

RuBackup

Система резервного копирования и восстановления данных

Резервное копирование и восстановление СУБД MySQL 5.7



RuBackup

Версия 1.9

2022 г.

Содержание

Введение.....	3
Подготовка сервера с СУБД MySQL 5.7 к резервному копированию с помощью RuBackup.....	4
Создание пользователя СУБД для безопасного выполнения резервного копирования MySQL.....	6
Принципы выполнения резервного копирования MySQL 5.7.....	7
Установка клиента RuBackup.....	8
Подготовка хоста с СУБД MySQL для выполнения резервного копирования средствами RuBackup.....	9
Защитное преобразование резервных копий.....	10
Локальный лист ограничений.....	12
Использование менеджера администратора RuBackup.....	13
Настройки правил глобального расписания RuBackup.....	18
Использование клиентского менеджера RuBackup.....	20
Утилиты командной строки клиента RuBackup.....	24
Восстановление MySQL 5.7 из резервных копий.....	25

Введение

Система резервного копирования RuBackup позволяет выполнять клиентам полное и инкрементальное резервное копирование СУБД MySQL 5.7 в ОС Linux.

Полное резервное копирование – это создание резервной копии всех данных из исходного набора, независимо от того, изменялись данные или нет с момента выполнения последней полной резервной копии.

Дифференциальное резервное копирование сохраняет только данные, изменённые со времени выполнения предыдущего полного резервного копирования.

Инкрементальное резервное копирование сохраняет только данные, изменённые со времени выполнения предыдущей инкрементальной резервной копии, а если такой нет, то со времени выполнения последней полной резервной копии.

Для возможности выполнения резервного копирования MySQL 5.7 на клиенте должен быть установлен модуль RuBackup:

```
- rb_module_mysql57.
```

Работа модуля на клиенте возможна только в том случае, если на нем установлены пакеты MySQL и Percona XtraBackup (подробнее см. <https://www.percona.com/software/mysql-database/percona-xtrabackup>).

Резервное копирование выполняется по заранее заданным правилам в глобальном расписании RuBackup. Так же клиенту доступно срочное резервное копирование.

Восстановление резервной копии возможно по инициативе клиента. Для восстановления данных пользователь должен ввести пароль, позволяющий выполнить восстановление.

Резервное копирование может быть выполнено с применением сжатия на стороне клиента или на стороне сервера RuBackup. При необходимости возможно выполнить защитное преобразование резервной копии выбранным алгоритмом (см. раздел «Защитное преобразование резервных копий»).

Подготовка сервера с СУБД MySQL

5.7 к резервному копированию с

помощью RuBackup

Для выполнения полного, инкрементального и дифференциального резервного копирования СУБД MySQL система резервного копирования RuBackup использует Percona XtraBackup. Percona XtraBackup – это свободно распространяемая утилита, позволяющая создать полную или инкрементную резервную копию баз данных на запущенном сервере MySQL.

Ограничения для операций резервного копирования СУБД MySQL с использованием Percona XtraBackup

См. подробную документацию

<https://www.percona.com/software/mysql-database/percona-xtrabackup>

<https://www.percona.com/doc/percona-xtrabackup/2.1/innobackupex/innobackup>

[ex_script.html](#)

1. Для работы модуля требуется Percona Xtrabackup версии не выше чем 2.4.

Т.к. модуль пытается получить доступ к `mysqladmin` под пользователем `root` без пароля, то необходимо выполнить следующие действия:

2. С помощью утилиты `mysql_config_editor` создать файл `.mylogin.cnf`, содержащий в себе учетные данные в обфусцированном виде. Для этого в терминале под пользователем `root` выполнить команду:

```
mysql_config_editor set --user=root --password
```

Примечание – Данную команду необходимо выполнять от пользователя `root`. В противном случае конфигурационный файл будет создан для текущего пользователя.

Утилита запросит ввод пароля, после чего создаст файл `/root/.mylogin.cnf` с доступом на чтение и запись только для владельца (`root`).

Для просмотра содержимого файла можно использовать команду:

```
mysql_config_editor print --all
```

Пароль в выводе будет замаскирован.

Если этого не сделать, то при запуске клиента модуль не запустится и выдаст ошибку.

Если у вас нет сервера с СУБД MySQL 5.7

В том случае, если MySQL 5.7 не установлен на вашем сервере, но вы хотите проверить как осуществляется резервное копирование СУБД при помощи RuBackup, вам потребуется установить эту СУБД. Для проверки рекомендуется использовать ОС Linux Ubuntu.

Установка СУБД осуществляется следующим образом (в случае использования других дистрибутивов Linux команды могут отличаться, см. соответствующие руководства по инсталляции MySQL в других ОС):

```
# sudo apt update  
# sudo apt install mysql-server
```

После завершения установки нужно запустить встроенный сценарий безопасности MySQL:

```
# sudo mysql_secure_installation
```

Проверить функционирование MySQL:

```
# systemctl status mysql.service
```

Установка Percona XtraBackup

```
# sudo apt-get install percona-xtrabackup
```

Создание пользователя СУБД для безопасного выполнения резервного копирования MySQL

Для создания пользователя СУБД необходимо выполнить следующие действия:

1. Открыть консоль MySQL:

```
# mysql -p -u root
```

2. Создать нового пользователя MySQL, в качестве пароля укажите желаемый пароль вместо *qwerty1234* для него при помощи команды:

```
CREATE USER rubackup_backuper@'localhost' IDENTIFIED BY 'qwerty1234';
```

3. Передать новому пользователю права на перезагрузку, блокировку и репликацию всех баз данных:

```
GRANT RELOAD, LOCK TABLES, PROCESS, REPLICATION CLIENT  
ON *.* TO 'rubackup_backuper'@'localhost';  
FLUSH PRIVILEGES;
```

Конфигурационный файл модуля `rb_module_mysql57`

В конфигурационном файле `/opt/rubackup/etc/rb_module_mysql57.conf` содержатся следующие настройки:

username – имя пользователя MySQL для безопасного резервного копирования;

password – пароль этого пользователя;

port – порт СУБД;

direct_restore – выполнять ли прямое восстановление резервной копии. При значении **yes** сервис СУБД будет выключен, файлы СУБД из *datadir* будут удалены и восстановлены на этом месте файлы из резервной копии, сервис СУБД будет включён. При значении по все резервные копии цепочки резервных копий окажутся в заданном каталоге *+ mysql_restore*;

use_memory – объем памяти, используемый Percona XtraBackup при восстановлении. Значение должно оканчиваться на М или G. 100M – 100 мегабайт, 10G – 10 гигабайт.

Принципы выполнения резервного копирования MySQL 5.7

Система резервного копирования RuBackup позволяет выполнять полное, инкрементальное или дифференциальное резервное копирование СУБД MySQL 5.7.

Получение полной резервной копии:

```
# innobackupex --user=DBUSER --password=DBUSERPASS  
/path/to/BACKUP-DIR/
```

Получение инкрементальное резервной копии:

```
# innobackupex --incremental /data/backups --incremental-lsn=1291135
```

Восстановление полной резервной копии:

```
# innobackupex --apply-log --redo-only /path/to/BACKUP-DIR  
# innobackupex --copy-back /path/to/BACKUP-DIR
```

Восстановление инкрементальной резервной копии происходит после восстановления полной резервной копии:

```
# innobackupex --apply-log --redo-only /path/to/BACKUP-DIR/  
--incremental-dir=INCREMENTAL-DIR-1  
# innobackupex --apply-log /path/to/BACKUP-DIR/  
# innobackupex --copy-back /path/to/BACKUP-DIR
```

Установка клиента RuBackup

Для возможности резервного копирования СУБД MySQL при помощи RuBackup на сервер должен быть установлен клиент RuBackup и модуль `rb_module_mysql57`. Подробно процедура установки клиента описана в «Руководстве по установке серверов резервного копирования и Linux клиентов RuBackup».

Клиент RuBackup представляет собой фоновое системное приложение (демон или сервис), обеспечивающее взаимодействие с серверной группировкой RuBackup. Для выполнения резервного копирования СУБД MySQL клиент RuBackup должен работать от имени суперпользователя (`root` для Linux и Unix).

Подготовка хоста с СУБД MySQL для выполнения резервного копирования средствами RuBackup

Каталог для создания резервных копий виртуальных машин и хранения временных файлов которые создаются при их восстановлении, требуется определённое пространство. Рекомендуется выделить для этой цели отдельный диск или устройство хранения достаточного размера и примонтировать к `/mysql-backup` (либо к иной удобной точке монтирования), во избежание переполнения системного диска. Необходимо определить этот каталог как значение параметра `use-local-backup-directory` в конфигурационном файле `/opt/rubackup/etc/config.file` и перезагрузить клиент RuBackup.

В исключительных случаях допустимо использование возможности сервера RuBackup предоставить клиенту NFS каталог для создания резервной копии. Для этого нужно определить значение параметра `nfs-share-mountpoint`, который определяет в какую точку файловой системы будет примонтирован NFS каталог. Параметр `use-local-backup-directory` в этом случае должен быть отключён, а на сервере RuBackup произведены соответствующие настройки для определения разделяемого каталога. Более подробно см. «Руководстве по установке серверов резервного копирования и Linux клиентов RuBackup» и «Руководство системного администратора RuBackup».

Защитное преобразование резервных копий

При необходимости, сразу после выполнения резервного копирования архивы 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 картриджной 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.

Если для правила глобального расписания необходимо выбрать особый режим защитного преобразования с длиной ключа, отличной от 256 бит, и с ключом, расположенным в другом месте, то вы можете 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 архива следует вручную при помощи утилиты `gbscrypt`. При таком режиме работы нет необходимости указывать алгоритм преобразования в правиле резервного копирования, иначе архив будет повторно преобразован 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	

Алгоритм	Длина ключа, бит	Примечание
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	

Локальный лист ограничений

В том случае, если какие-либо конкретные ресурсы клиента не должны попасть в резервную копию, их можно включить в локальный лист ограничений на клиенте. Лист ограничений располагается в каталоге `/opt/rubackup/etc`:

```
rubackup_restriction.list.mysql57
```

Наименование ресурса, для которого нет необходимости выполнять резервное копирование, должно быть указано в отдельной строке соответствующего файла.

Для того, чтобы листы ограничений имели силу, необходимо включить эту возможность для клиента в конфигурации RuBackup (см. Руководство системного администратора RuBackup).

Использование менеджера администратора RuBackup

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 пр.

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

Запуск менеджера администратора RBM:

Вариант 1:

```
# sudo LD_LIBRARY_PATH=/opt/rubackup/lib /opt/rubackup/bin/rbm
```

Вариант 2:

```
# ssh -X you_rubackup_server
```

```
# sudo LD_LIBRARY_PATH=/opt/rubackup/lib /opt/rubackup/bin/rbm
```

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 СУБД MySQL на хосте должен быть установлен клиент RuBackup и соответствующий модуль, обеспечивающий резервное копирование. Клиент должен быть авторизован администратором RuBackup (см. раздел «Клиенты» менеджера администратора RuBackup).

При помощи менеджера администратора RuBackup можно создать в глобальном расписании одно или несколько правил резервного копирования СУБД MySQL.

Для этого необходимо выполнить следующие действия:

1. Выбрать клиентский хост, на котором установлена СУБД MySQL и добавить правило резервного копирования (рисунок 2).

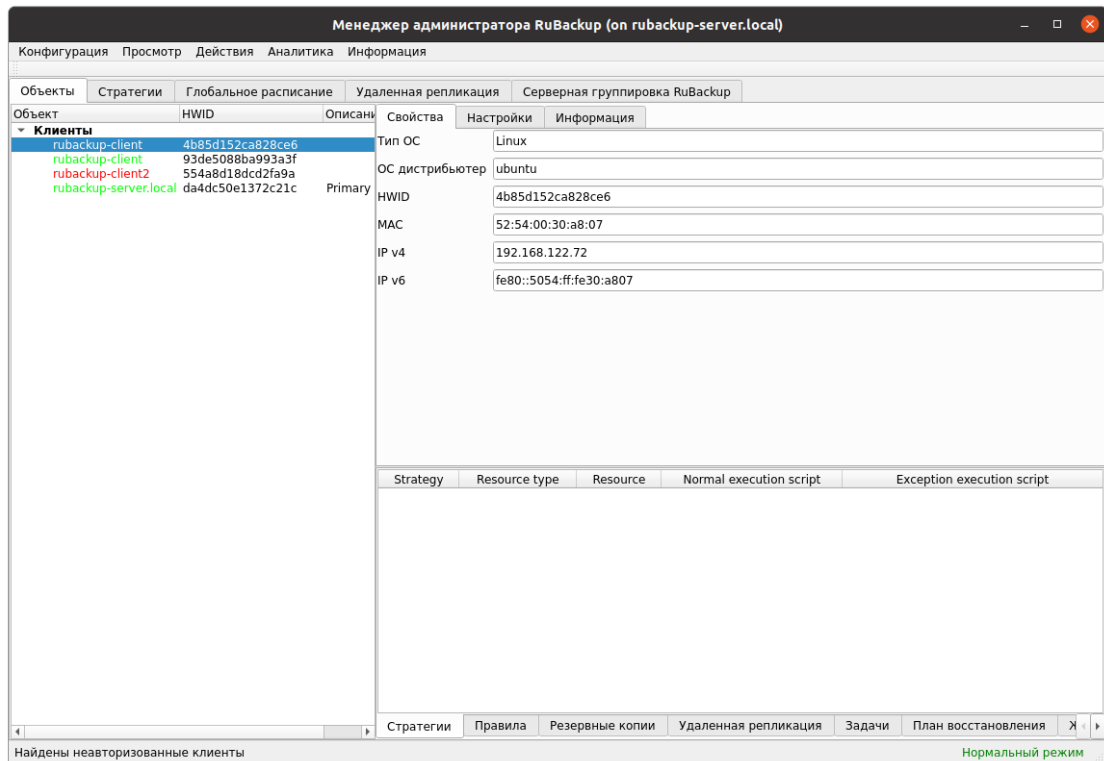


Рисунок 1

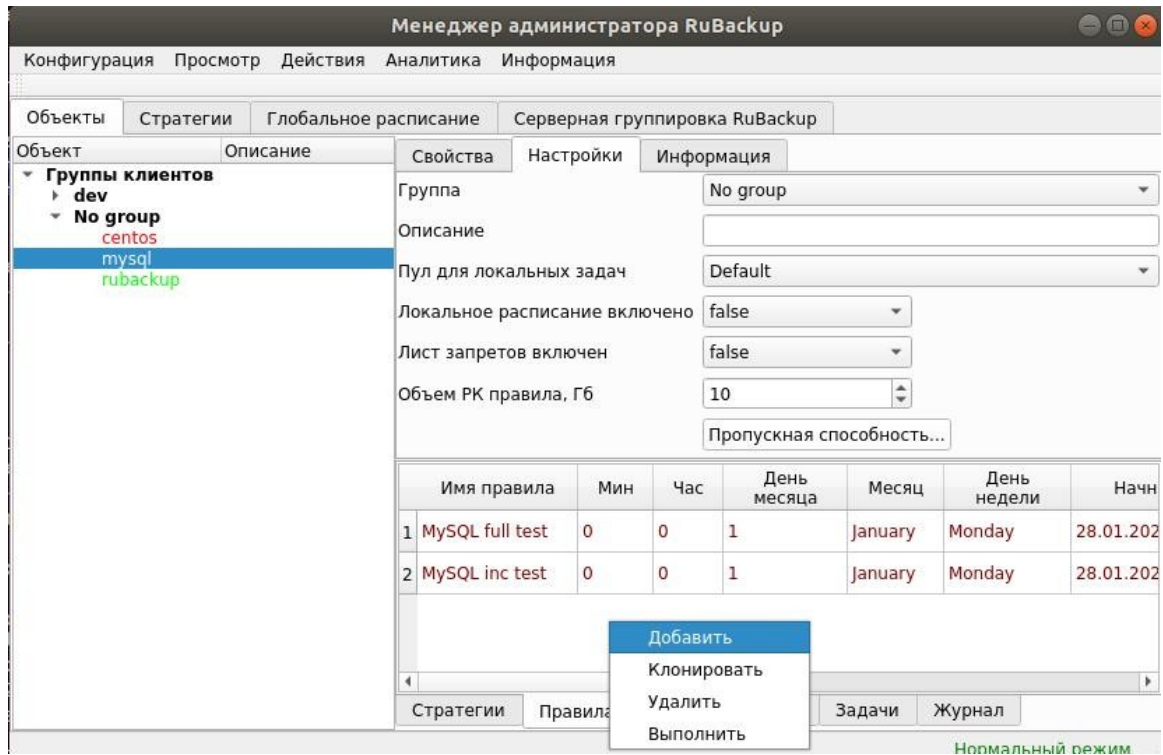


Рисунок 2

2. Выбрать тип ресурса «MySQL 5.7» (рисунок 3):

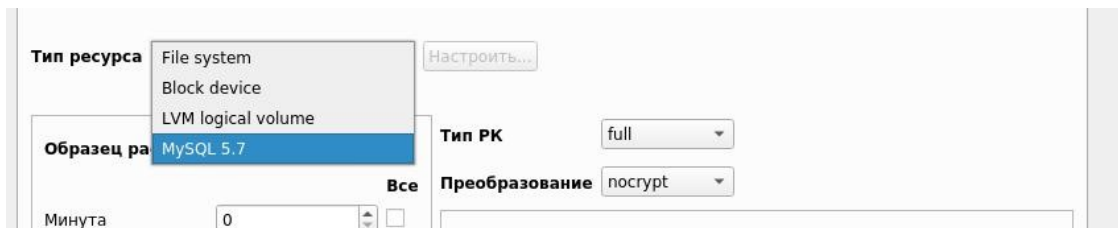


Рисунок 3

3. Выбрать ресурс (конфигурационный файл СУБД), для которого будет выполняться правило (рисунок 4):

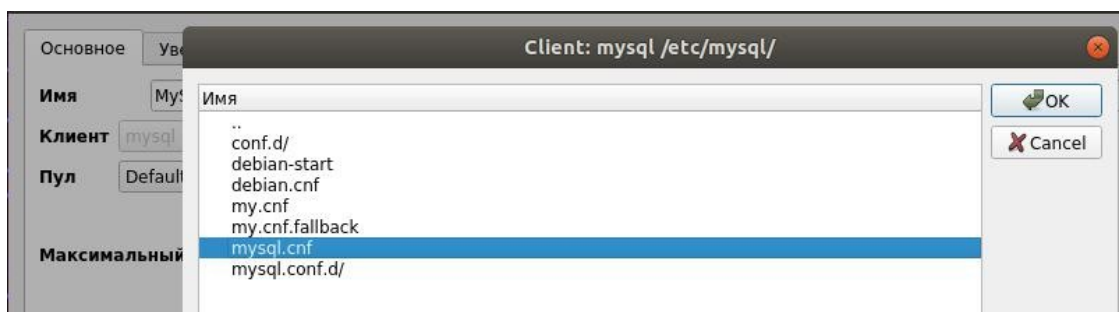


Рисунок 4

4. Установить прочие настройки: тип резервного копирования (Full), максимальный объем для резервных копий данного правила (50Гб), срок хранения (2 месяца), через какой промежуток времени требуется выполнить проверку резервной копии или не проверять её вовсе (рисунок 5).
5. На вкладке «Дополнительно» можно установить разрешение для клиента удалять резервные копии, установить автоматическое удаление устаревших резервных копий или определить условие их перемещения в другой пул (рисунок 6).

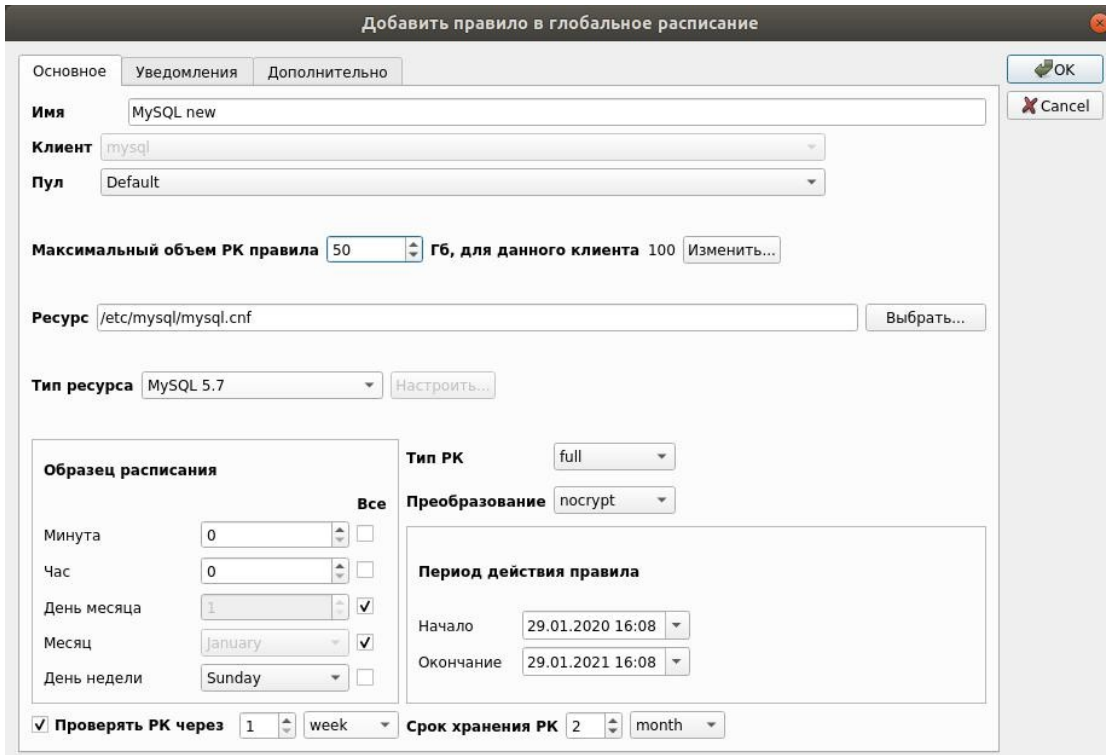


Рисунок 5

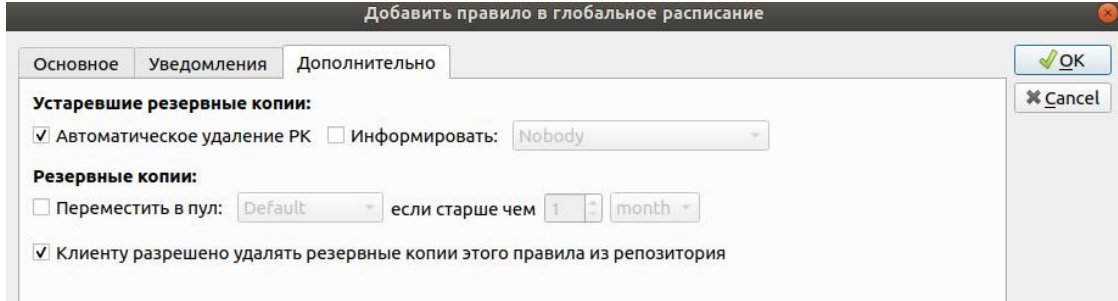


Рисунок 6

Вновь созданное правило будет обладать статусом «wait», это означает что оно не будет порождать задач на выполнение резервного копирования до той поры, пока администратор RuBackup не запустит его и оно изменит свой статус на «run». При необходимости работу правила можно будет приостановить или запустить в любой момент времени по желанию администратора. Так же администратор может инициировать немедленное создание задачи при статусе правила «wait».

Правило глобального расписания имеет срок жизни, определяемый при его создании, а так же предусматривает следующие возможности:

1) Выполнить скрипт на клиенте перед началом резервного копирования.

2) Выполнить скрипт на клиенте после успешного окончания резервного копирования.

3) Выполнить скрипт на клиенте после неудачного завершения резервного копирования.

4) Выполнить преобразование резервной копии на клиенте.

5) Периодически выполнять проверку целостности резервной копии.

6) 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.

7) Через определённый срок после создания резервной копии автоматически переместить её на другой пул хранения резервных копий, например на картридж ленточной библиотеки.

8) 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 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.

Настройки правил глобального расписания RuBackup

Для выполнения резервного копирования СУБД MySQL необходимо при помощи менеджера администратора RuBackup создать правило в глобальном расписании, в котором указать соответствующий тип ресурса. При создании правила в глобальном расписании администратор RuBackup будет иметь возможность выбрать необходимый конфигурационный файл СУБД MySQL на клиенте (для этого необходимо, чтобы на клиенте работал клиентский фоновый процесс).

При создании правила резервного копирования можно определить следующие параметры:

- тип резервного копирования (полный, дифференциальный или инкрементальный). Для образов и контейнеров выполняется только полный тип резервного копирования, для томов — полный, дифференциальный или инкрементальный;
- разрешенный максимальный объем для всех резервных копий правила;
- необходимость преобразования резервной копии тем или иным алгоритмом. Преобразование будет выполняться на стороне клиента;
- шаблон времени и даты создания задачи резервного копирования;
- флаг и период автоматической проверки резервной копии;
- срок хранения резервных копий создаваемого правила;
- пул хранения, в котором будут размещены резервные копии;
- необходимость автоматического удаления резервной копии, срок хранения которой истёк;
- перемещение резервной копии в другой пул, при достижении определённого срока с момента её создания;

- возможность для клиента удалять резервные копии из репозитория;
- настройки системы уведомления RuBackup для создаваемого правила;

Уведомления могут происходить в следующих случаях:

- нормальное исполнение процедуры резервного копирования;
- исполнение процедуры резервного копирования с ошибками;
- проверка резервной копии;
- окончание периода действия создаваемого правила;
- окончание выделенного объёма для хранения резервных копий правила;
- окончание срока хранения резервной копии.

Использование клиентского менеджера RuBackup

Принцип взаимодействия клиентского менеджера 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 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 вы отдали ту или иную команду при помощи клиентского менеджера, вы можете просто закрыть приложение, все действия будут выполнены системой резервного копирования (однако стоит дождаться сообщения что задание принято к исполнению и проконтролировать это во вкладке «Задачи»).

Графический интерфейс клиентского менеджера поддерживает русский и английский языки.

Запуск клиентского менеджера:

Вариант 1:

```
# sudo LD_LIBRARY_PATH=/opt/rubackup/lib/opt/rubackup/bin/rbc
```

Вариант 2:

```
# ssh -X you_rubackup_client
```

```
# sudo LD_LIBRARY_PATH=/opt/rubackup/lib /opt/rubackup/bin/rbc
```

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**»).

На главной странице клиентского менеджера расположены переключающиеся вкладки, позволяющие управлять резервными копиями, расписанием резервного копирования и просматривать текущие задачи клиента.

Вкладка «Резервные копии»

В таблице вкладки «Резервные копии» содержится информация обо всех резервных копиях клиента, которые хранятся в репозитории RuBackup (рисунок 7). 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	Created	Cr	
1	24	72	MySQL 5.7	/etc/mysql/mysql.cnf	full	Default	693076	113	2020-01-29 13:25:42+03	00:0	
2	25	73	24	MySQL 5.7	/etc/mysql/mysql.cnf	incremental	Default	152659	117	2020-01-29 13:29:05+03	00:0
3	26	86	25	MySQL 5.7	/etc/mysql/mysql.cnf	incremental	Default	152820	117	2020-01-29 13:58:09+03	00:0

Рисунок 7

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 потребуются вести пароль клиента.

Восстановить цепочку резервных копий.

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

Клиентский менеджер не ожидает окончания восстановления всех резервных копий, пользователь должен проконтролировать во вкладке «Задачи» что все созданные задачи 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 (см.опцию use-local-backup-directory).

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 8).

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	Validity start period		Validity end period	Resource type	
26	MySQL full test	2	0	0	1	January	Monday	2020-01-28 12:55:00+03		2021-01-28 12:55:00+03	MySQL 5.7	/etc
27	MySQL inc test	2	0	0	1	January	Monday	2020-01-28 12:56:00+03		2021-01-28 12:56:00+03	MySQL 5.7	/etc

Рисунок 8

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.

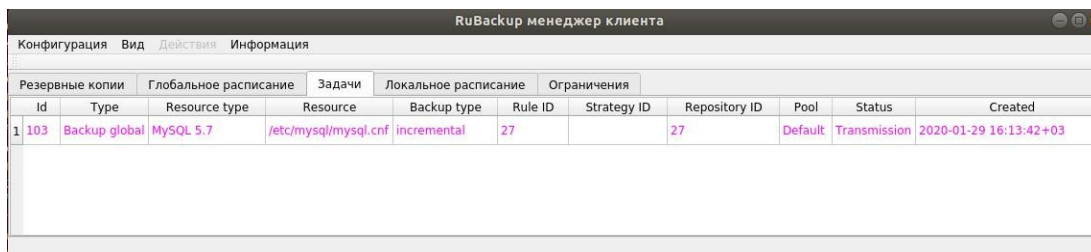
Запросить удалить правило из глобального расписания.

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 9). 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	Created
1	103	Backup global	MySQL 5.7	/etc/mysql/mysql.cnf	incremental	27	27	Default	Transmission	2020-01-29 16:13:42+03

Рисунок 9

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_archive

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

```
root@mysql:/backup# rb_archives
```

Id	Ref ID	Resource	Resource type	Backup type	Created	Crypto	Signed	Status
24		/etc/mysql/mysql.cnf	MySQL 5.7	full	2020-01-29 13:25:42+03	nocrypt	True	Not Ve
25	24	/etc/mysql/mysql.cnf	MySQL 5.7	incremental	2020-01-29 13:29:05+03	nocrypt	True	Not Ve
26	25	/etc/mysql/mysql.cnf	MySQL 5.7	incremental	2020-01-29 13:58:09+03	nocrypt	True	Not Ve
27	26	/etc/mysql/mysql.cnf	MySQL 5.7	incremental	2020-01-29 16:13:52+03	nocrypt	True	Not Ve

```
root@mysql:/backup#
```

rb_schedule

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

```
root@mysql:/backup# rb_schedule
```

Id	Name	Resource type	Resource	Backup type	Status
26	MySQL full test	MySQL 5.7	/etc/mysql/mysql.cnf	full	wait
27	MySQL inc test	MySQL 5.7	/etc/mysql/mysql.cnf	incremental	wait

```
root@mysql:/backup#
```

rb_tasks

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

```
root@mysql:/backup# rb_tasks
```

Id	Task type	Resource	Backup type	Status	Created
103	Backup global	/etc/mysql/mysql.cnf	incremental	Done	2020-01-29 16:13:42+03

```
root@mysql:/backup#
```

Ознакомиться с функциями утилит командной строки можно при помощи команды `man` или в руководстве «Утилиты командной строки RuBackup».

Восстановление MySQL 5.7 из резервных копий

Восстановление резервных копий можно инициировать при помощи RBC или из командной строки при помощи следующей команды:

```
# rb_archive -x ID -d /path_to_restore,
```

где ID - идентификатор резервной копии, которую необходимо восстановить (при восстановлении цепочки резервных копий достаточно дать команду на восстановление последней резервной копии, все предыдущие будут восстановлены автоматически),

path_to_restore - каталог, в котором будут восстановлены резервные копии. После выполнения операции в этом каталоге будет создан каталог mysql_restore. В том случае, если этот каталог существует, то операция восстановления будет завершена с ошибкой, перед восстановлением этот каталог следует удалить или переименовать. В том случае, если опция -d не была указана, то восстановление произойдет в текущий каталог.

В том случае, если в конфигурационном файле /opt/rubackup/etc/rb_module_mysql57.conf настройка direct_restore имеет значение **yes**, то резервная копия (или цепочка резервных копий, если вы восстанавливаете полную и несколько инкрементальных резервных копий) будет восстановлена на место существующей и работающей СУБД. В ходе восстановления сервис СУБД MySQL будет остановлен, файлы в каталоге datadir будут уничтожены, на их место будут восстановлены данные из резервных копий, сервис СУБД MySQL будет запущен вновь, данные из каталога mysql_restore будут удалены вместе с ним.

В том случае, если в момент восстановления сервис СУБД MySQL не будет функционировать, то резервные копии будут восстановлены в заданный каталог + mysql_restore, дальнейшее восстановление данных необходимо выполнять в ручном режиме при помощи следующего алгоритма:

1. Убедится в том, что сервис СУБД MySQL не функционирует.
2. Очистить каталог datadir (обычно это /var/lib/mysql).
3. Выполнить команду

```
# innobackupex --copy-back /path/to/BACKUP-DIR
```

В том случае, если в конфигурационном файле /opt/rubackup/etc/rb_module_mysql57.conf настройка direct_restore имеет значение **no**, то резервные копии будут восстановлены в заданный каталог

+ `mysql_restore`. Дальнейшее восстановление следует производить по алгоритму, предложенному выше.