

RuBackup

Система резервного копирования и восстановления данных

Резервное копирование и восстановление баз данных РСУБД Microsoft SQL Server



RuBackup

Версия 1.9

2022 г.

Содержание

Введение.....	3
Подготовка хоста PCУБД Microsoft SQL Server.....	5
Принцип базового резервного копирования Microsoft SQL Server.....	8
Принцип дифференциального резервного копирования Microsoft SQL Server.....	9
Принцип восстановления резервной копии Microsoft SQL Server.....	10
Мастер-ключ.....	11
Защитное преобразование резервных копий.....	12
Менеджер администратора RuBackup (RBM).....	14
Менеджер клиента RuBackup (RBC).....	22
Утилиты командной строки клиента RuBackup.....	26
Восстановление резервной копии PCУБД Microsoft SQL Server.....	27
Восстановление системной базы данных 'master'.....	31

Введение

Система резервного копирования (СРК) RuBackup поддерживает базы данных системы управления реляционными базами данных (РСУБД) Microsoft SQL Server.

Принцип резервного копирования баз данных РСУБД Microsoft SQL Server с использованием RuBackup состоит в периодическом создании базовых резервных копий экземпляра РСУБД по определённому расписанию.

В репозитории RuBackup базовые резервные копии баз данных вместе с резервной копией журнала транзакций будут храниться как **полные резервные копии** (full). На основе изменений в копируемой базе данных, с момента создания последней полной резервной копии создаётся **дифференциальная резервная копия** (differential). Инкрементальное резервное копирование баз данных РСУБД Microsoft SQL Server не предусмотрено. Файл с расширением «.snap» создаётся модулем в автоматическом режиме. Данный файл содержит в себе имя копируемой базы данных, дату и время момента создания резервной копии, а в отдельных случаях - запись COPY_ONLY.

В СРК RuBackup предусмотрена возможность создания резервных копий файлов баз данных. Такой способ резервного копирования несколько быстрее обычного резервного копирования. Есть возможность создания как **полных резервных копий** файлов баз данных, так и **дифференциальных резервных копий**. Резервное копирование и в том, и в другом случае подразумевает так же резервное копирование журнала транзакций.

После окончания операции резервного копирования будут созданы два файла – архивный и snap-файл – на медиасervere, которому принадлежит пул, указанный в правиле резервного копирования. Точное расположение файлов указано в записи репозитория системы резервного копирования RuBackup.

При необходимости архивный файл может быть преобразован при помощи алгоритма защитного преобразования на клиенте и сжат. Snap-файл не преобразуется так как в нём отсутствуют значимые данные.

Для выполнения резервного копирования базы данных РСУБД Microsoft SQL Server на хосте клиента должно быть достаточно свободного места. Локальное местоположение временного каталога для создания резервных копий определено в файле /opt/rubackup/etc/config.file (параметр use-local-backup-directory). Если на хосте клиента недостаточно места для создания резервной копии, ему может быть предоставлена сетевая файловая система NFS с сервера резервного копирования во временное пользование (см. «Руководство системного администратора RuBackup»).

Для выполнения резервного копирования администратор RuBackup может настраивать правила глобального расписания в оконном Менеджере Администратора RuBackup (RBM). Также доступно восстановление резервной копии при помощи (RBM), если настроена функция централизованного восстановления.

Клиенты RuBackup могут осуществлять восстановление данных резервных копий и создание срочных резервных копий при помощи оконного Менеджера Клиента RuBackup (RBC), а также при помощи утилит командной строки RuBackup.

Подготовка хоста РСУБД Microsoft

SQL Server

Для возможности резервного копирования данных кластера РСУБД Microsoft SQL Server при помощи СРК RuBackup на сервер следует установить следующие пакеты:

- `gubackup-common.deb` — пакет зависимостей RuBackup;
- `gubackup-client.deb` — клиент резервного копирования;
- `gubackup-mssql.deb` — модуль резервного копирования баз данных Microsoft SQL Server;
- `python3-pymssql` — простой интерфейс базы данных для Python, который построен поверх FreeTDS для обеспечения интерфейса Python DB-API для Microsoft SQL Server.

Установка клиента RuBackup

Для осуществления резервного копирования и восстановления баз данных РСУБД Microsoft SQL Server при помощи RuBackup на хост должен быть установлен клиент RuBackup со всеми необходимыми модулями. Клиент RuBackup представляет собой фоновое системное приложение (демон или сервис), обеспечивающее взаимодействие с серверной группировкой RuBackup. Для выполнения резервного копирования баз данных РСУБД Microsoft SQL Server клиент RuBackup должен работать от имени суперпользователя (`root` в Linux и Unix)

Подробно процедура установки клиента описана в «Руководстве по установке Windows клиентов RuBackup».

Установка пакетов модулей резервного копирования

Установка пакета модулей резервного копирования RuBackup производится из учётной записи с административными правами на хосте с РСУБД Microsoft SQL Server **после** установки на него клиента RuBackup.

Для установки пакета модулей используйте следующий вызов:

```
# dpkg -i gubackup-mssql.deb
```

```
Выбор ранее не выбранного пакета gubackup-mssql.
```

```
(Чтение базы данных ... на данный момент установлено 137334 файла и каталога.)
```

```
Подготовка к распаковке gubackup-mssql.deb ...
```

```
Распаковывается gubackup-mssql (2021-02-20) ...
```

```
Настраивается пакет gubackup-mssql (2020-02-20) ...
```

Подготовка к использованию sqlcmd

Для выполнения резервного копирования и восстановления баз данных или файлов баз данных, необходимо установить консольную утилиту sqlcmd. Данная утилита позволит выполнять инструкции Transact-SQL в SQL Server. Ниже приведены инструкции по установке программы командной строки SQL Server: sqlcmd.

Чтобы установить mssql-tools в Ubuntu, необходимо выполнить следующие действия:

1. Импортируйте открытые ключи GPG из репозитория.

```
# curl https://packages.microsoft.com/keys/microsoft.asc | sudo apt-key add -
```

2. Зарегистрируйте репозиторий Ubuntu для Microsoft.

Для Ubuntu 20.04:

```
# curl https://packages.microsoft.com/config/ubuntu/20.04/prod.list | sudo tee /etc/apt/sources.list.d/msprod.list
```

3. Обновите список источников и выполните команду установки с помощью пакета разработчика unixODBC. Дополнительные сведения см. в разделе Установка драйвера Microsoft ODBC для SQL Server (Linux).

```
# sudo apt-get update
```

```
# sudo apt-get install mssql-tools unixodbc-dev
```

Примечание – Чтобы произвести обновление до последней версии mssql-tools, выполните следующие команды:

```
# sudo apt-get update
```

```
# sudo apt-get install mssql-tools
```

Необязательно: Добавьте путь /opt/mssql-tools/bin/ в переменную среды PATH в оболочке bash.

Чтобы утилита sqlcmd была доступна из оболочки bash в рамках интерактивных сеансов и сеансов без входа в систему, измените переменную среды PATH в файле ~/.bashrc с помощью следующей команды:

```
# echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >> ~/.bashrc
```

Удаление клиента RuBackup

При необходимости вы можете удалить с хоста клиент RuBackup и установленные модули резервного копирования.

Удаление клиента RuBackup возможно из учётной записи с административными правами.

Для удаления сервиса `rubackup-client` используйте команды:

```
# systemctl disable rubackup-client  
# systemctl daemon-reload
```

Для удаления клиента RuBackup и модуля `rubackup-mssql` используйте команды:

```
# apt remove rubackup-mssql  
# apt remove rubackup-client
```

При необходимости удалить клиент RuBackup из конфигурации системы резервного копирования, это может сделать системный администратор RuBackup с помощью оконного Менеджера Администратора (RBM).

После удаления клиента RuBackup в ОС Astra Linux SE 1.6 с активированным режимом защитной программной среды следует:

1. Выполнить команду:

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

2. Перезагрузить операционную систему

```
$ sudo init 6
```

Принцип базового резервного копирования Microsoft SQL Server

В ходе базового резервного копирования выполняется взаимодействие с утилитой sqlcmd.

Команда на выполнение полного резервного копирования базы данных:

```
# sqlcmd -S localhost -U server_admin_login -P 'YourPassword' -Q  
"BACKUP DATABASE [your_db] TO DISK = N'backup_file' WITH NOINIT,  
NO_COMPRESSION, SKIP, NOFORMAT, NOREWIND, NOUNLOAD, STATS = 10",
```

где опция -S – имя или IP-адрес сервера MS SQL;

-U - логин пользователя сервера MS SQL;

-P - пароль пользователя сервера MS SQL;

-Q - инструкции Transact-SQL.

Команда на выполнение резервного копирования файлов базы данных:

```
# sqlcmd -S localhost -U server_admin_login -P 'YourPassword' -Q  
"BACKUP DATABASE [your_db] FILE = N'DB_file_logical_name' TO DISK =  
N'backup_file' WITH NO_COMPRESSION, NOINIT, SKIP, NOFORMAT, STATS = 10"
```

Команда на выполнение резервного копирования журнала транзакций:

```
# sqlcmd -S localhost -U server_admin_login -P 'YourPassword' -Q  
"BACKUP LOG [your_db] TO DISK = N'backup_file' WITH NO_COMPRESSION,  
NOINIT, SKIP, NOFORMAT, STATS = 10"
```

Принцип дифференциального резервного копирования Microsoft SQL Server

SQL Server

В ходе выполнения дифференциального резервного копирования выполняется взаимодействие с утилитой sqlcmd:

Команда на выполнение дифференциального резервного копирования базы данных:

```
# sqlcmd -S localhost -U server_admin_login -P 'YourPassword' -Q "BACKUP DATABASE [your_db] TO DISK = N'backup_file' WITH DIFFERENTIAL, NO_COMPRESSION, NOINIT, SKIP, NOFORMAT, STATS = 10",
```

где опция -S – имя или IP-адрес сервера MS SQL;

-U - логин пользователя сервера MS SQL;

-P - пароль пользователя сервера MS SQL;

-Q - инструкции Transact-SQL.

Команда на выполнение дифференциального резервного копирования файлов базы данных:

```
# sqlcmd -S localhost -U server_admin_login -P 'YourPassword' -Q "BACKUP DATABASE [your_db] FILE = N'DB_file_logical_name' TO DISK = N'backup_file' WITH DIFFERENTIAL, NO_COMPRESSION, NOINIT, SKIP, NOFORMAT, STATS = 10"
```

Команда на выполнение резервного копирования журнала транзакций выглядит тем же образом, что и при выполнении полного резервного копирования.

Принцип восстановления резервной копии Microsoft SQL Server

При восстановлении базы данных из полной резервной копии системный администратор должен быть единственным пользователем, работающим с базой данных.

Для восстановления базы данных PCYБД Microsoft SQL Server необходимо выполнить следующие действия:

1. Если в кластере имеется база данных с таким же именем, что и восстанавливаемая, то предлагается либо удалить ее, либо сделать недоступной для всех пользователей:

```
$ sqlcmd -S localhost -U server_admin_login -P 'YourPassword' -Q "IF DB_ID('your_db') IS NOT NULL ALTER DATABASE [your_db] SET OFFLINE WITH ROLLBACK IMMEDIATE"
```

2. Выполнить восстановление резервной копии базы данных или файлов базы данных:

```
$ sqlcmd -S localhost -U server_admin_login -P 'YourPassword' -Q "RESTORE DATABASE [your_db] FROM DISK = N'backup_file' WITH NORECOVERY, STATS = 10"
```

3. Выполнить восстановление журнала транзакций если он был скопирован:

```
$ sqlcmd -S localhost -U server_admin_login -P 'YourPassword' -Q "RESTORE LOG [your_db] FROM DISK = N'backup_log_file' WITH STATS = 10"
```

4. Восстановить доступ к базе данных:

```
$ sqlcmd -S localhost -U server_admin_login -P 'YourPassword' -Q "IF DB_ID('your_db') IS NOT NULL ALTER DATABASE [your_db] SET ONLINE; IF DB_ID('your_db') IS NOT NULL ALTER DATABASE [your_db] SET MULTI_USER"
```

Мастер-ключ

В ходе установки клиента RuBackup будет создан мастер-ключ для защитного преобразования резервных копий, а также ключи для электронной подписи, если предполагается использовать электронную подпись.

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

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

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

```
$ hexdump /opt/rubackup/keys/master-key  
00000000 79d1 4749 7335 e387 9f74 c67e 55a7 20ff  
00000100 6284 54as 83a3 2053 4818 e183 1528 a343  
00000200
```

Защитное преобразование резервных копий

При необходимости, сразу после выполнения резервного копирования ваши резервные копии могут быть преобразованы на хосте клиента. Таким образом, важные данные будут недоступны для администратора RuBackup или других лиц, которые могли бы получить доступ к резервной копии (например, на внешнем хранилище картриджей ленточной библиотеки или на площадке провайдера облачного хранилища для ваших резервных копий).

Защитное преобразование осуществляется входящей в состав RuBackup утилитой `gbscrypt`. Ключ для защитного преобразования резервных копий располагается на хосте клиента в файле `/opt/rubackup/keys/master-key`. Защитное преобразование данных при помощи `gbscrypt` возможно с длиной ключа 256 бит (по умолчанию), а также 128, 512 или 1024 бита в зависимости от выбранного алгоритма преобразования.

Автоматическое защитное преобразование и обратное преобразование резервных копий клиентом RuBackup возможны при помощи ключей длиной 256 бит, однако утилита `rbcrypt` поддерживает ключи длиной 128, 256, 512 и 1024 бита (в зависимости от выбранного алгоритма преобразования). Если необходимо для правила глобального расписания выбрать особый режим преобразования, с длиной ключа, отличной от 256 бит и с ключом, располагающемся в другом месте, то вы можете воспользоваться возможностью сделать это при помощи скрипта, выполняющегося после выполнения резервного копирования (определяется в правиле глобального расписания администратором RuBackup). При этом необходимо, чтобы имя преобразованного файла осталось таким же, как и ранее, иначе задача завершится с ошибкой. Провести обратное преобразование такого файла после восстановления его из резервной копии следует вручную при помощи утилиты преобразования. При таком режиме работы нет необходимости указывать алгоритм преобразования в правиле резервного копирования, либо архив будет преобразован ещё раз автоматически с использованием мастер-ключа.

Для выполнения защитного преобразования доступны алгоритмы, представленные в таблице 1.

Таблица 1 – Алгоритмы защитного преобразования, доступные в утилите gbscrypt

Алгоритм	Длина ключа, бит	Примечание
Anubis	128, 256	
Aria	128, 256	
CAST6	128, 256	
Camellia	128, 256	
Kalyna	128, 256, 512	Украинский национальный стандарт <u>ДСТУ 7624:2014</u>
Kuznyechik	256	Российский национальный стандарт ГОСТ Р 34.12-2015
MARS	128, 256	
Rijndael	128, 256	Advanced Encryption Standard (AES)
Serpent	128, 256	
Simon	128	
SM4	128	Китайский национальный стандарт для беспроводных сетей
Speck	128, 256	
Threefish	256, 512, 1024	
Twofish	128, 256	

Менеджер администратора RuBackup

(RBM)

Оконное приложение «Менеджер администратора RuBackup» (RBM) предназначено для общего администрирования серверной группировки RuBackup, управления клиентами резервного копирования, глобальным расписанием резервного копирования, хранилищами резервных копий и другими параметрами RuBackup.

RBM может быть запущено администратором на основном сервере резервного копирования RuBackup.

Для запуска менеджера администратора RBM необходимо выполнить команду:

```
# ssh -X user@rubackup_server
```

```
# /opt/rubackup/bin/rbm&
```

Пользователь, запускающий RBM, должен входить в группу rubackup.

На вкладке **Объекты** в левой части представлен список клиентов системы резервного копирования, в котором указано имя, уникальный HWID и описание. Клиенты, которые в данный момент находятся в online, будут отмечены зеленым цветом. Клиенты в состоянии offline – красным (рисунок 1).

Для резервного копирования баз данных PCУБД Microsoft SQL Server на хосте должен быть установлен клиент RuBackup и необходимые модули. Клиент должен быть авторизован администратором RuBackup.

В том случае, если клиент RuBackup был установлен, но не авторизован, в нижней части окна RBM будет сообщение о том, что найдены неавторизованные клиенты. Все новые клиенты должны быть авторизованы в системе резервного копирования.

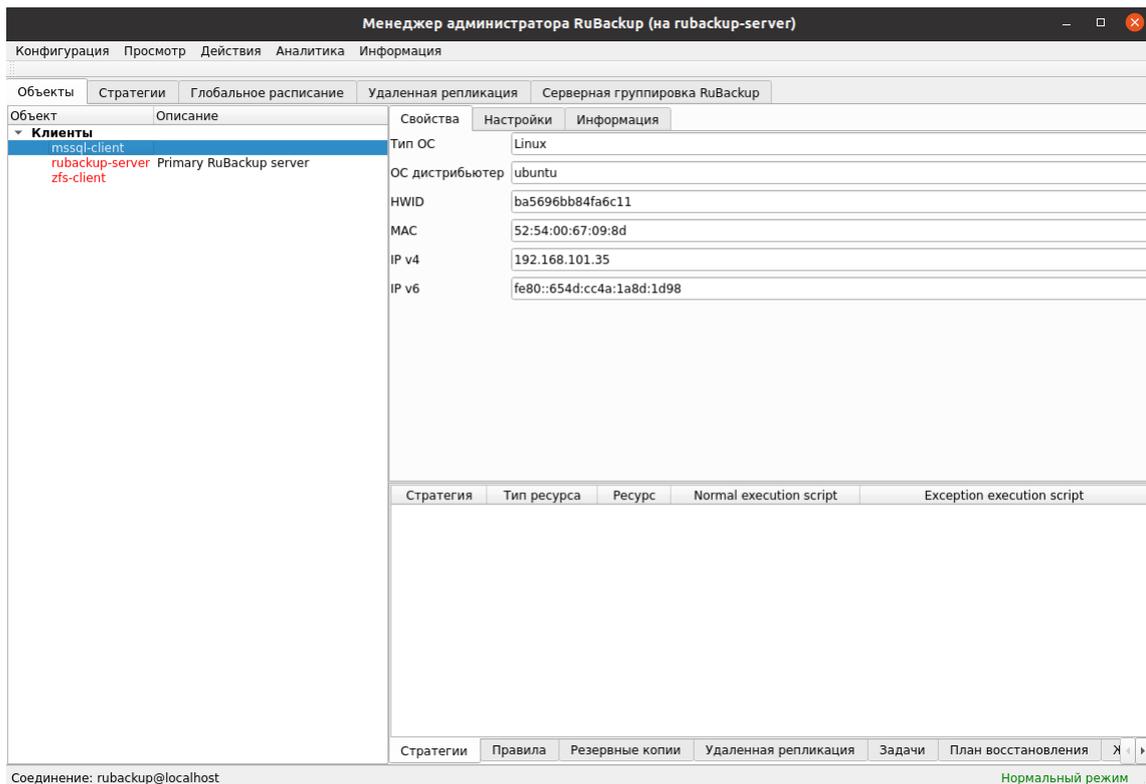


Рисунок 1

Для авторизации неавторизованного клиента в RBM выполните следующие действия:

- 1) Откройте меню **Действия** → **Клиенты** → **Авторизовать клиентов** (рисунок 2):

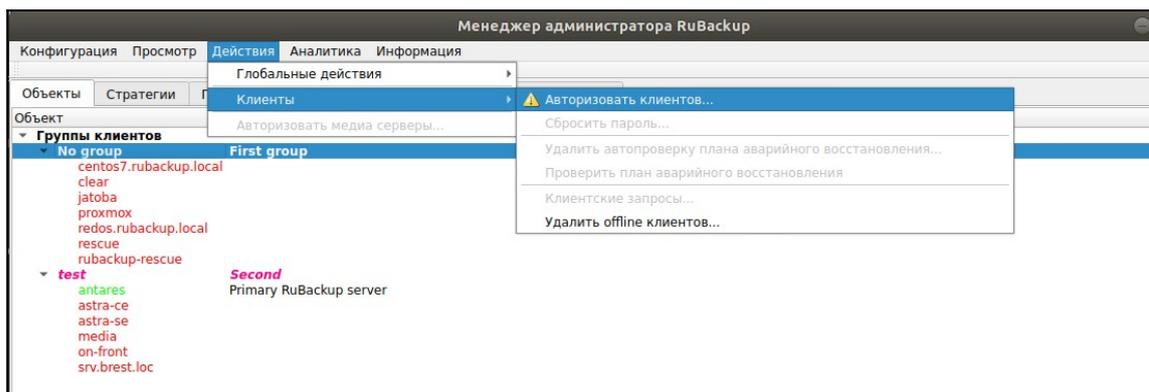


Рисунок 2

- 2) Выберите нужного неавторизованного клиента и нажмите **Авторизовать** (рисунок 3):

Неавторизованные клиенты							
Имя хоста	Тип ОС	ОС дистрибьютер	MAC	IPv4	IPv6	Последний раз на	
1 postgresql	Linux	ubuntu	52:54:00:9a:ce:f4	192.168.0.31	fe80::5054:ff:fe9a:cef4	26.10.2020	13:09

Заккрыть Авторизовать Удалить

Рисунок 3

После авторизации новый клиент будет виден в главном окне RBM (рисунок 4):

Менеджер администратора RuBackup	
Конфигурация Просмотр Действия Аналитика Информация	
Объекты Стратегии Глобальное расписание Серверная группировка RuBackup	
Объект	Описание
Группы клиентов	
<ul style="list-style-type: none"> ▼ <i>No group</i> First group <ul style="list-style-type: none"> centos7.rubackup.local clear jatoba postgresql proxmox redos.rubackup.local rescue rubackup-rescue ▼ test Second <ul style="list-style-type: none"> antares astra-ce astra-se media on-front srv.brest.loc 	<ul style="list-style-type: none"> Primary RuBackup server

Рисунок 4

Клиенты могут быть сгруппированы администратором по какому-либо общему признаку. В случае необходимости восстанавливать резервные копии на другом хосте клиенты должны принадлежать к разделяемой группе (такая группа отмечается шрифтом *italic*).

Чтобы выполнять регулярное резервное копирование баз данных РСУБД Microsoft SQL Server, необходимо создать правило в глобальном расписании. Для этого выполните следующие действия:

1. Выбрать клиентский хост, на котором находится баз данных РСУБД Microsoft SQL Server и добавить правило резервного копирования (рисунок 5).

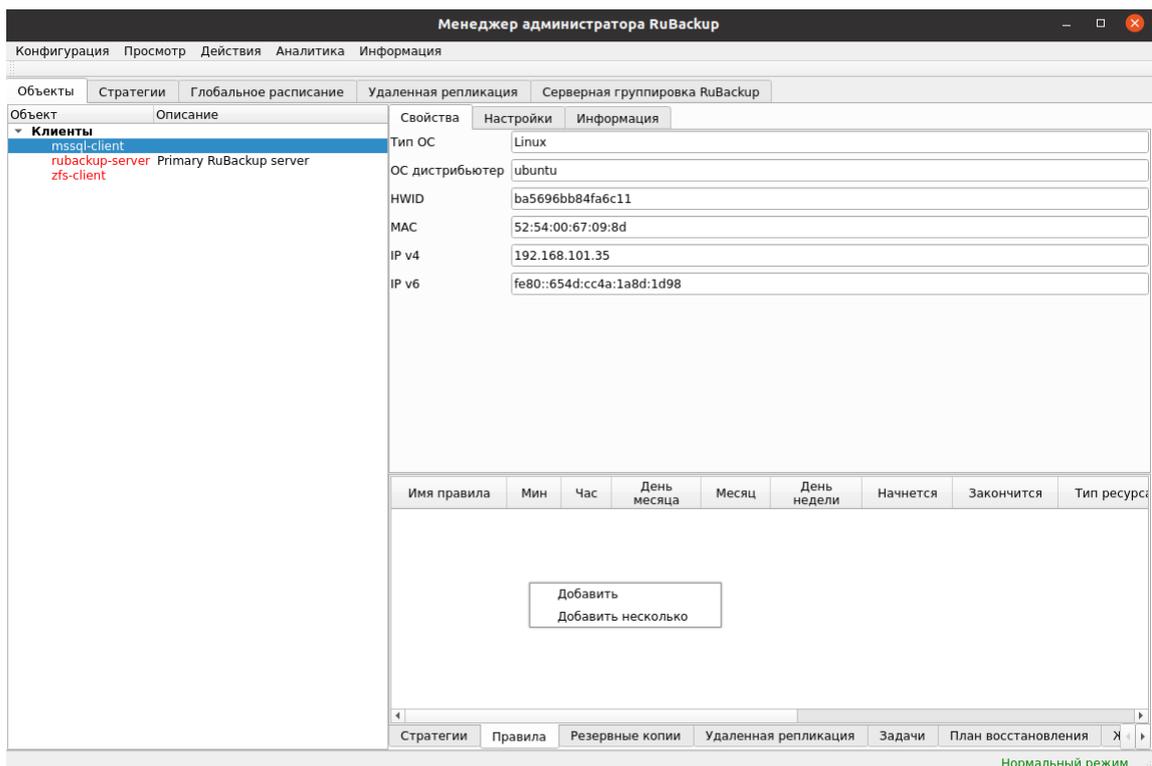


Рисунок 5

- Выбрать тип ресурса «**Microsoft SQL Server**» для выполнения резервного копирования баз данных или «**Microsoft SQL Server (Files and Filegroups)**» для резервного копирования файлов баз данных(рисунок 6):

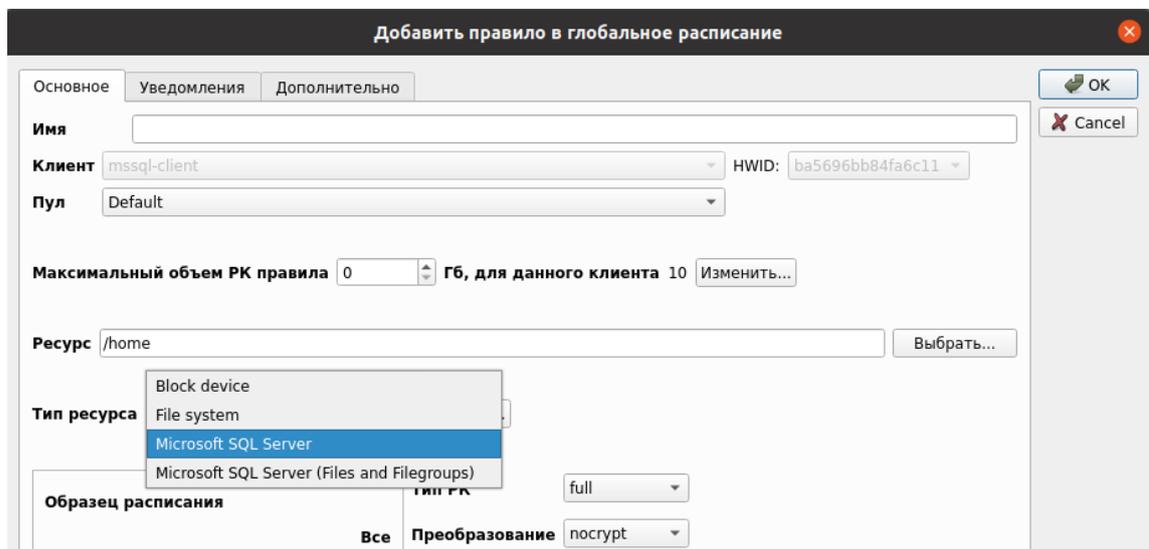


Рисунок 6

- Напротив поля **Ресурс** нажмите на кнопку «Выбрать...» и выберите одну из баз данных(рисунок 7):

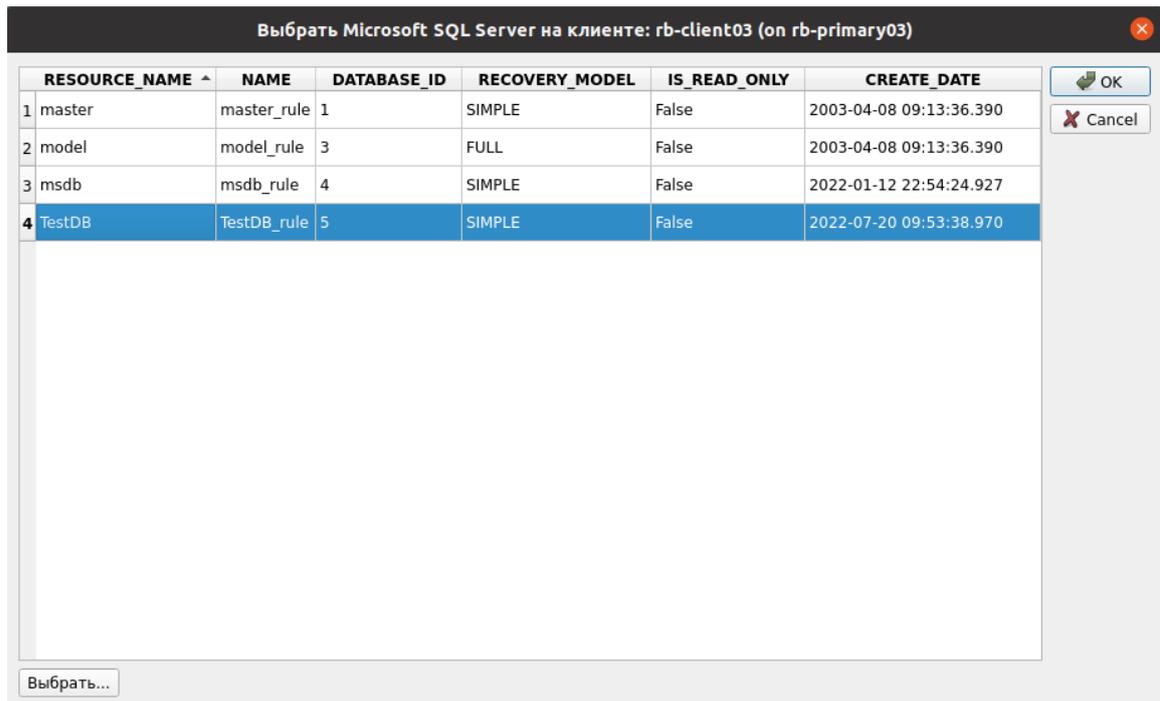


Рисунок 7

В окне выбора ресурса представлены следующие колонки:

- имя базы данных;
- имя правила. Данное значение будет автоматически применено для имени правила если оно не заполнено;
- модель восстановления базы данных;
- аргумент определяющий разрешение базы данных (только для чтения);
- дата и время создания базы данных.

Для модулей `rb_module_mssql` и `rb_module_files` запрещено копирование БД `<tempdb>`, т. к. данная процедура запрещена самим SQL Server. Также для модуля `rb_modules_mssql_files` запрещено выполнение копирования БД `<master>`, поскольку данная БД не поддерживает копирование своих файлов.

Если, при создании правила, поле **Имя** не было заполнено пользователем, то оно автоматически будет определено информацией из колонки NAME после того, как вы нажмёте кнопку ОК в текущем окне.

4. Установите настройки правила: имя правила, пул хранения данных, максимальный объём для резервных копий правила в ГБ (изначально не ограничен и задаётся в «Глобальных настройках» СРК), тип резервного копирования, расписание резервного копирования, срок хранения,

необязательный временной промежуток проверки резервной копии, и автозапуск правила (рисунок 8).

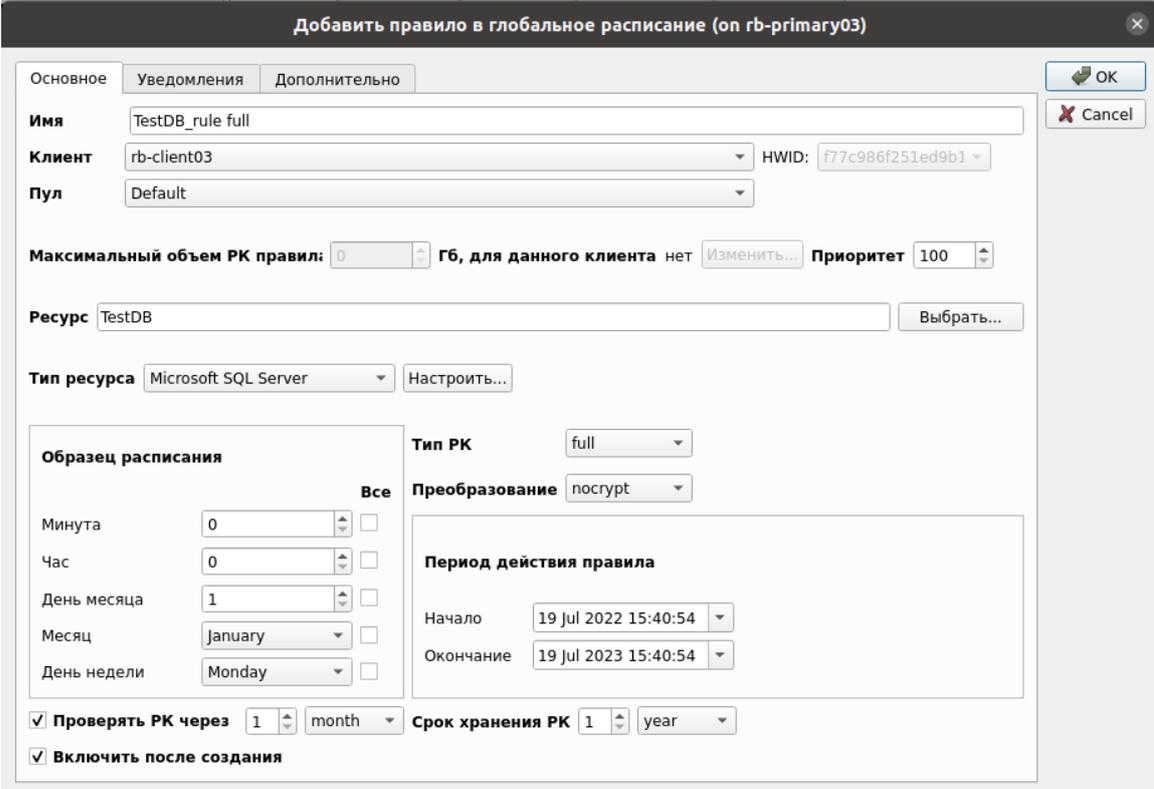


Рисунок 8

5. На вкладке «Дополнительно» можно настроить автоматическое удаление устаревших резервных копий, определить условие их перемещения в другой пул и установить разрешение для клиента удалять резервные копии (рисунок 9):

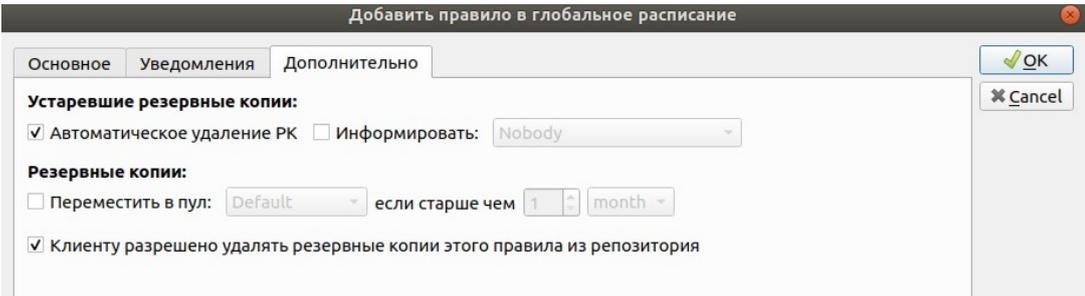


Рисунок 9

В том случае, если при создании правила был выключен чек-бокс «Включить после создания» - вновь созданное правило будет иметь статус wait. Это означает, что оно не будет порождать задач на выполнение резервного копирования, пока администратор RuBackup не запустит его (тогда его статус сменится на run). При необходимости, администратор может приостановить работу правила или немедленно запустить его (т. е. инициировать немедленное создание задачи при статусе правила wait).

6. В системе предусмотрена возможность тонкой настройки правила резервного копирования, для их редактирования нажмите на кнопку «Настроить...» (рисунок 10):

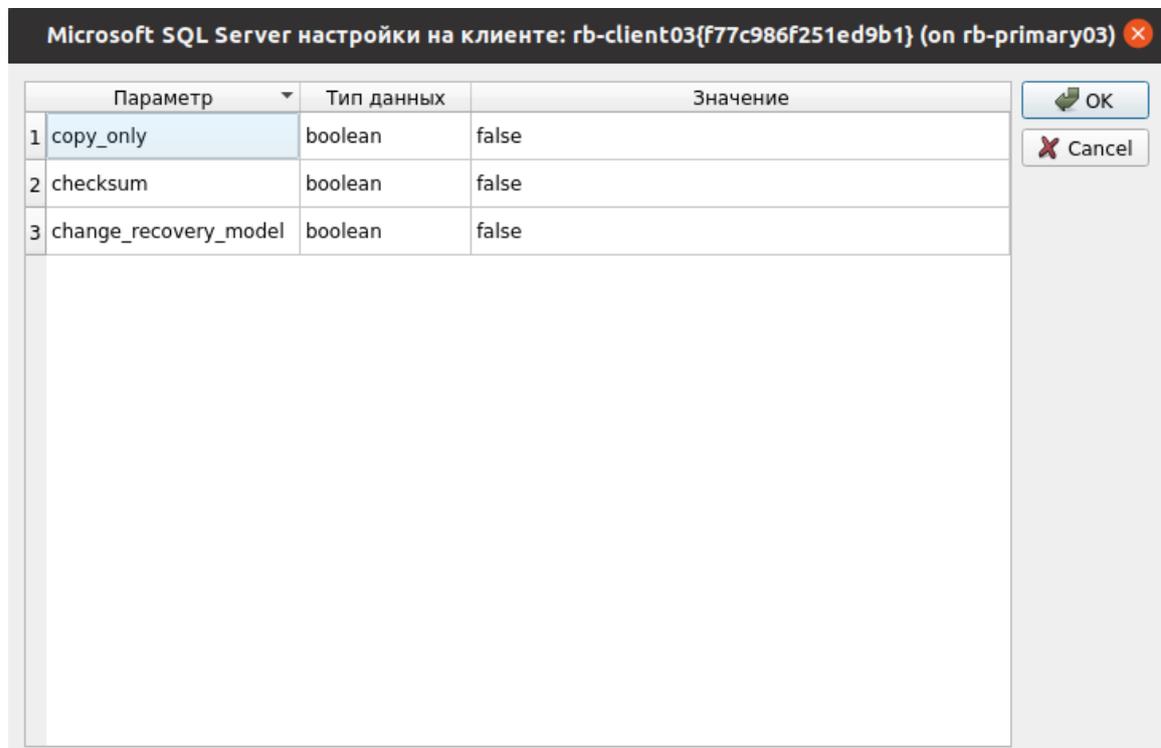


Рисунок 10

Здесь параметр `copy_only` определяет режим выполнения резервной копии (`true` - выполнить резервную копию в формате «только для копирования»). **COPY_ONLY** - Резервная копия «только для копирования» — это резервная копия SQL Server, которая не зависит от обычной последовательности создания традиционных резервных копий SQL Server. Обычно создание резервного копирования приводит к изменению базы данных и влияет на то, как будут восстанавливаться последующие резервные копии. Однако иногда приходится выполнять резервное копирование базы данных для особых нужд, когда это не сказывается на общем процессе резервного копирования и восстановления. Этой цели служат резервные копии «только для копирования».

Внимание! Если полная резервная копия была выполнена с параметром «только для копирования», то выполнение дифференциального резервного копирования на основе этой копии будет запрещено!

Параметр `checksum` - указывает, что во время операции резервного копирования выполняется проверка контрольной суммы и наличия разрывов на каждой странице (если эти проверки включены и доступны), а также будет создаваться контрольная сумма для всей резервной копии.

Внимание! Использование контрольных сумм резервных копий может повлиять на производительность рабочей нагрузки и пропускной способности резервного копирования.

Параметр `change_recovery_model` — изменяет модель восстановления с `simple` на `full`. Это требуется для тех ресурсов, где для процесса восстановления необходимо копировать журналы базы данных.

Эти настройки одинаковы для обоих типов ресурса: **Microsoft SQL Server** и **Microsoft SQL Server (Files and Filegroups)**.

Правила глобального расписания имеют срок жизни, определяемый при их создании, а также предоставляют следующие возможности:

- 1) Выполнить защитное преобразование резервной копии на клиенте.
- 2) Периодически выполнять проверку целостности резервной копии.
- 3) Хранить резервные копии определённый срок, а после его окончания удалять их из хранилища резервных копий и из записей репозитория, либо просто уведомлять пользователей системы резервного копирования об окончании срока хранения.
- 4) Через определённый срок после создания резервной копии автоматически переместить её на другой пул хранения резервных копий, например на картридж ленточной библиотеки.
- 5) Уведомлять пользователей системы резервного копирования о результатах выполнения тех или иных операций, связанных с правилом глобального расписания.

При создании задачи RuBackup она появляется в главной очереди задач. Отслеживать исполнение правил может как администратор, с помощью RBM, так и клиент при помощи RBC.

После успешного завершения резервного копирования резервная копия будет размещена в хранилище резервных копий, а информация о ней будет размещена в репозитории RuBackup.

Менеджер клиента RuBackup (RBC)

Принцип взаимодействия клиентского менеджера (RBC) с системой резервного копирования состоит в том, что пользователь может сформировать ту или иную команду (желаемое действие) и отправить его серверу резервного копирования RuBackup. Взаимодействие пользователя с сервером резервного копирования производится через клиента (фоновый процесс) резервного копирования. Клиентский менеджер отправляет команду пользователя клиенту, клиент отправляет её серверу. В том случае, если действие допустимо, то сервер RuBackup отдаст обратную команду клиенту и/или перенаправит её медиасерверу RuBackup для дальнейшей обработки. Это означает, что, как правило, клиентский менеджер обычно не ожидает завершения того или иного действия, но ожидает ответа от клиента, что задание принято. Это позволяет инициировать параллельные запросы клиента к серверу резервного копирования, но требует от пользователя самостоятельно контролировать чтобы не было «встречных» операций, когда происходит восстановление данных, и в этот же момент эти же данные требуются для создания новой резервной копии. После того, как клиент отдал какую-либо команду при помощи RBC, он может просто закрыть приложение, все действия будут выполнены системой резервного копирования (тем не менее, стоит дождаться сообщения о том, что задание принято к исполнению, и проконтролировать это на вкладке «Задачи»).

Графический интерфейс клиентского менеджера поддерживает русский и английский языки.

Для запуска RBC следует выполнить команды:

```
# ssh X user@mssql-host
```

```
# /opt/rubackup/bin/rbc&
```

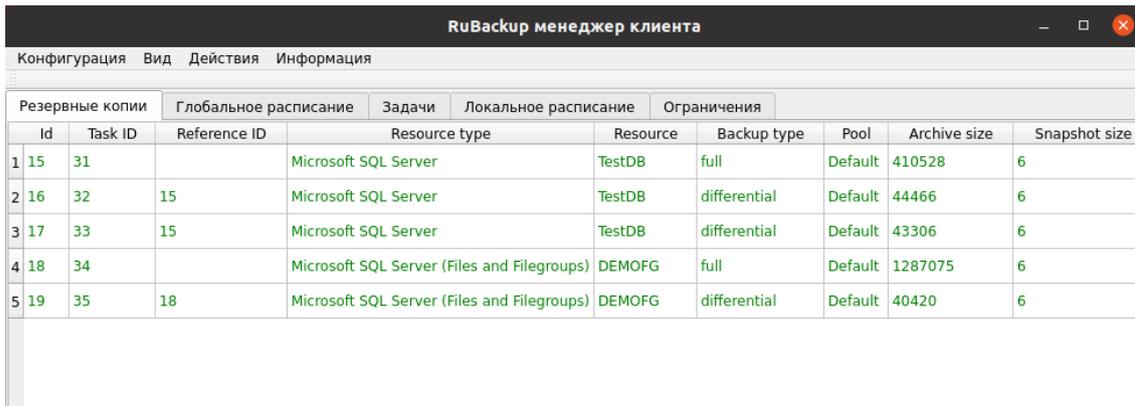
Пользователь, запускающий RBC, должен входить в группу rubackup.

При первом запуске клиентского менеджера необходимо задать пароль, при помощи которого впоследствии можно будет запросить восстановление резервной копии. Без ввода пароля получить резервную копию для клиента из хранилища невозможно. Хэш пароля восстановления хранится в базе данных RuBackup сервера. При необходимости можно изменить пароль при помощи клиентского менеджера (меню «**Конфигурация**» → «**Изменить пароль**»).

Главная страница RBC содержит переключающиеся вкладки, позволяющие управлять резервными копиями, расписанием резервного копирования, а также просматривать текущие задачи клиента, локальное расписание и ограничения.

Вкладка «Резервные копии»

В таблице вкладки «Резервные копии» содержится информация обо всех резервных копиях клиента, которые хранятся в репозитории RuBackup (рисунок 11). Дифференциальные резервные копии ссылаются на полные резервные копии, инкрементальные резервные копии ссылаются на полные резервные копии или предыдущие инкрементальные, так что при необходимости восстановить данные можно одной командой инициировать восстановление всей цепочки резервных копий.



RuBackup менеджер клиента									
Конфигурация Вид Действия Информация									
Резервные копии		Глобальное расписание	Задачи	Локальное расписание	Ограничения				
Id	Task ID	Reference ID	Resource type	Resource	Backup type	Pool	Archive size	Snapshot size	
1	15	31	Microsoft SQL Server	TestDB	full	Default	410528	6	
2	16	32	15	Microsoft SQL Server	TestDB	differential	Default	44466 6	
3	17	33	15	Microsoft SQL Server	TestDB	differential	Default	43306 6	
4	18	34	Microsoft SQL Server (Files and Filegroups)	DEMOFG	full	Default	1287075 6		
5	19	35	18	Microsoft SQL Server (Files and Filegroups)	DEMOFG	differential	Default	40420 6	

Рисунок 11

Во вкладке «Резервные копии» пользователю доступны следующие действия:

Удалить выбранную резервную копию.

Это действие возможно в том случае, если в правиле глобального расписания есть соответствующее разрешение. Кроме того, при необходимости выполнить удаление резервной копии потребуется ввести пароль клиента.

Восстановить цепочку резервных копий.

Это действие запускает процесс восстановления цепочки резервных копий на локальной файловой системе клиента.

При восстановлении резервной копии или цепочки резервных копий клиент должен выбрать место для восстановления файлов резервной копии. Рекомендуется использовать временный каталог для операций с резервными копиями (например, /rubackup-tmp).

Если в файле /opt/rubackup/etc/rb_module_mssql.conf параметр **direct_restore** имеет значение **yes**, то произойдет восстановление резервной копии непосредственно в ресурс. Если в файле /opt/rubackup/etc/rb_module_mssql.conf параметр **direct_restore** имеет значение **no**, то восстановленные резервные копии будут расположены в выбранном для восстановления каталоге и далее вы сможете провести восстановление базы данных в ручном режиме.

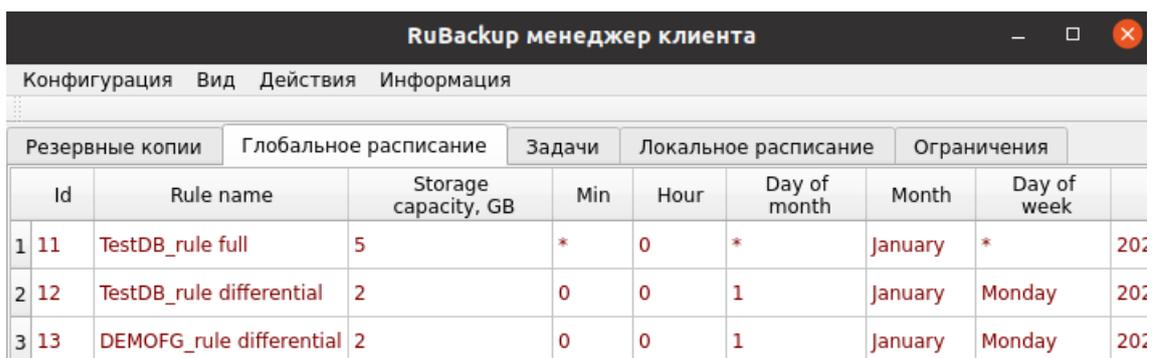
RBC не ожидает окончания восстановления всех резервных копий. Клиент должен проконтролировать на вкладке «Задачи» успешное завершение созданных задач на восстановление данных завершились успешно (статус задач Done). Для успешного выполнения этого действия требуется наличие достаточного свободного места в каталоге, предназначенном для создания и временного хранения резервных копий.

Проверить резервную копию.

Это действие инициирует создание задачи проверки резервной копии. В том случае, если резервная копия была подписана цифровой подписью, то будут проверены размер файлов резервной копии, md5 сумма и проверена сама резервная копия. Если резервная копия не была подписана цифровой подписью, то будут проверены размер файлов резервной копии и md5 сумма.

Вкладка «Глобальное расписание»

В таблице вкладки «Глобальное расписание» содержится информация обо всех правилах в глобальном расписании RuBackup для этого клиента. (рисунок 12).



RuBackup менеджер клиента									
Конфигурация Вид Действия Информация									
Резервные копии Глобальное расписание Задачи Локальное расписание Ограничения									
	Id	Rule name	Storage capacity, GB	Min	Hour	Day of month	Month	Day of week	
1	11	TestDB_rule full	5	*	0	*	January	*	202
2	12	TestDB_rule differential	2	0	0	1	January	Monday	202
3	13	DEMOFG_rule differential	2	0	0	1	January	Monday	202

Рисунок 12

Во вкладке «Глобальное расписание» пользователю доступны следующие действия:

Запросить новое правило.

Это действие вызывает диалог подготовки нового правила в глобальном расписании RuBackup для данного клиента. Запрос на добавление правила требует одобрения администратора RuBackup, одобрение может быть сделано в оконном менеджере администратора RuBackup.

Запросить удалить правило из глобального расписания.

Это действие формирует запрос к администратору RuBackup об удалении выбранного пользователем правила из глобального расписания RuBackup. Запрос на удаление правила требует одобрения администратора RuBackup, одобрение может быть сделано в оконном менеджере администратора RuBackup.

Вкладка «Задачи»

В таблице вкладки «Задачи» содержится информация обо всех задачах в главной очереди заданий RuBackup для этого клиента (рисунок 13). В зависимости от настроек резервного сервера RuBackup выполненные задачи и задачи, завершившиеся неудачно, через какое-то время могут быть автоматически удалены из главной очереди задач. Информация о выполнении заданий фиксируется в специальном журнале задач сервера RuBackup, при необходимости статус любой задачи, даже удалённой из очереди, можно уточнить у администратора RuBackup. Так же информация о выполнении задач клиента заносится в локальный журнальный файл на клиенте. В клиентском менеджере можно открыть окно отслеживания журнального файла (меню «Информация» → «Журнальный файл»).



RuBackup менеджер клиента											
Конфигурация Вид Действия Информация											
Резервные копии		Глобальное расписание		Задачи		Локальное расписание		Ограничения			
ID	Type	Resource type	Resource	Backup type	Rule ID	Strategy ID	Repository ID	Pool	Status		
1 31	Backup global	Microsoft SQL Server	TestDB	full	11	0	15	Default	Done	202	
2 32	Backup global	Microsoft SQL Server	TestDB	differential	12	0	16	Default	Done	202	
3 33	Backup global	Microsoft SQL Server	TestDB	differential	12	0	17	Default	Done	202	
4 34	Backup global	Microsoft SQL Server (Files and Filegroups)	DEMOFG	differential	13	0	18	Default	Done	202	
5 35	Backup global	Microsoft SQL Server (Files and Filegroups)	DEMOFG	differential	13	0	19	Default	Done	202	

Рисунок 13

Вкладка «Локальное расписание»

Во вкладке «Локальное расписание» можно определить правила, задаваемые клиентом для тех или иных локальных ресурсов. Для работы локального расписания эта возможность должна быть включена администратором RuBackup для клиента.

Вкладка «Ограничения»

Во вкладке «Ограничения» могут быть определены локальные ресурсы, резервное копирование которых нежелательно. Для работы локальных ограничений эта возможность должна быть включена администратором RuBackup для клиента.

Утилиты командной строки клиента

RuBackup

Для управления RuBackup со стороны клиента, помимо клиентского оконного менеджера, можно воспользоваться утилитами командной строки:

rb_archives

Утилита предназначена для просмотра списка резервных копий клиента в системе резервного копирования, создания срочных резервных копий, их удаления, проверки и восстановления.

rb_archives

```
root@mssql-client:~# rb_archives
```

Id	Ref ID	Resource	Resource type	Backup type	Created	Crypto	Signed	Status
15		TestDB	Microsoft SQL Server	full	2021-10-22 21:23:06+03	nocrypt	True	Trusted
16	15	TestDB	Microsoft SQL Server	differential	2021-10-22 21:23:09+03	nocrypt	True	Trusted
17	15	TestDB	Microsoft SQL Server	differential	2021-10-22 21:23:18+03	nocrypt	True	Trusted
18		DEMOFG	Microsoft SQL Server (Files and Filegroups)	full	2021-10-22 21:23:24+03	nocrypt	True	Trusted
19	18	DEMOFG	Microsoft SQL Server (Files and Filegroups)	differential	2021-10-22 21:23:27+03	nocrypt	True	Trusted

rb_schedule

Утилита предназначена для просмотра имеющихся правил клиента в глобальном расписании резервного копирования.

#rb_schedule

```
root@mssql-client:~# rb_schedule
```

Id	Name	Resource type	Resource	Backup type	Status
11	TestDB_rule full	Microsoft SQL Server	TestDB	full	wait
12	TestDB_rule differential	Microsoft SQL Server	TestDB	differential	wait
13	DEMOFG_rule differential	Microsoft SQL Server (Files and Filegroups)	DEMOFG	differential	wait

rb_tasks

Утилита предназначена для просмотра задач клиента, которые присутствуют в главной очереди задач системы резервного копирования.

#rb_tasks

```
root@mssql-client:~# rb_tasks
```

Id	Task type	Resource	Backup type	Status	Created
31	Backup global	TestDB	full	Done	2021-10-22 21:23:03+03
32	Backup global	TestDB	differential	Done	2021-10-22 21:23:07+03
33	Backup global	TestDB	differential	Done	2021-10-22 21:23:16+03
34	Backup global	DEMOFG	differential	Done	2021-10-22 21:23:20+03
35	Backup global	DEMOFG	differential	Done	2021-10-22 21:23:25+03

Восстановление резервной копии

РСУБД Microsoft SQL Server

Ход восстановления резервной копии РСУБД Microsoft SQL Server зависит от значения параметра `direct_restore` в файле конфигурации модуля резервного копирования `/opt/rubackup/etc/rb_module_mssql.conf`.

Если параметр `direct_restore` имеет значение `yes`, то произойдёт восстановление резервной копии непосредственно в ресурс.

Если в файле `/opt/rubackup/etc/rb_module_mssql.conf` параметр `direct_restore` имеет значение `no`, то восстановленные резервные копии будут расположены в выбранном для восстановления каталоге, и восстановление СУБД можно будет провести вручную.

Клиент может осуществить восстановление данных резервной копии в оконном Менеджере Клиента RuBackup (RBC), либо при помощи утилиты командной строки `rb_archives`.

В случае восстановления дифференциальной резервной копии будет сформирована цепочка восстановления: вначале будет восстановлена полная резервная копия, на которую будут наложены изменения из дифференциальной резервной копии.

Восстановление резервной копии в RBC

Для восстановления данных резервной копии в оконном Менеджере Клиента RuBackup (RBC) необходимо выполнить следующие действия:

1. Выделить нужную резервную копию и в контекстном меню выбрать «Восстановить».
2. Ввести пароль клиента и далее RBC выведет информационное сообщение о дальнейших действиях.
3. Указать в качестве временного места восстановления резервных копий каталог, отдельный от копируемого каталога кластера баз данных (`/var/lib/pgpro/std-13/data/`).
4. RBC выведет информационное сообщение о создании задачи на восстановление.

5. Проконтролировать результат процесса восстановления можно после автоматического переключения RBC на вкладку «Задачи» (рисунок 14):

RuBackup менеджер клиента											
Конфигурация Вид Действия Информация											
Резервные копии Глобальное расписание Задачи Локальное расписание Ограничения											
	Id	Type	Resource type	Resource	Backup type	Rule ID	Strategy ID	Repository ID	Pool	Status	Created
1	31	Backup global	Microsoft SQL Server	TestDB	full	11		15	Default	Done	2021-10-22 ...
2	32	Backup global	Microsoft SQL Server	TestDB	differential	12		16	Default	Done	2021-10-22 ...
3	33	Backup global	Microsoft SQL Server	TestDB	differential	12		17	Default	Done	2021-10-22 ...
4	34	Backup global	Microsoft SQL Server (Files and Filegroups)	DEMOFG	differential	13		18	Default	Done	2021-10-22 ...
5	35	Backup global	Microsoft SQL Server (Files and Filegroups)	DEMOFG	differential	13		19	Default	Done	2021-10-22 ...
6	41	Restore	Microsoft SQL Server	TestDB	full			15	Default	Done	2021-10-22 ...
7	42	Restore	Microsoft SQL Server	TestDB	differential			16	Default	Done	2021-10-22 ...

Рисунок 14

Восстановление при помощи утилиты rb_archives

Для восстановления резервных копий клиент может использовать утилиту командной строки rb_archives. Вызов следующий:

```
# rb_archives
```

```
root@mssql-client:~# rb_archives
Id | Ref ID | Resource | Resource type | Backup type | Create
-----+-----+-----+-----+-----+-----
15 |  | TestDB | Microsoft SQL Server | full | 2021-1
16 | 15 | TestDB | Microsoft SQL Server | differential | 2021-1
17 | 15 | TestDB | Microsoft SQL Server | differential | 2021-1
18 |  | DEMOFG | Microsoft SQL Server (Files and Filegroups) | full | 2021-1
19 | 18 | DEMOFG | Microsoft SQL Server (Files and Filegroups) | differential | 2021-1
```

В приведённом примере в системе резервного копирования присутствуют пять резервных копий с идентификаторами 15, 16, 17, 18 и 19. Для восстановления резервной копии 17 необходимо выполнить команду:

```
# rb_archives -x 17
```

```
root@mssql-client:~# rb_archives -x 17
Password:
----> Restore archive chain: 15 17 < ----
Record ID: 15 has status: Trusted
Record ID: 17 has status: Trusted
[RBC] Request to restore next archive(s) ID from repository: 15 17 to: /root
TASK WAS ADDED TO QUEUE:43 44
```

В случае успешно принятой задачи команда вернёт список созданных задач, а восстановление будет происходить в фоновом режиме.

Проконтролировать процесс восстановления можно при помощи утилиты rb_tasks:

```
#rb_tasks
```

```
root@mssql-client:~# rb_tasks
Id | Task type | Resource | Backup type | Status | Created
-----+-----+-----+-----+-----+-----
31 | Backup global | TestDB | full | Done | 2021-10-22 21:23:03+03
32 | Backup global | TestDB | differential | Done | 2021-10-22 21:23:07+03
33 | Backup global | TestDB | differential | Done | 2021-10-22 21:23:16+03
34 | Backup global | DEMOFG | differential | Done | 2021-10-22 21:23:20+03
35 | Backup global | DEMOFG | differential | Done | 2021-10-22 21:23:25+03
41 | Restore | TestDB | full | Done | 2021-10-22 21:42:06+03
42 | Restore | TestDB | differential | Done | 2021-10-22 21:42:06+03
43 | Restore | TestDB | full | Done | 2021-10-22 21:45:25+03
44 | Restore | TestDB | differential | Done | 2021-10-22 21:45:25+03
```

Вы можете проконтролировать процесс восстановления в файле журнала при помощи вызова:

```
# tail -f /opt/rubackup/log/RuBackup.log
```

```
root@mssql-client:~# tail -f /opt/rubackup/log/RuBackup.log
Fri Oct 22 21:45:27 2021: RuBackup server commands: Run task ID: 44 Resource type: 41 Module: 'Microsoft SQL Server' Resource: 'TestDB' Media server: rubackup-server
Fri Oct 22 21:45:27 2021: Set unlimited bandwidth for task ID: 44
Fri Oct 22 21:45:27 2021: Create a file: /root/mssql-client_TaskID_33_RuleID_12_D2021_10_22H21_23_17_BackupType_3_ResourceType_41.tar.gz
Fri Oct 22 21:45:28 2021: md5sum of transferred file is ok: 18dab7c579be80b520c9daa8bbc33207
Fri Oct 22 21:45:28 2021: Transfer file is succeeded: /root/mssql-client_TaskID_33_RuleID_12_D2021_10_22H21_23_17_BackupType_3_ResourceType_41.tar.gz
Fri Oct 22 21:45:28 2021: Execute OS command: /opt/rubackup/modules/rb_module_mssql -r /root/mssql-client_TaskID_33_RuleID_12_D2021_10_22H21_23_17_BackupType_3_ResourceType_41.tar.gz -z 1 -e last:true,tnp_catalog:/rubackup-tnp,rbd_hash_algorithm:sha2,rbd_hash_length:256,rbd_block_size:10384,granular_restore:no,without_deployment_restore:no,resource_name:testdb,copy_only:r,checksum:-d /root -s mssql-client_TaskID_31_RuleID_11_D2021_10_22H21_23_04_BackupType_1_ResourceType_41 2>&1
Fri Oct 22 21:45:28 2021: mssql-client_TaskID_33_RuleID_12_D2021_10_22H21_23_17_BackupType_3_ResourceType_41_bak
Fri Oct 22 21:45:28 2021: mssql-client_TaskID_33_RuleID_12_D2021_10_22H21_23_17_BackupType_3_ResourceType_41.log_bak
Fri Oct 22 21:45:28 2021: Remove obsoleted: /root/mssql-client_TaskID_33_RuleID_12_D2021_10_22H21_23_17_BackupType_3_ResourceType_41.tar.gz
Fri Oct 22 21:45:28 2021: Task was done. ID: 44
```

Восстановление системной базы данных 'master'

Для того, чтобы восстановить системную базу данных 'master' необходимо выполнить следующие действия:

1. Остановите работу SQL Server и убедитесь в том, что сервер остановлен при помощи следующих команд:

```
# sudo systemctl stop mssql-server.service  
# systemctl status mssql-server.service
```

```
root@mssql-client:~# systemctl stop mssql-server.service  
root@mssql-client:~# systemctl status mssql-server.service  
● mssql-server.service - Microsoft SQL Server Database Engine  
   Loaded: loaded (/lib/systemd/system/mssql-server.service; enabled; vendor preset: enabled)  
   Active: inactive (dead) since Wed 2022-05-11 18:56:38 MSK; 11s ago  
     Docs: https://docs.microsoft.com/en-us/sql/linux  
   Process: 661 ExecStart=/opt/mssql/bin/sqlservr (code=exited, status=0/SUCCESS)  
   Main PID: 661 (code=exited, status=0/SUCCESS)  
  
мая 11 18:41:51 mssql-client sqlservr[751]: [195B blob data]  
мая 11 18:56:34 mssql-client systemd[1]: Stopping Microsoft SQL Server Database Engine...  
мая 11 18:56:34 mssql-client sqlservr[751]: [200B blob data]  
мая 11 18:56:34 mssql-client sqlservr[751]: [190B blob data]  
мая 11 18:56:35 mssql-client sqlservr[751]: [73B blob data]  
мая 11 18:56:35 mssql-client sqlservr[751]: [76B blob data]  
мая 11 18:56:38 mssql-client sqlservr[751]: [164B blob data]  
мая 11 18:56:38 mssql-client sqlservr[751]:  
мая 11 18:56:38 mssql-client systemd[1]: mssql-server.service: Succeeded.  
мая 11 18:56:38 mssql-client systemd[1]: Stopped Microsoft SQL Server Database Engine.  
root@mssql-client:~#
```

2. Далее выполнение команд необходимо производить от имени пользователя 'mssql':

```
# sudo su mssql -
```

3. Запустите SQL Server в однопользовательском режиме:

```
# /opt/mssql/bin/sqlservr -m
```

```
root@mssql-client:~# sudo su mssql -  
mssql@mssql-client:/root$ /opt/mssql/bin/sqlservr -m  
2022-05-11 19:00:52.38 Server Setup step is FORCE copying system data file 'C:\templatedata\model_replicatedmaster.ndf' to '/var/opt/mssql/data/model_replicatedmaster.ndf'.  
2022-05-11 19:00:52.44 Server Setup step is FORCE copying system data file 'C:\templatedata\model_replicatedmaster.ldf' to '/var/opt/mssql/data/model_replicatedmaster.ldf'.  
2022-05-11 19:00:52.47 Server Setup step is FORCE copying system data file 'C:\templatedata\model_msdbdata.ndf' to '/var/opt/mssql/data/model_msdbdata.ndf'.  
2022-05-11 19:00:52.53 Server Setup step is FORCE copying system data file 'C:\templatedata\model_msdblog.ldf' to '/var/opt/mssql/data/model_msdblog.ldf'.  
2022-05-11 19:00:52.79 Server Microsoft SQL Server 2019 (RTM-CU15) (KB5008996) - 15.0.4198.2 (X64)  
Jan 12 2022 22:30:08  
Copyright (c) 2019 Microsoft Corporation
```

4. После проделанных операций запустите восстановление резервной копии. Это можно сделать через менеджер клиента rbc или при помощи консольной

утилиты `rb_archives`. Команда на восстановление системной базы данных 'master':

```
# rb_archives -x 43 -d /rubackup-tmp
```

где 43 – id резервной копии базы данных master в репозитории,

`/rubackup-tmp` – директория восстановления.

```
root@mssql-client:~# rb_archives -x 43 -d /rubackup-tmp/  
Password:  
The archive will be restored in the directory: /rubackup-tmp/  
----> Restore archive chain: 43 < ----  
Record ID: 43 has status: Trusted  
TASK WAS ADDED TO QUEUE:200
```

```
root@mssql-client:~# [RBC] Request to restore next archive(s) ID from repository: 43 to: /rubackup-tmp/  
RuBackup server commands: Run task ID: 200 Resource type: 21 Module: 'Microsoft SQL Server' Resource: '  
Set status for task ID: 200 from: Assigned to: At_Client  
Set status for task ID: 200 from: At_Client to: Start_Transfer  
Restore system module name: mssql  
Set unlimited bandwidth for task ID: 200  
Create a file: /rubackup-tmp/mssql-client_TaskID_197_RuleID_9_D2022_5_11H19_03_15_BackupType_1_Resource  
*****  
Wait for change status of task ID: 200 from: Start_Transfer to Transmission  
Set status for task ID: 200 from: Start_Transfer to: Transmission  
Set status for task ID: 200 from: Transmission to: Finish_Transfer  
md5sum of transferred file is ok: 51d45ee8b69d03725842f637313444d3  
Transfer file is succeeded: /rubackup-tmp/mssql-client_TaskID_197_RuleID_9_D2022_5_11H19_03_15_BackupType  
Execute OS command: /opt/rubackup/modules/rb_module_mssql -r /rubackup-tmp/mssql-client_TaskID_197_Rule  
talog:/rubackup-tmp,rbd_hash_algorithm:sha2,rbd_hash_length:256,rbd_block_size:16384,granular_restore:n  
backup-tmp/ 2>&1  
Set status for task ID: 200 from: Finish_Transfer to: Execution  
mssql-client_TaskID_197_RuleID_9_D2022_5_11H19_03_15_BackupType_1_ResourceType_21.bak  
Remove obsoleted: /rubackup-tmp/mssql-client_TaskID_197_RuleID_9_D2022_5_11H19_03_15_BackupType_1_Resou  
Service 'mssql-server.service' is inactive.  
Five seconds delay  
10 percent processed.  
20 percent processed.  
30 percent processed.  
41 percent processed.  
51 percent processed.  
61 percent processed.  
70 percent processed.  
80 percent processed.  
90 percent processed.  
100 percent processed.  
Processed 480 pages for database 'master', file 'master' on file 1.  
Processed 2 pages for database 'master', file 'mastlog' on file 1.  
The master database has been successfully restored. Shutting down SQL Server.  
SQL Server is terminating this process.  
Set status for task ID: 200 from: Execution to: Done  
Task was done. ID: 200
```

```
root@mssql-client:~# [RBC] Request to restore next archive(s) ID from repository: 43 to: /rubackup-tmp/
RuBackup server commands: Run task ID: 200 Resource type: 21 Module: 'Microsoft SQL Server' Resource: '
Set status for task ID: 200 from: Assigned to: At_Client
Set status for task ID: 200 from: At_Client to: Start_Transfer
Restore system module name: mssql
Set unlimited bandwidth for task ID: 200
Create a file: /rubackup-tmp/mssql-client_TaskID_197_RuleID_9_D2022_5_11H19_03_15_BackupType_1_Resource
*****
Wait for change status of task ID: 200 from: Start_Transfer to Transmission
Set status for task ID: 200 from: Start_Transfer to: Transmission
Set status for task ID: 200 from: Transmission to: Finish_Transfer
md5sum of transferred file is ok: 51d45ee8b69d03725842f637313444d3
Transfer file is succesd: /rubackup-tmp/mssql-client_TaskID_197_RuleID_9_D2022_5_11H19_03_15_BackupType
Execute OS command: /opt/rubackup/modules/rb_module_mssql -r /rubackup-tmp/mssql-client_TaskID_197_Rule
talog:/rubackup-tmp,rbd_hash_algorithm:sha2,rbd_hash_length:256,rbd_block_size:16384,granular_restore:n
backup-tmp/ 2>&1
Set status for task ID: 200 from: Finish_Transfer to: Execution
mssql-client_TaskID_197_RuleID_9_D2022_5_11H19_03_15_BackupType_1_ResourceType_21.bak
Remove obsoleted: /rubackup-tmp/mssql-client_TaskID_197_RuleID_9_D2022_5_11H19_03_15_BackupType_1_Resou
Service 'mssql-server.service' is inactive.
Five seconds delay
10 percent processed.
20 percent processed.
30 percent processed.
41 percent processed.
51 percent processed.
61 percent processed.
70 percent processed.
80 percent processed.
90 percent processed.
100 percent processed.
Processed 480 pages for database 'master', file 'master' on file 1.
Processed 2 pages for database 'mastlog', file 'mastlog' on file 1.
The master database has been successfullly restored. Shutting down SQL Server.
SQL Server is terminating this process.
Set status for task ID: 200 from: Execution to: Done
Task was done. ID: 200
```

5. Когда восстановление завершится, запущенный ранее SQL Server в однопользовательском режиме самостоятельно завершит работу. Далее, вернитесь к первоначальному пользователю и запустите сервер. Затем проверьте, что он действительно запустился без ошибок при помощи следующих команд:

```
# sudo systemctl start mssql-server.service
```

```
# systemctl status mssql-server.service
```

```
mssql@mssql-client:/root$ exit
root@mssql-client:~#
root@mssql-client:~# systemctl start mssql-server.service
root@mssql-client:~# systemctl status mssql-server.service
● mssql-server.service - Microsoft SQL Server Database Engine
   Loaded: loaded (/lib/systemd/system/mssql-server.service; enabled; vendor
   Active: active (running) since Wed 2022-05-11 19:06:29 MSK; 5min ago
     Docs: https://docs.microsoft.com/en-us/sql/linux
   Main PID: 2067 (sqlservr)
     Tasks: 157
    Memory: 658.2M
    CGroup: /system.slice/mssql-server.service
            └─2067 /opt/mssql/bin/sqlservr
              └─2069 /opt/mssql/bin/sqlservr

мая 11 19:06:33 mssql-client sqlservr[2069]: [155B blob data]
мая 11 19:06:33 mssql-client sqlservr[2069]: [159B blob data]
мая 11 19:06:33 mssql-client sqlservr[2069]: [61B blob data]
мая 11 19:06:34 mssql-client sqlservr[2069]: [96B blob data]
мая 11 19:06:34 mssql-client sqlservr[2069]: [66B blob data]
мая 11 19:06:34 mssql-client sqlservr[2069]: [75B blob data]
мая 11 19:06:34 mssql-client sqlservr[2069]: [96B blob data]
мая 11 19:06:34 mssql-client sqlservr[2069]: [100B blob data]
мая 11 19:06:34 mssql-client sqlservr[2069]: [71B blob data]
мая 11 19:06:34 mssql-client sqlservr[2069]: [124B blob data]
```