RuBackup
Система резервного копирования и восстановления данных

Дедупликация в RuBackup

RuBackup

Версия 1.7

2021 г.

Содержание

Введение

Система резервного копирования RuBackup 1.7 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 в RuBackup

При выполнении дедупликации происходит вычисление хеша для всех блоков данных, которые должны попасть в резервную копию. Хеш-алгоритмы, поддерживаемые RuBackup, приведены в таблице 1.

Алгоритм	Длина хеш, 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

Таблица 1. Алгоритмы хеш-функций, поддерживаемые RuBackup.

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 дедупликационном буфере в оперативной 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.

Внимание! 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**.

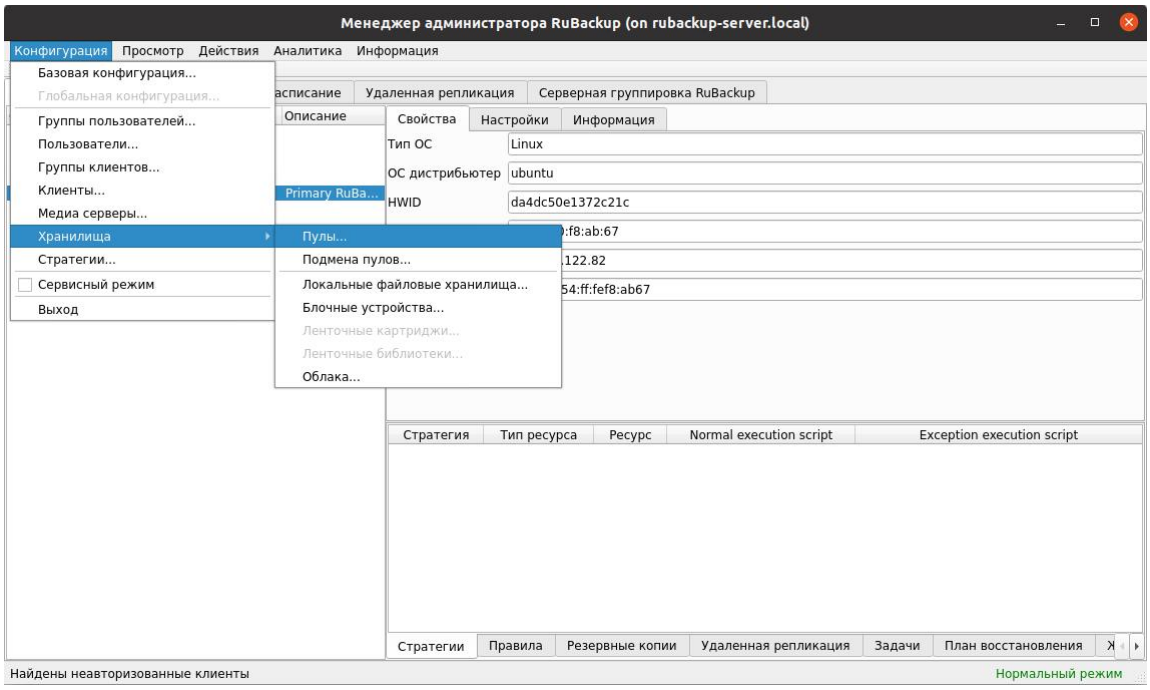

Рис. 1. Пункт «Пулы» главного меню RBM.

2. В окне «**Пулы**» нажать кнопку **Добавить**.

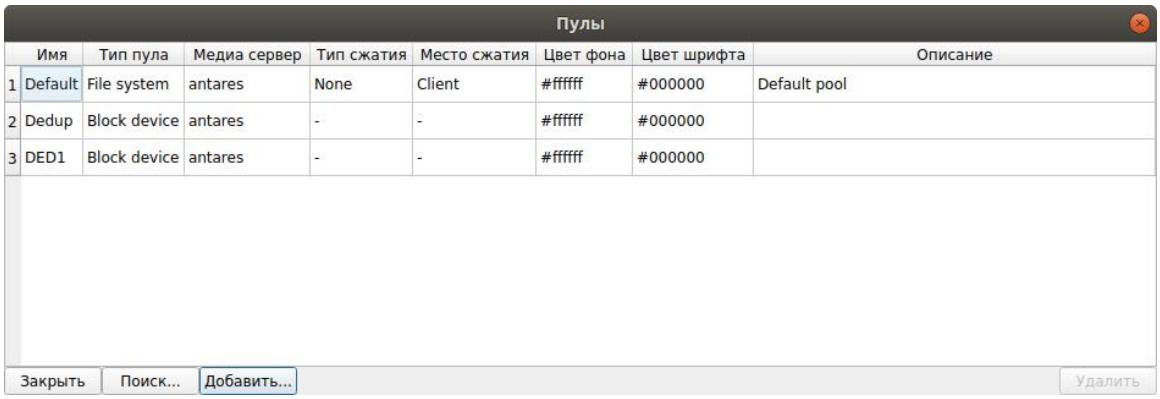

Рис. 2. Окно «Пулы» в RBM.

3. 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).

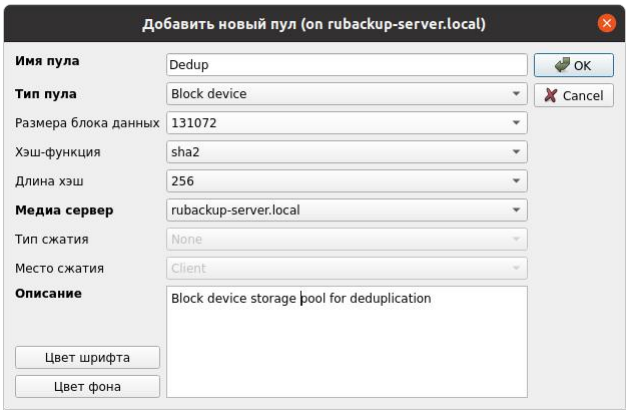

Рис. 3. Добавление нового пула в RBM.

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**.

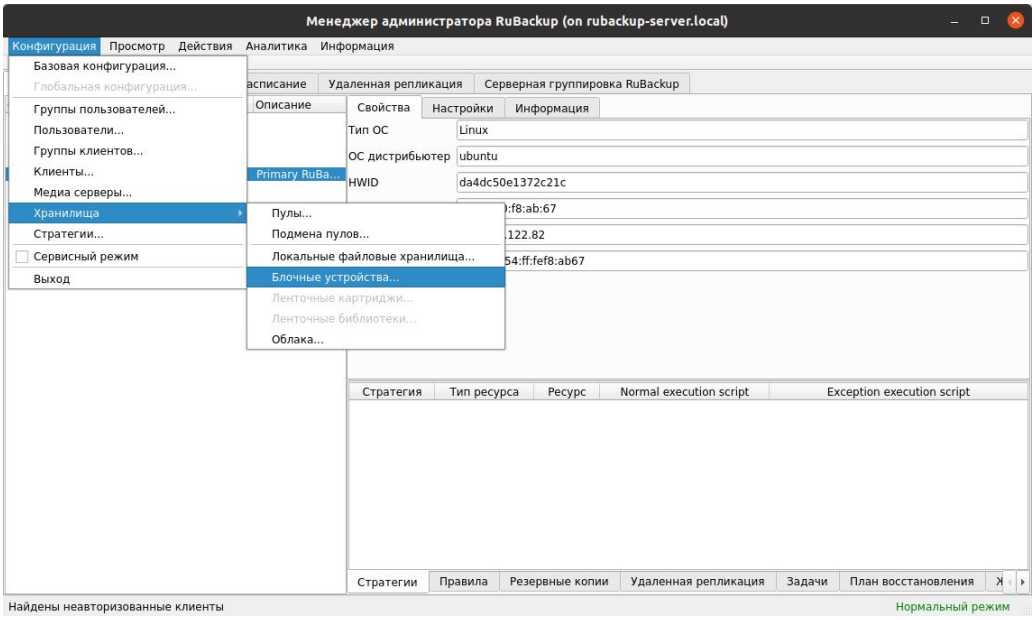

Рис. 4. Пункт «Блочные устройства» главного меню RBM.

2. В окне «**Пулы**» нажать кнопку **Добавить**.

Рис. 5. Окно «Блочные устройства» в RBM.

3. 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**».

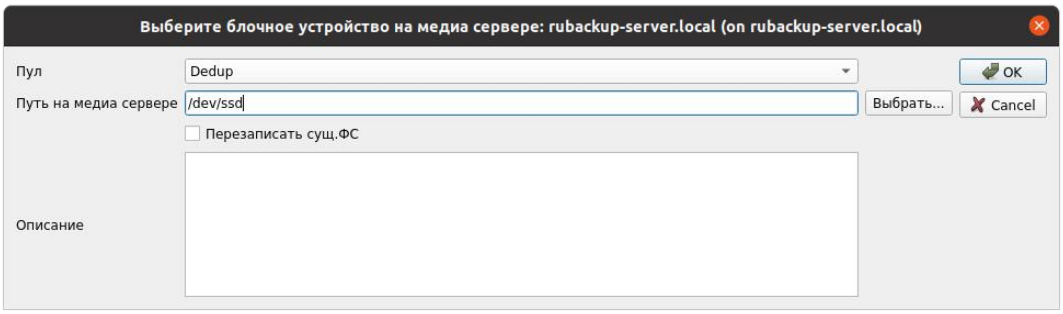

Рис. 6. Добавление блочного устройства в пул хранения данных.

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. руководство *«Утилиты командной строки RuBackup»*.

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**.

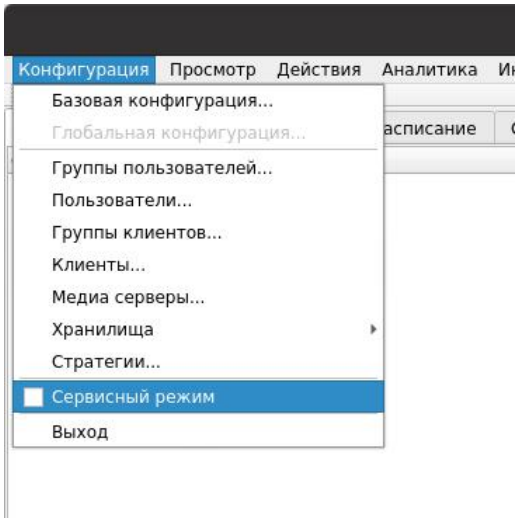

Рис. 7. Переключение сервисного режима.

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

Настройки глобальной конфигурации доступны в меню **Конфигурация > Глобальная конфигурация** на вкладке «Дедупликация». Там вы можете настроить следующие параметры:

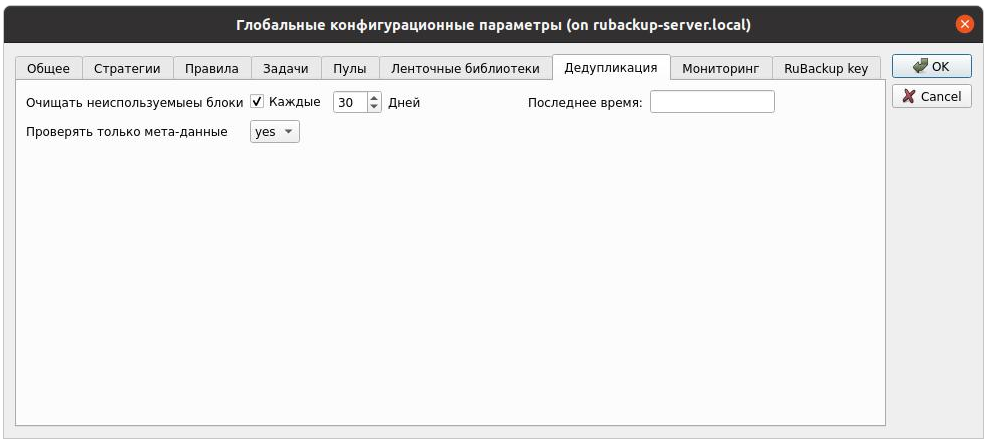

Рис. 8. Параметры дедупликации в настройках глобальной конфигурации RuBackup.

- 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».

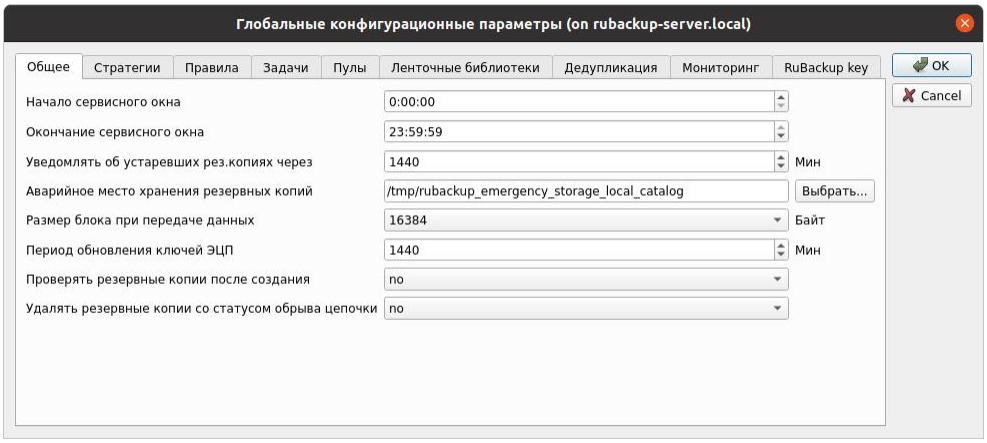

Рис. 9. Общие параметры в настройках глобальной конфигурации RuBackup.

Внимание! 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
```

Также для получения информации о поддержке дедупликации см. руководство *«Матрица совместимости»*.

- 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.