QEMU    QEMU DVD-ROM    2.5+  /dev/sr0  /dev/sg0
[2:0:0:0]    disk    QEMU    QEMU HARDDISK   2.5+  /dev/sda  /dev/sg1
[3:0:0:0]    mediumx IBM     3573-TL        D.00  /dev/sch0 /dev/sg2
[4:0:0:0]    tape    IBM     ULT3580-TD9     HB91  /dev/st0  /dev/sg3
[5:0:0:0]    tape    IBM     ULT3580-TD9     HB91  /dev/st1  /dev/sg4
[root@rubackup-media-vtl ltfs]#
```

Рисунок 45. Пример вывода `lsscsi -g`

Если в крайнем правом столбце отображаются sg-пути, то это значит, что sg-драйвер уже установлен и запущен.

Если в крайнем правом столбце отсутствуют sg-пути, то установите sg-драйвер для вашей операционной системы (см. [Установка sg-драйвера](#)).

Установка sg-драйвера

Astra Linux 1.6 и 1.7

```
sudo apt install libsgutils2-dev
```

Ubuntu 18.04 и 20.04

```
sudo apt install libsgutils2-dev
```

CentOS 7 и 8

```
sudo yum install sg3_utils
```



Для CentOS 7 необходимо установить пакет mt-st:

```
sudo yum install mt-st
```

Alt Linux 10

Установите пакет mt-st:

```
sudo yum install mt-st
```

Для установки sg-драйвера выполните команду:

```
sudo apt-get install udev-rules-sgutils
```

Для корректной работы sg-драйвера выполните команды:

```
sudo sg_scan  
sudo modprobe sg  
sudo find /dev/ -name "sg"
```

Убедитесь, что sg-драйвер установлен и запущен (см. [Проверка наличия sg-драйвера](#)):

```
sudo lsscsi -g
```

РЕД ОС 7.3

```
sudo dnf install sg3_utils-libs
```

Настройки автоматического запуска sg-драйвера

Создайте скрипт запуска sg-драйвера:

```
sudo touch /etc/sg_driver_startup.sh
```

Содержание скрипта `sg_driver_startup.sh`

```
sudo !/bin/sh -e
echo 'init sg-driver'
sg_scan
modprobe sg
echo 'done'
exit 0
```

Сделайте скрипт исполняемым:

```
sudo chmod a+x /etc/sg_driver_startup.sh
```

Создайте конфигурационный файл для службы `systemd`:

```
sudo touch /lib/systemd/system/sg_driver_startup.service
```

Содержание скрипта `sg_driver_startup.service`

```
[Unit]
Description=sg driver startup script
[Service]
ExecStart=/etc/sg_driver_startup.sh
[Install]
WantedBy=multi-user.target
```

Запустите сервис:

```
sudo systemctl enable sg_driver_startup.service --now
```

Добавьте зависимость от сервиса `sg_driver_startup.service` в `rubackup_server.service`:

```
sudo systemctl edit --full rubackup_server.service
```

[Unit]**Description**=RuBackup server**Requires**=network.target**After**=network.target postgresql.service sg_driver_startup.service

После перезагрузки проверьте статус сервиса `rubackup_server.service`:

```
sudo systemctl status rubackup_server.service
```

Сборка LTFS



Информация в данном пункте необходима для использования пулов *Tape Library*, *LTFS*.

Страница проекта: <https://github.com/LinearTapeFileSystem/ltfs>

Зависимости, которые должны быть установлены перед сборкой:
<https://github.com/LinearTapeFileSystem/ltfs/wiki/Build-Environments>

Общая сборочная инструкция: <https://github.com/LinearTapeFileSystem/ltfs#build-and-install-on-linux>

Поддерживаемые устройства:
<https://github.com/LinearTapeFileSystem/ltfs#supported-tape-drives>

1. Установите необходимые пакеты для сборки, исходя из операционной системы, на которую устанавливается LTFS:

- a. RPM-based OS

Чтобы собрать LTFS, выполните:

```
sudo yum install perl
sudo yum install make
sudo yum install gcc
sudo yum install git
sudo yum install pkg-config
sudo yum install libxml2-dev
sudo yum install automake
sudo yum install autoconf
sudo yum install libtool
sudo yum install uuid
sudo yum install uuid-devel
sudo yum install libuuid-devel
```

```
sudo yum install icu
sudo yum install fuse
sudo yum install fuse-devel
sudo yum install libicu-devel
sudo yum install net-snmp
sudo yum install net-snmp-devel
```

b. DEB-based OS

Чтобы собрать LTFS, выполните следующие команды (для всех поддерживаемых систем, кроме Astra Linux 1.8 и Debian 12):

```
sudo apt install make
sudo apt install git
sudo apt install pkg-config
sudo apt install libxml2-dev
sudo apt install automake
sudo apt install autoconf
sudo apt install libtool
sudo apt install uuid
sudo apt install uuid-dev
sudo apt install fuse
sudo apt install libfuse-dev
sudo apt install libsnmp-dev
sudo apt install icu-devtools
sudo apt install libicu-dev
```



Для ОС Astra Linux 1.8 и Debian 12 замените команду:

```
sudo apt install fuse
```

командой:

```
sudo apt install libfuse2
```

Создайте файл `/usr/bin/icu-config` со следующим содержимым:

```
#!/bin/sh
opts=$1
case $opts in '--cppflags')
echo '' ;;
```



```
'--ldflags' )  
echo '-licuuc -licudata -ldl' ;;  
*)  
echo '/usr/lib/x86_64-linux-gnu/icu/pkgdata.inc' ;;  
esac
```

и выполните команду:

```
chmod 755 /usr/bin/icu-config
```

2. Сборка

Для сборки выполните:

```
git clone https://github.com/LinearTapeFileSystem/ltfs.git  
cd ltfs  
sudo ./autogen.sh  
sudo ./configure  
sudo make  
sudo make install  
sudo ldconfig -v
```

3. Проверьте, подключена ли ленточная библиотека к хосту:

```
sudo lsscsi -g
```

Команда должна показать подключённые устройства, в их числе привод (приводы) ленточной библиотеки и робота ленточной библиотеки:

```
[1:0:0:0] tape IBM ULT3580-TD6 D8E4 /dev/st0 /dev/sg5
```

```
[1:0:0:1] mediumx IBM3573-TL C.20 /dev/sch0 /dev/sg6
```

В данном случае у библиотеки есть один ленточный привод (магнитофон) и робот, к которому можно обращаться через `/dev/sg6`.

4. Получите информацию о ленточной библиотеке, обращаясь к роботу ленточной библиотеки:

```
sudo mtx -f /dev/sg6 status
```

Эта команда должна показать информацию о слотах ленточной библиотеки, о

загруженных в них картриджах и о приводах ленточной библиотеки.

Пример вывода `sudo mtx -f /dev/sg6 status`

```
Storage Changer /dev/sg6:1 Drives, 24 Slots ( 1 Import/Export )
Data Transfer Element 0:Empty
Storage Element 1:Full :VolumeTag=INT020L6
Storage Element 2:Full :VolumeTag=INT023L6
Storage Element 3:Full :VolumeTag=INT033L6
Storage Element 4:Full :VolumeTag=INT026L6
Storage Element 5:Full :VolumeTag=INT029L6
Storage Element 6:Full :VolumeTag=INT022L6
Storage Element 7:Full :VolumeTag=INT034L6
Storage Element 8:Full :VolumeTag=INT025L6
Storage Element 9:Full :VolumeTag=INT028L6
Storage Element 10:Full :VolumeTag=INT021L6
Storage Element 11:Full :VolumeTag=INT024L6
Storage Element 12:Full :VolumeTag=INT039L6
Storage Element 13:Full :VolumeTag=INT012L6
Storage Element 14:Full :VolumeTag=INT011L6
Storage Element 15:Empty
Storage Element 16:Full :VolumeTag=INT036L6
Storage Element 17:Full :VolumeTag=INT014L6
Storage Element 18:Full :VolumeTag=INT010L6
Storage Element 19:Empty
Storage Element 20:Full :VolumeTag=INT038L6
Storage Element 21:Full :VolumeTag=INT037L6
Storage Element 22:Empty
Storage Element 23:Full :VolumeTag=CLNU41L1
Storage Element 24
IMPORT/EXPORT:Full :VolumeTag=INT027L6
```

В данном случае библиотека состоит из 24 слотов, один из которых — слот ввода-вывода, через который можно импортировать или экспортировать ленточные картриджи, и один ленточный привод (сейчас пуст). Слоты ленточной библиотеки заполнены картриджами с определенными VolumeTag, один из картриджей — чистящий, три слота в ленточной библиотеке пусты.

5. Загрузите картридж в ленточную библиотеку, создайте на нем файловую систему LTFS и проверьте её работу:

```
sudo mtx -f /dev/sg6 load 1 0
```

В результате выполнения этой команды картридж из слота 1 будет загружен в

единственный магнитофон 0.

6. Создайте файловую систему LTFS на картридже:

```
sudo mklfts -f -d /dev/sg5
```

7. Проверьте файловую систему LTFS:

```
sudo lftscck /dev/sg5
```

8. Создайте точку монтирования:

```
sudo mkdir /lfts
```

9. Монтируйте файловую систему LTFS:

```
sudo lfts -o devname=/dev/sg5 /lfts/
```

10. Убедитесь, что файловая система примонтирована:

```
sudo df -k /lfts
```

```
Filesystem 1K-blocks Used Available Use% Mounted on
lfts:/dev/sg5 2351648768 0 2351648768 0% /lfts
```

11. Отмонтировать файловую систему LTFS:

```
sudo umount /lfts
```

12. Возвратить картридж из магнитофона 0 в слот 1:



```
sudo mtx -f /dev/sg6 unload 1 0
```

Если все действия завершились успешно, то ленточная библиотека готова к работе с сервером RuBackup.

Конфигурация ленточной библиотеки

Для конфигурации ленточной библиотеки необходимо воспользоваться Менеджером администратора RuBackup (RBM).

Перед началом настройки новой или существующей ленточной библиотеки переведите RuBackup в сервисный режим.

Чтобы переключить RuBackup в сервисный режим, нажмите на кнопку  (**Настройки**) в правом верхнем углу ([Рисунок 46](#)) и включите флаг  ([Рисунок 47](#)).

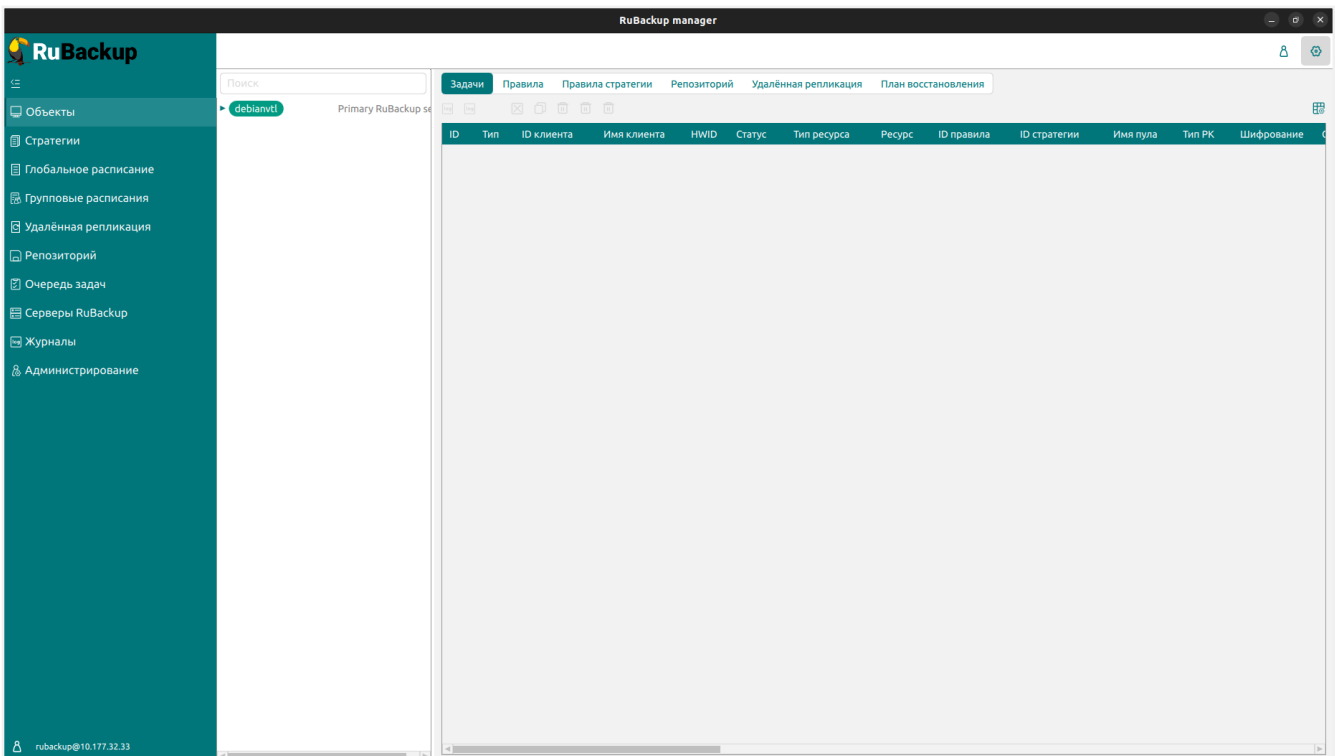


Рисунок 46.

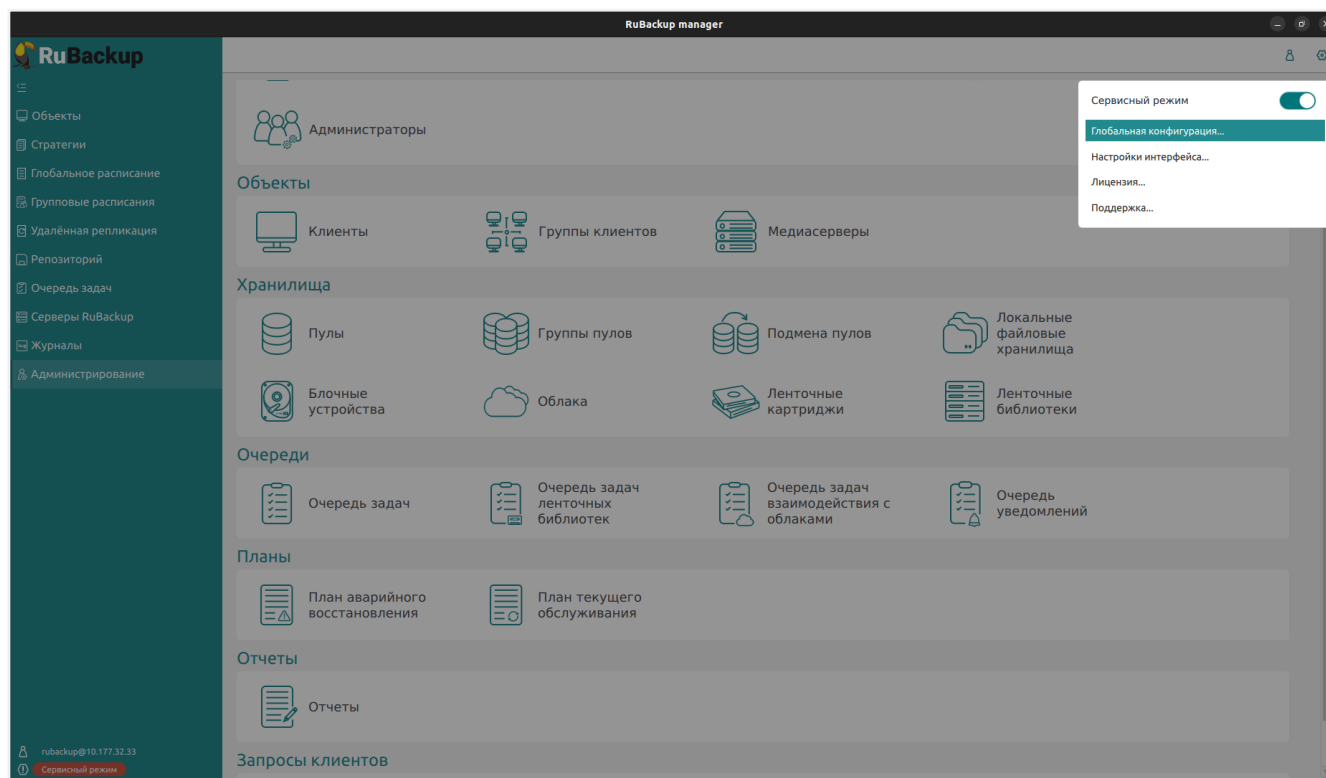


Рисунок 47.

Во время сервисного режима не создаются новые задачи в системе резервного копирования. Если в главной очереди задач остались какие-либо задачи, рекомендуется дождаться окончания их выполнения перед настройкой ленточной библиотеки.

Если ленточная библиотека располагается не на основном сервере RuBackup, предварительно необходимо создать для этого сервера пул типа «Tape library, LTFS» либо «Tape library, Native». Для этого выберите **Администрирование** → **Хранилища** → **Пулы** и нажмите на кнопку **Добавить** (Рисунок 48).

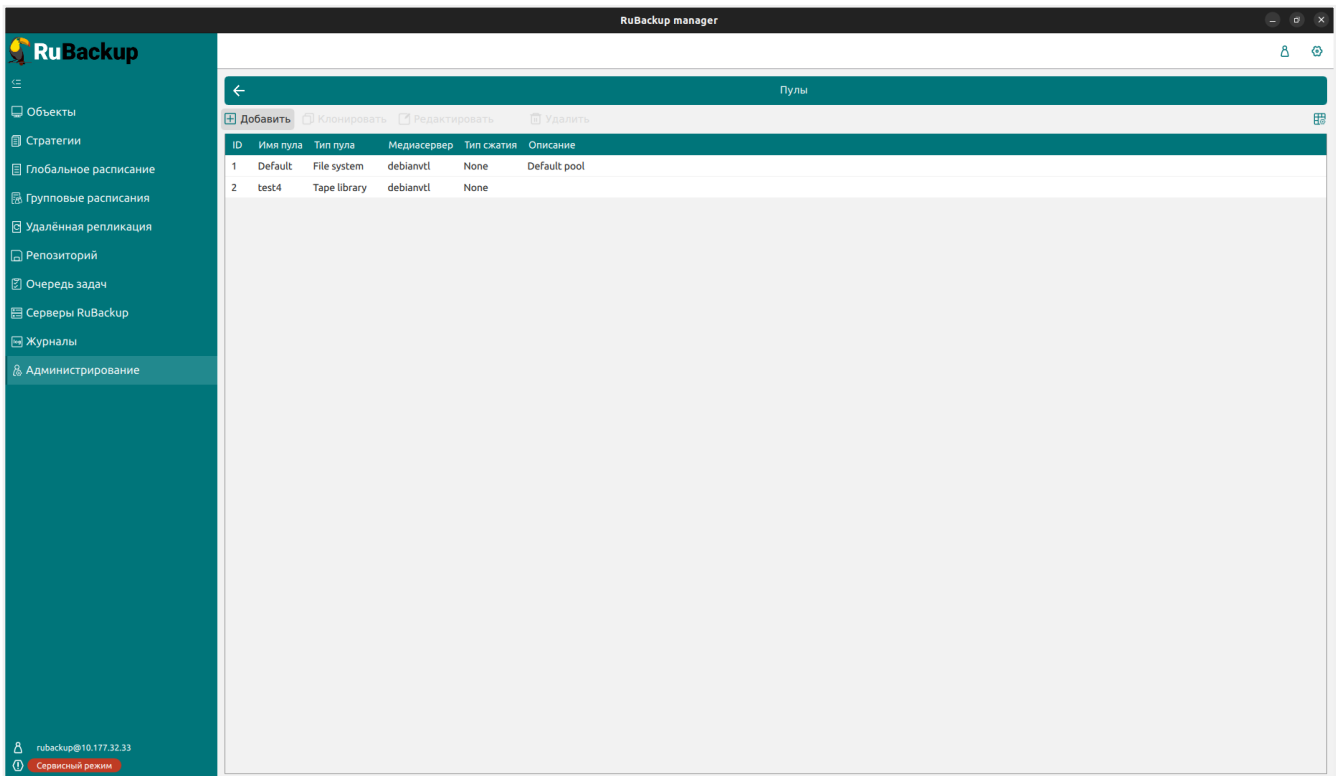


Рисунок 48.

При добавлении нового пула его необходимо привязать к медиасерверу, на котором находится ленточная библиотека. Также для этого пула можно выбрать тип сжатия и ввести описание (Рисунок 49).

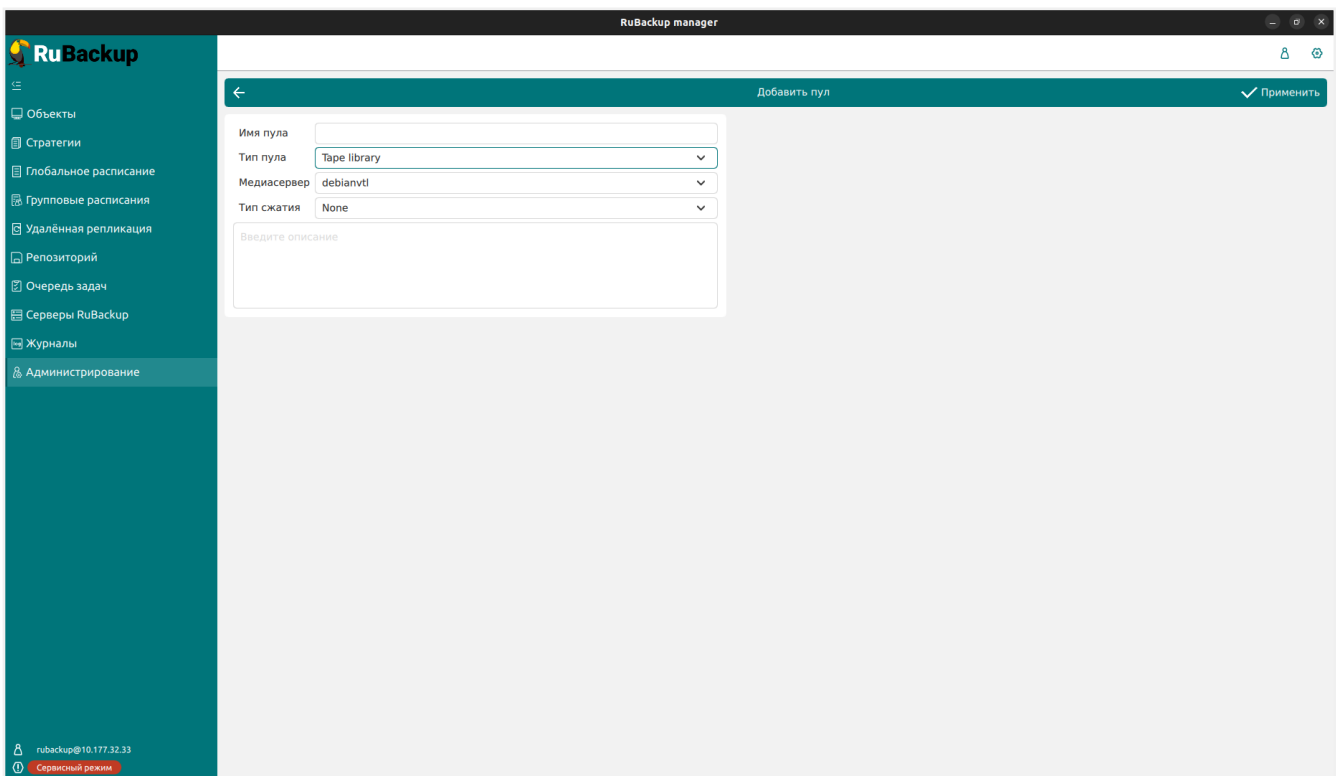


Рисунок 49.

Добавить ленточную библиотеку в конфигурацию RuBackup можно из главного

меню:  **Администрирование** →  **Хранилища** → **Ленточные библиотеки** (Рисунок 50).

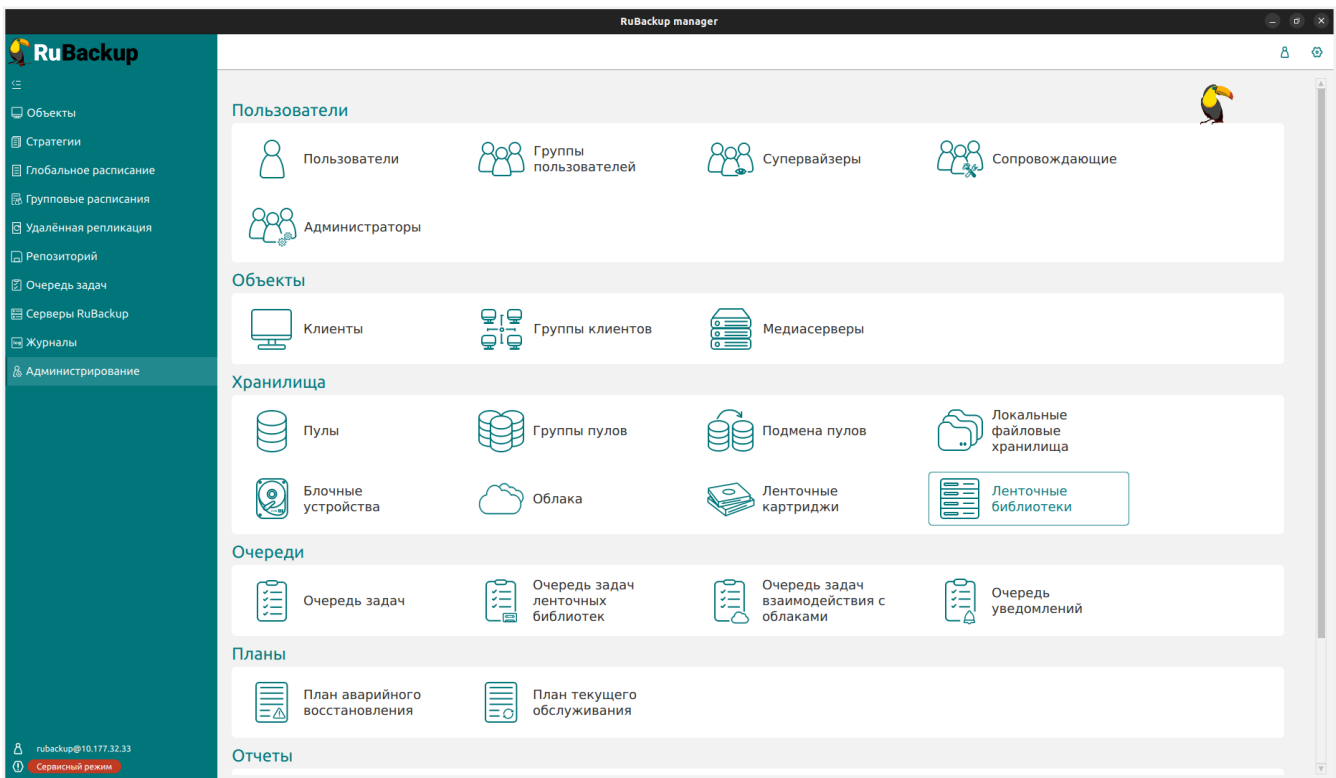


Рисунок 50.

Откроется окно (Рисунок 51).

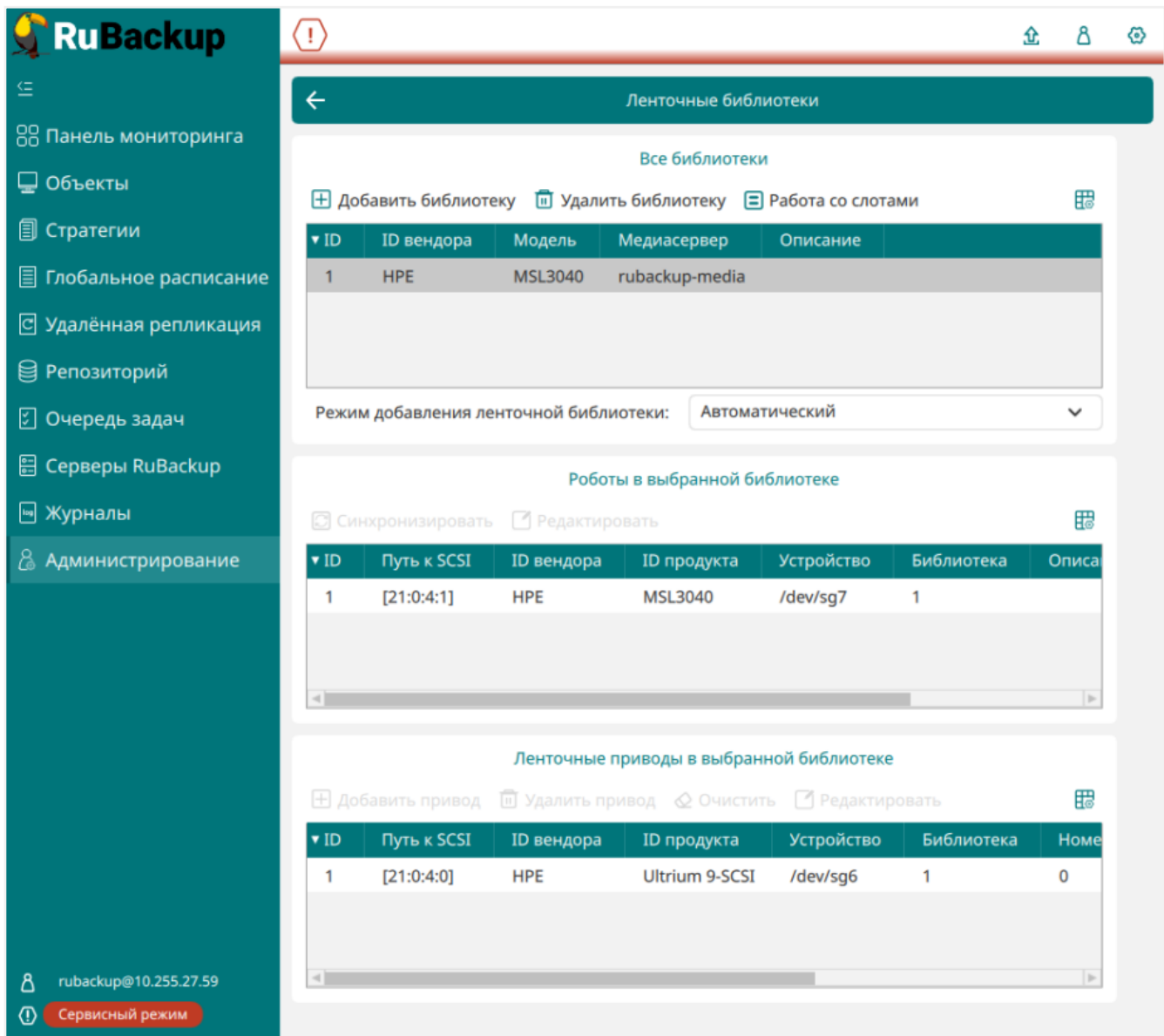


Рисунок 51.

При нажатии кнопки **Добавить библиотеку** в том случае, если на основном сервере резервного копирования нет ленточной библиотеки, появится окно с предупреждением, так как по умолчанию RBM пытается обнаружить ленточную библиотеку на основном сервере RuBackup ([Рисунок 52](#)).

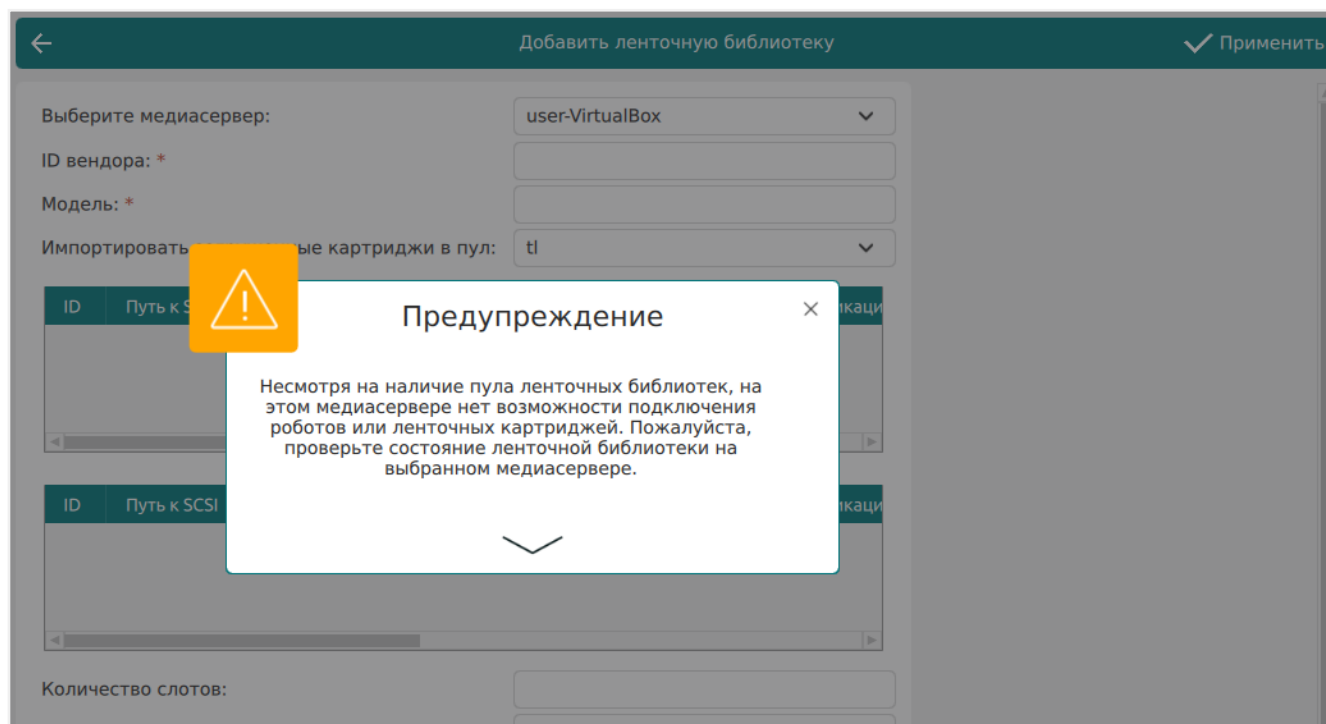


Рисунок 52.

Предупреждение можно закрыть, после этого откроется окно добавления ленточной библиотеки (Рисунок 53).

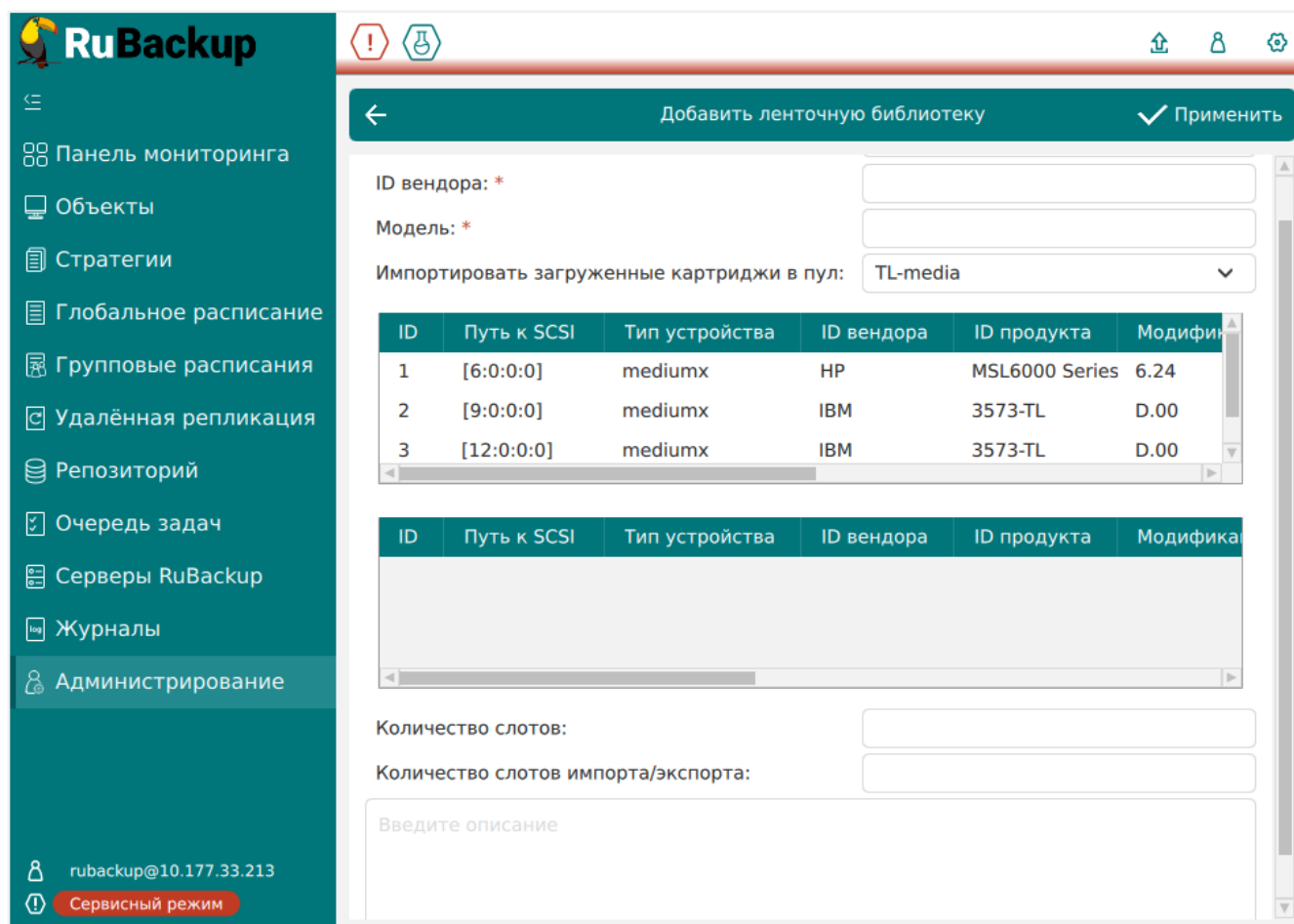


Рисунок 53.

В открывшемся окне нужно выбрать медиасервер, к которому подключена ленточная библиотека, и нужный ленточный пул. Картриджи, которые обнаружатся в ленточной библиотеке, будут добавлены в выбранный в этом окне пул.

После добавления в конфигурацию RuBackup в окне **Ленточные библиотеки** можно выбрать библиотеку и нажать кнопку **Работа со слотами** (Рисунок 54). Откроется окно (Рисунок 55), в котором можно посмотреть все загруженные ленты и слоты библиотеки, а также узнать, находятся ли в слотах картриджи.

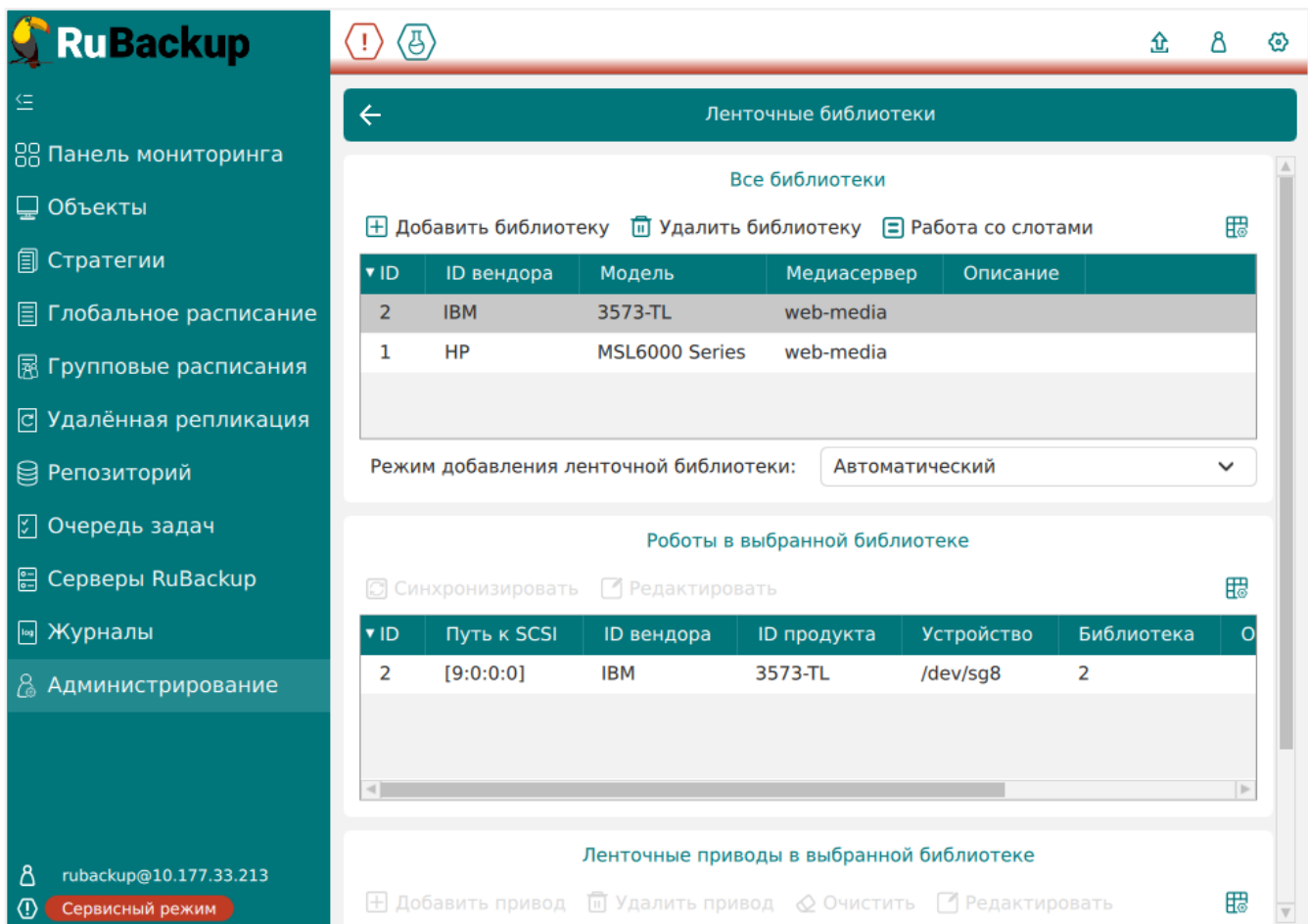


Рисунок 54.

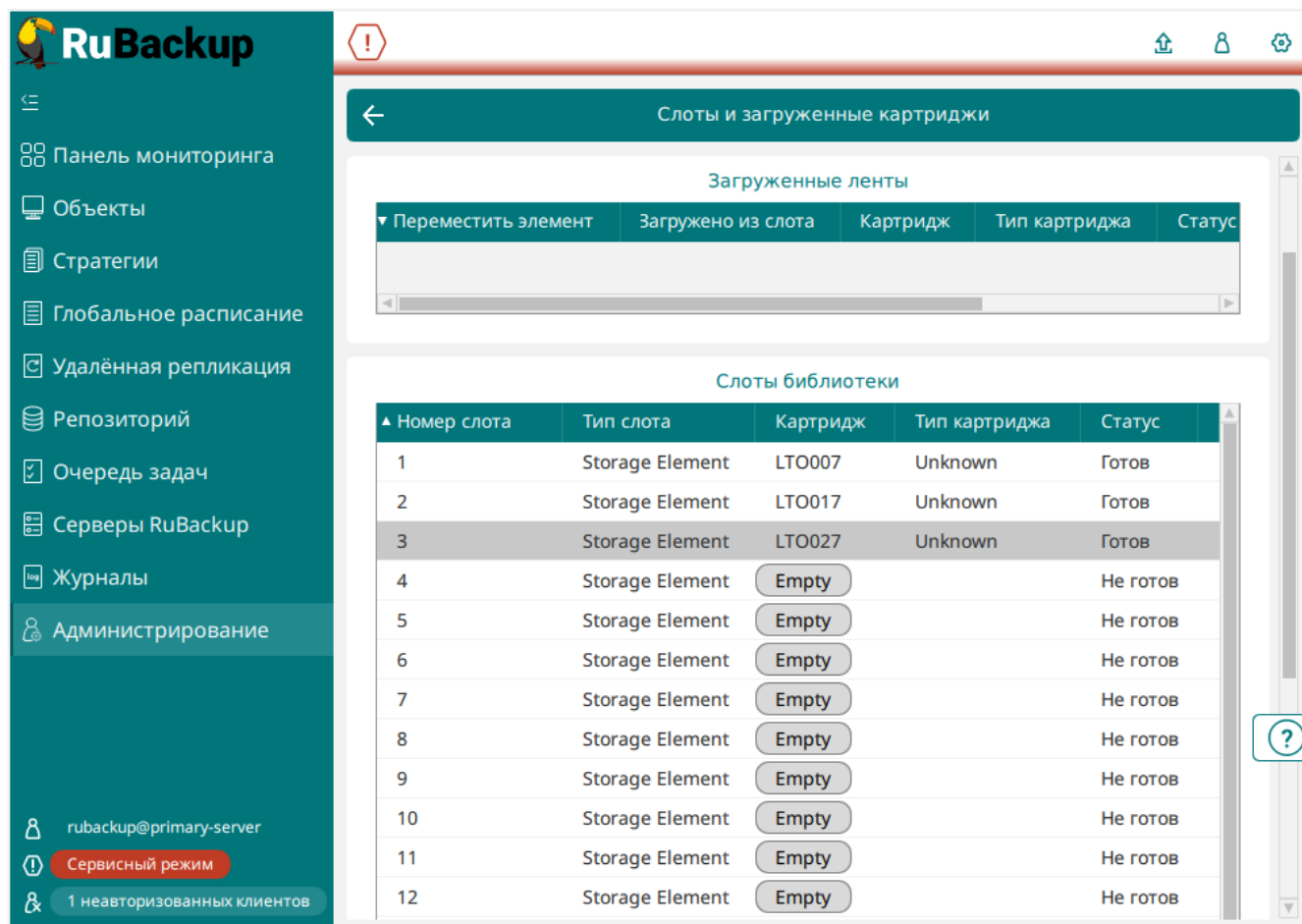


Рисунок 55.

Также в данном окне доступны кнопки **Форматировать** и **Проверка**. Для обеих кнопок доступен множественный выбор картриджей при нажатии CTRL.

Для работы с картриджем он должен быть в статусе *Готов*. Если картридж находится в статусе *Не готов*, его нужно отформатировать. Для этого выберите картридж и нажмите на кнопку **Форматировать**. При этом на картридже будет создана файловая система. Подробнее о статусах картриджей см. раздел [Статусы ленточных библиотек](#).

При нажатии кнопки **Проверка** создается задача на выполнение проверки файловой системы картриджа на наличие ошибок. Статус задачи можно увидеть в очереди задач ленточных библиотек, а логи по задаче — в общем журнале `/opt/rubackup/log/RuBackup.log`.

Для включения дополнительного журналирования при использовании LTFS в файле `RuBackup.log` для операций с ленточной библиотекой, такими как форматирование, проверка, монтирование картриджа при выполнении резервного копирования, восстановления из резервной копии или верификации резервной копии, необходимо указать значение `yes` или `true` для параметра `ltfs-debug-log`. При значении для этого параметра `no` или `false` дополнительное журналирование для операций с ленточной библиотекой выключено. После установки значения для параметра `ltfs-debug-log` на узле медиасервера, к которому подключена ленточ-

ная библиотека, необходимо перезапустить сервис `rubackup_server`.

После завершения настройки необходимо выйти из сервисного режима, в противном случае новые задания не будут выполняться в главной очереди задач ([Рисунок 46](#)).

Работа с ленточной библиотекой

Синхронизация ленточной библиотеки и RuBackup

Выполняйте синхронизацию в случае любых изменений состояния ленточной библиотеки, которые не отслеживаются СРК. Это поможет избежать ошибок в работе RuBackup с ленточной библиотекой.

При необходимости можно изъять или добавить картриджи в ленточную библиотеку вручную. Синхронизация ленточной библиотеки не требует включения сервисного режима.

Для синхронизации ленточной библиотеки и RuBackup нужно в окне ленточных библиотек в таблице «Роботы в выбранной библиотеке» выбрать требующее синхронизации устройство и нажать кнопку «Синхронизировать» ([Рисунок 56](#)). Синхронизация — длительный процесс.

В результате в коллекцию картриджей будут внесены все новые картриджи, загруженные в ленточную библиотеку, а данные о расположении картриджей в базе данных и в ленточной библиотеке будут актуализированы.

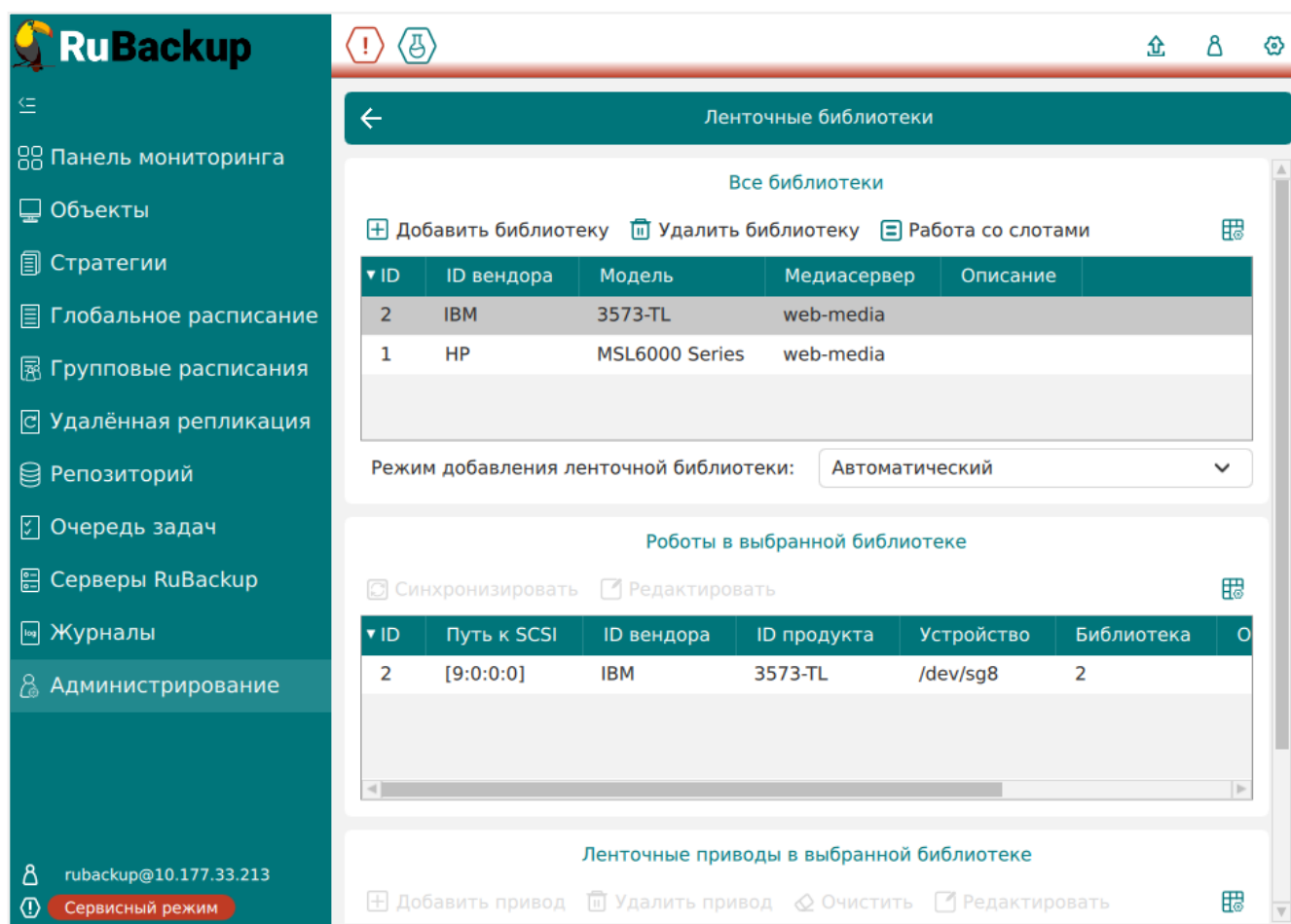


Рисунок 56.

Перемещение ленточного картриджа в другой слот

Чтобы переместить картридж в другой слот, выберите его в списке картриджей и нажмите на кнопку «Переместить» ([Рисунок 57](#)).

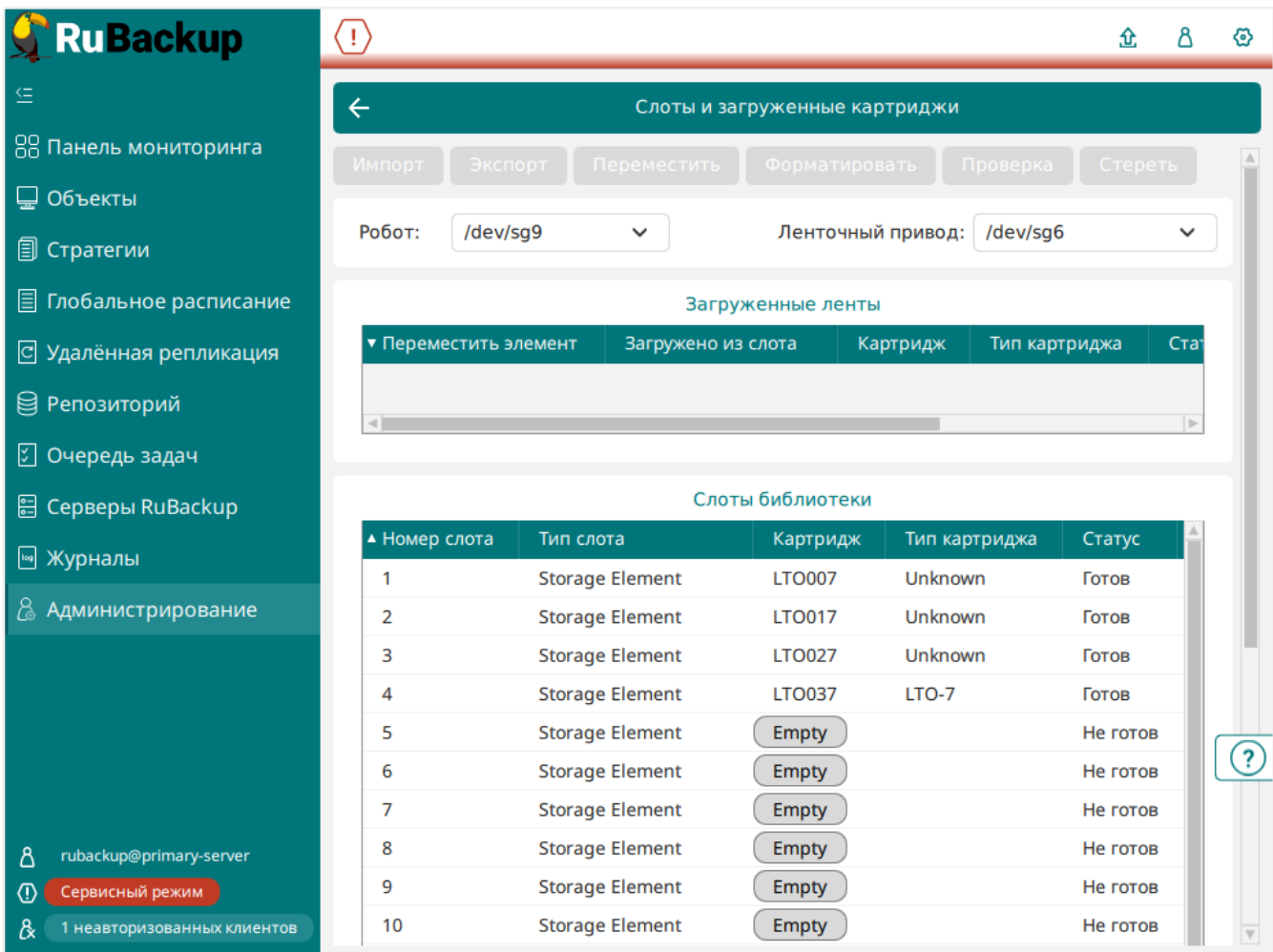


Рисунок 57.

В появившемся окне (Рисунок 58) выберите из выпадающего списка слот, в который следует переместить картридж. Нажмите «ОК».

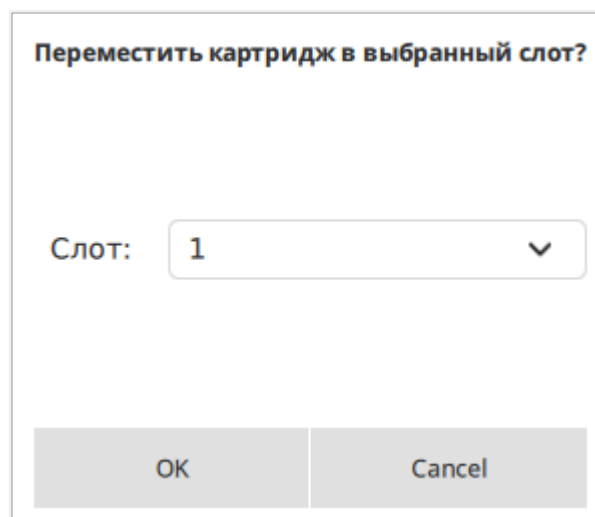


Рисунок 58.

Перемещение ленточного картриджа в другой пул

Чтобы переместить картридж в другой пул, воспользуйтесь консольной утилитой

`rb_tape_cartridges:`

```
rb_tape_cartridges -c ID_картриджа [ -p ID_пула ] [ -d 'Новое описание' ]
```

Импорт и экспорт ленточных картриджей

Используя приёмный слот ленточной библиотеки («Storage Element Export/Import»), можно импортировать и экспортировать ленточные картриджи. Импорт и экспорт ленточных картриджей не требует включения сервисного режима.

Для импорта картриджа приёмный слот должен быть пустым. Необходимо поместить в него картридж и нажать кнопку «Импорт» ([Рисунок 59](#)).

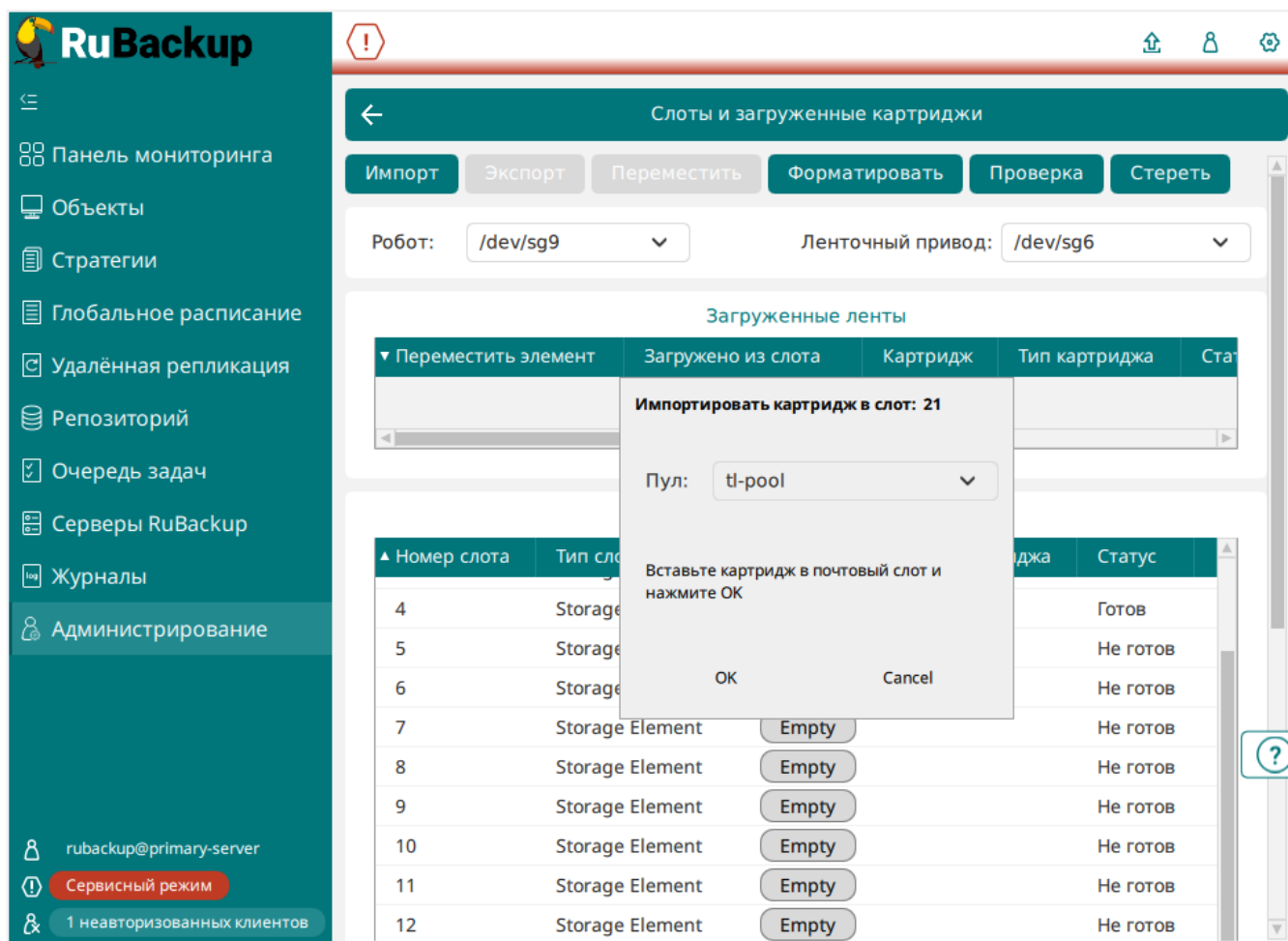


Рисунок 59.

Для экспорта картриджа его нужно переместить в приёмный слот, нажать кнопку «Экспорт» и извлечь из приёмного слота ([Рисунок 60](#)):

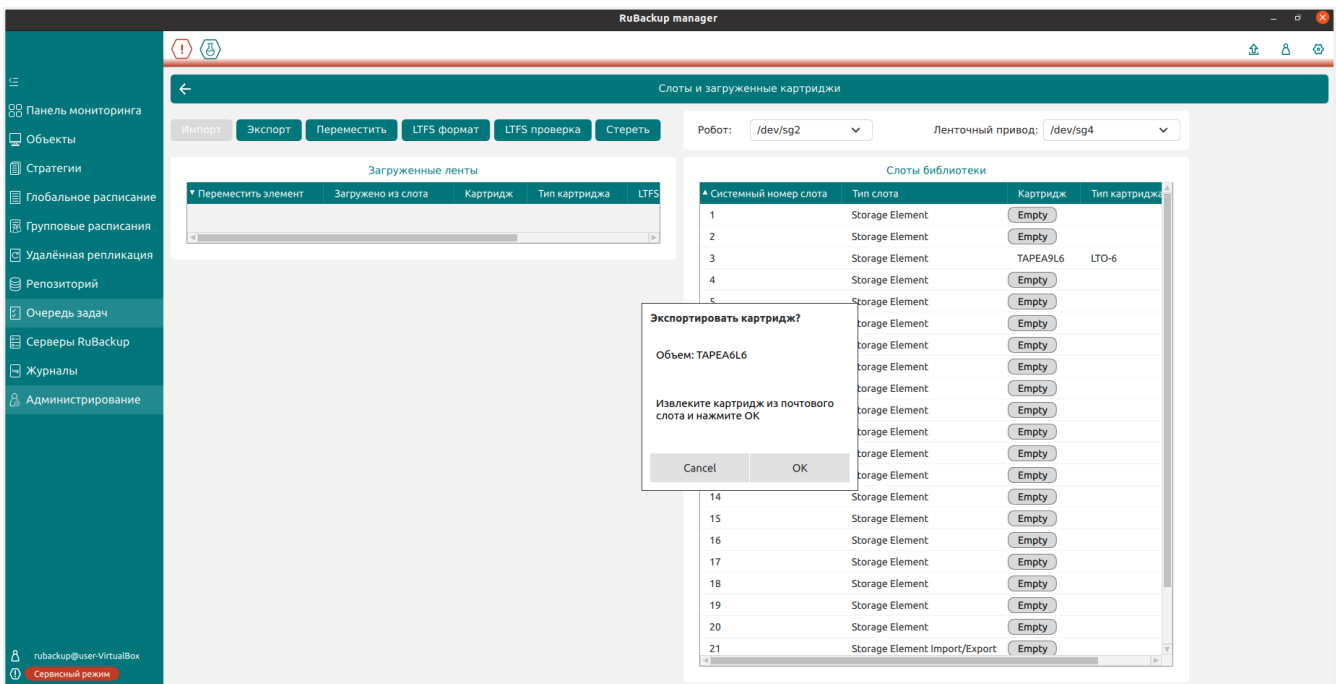


Рисунок 60.

Во время экспорта картриджа новые задачи, использующие этот картридж, не создаются.

Если в главной очереди задач есть активные задачи, использующие картридж ленточной библиотеки, то экспорт такого картриджа невозможен. Дождитесь завершения всех задач, связанных с этим картриджем, либо завершите эти задачи в очереди задач вручную.

Если картридж был экспортирован, система не будет предпринимать попытки удалить резервные копии, срок хранения которых истек. Пользователю потребуется повторно импортировать картридж в ленточную библиотеку. После этого система сможет обработать резервные копии с истекшим сроком хранения и удалить их с ленточных картриджей.

Информацию о том, как физически извлечь картридж из приёмного слота ленточной библиотеки смотрите в эксплуатационной документации на ленточную библиотеку.

Инвентаризация резервных копий

Чтобы внести записи о РК с картриджа в служебную БД и иметь возможность совершать с ними все [доступные операции](#), произведите инвентаризацию РК. Воспользуйтесь для этого утилитой `rb_tape_libraries`.

Ход инвентаризации можно проследить по статусам в разделе **Задачи** → **Ленточные библиотеки**:

New

Только что поставленная задача на инвентаризацию, выполнение которой еще не началось

Wait



Задача не может быть запущена (например, потому что не было


свободных приводов для чтения картриджей) и ожидает перезапуска сервером

Execution	Идет процесс инвентаризации
Done	Инвентаризация успешно завершилась
Error	При инвентаризации возникла критическая ошибка

Если картридж отформатирован и в ходе инвентаризации не было ошибок, то статус картриджа поменяется на *Готов*, а в БД появятся записи о РК.

Проинвентаризированные РК можно найти:

- в разделе  **Репозиторий**, если на картридже используется файловая система LTFS. Для этого в столбце **Файл РК** (необходимо добавить в настройках таблицы) кликните по заголовку таблицы левой кнопкой мыши и введите тэг картриджа в поле **Фильтр**;
- в разделе **Ленточные картриджи** ( **Хранилища** → **Ленточные картриджи**), если на картридже используется нативное хранение. Для этого кликните правой кнопкой мыши по картриджу и в меню **Перейти к...** выберите **Список РК**.



Из раздела  **Репозиторий** можно перейти к картриджам, содержащим инвентаризированные РК. Для этого вызовите контекстное меню резервной копии и в меню **Перейти к...** выберите **Ленточные картриджи**.

Если картридж не отформатирован, то после инвентаризации его статус останется *Не готов* и для дальнейшей работы с ним его нужно [отформатировать](#).

Удаление ленточной библиотеки

Для удаления ленточной библиотеки необходимо:

1. Включить сервисный режим.

Чтобы переключить RuBackup в сервисный режим, нажмите на кнопку  (**Настройки**) в правом верхнем углу ([Рисунок 61](#)) и включите флаг  ([Рисунок 62](#)).

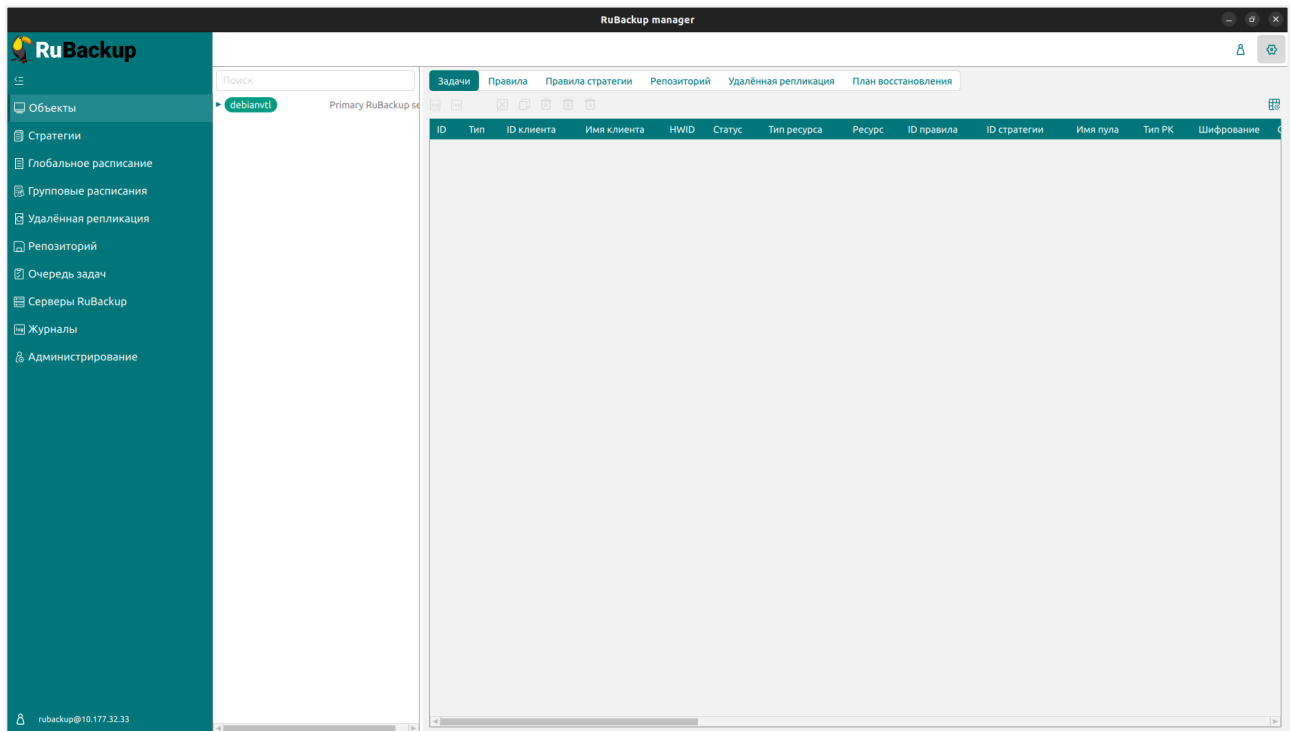


Рисунок 61.

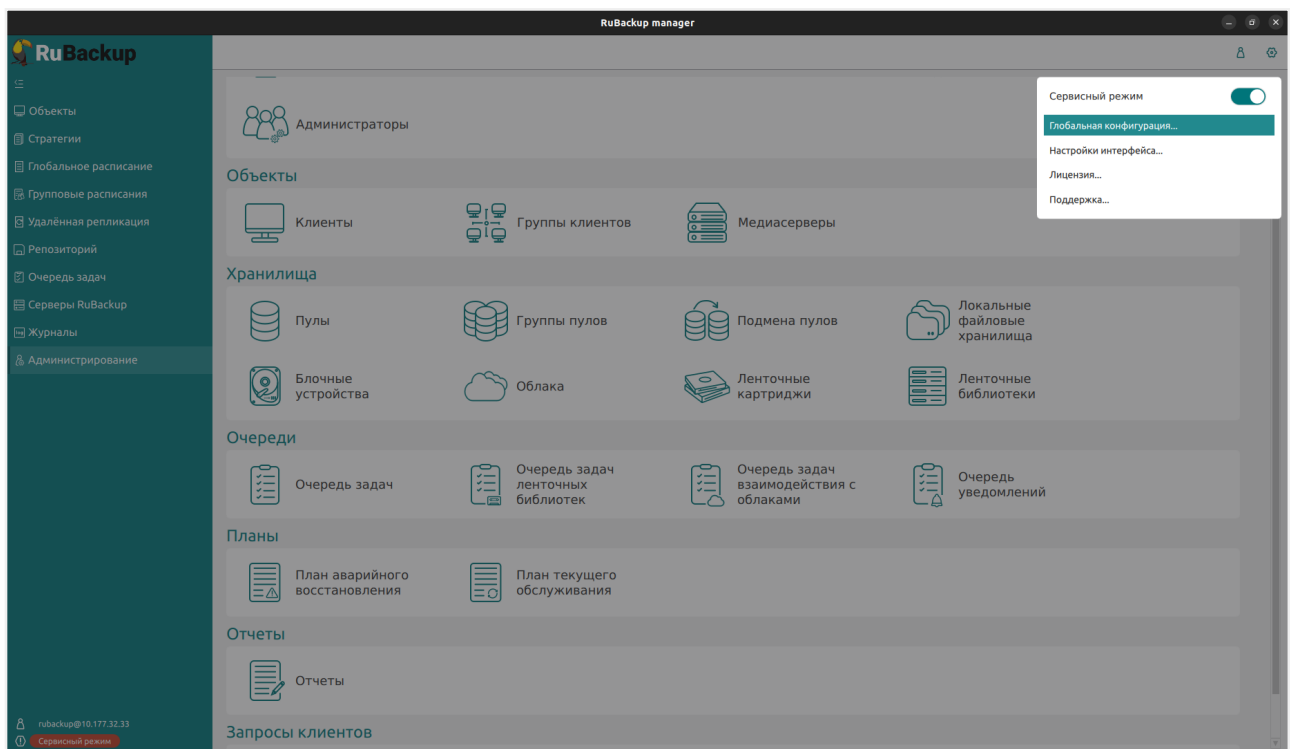



Рисунок 62.

2. В окне «Все библиотеки» выбрать необходимую библиотеку.
3. Нажать кнопку  **Удалить библиотеку** (Рисунок 63).

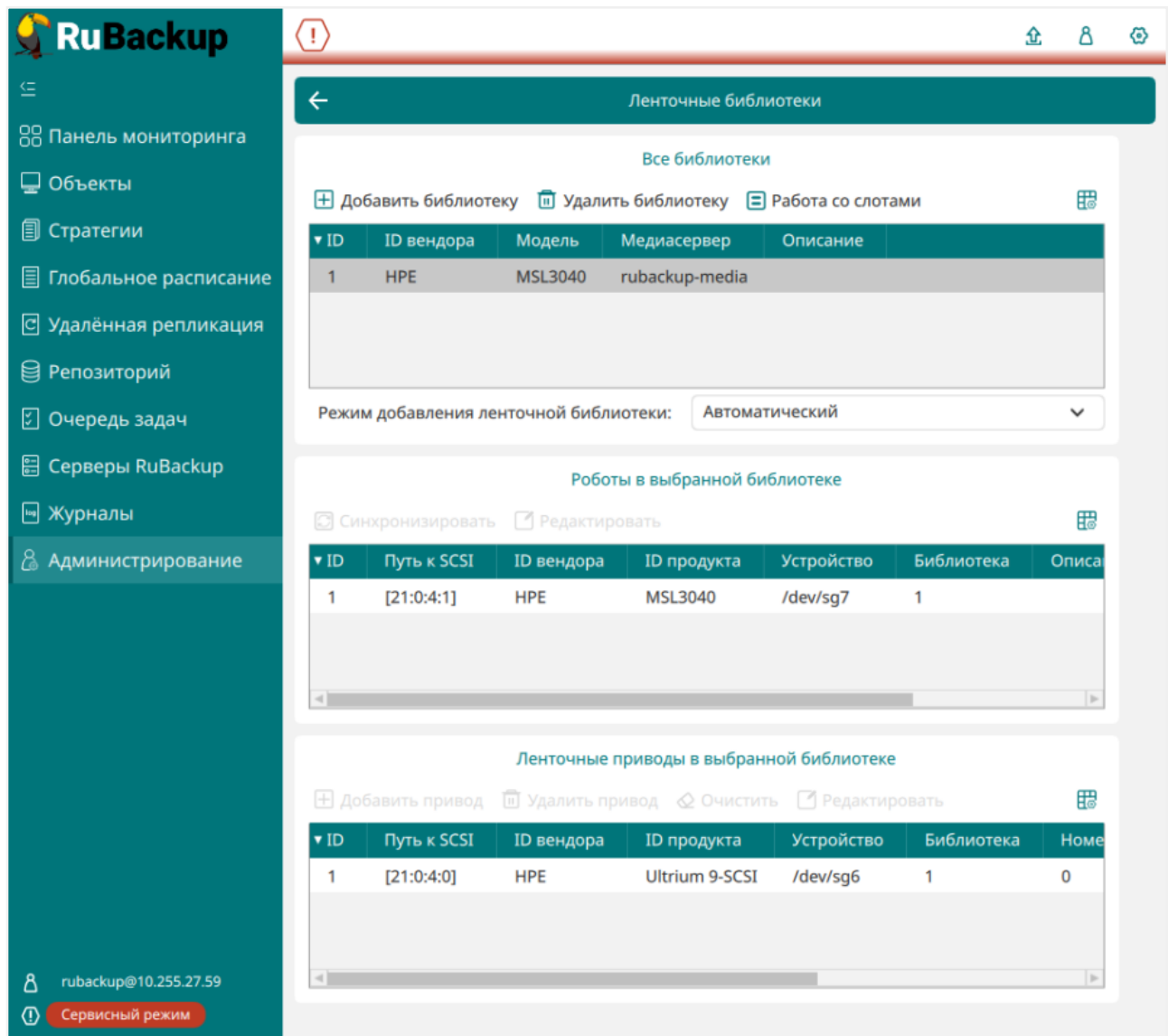


Рисунок 63.

4. Во всплывающем окне нажать кнопку «Да» (Рисунок 64).

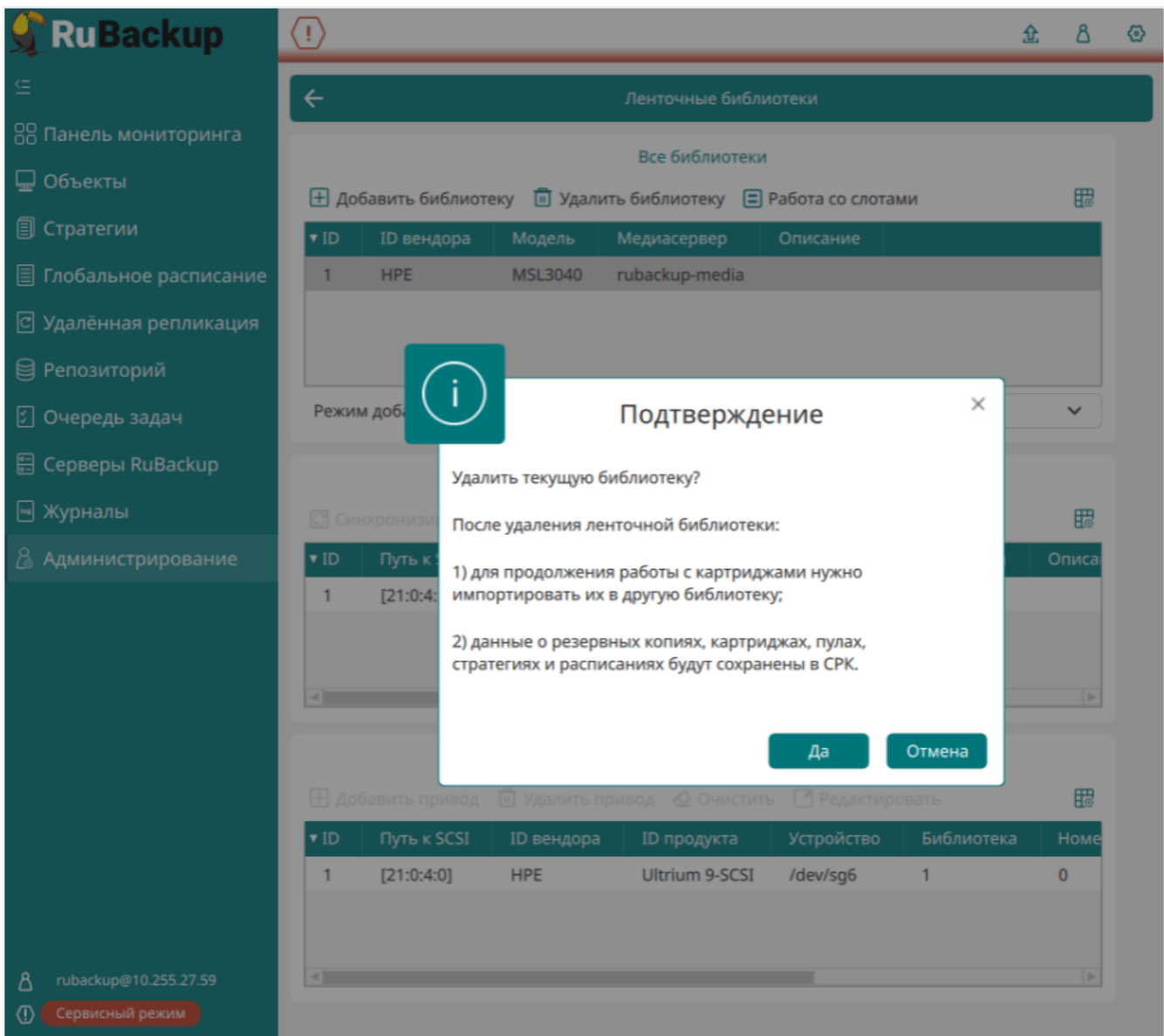


Рисунок 64.

В результате выбранная ленточная библиотека будет удалена ([Рисунок 65](#)).

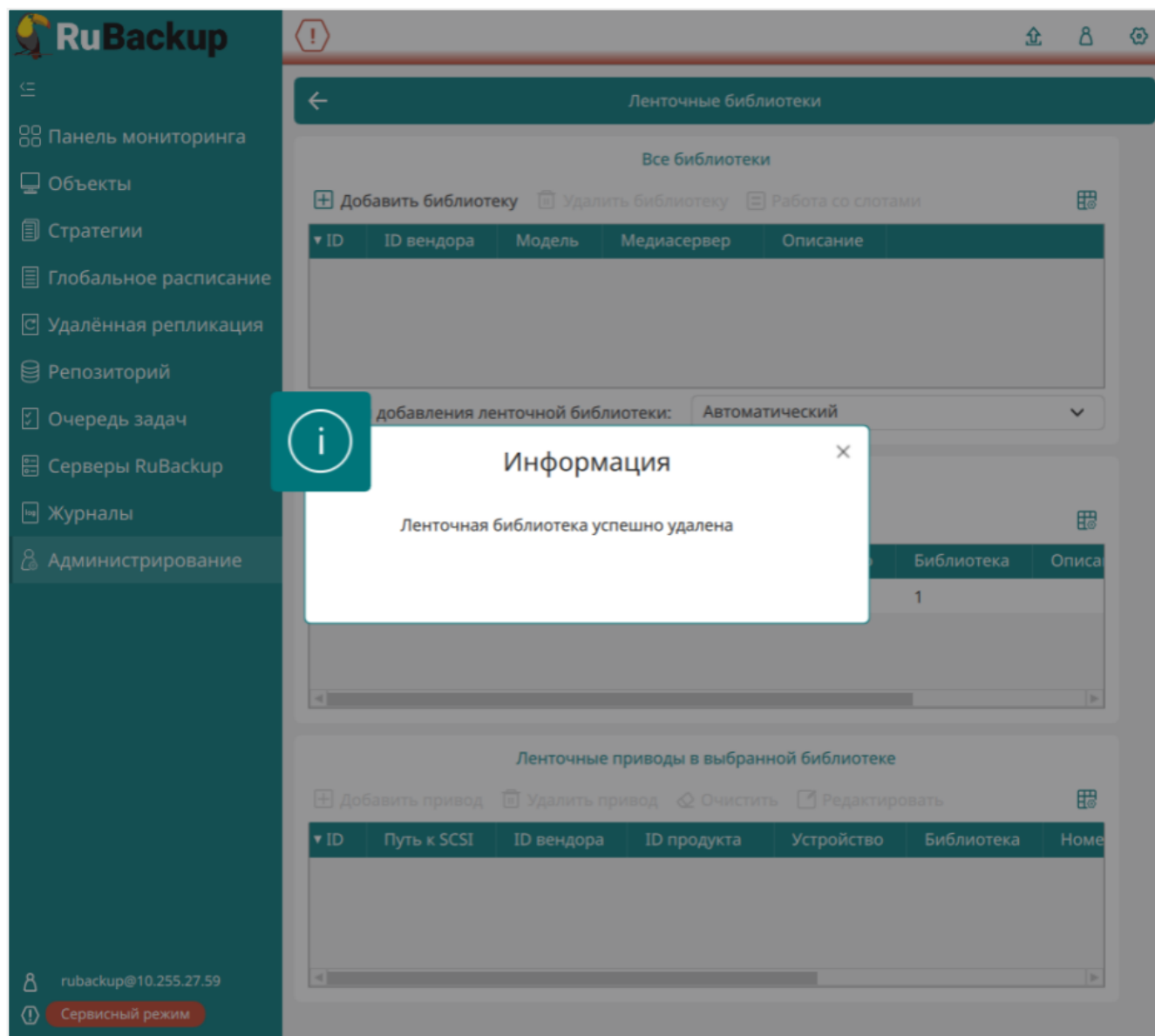


Рисунок 65.

Данные о резервных копиях, картриджах, пулах, стратегиях и расписаниях будут сохранены. Для продолжения работы с картриджами нужно импортировать их в другую библиотеку.

Если в очереди задач существуют задачи, связанные с удаляемой ленточной библиотекой, появится окно с предупреждением ([Рисунок 66](#)). В таком случае необходимо вручную завершить задачи, связанные с удаляемой ленточной библиотекой и повторить процесс удаления.

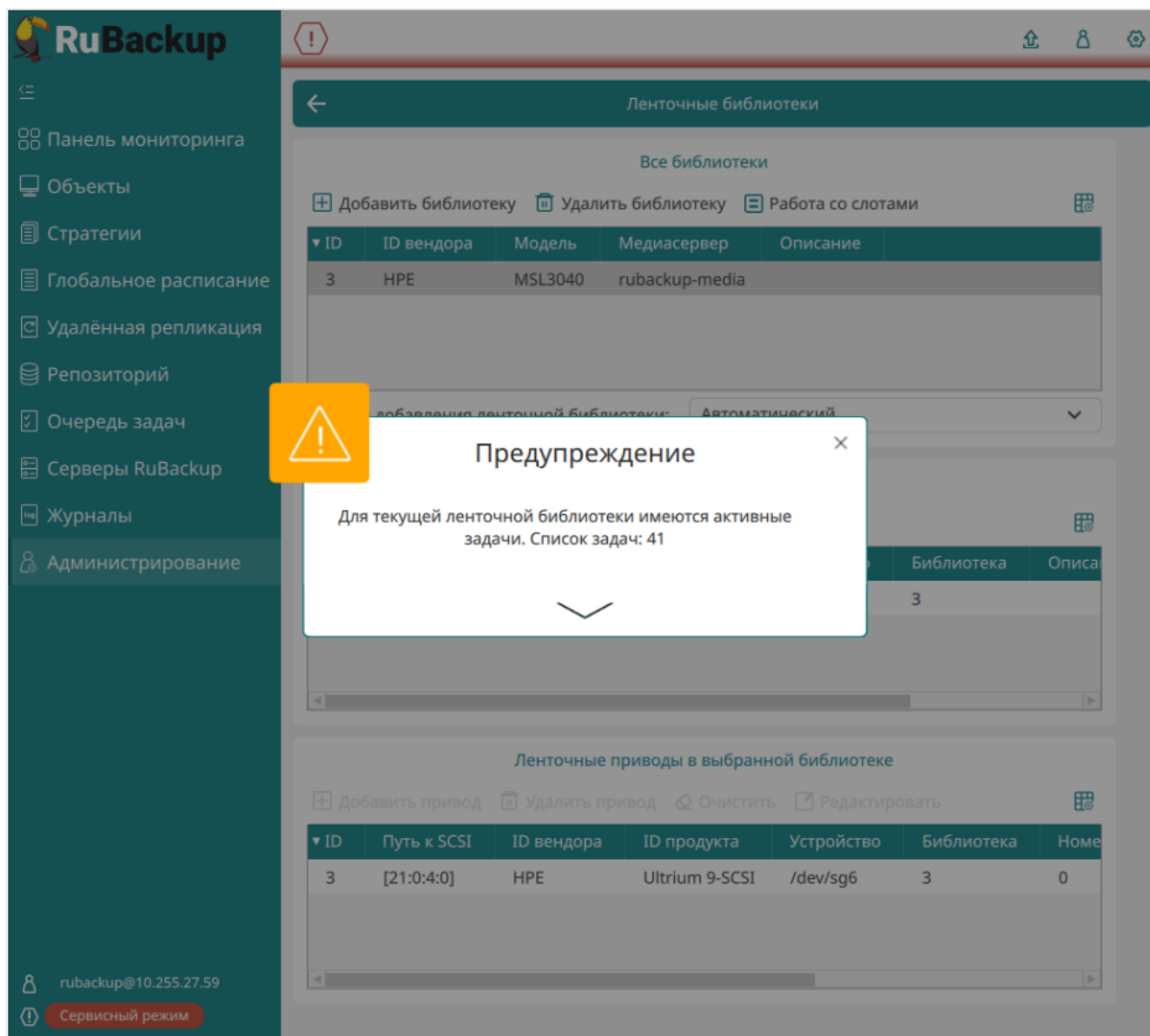


Рисунок 66.

Коллекция картриджей ленточных библиотек

Ознакомиться с коллекцией ленточных картриджей RuBackup можно, выбрав в меню пункт **Ленточные картриджи** (Рисунок 67):

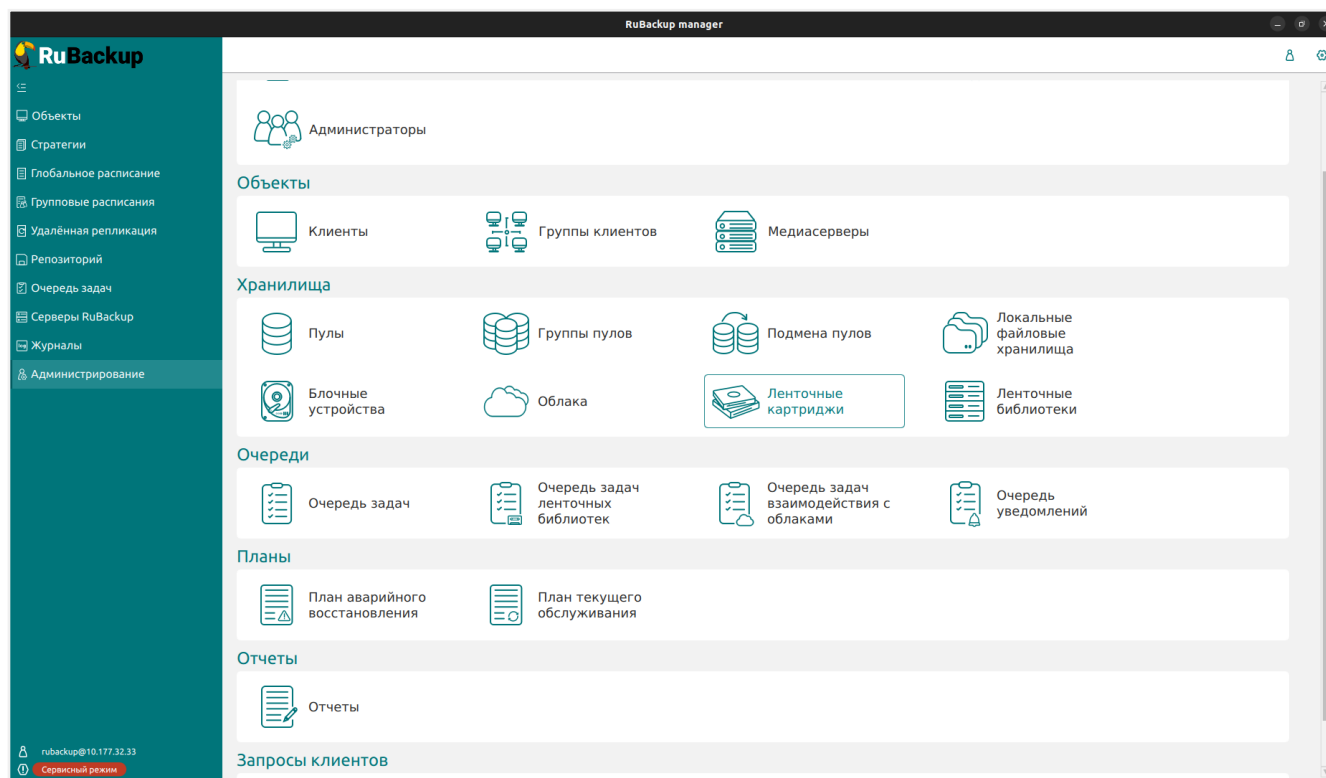


Рисунок 67.

В открывшемся окне можно увидеть ID картриджа, его тип, имя пула, статус картриджа, количество некритических ошибок, свободная ёмкость на картридже в байтах, объём занятого места в ГБ, метку тома, количество резервных копий на картридже, количество неистекших резервных копий, количество монтирований и описание картриджа.

Также можно добавить, редактировать, удалить картриджи и изменить статус картриджа. Удалить картриджи из коллекции можно только после того, как они были экспортированы из ленточной библиотеки ([Рисунок 68](#)).

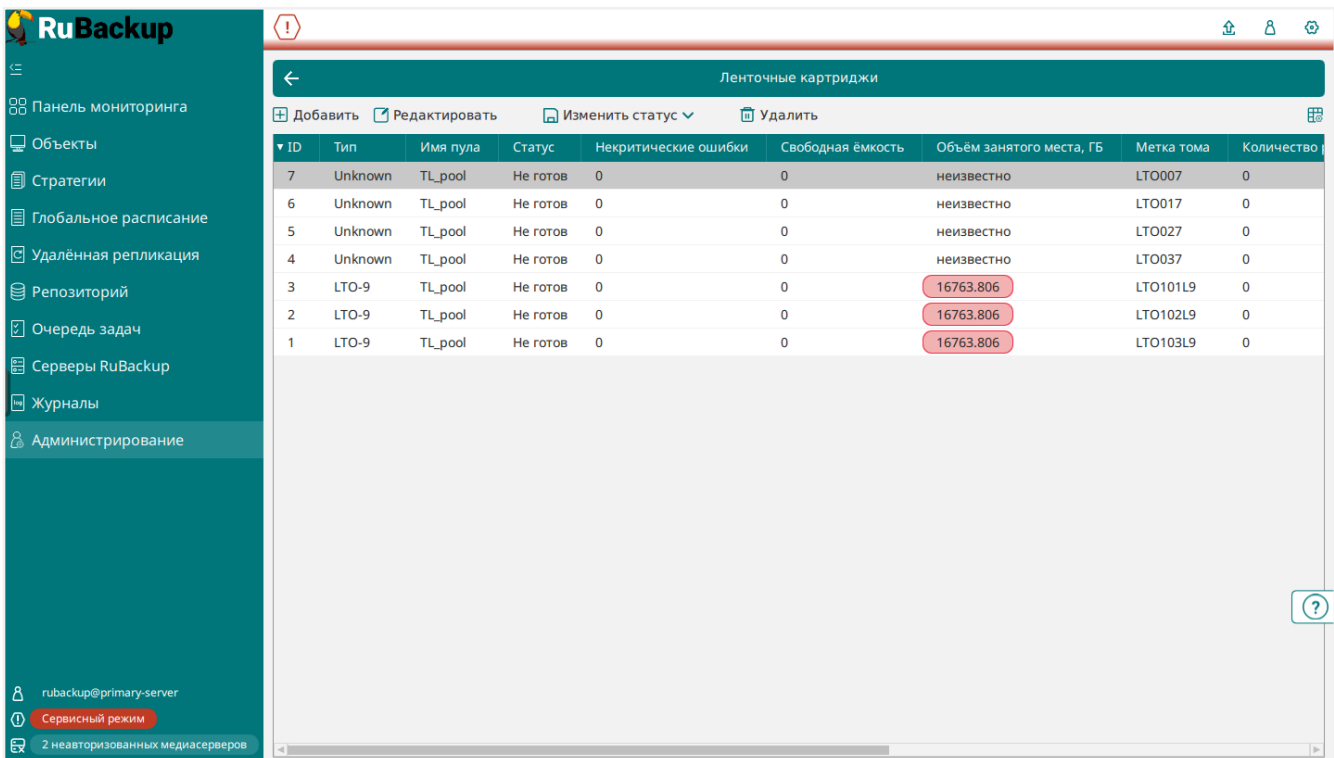


Рисунок 68.

С помощью меню «Изменить статус» можно заморозить и разморозить, а также приостановить и возобновить работу картриджа (Рисунок 69). Подробнее о статусах, которые могут принимать картриджи, см. в разделе [Статусы ленточных картриджей](#).

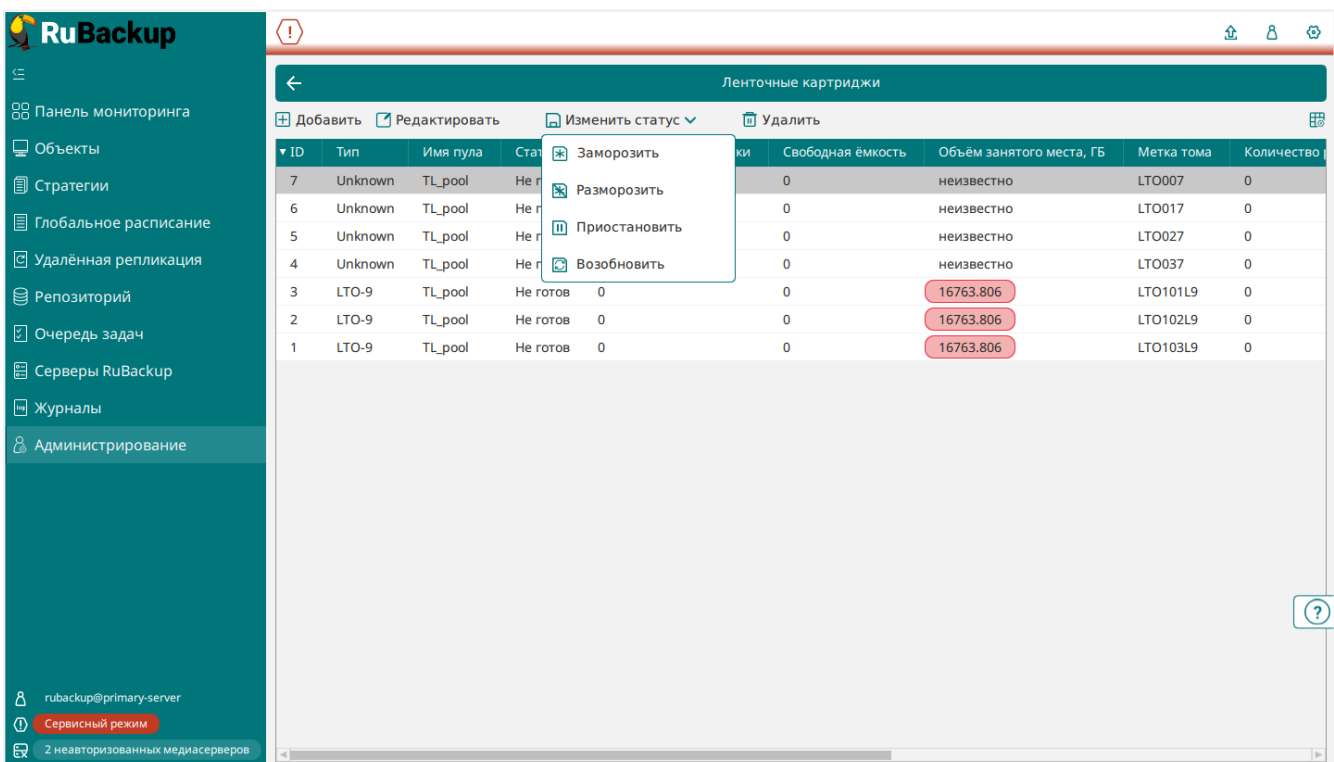


Рисунок 69.

Дополнительные настройки

В глобальных конфигурационных настройках присутствует раздел, имеющий отношение к общим настройкам для всех ленточных библиотек, входящих в конфигурацию RuBackup. Для изменения глобальных конфигурационных настроек RuBackup должен находиться в сервисном режиме. Войти в окно глобальных конфигурационных настроек можно из главного меню RBM ([Рисунок 70](#)).

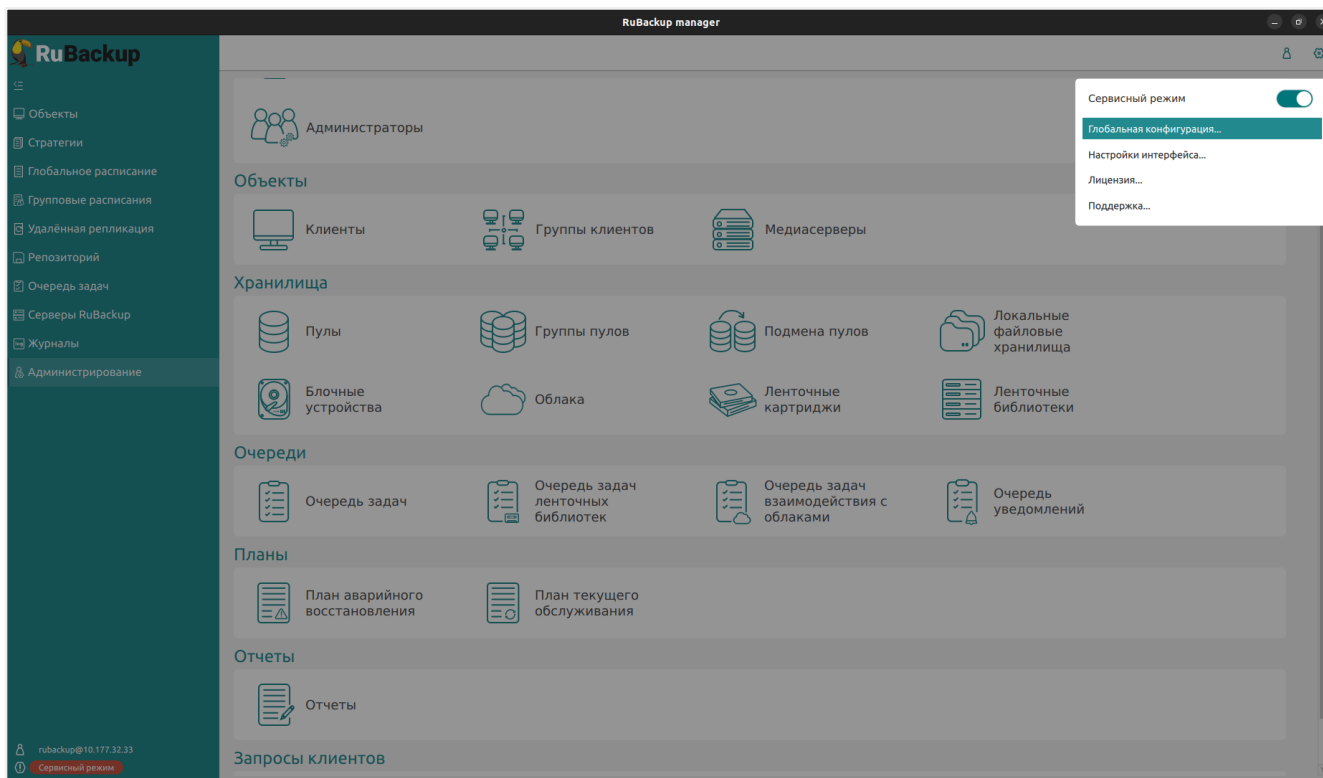


Рисунок 70.

В глобальных настройках ([Рисунок 71](#)) можно изменить:

- точку монтирования картриджей ленточной библиотеки. Если вы изменяете этот параметр, то вам необходимо позаботиться о том, чтобы эта точка монтирования присутствовала на всех серверах RuBackup, где предполагается использование ленточных библиотек;
- для нормальной работы RuBackup при старте пытается выгрузить картриджи из ленточных приводов LTFS-библиотек. Вы можете изменить этот параметр, но Вам придётся позаботиться о том, чтобы самостоятельно выгружать картридж из привода ленточной библиотеки, если он случайно оказался в ленточном приводе, при старте медиасервера;
- после выполнения любой задачи, связанной с использованием ленточного картриджа, RuBackup выгружает картридж из ленточного привода в слот ленточной библиотеки. Файловой системе LTFS при отмонтировании требуется некоторое время для завершения работы. Минимальный таймаут для размонтирования определяется параметром «Таймаут размонтирования LTFS». Если этого таймаута не хватит для завершения работы, RuBackup будет дожидаться, когда

LTFS сбросит все данные на ленту, и только после этого выгрузит картридж из ленточного привода в слот ленточной библиотеки (Рисунок 71).

- параметр «Время выгрузки картриджа нативных библиотек» определяет, через сколько минут после последней работы с картриджем он будет выгружен из драйва. Параметр необходим для оптимизации загрузки и выгрузки картриджа.

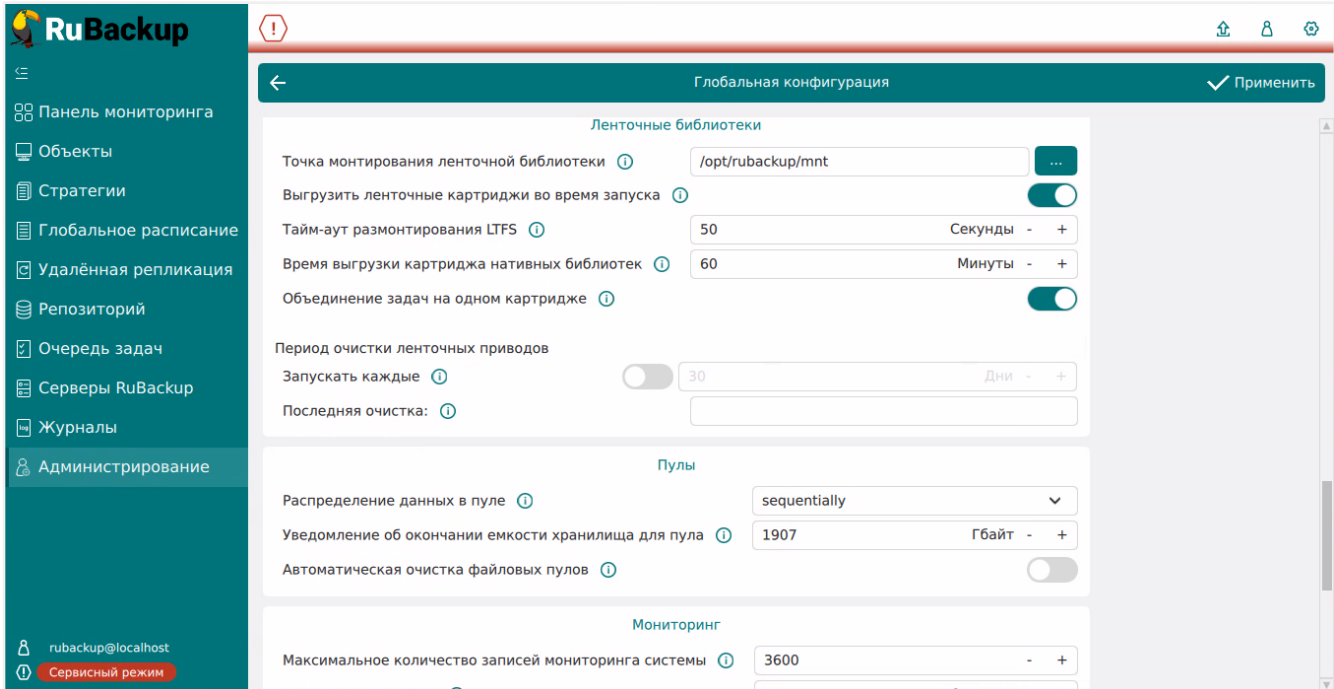


Рисунок 71.

При выполнении задачи, связанной с картриджем ленточной библиотеки он должен быть загружен в ленточный привод. На время загрузки картриджа, а это может занимать десятки секунд, задача переходит в статус «Suspended» («Приостановлено»).

Параметр «Период перезапуска для приостановленных задач» (Рисунок 72) определяет минимальный период в минутах, по прошествии которого приостановленные задачи будут возобновлены. Если к тому времени в очереди задач ленточных библиотек задача загрузки картриджа для основной приостановленной задачи будет со статусом «Ready», то задача продолжит свою работу. Если задача в очереди ленточных библиотек будет находиться в статусе ожидания, то основная задача будет вновь приостановлена. С помощью очереди ленточных библиотек осуществляется диспетчеризация задач главной очереди (это задачи на выполнение резервного копирования, восстановление, перемещение, проверку, удаление резервных копий) во избежании конфликтов за один и тот же картридж ленточной библиотеки.

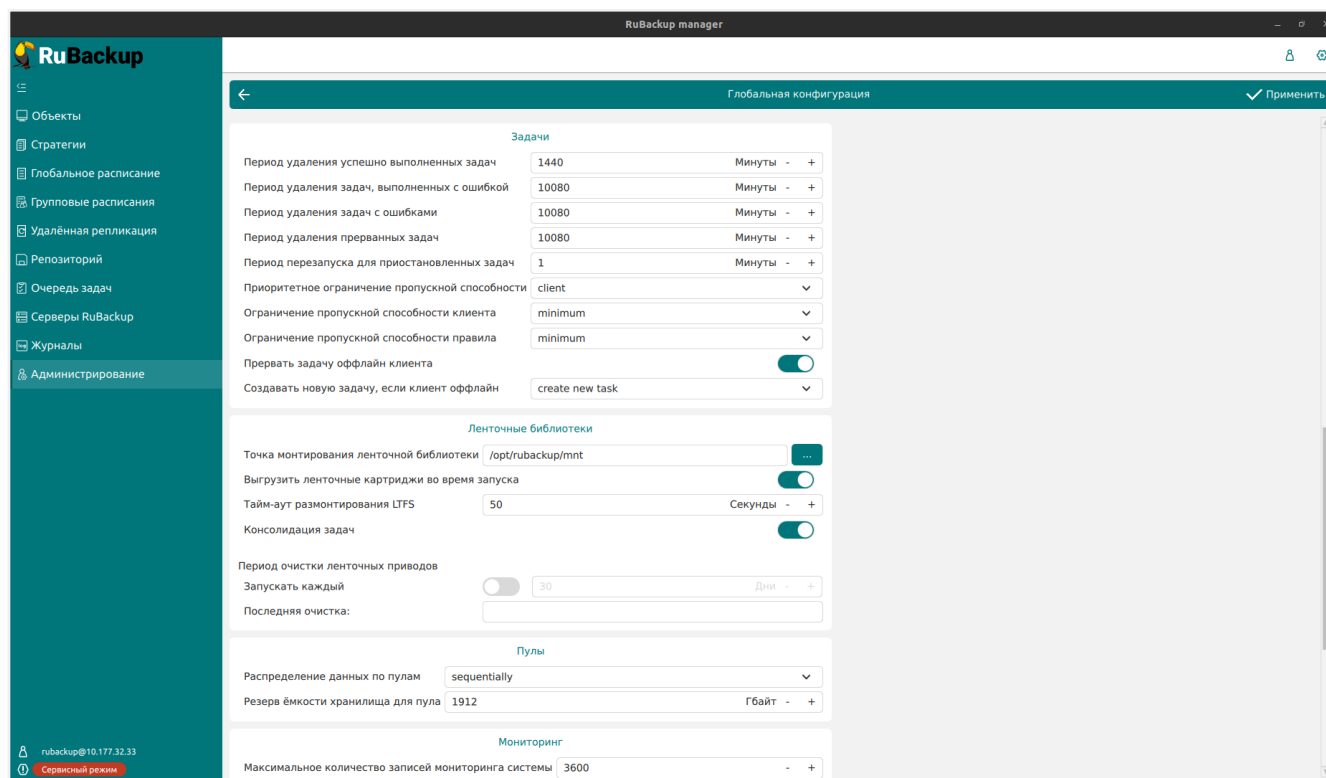


Рисунок 72.

На вкладке **Пулы** можно настроить следующие параметры (Рисунок 73):

- метод распределения данных по пулам: последовательно или параллельно. Если в пуле есть несколько устройств хранения резервных копий, то можно выбрать стратегию заполнения устройств резервными копиями;
- резерв ёмкости хранилища для пула (Гбайт). Когда в пуле останется пространства для хранения резервных копий меньше этого значения, будет создана задача на уведомление.

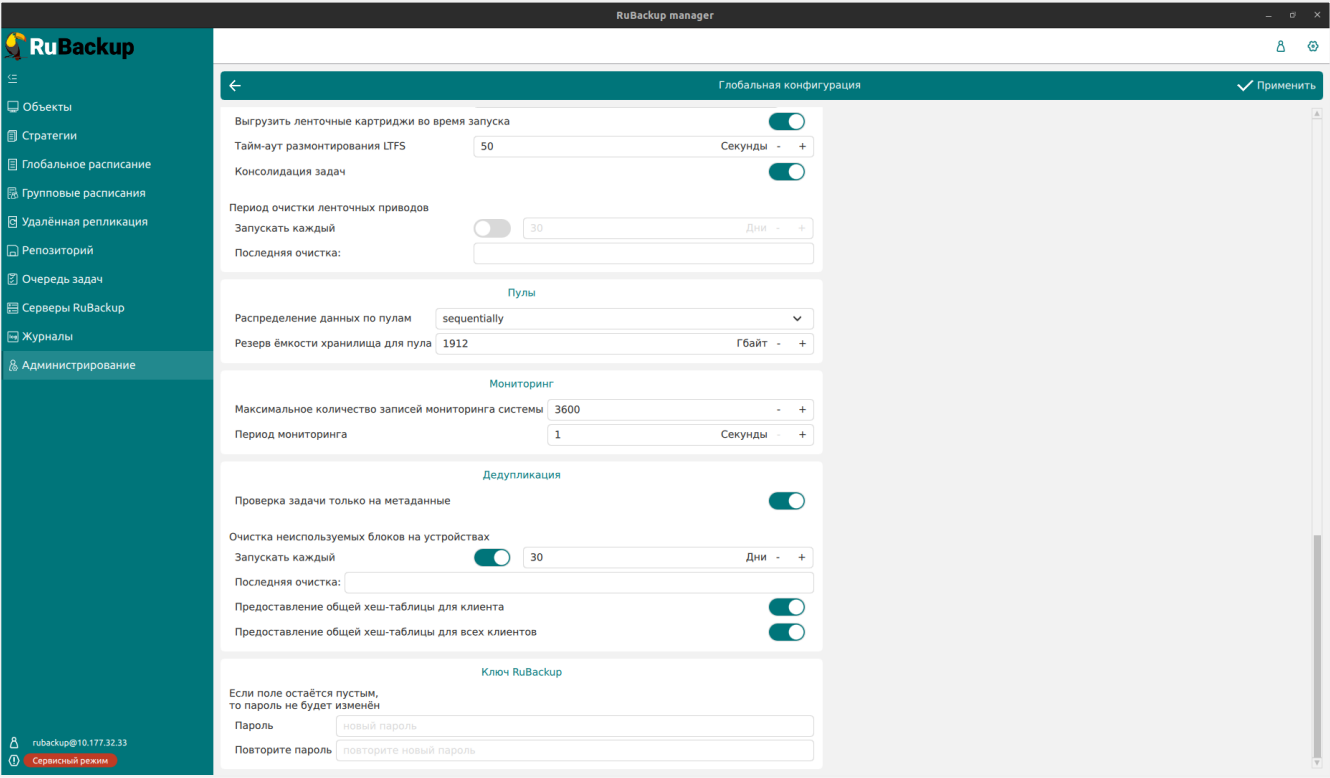


Рисунок 73.

Статусы ленточных картриджей

При работе с ленточными картриджами необходимо учитывать их статусы. Ниже приведена таблица Таблица 2 со статусами, их описанием и на какие статусы они могут быть изменены.

Таблица 2. Статусы ленточных картриджей

Статус	Описание	На какой статус могут быть изменены
Готов (Ready)	Картридж готов к использованию и находится внутри ленточной библиотеки, с ним можно производить все доступные действия. Статусом Готов отмечается, в частности, только что отформатированный картридж.	Заморожен Приостановлен Экспортирован Ошибка Не готов

Статус	Описание	На какой статус могут быть изменены
Не готов (NotReady)	Картридж не готов к использованию (отсутствует информация о наличии файловой системы). Требуется форматирование картриджа для использования, проверка на наличие файловой системы (LTFS или нативной) или инвентаризация (утилита rb_tape_libraries). Статусом <i>Не готов</i> отмечается, в частности, только что стертый картридж.	Готов Ошибка Экспортирован
Заморожен (Frozen)	Статус выставляется в случае проблем записи на картридж после нескольких неудачных попыток записать РК на ленту. Картридж доступен для чтения и недоступен для записи. Срок хранения резервной копии не истекает, пока администратор не разморозит картридж вручную. Если у картриджа стоит данный статус, то с него не будут удаляться просроченные резервные копии специальным механизмом. Администратор может вручную выставить статус для сохранения РК.	Готов Ошибка Экспортирован
Приостановлен (Suspended)	Данный статус можно выставить только вручную. Картридж доступен для чтения и недоступен для записи. РК могут удаляться, срок хранения РК продолжает отсчитываться.	Готов Ошибка Экспортирован
Экспортирован (Exported)	Картридж экспортирован из ленточной библиотеки и не может быть использован.	—

Статус	Описание	На какой статус могут быть изменены
Ошибка (Error)	Ошибки монтирования LTFS, ошибка выгрузки картриджа из драйва. Статус выставляется, если нет технической возможности работать с картриджем. Статус также выставляется при достижении лимита количества некритических ошибок (раздел «Настройки» → «Глобальная конфигурация» → «Ленточные библиотеки» → поле «Ограничение количества некритических ошибок»)	Готов Не готов Экспортирован



Невозможно изменить статус, пока картридж находится в приводе.



После разморозки картриджей продолжится отсчёт срока годности РК. Некоторые РК могут быть удалены.

Утилиты командной строки RuBackup для работы с ленточной библиотекой

Для управления ленточной библиотекой из командной строки предназначены следующие утилиты:

`rb_tape_libraries` — информация о ленточных библиотеках в системе резервного копирования и управление ими.

`rb_tape_cartridges` — информация о коллекции ленточных картриджей, зарегистрированных в системе резервного копирования, и управление ими.

`rb_tl_task_queue` — информация о заданиях в очереди ленточных библиотек.

`rb_inventory` - информации о резервных копиях, которые были сделаны вне текущей конфигурации RuBackup, например, в другой серверной группировке RuBackup.

Утилиты командной строки не предназначены для настройки новой библиотеки в системе резервного копирования RuBackup. Для этого воспользуйтесь оконным менеджером системного администратора RBM.

Описание утилит командной строки содержится в руководстве [Утилиты командной строки](#).