

**RuBackup**

Система резервного копирования и восстановления данных

# **Резервное копирование и восстановление шаблонов и виртуальных машин OpenNebula**



**RuBackup**

Версия 1.9

2022 г.

# Содержание

Введение.....	3
Установка клиента RuBackup.....	5
Мастер-ключ.....	6
Подготовка виртуальной машины для выполнения резервного копирования средствами RuBackup.....	7
Защитное преобразование резервных копий.....	8
Локальные листы ограничений.....	10
Использование оконного менеджера администратора RuBackup.....	11
Использование клиентского менеджера RuBackup (RBC).....	21
Утилиты командной строки клиента RuBackup.....	26
Восстановление резервной копии виртуальной машины.....	27
Поддерживаемые конфигурации.....	29

## Введение

Система резервного копирования RuBackup позволяет выполнять клиентам полное, инкрементальное и дифференциальное резервное копирование шаблонов (template) и виртуальных машин платформы виртуализации OpenNebula.

Для шаблонов доступно полное резервное копирование, для виртуальных машин – полное, инкрементальное и дифференциальное.

**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 выполнения резервного копирования виртуальных машин OpenNebula на хост *front* требуется установить клиента RuBackup и модули *opennebula\_template*, *opennebula\_vm*. На виртуальные машины, для которых предполагается выполнять резервное копирование, должны быть установлены дополнения гостевой системы.

Резервное копирование выполняется по заранее заданным правилам в глобальном расписании RuBackup. Клиенту доступно срочное резервное копирование виртуальных машин OpenNebula, но в этом случае выполняется полное резервное копирование выбранного ресурса. Кроме того, клиенту может быть доступно локальное расписание, если это разрешено администратором системы резервного копирования.

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, возможно выполнить защитное преобразование резервной копии выбранным алгоритмом (см. раздел «Защитное преобразование резервных копий»).

Резервное копирование шаблона может быть выполнено как только для его конфигурации, так и совместно с образами, ассоциированными с

шаблоном. В ходе выполнения резервного копирования шаблона используется технология клонирования.

Резервное копирование виртуальной машины возможно в трех вариантах:

1) Резервное копирование только конфигурации виртуальной машины. При восстановлении такой резервной копии виртуальная машина будет создана точно такой, какой она создается из шаблона. Восстановить такую резервную копию можно только в том случае, если в системе присутствуют образы, которыми она должна пользоваться.

2) Резервное копирование конфигурации и частных данных виртуальной машины, которые образовались с момента ее создания. Восстановить такую резервную копию можно только в том случае, если в системе присутствуют образы, которыми она должна пользоваться.

3) Резервное копирование конфигурации, частных данных виртуальной машины и образов, которые она использует.

В ходе выполнения резервного копирования используется технология создания моментальных снимков виртуальной машины. Перед созданием снимка и сразу после создания снимка, внутри виртуальной машины может быть выполнен скрипт, который обеспечит консистентность данных приложения, функционирующего в виртуальной машине.

Для выполнения резервного копирования работающей виртуальной машины на ней должны быть установлены гостевые расширения, а так же при ее создании в OpenNebula необходимо включить функцию QEMU guest agent communication (это может быть включено как для всего комплекса OpenNebula, так и для отдельного шаблона из которого создаются виртуальные машины). Без гостевых расширений резервное копирование возможно только для выключенных виртуальных машин.

## Установка клиента RuBackup

Для резервного копирования OpenNebula необходимы следующие пакеты:

`rubackup-client.deb` – клиент резервного копирования

`rubackup-opennebula.deb` – модули резервного копирования

Установка пакетов клиента RuBackup производится из-под учетной записи с административными правами на узел **front** OpenNebula при помощи следующих команд:

```
# dpkg -i rubackup-client.deb
```

```
# dpkg -i rubackup-opennebula.deb
```

```
root@on-front:/rubackup-tmp# dpkg -i ./rubackup-client.deb
Выбор ранее не выбранного пакета rubackup-client.
(Чтение базы данных ... на данный момент установлено 191835 файлов и каталогов.)
Подготовка к распаковке ./rubackup-client.deb ...
Распаковывается rubackup-client (2020-03-10) ...
Настраивается пакет rubackup-client (2020-03-10) ...
root@on-front:/rubackup-tmp# dpkg -i ./rubackup-opennebula.deb
Выбор ранее не выбранного пакета rubackup-opennebula.
(Чтение базы данных ... на данный момент установлено 191913 файлов и каталогов.)
Подготовка к распаковке ./rubackup-opennebula.deb ...
Распаковывается rubackup-opennebula (2020-04-28) ...
Настраивается пакет rubackup-opennebula (2020-04-28) ...
```

Для возможности резервного копирования файловых систем при помощи RuBackup на сервер должен быть установлен клиент RuBackup. Подробно процедура установки клиента описана в «Руководстве по установке серверов резервного копирования и Linux клиентов RuBackup», для операционной системы Windows — в «Руководстве по установке Windows клиентов RuBackup».

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

## 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 e973 053d 10a1 c0c1 40e8 d332 9463 a7ee
00000100 8965 f275 d5e4 a04a d07d a625 d4e8 755f
00000200
```

# Подготовка виртуальной машины для выполнения резервного копирования средствами RuBackup

Для шаблона, на базе которого будут создаваться виртуальные машины, необходимо включить возможность взаимодействия с гостевыми дополнениями (рисунок 1):

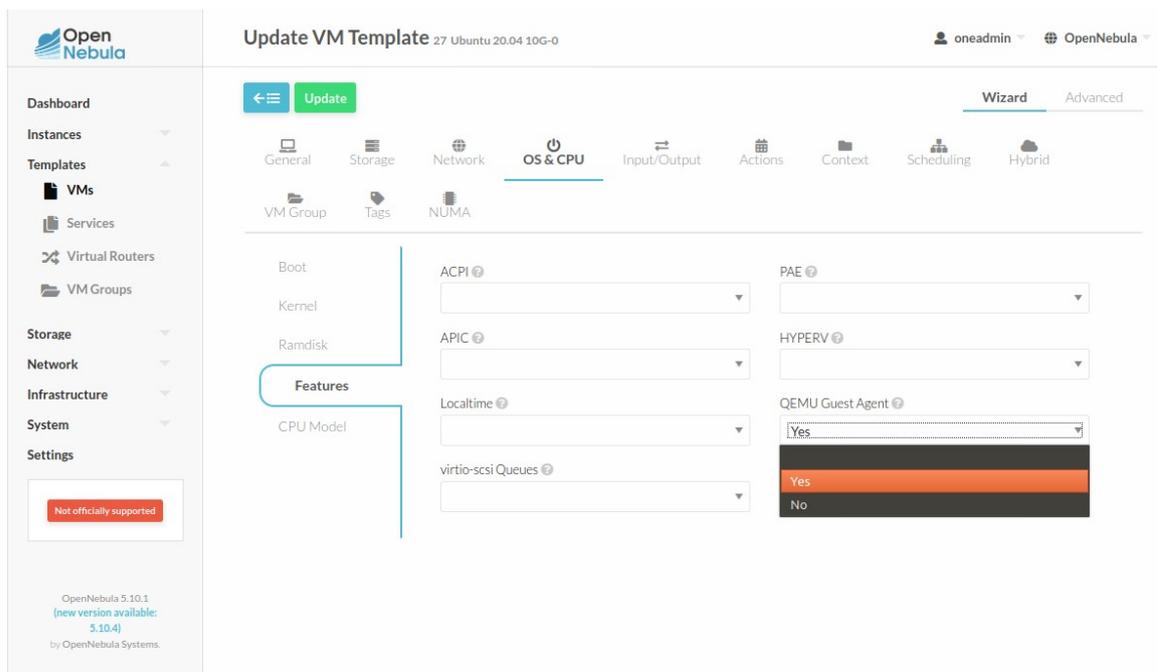


Рисунок 1

В операционной системе виртуальной машины необходимо установить пакет *qemu-guest-agent*:

```
# apt-get install qemu-guest-agent
```

или

```
# yum install qemu-guest-agent
```

в зависимости от типа операционной системы.

# 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	

## Локальные листы 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 файлах:

*/opt/rubackup/etc/rubackup\_restriction.list.opennebula\_vm*

*/opt/rubackup/etc/rubackup\_restriction.list.opennebula\_template*

Наименование ресурса (ID виртуальной машины или шаблона), для которого нет необходимости выполнять резервное копирование, должно быть указано в отдельной строке листа ограничений.

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

По умолчанию в предустановленных пакетах нет вышеуказанных файлов. При необходимости использовать листы ограничений их необходимо создать из-под учетной записи с административными привилегиями.

# Использование оконного 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.

Для запуска менеджера администратора RBM необходимо выполнить команду:

```
# ssh -X user@rbackup_server
```

```
# /opt/rbackup/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 2).

Клиент должен быть авторизован администратором RuBackup (см. раздел «Клиенты» менеджера администратора 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 (рисунок 3). Все новые клиенты должны быть авторизованы в системе резервного копирования:

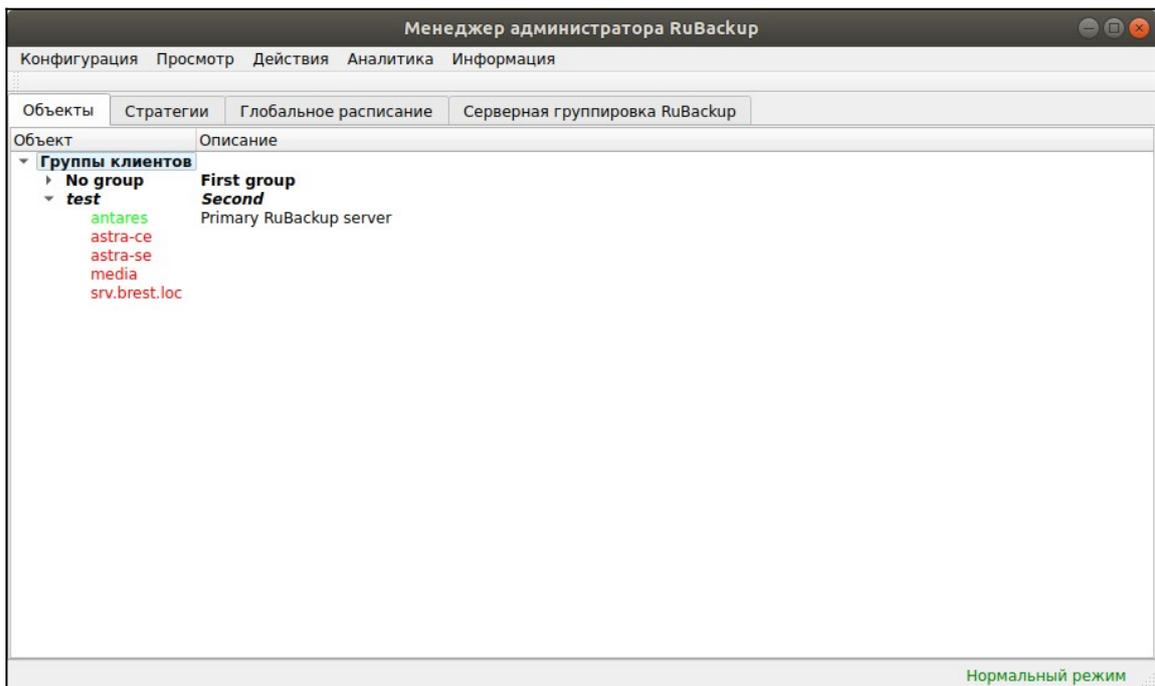


Рисунок 2

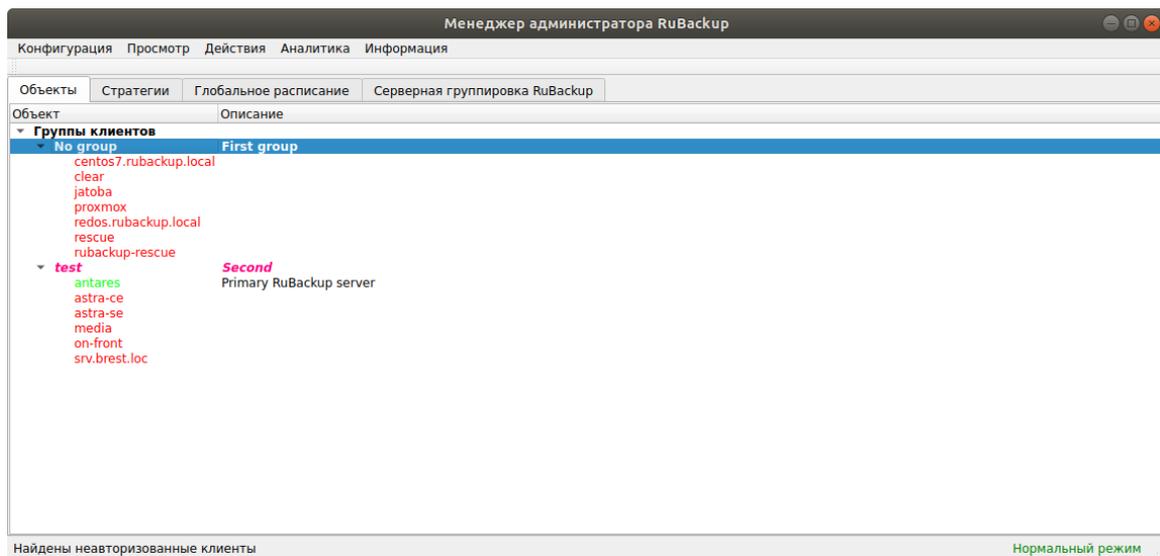


Рисунок 3

Для авторизации неавторизованного клиента в RBM выполните следующие действия:

1. Откройте меню **Действия** → **Клиенты** → **Авторизовать клиентов** (рисунок 4).

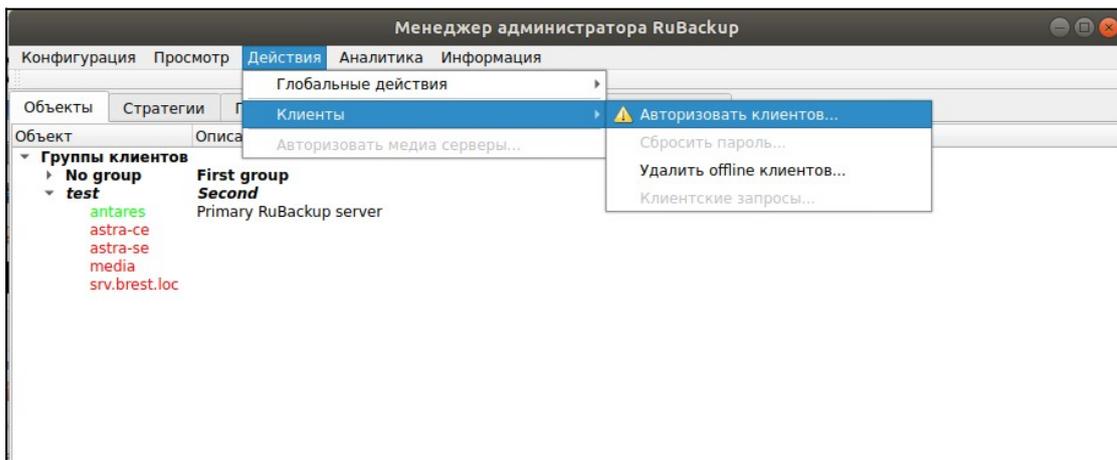


Рисунок 4

- Выберите нужного неавторизованного клиента и нажмите **Авторизовать** (рисунок 5).

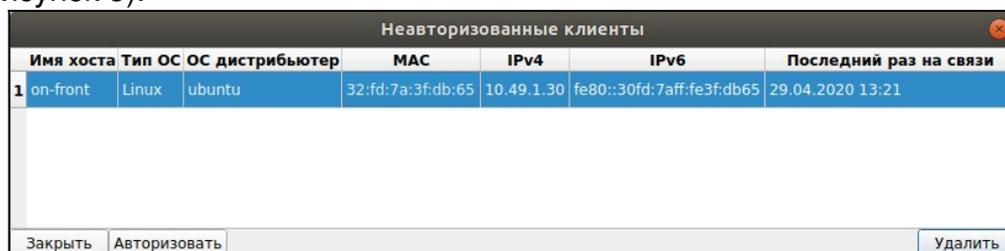


Рисунок 5

- После авторизации новый клиент будет виден в главном окне RBM (рисунок 6):

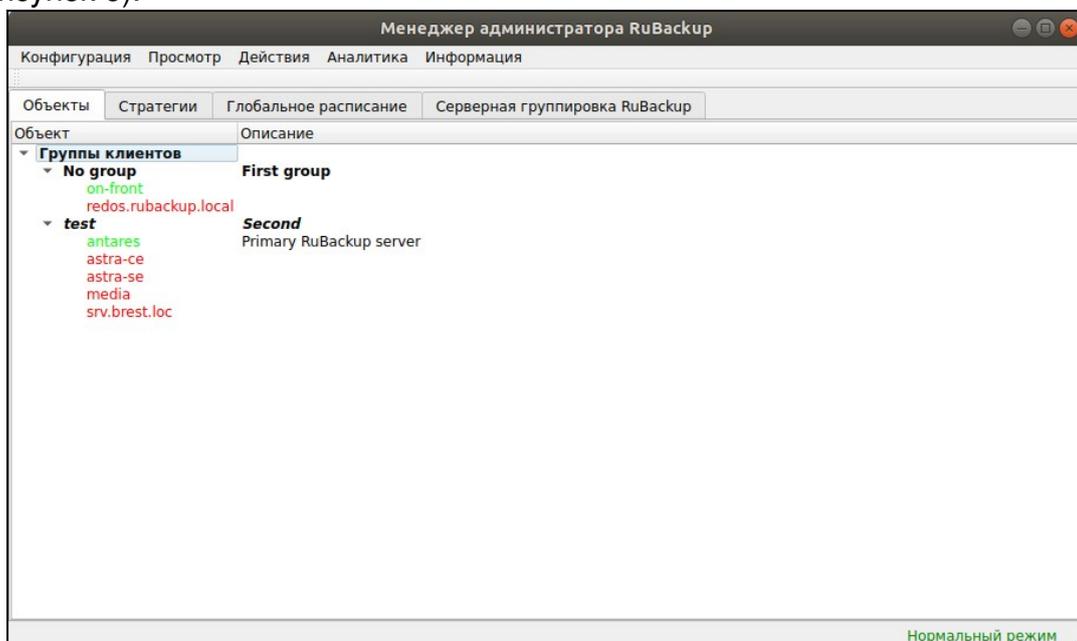


Рисунок 6

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*). Например, если в такую группу включить два сервера front двух разных комплексов OpenNebula, то можно реплицировать между ними шаблоны и виртуальные машины или переносить их с одного комплекса на другой. Перевести клиента из одной группы в другую можно следующим образом (рисунок 7):

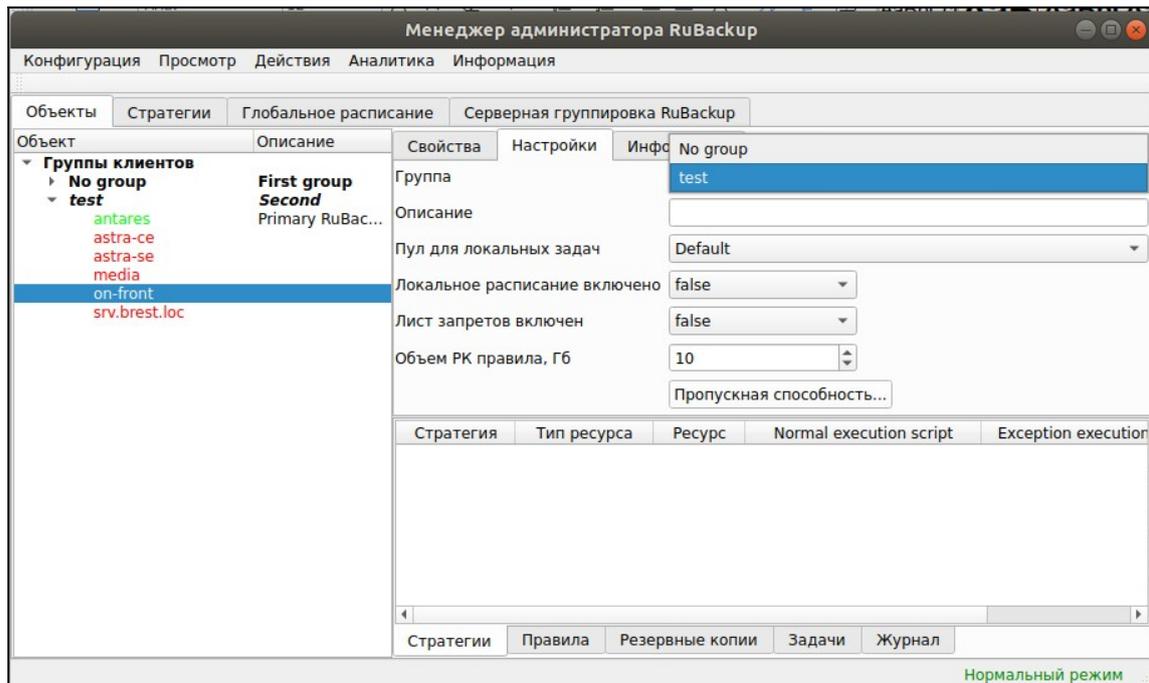


Рисунок 7

Для того, чтобы выполнять регулярное резервное копирование шаблона или виртуальной машины, необходимо создать правило в глобальном расписании. С этой целью необходимо выполнить следующие действия:

1. Выбрать клиентский хост, на котором установлен front OpenNebula и добавить правило резервного копирования (рисунок 8).
2. Выбрать тип ресурса «OpenNebula VM» для виртуальных машин или «OpenNebula template» для шаблона (рисунок 9).

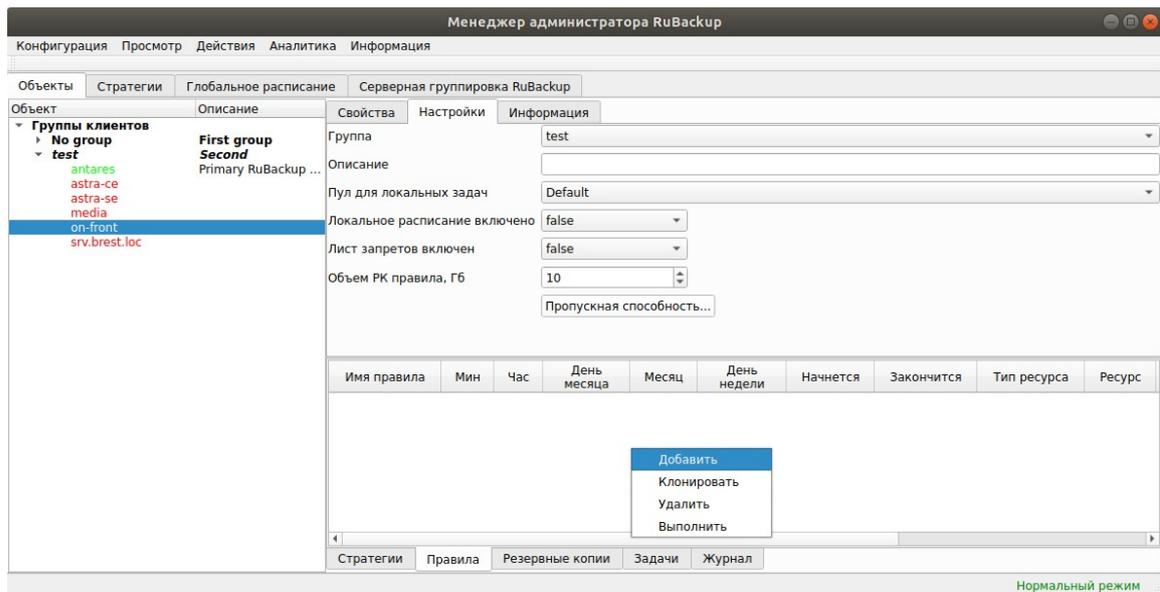


Рисунок 8

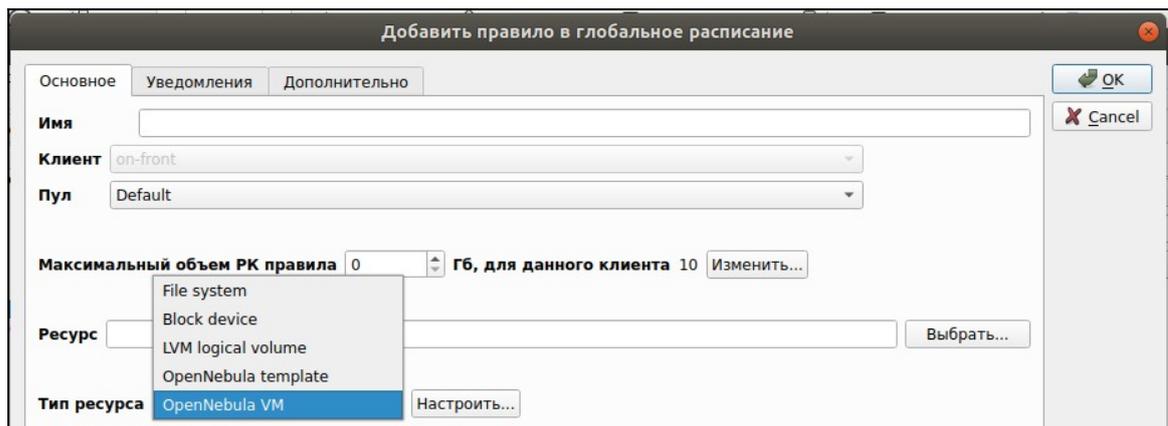


Рисунок 9

3. Выбрать ресурс, для которого будет выполняться правило (рисунок 10):

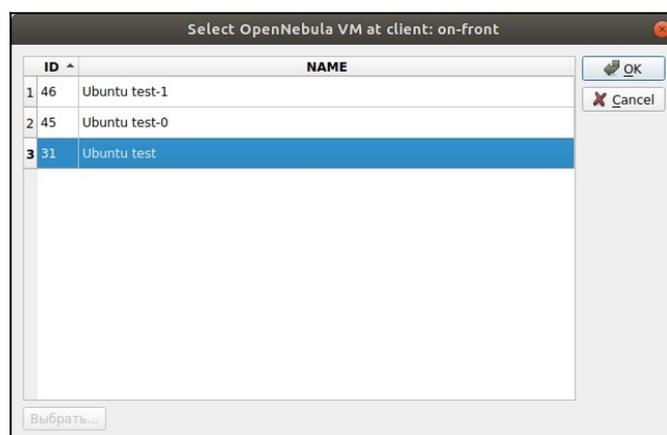
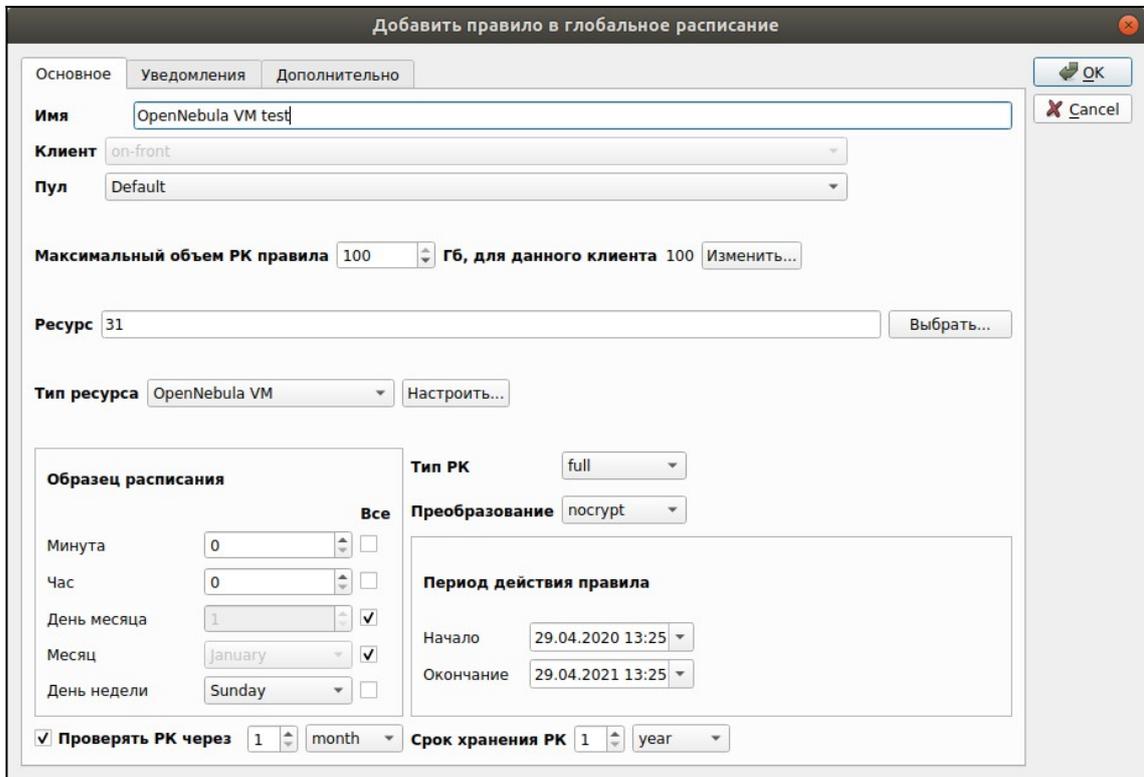


Рисунок 10

- Установить прочие настройки: тип резервного копирования, максимальный объем для резервных копий данного правила (100 Гб), срок хранения, через какой промежуток времени требуется выполнить проверку резервной копии или не проверять её вовсе (рисунок 11).



Добавить правило в глобальное расписание

Основное | Уведомления | Дополнительно

Имя: OpenNebula VM test

Клиент: on-front

Пул: Default

Максимальный объем РК правила: 100 Гб, для данного клиента: 100 Изменить...

Ресурс: 31

Тип ресурса: OpenNebula VM

Образец расписания:
 

Минута	0	<input type="checkbox"/>
Час	0	<input type="checkbox"/>
День месяца	1	<input checked="" type="checkbox"/>
Месяц	January	<input checked="" type="checkbox"/>
День недели	Sunday	<input type="checkbox"/>

Тип РК: full

Преобразование: nocrypt

Период действия правила:
 

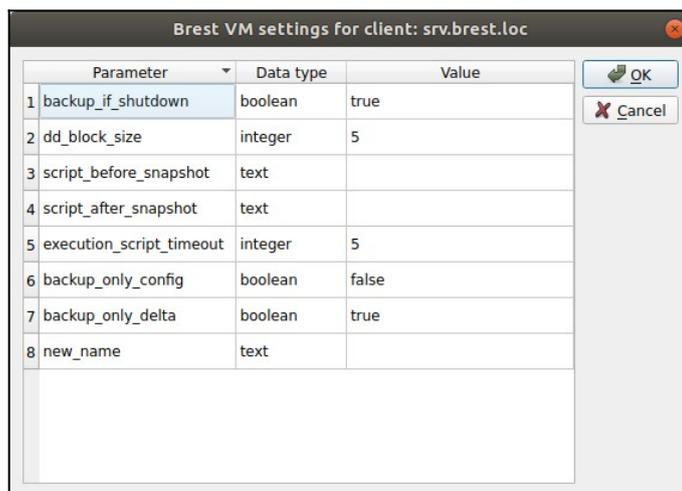
Начало	29.04.2020 13:25
Окончание	29.04.2021 13:25

Проверять РК через: 1 month

Срок хранения РК: 1 year

Рисунок 11

- Правила для выполнения резервных копий виртуальных машин могут иметь следующие дополнительные настройки (рисунок 12):



Parameter	Data type	Value
1 backup_if_shutdown	boolean	true
2 dd_block_size	integer	5
3 script_before_snapshot	text	
4 script_after_snapshot	text	
5 execution_script_timeout	integer	5
6 backup_only_config	boolean	false
7 backup_only_delta	boolean	true
8 new_name	text	

Рисунок 12

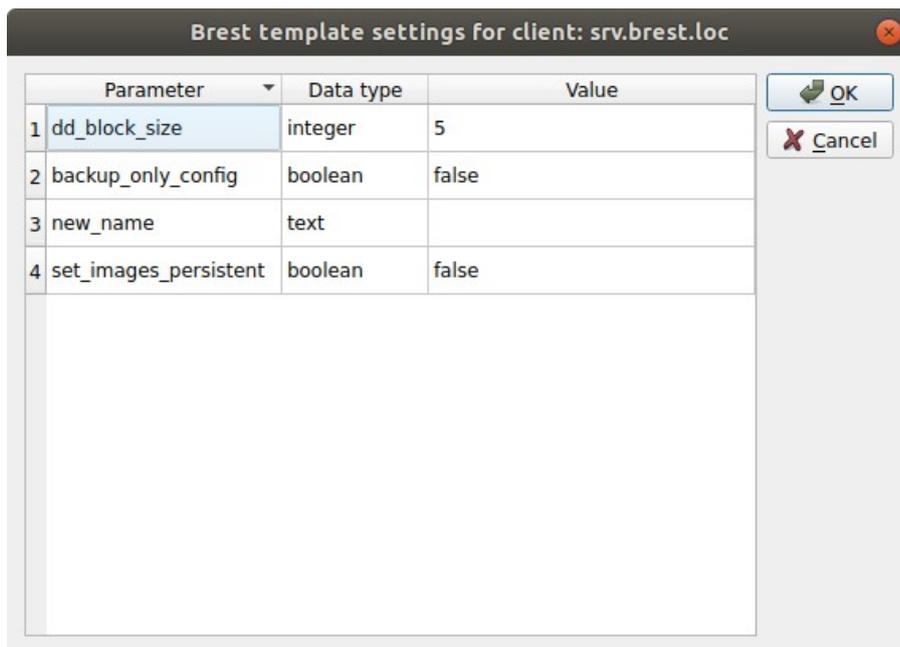
Описание параметров и их значения представлены в таблице 2.

Таблица 2 – Описание параметров настройки виртуальных машин

Параметр	Описание	Значение по умолчанию	Допустимые значения
backup_if_shutdown	Выполнять ли резервное копирование, если ВМ выключена	true	true, false
dd_block_size	Размер блока (в МБ) для операций DD	5	>1
script_before_snapshot	Скрипт внутри ВМ, который будет выполнен перед операцией мгновенного снимка		
script_after_snapshot	Скрипт внутри ВМ, который будет выполнен после операции мгновенного снимка		
execution_script_timeout	Период (в сек), в течение которого скрипт должен быть завершён. Если скрипт не будет завершён, операция резервного копирования будет прервана	5	>1
backup_only_config	Выполнять резервное копирование только конфигурации ВМ. В данном случае всегда выполняется полное резервное копирование. В случае true перекрывает значение параметра backup_only_delta	false	true, false
backup_only_delta	В случае true выполняет резервное копирование только частных данных виртуальной машины, которые появились после ее создания, данные из образов в резервную копию не попадают. В случае false резервная копия будет выполнена в том числе для образов виртуальной машины, исключая CDROM	true	true, false
new_name	Имя, с которым создавать виртуальную машину при восстановлении из резервной копии. В том случае, если этот параметр пуст, то виртуальная машина будет создана с прежним именем. Если такое имя уже есть в системе, то к нему будет добавлено число		

В том случае, если дополнительными настройками не заданы скрипты, которые могли бы выполняться в виртуальной машине, но в ней существует исполняемый скрипт `/opt/rubackup/scripts/rubackup-opennebula.sh`, то перед выполнением моментального снимка он будет выполнен с параметром **before**, а после выполнения моментального снимка он будет выполнен с параметром **after**.

6. Правила для выполнения резервных копий шаблонов могут иметь следующие дополнительные настройки (рисунок 13):

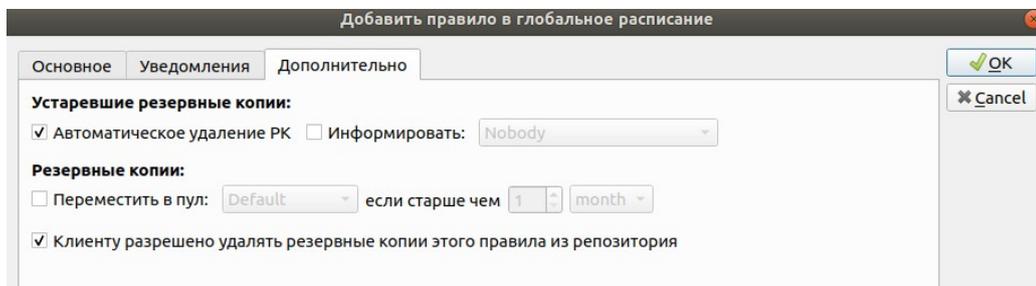


Parameter	Data type	Value
1 dd_block_size	integer	5
2 backup_only_config	boolean	false
3 new_name	text	
4 set_images_persistent	boolean	false

Рисунок 13

Описание параметров и их значения представлены в таблице 3.

7. 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 14):



Добавить правило в глобальное расписание

Основное | Уведомления | **Дополнительно**

**Устаревшие резервные копии:**

Автоматическое удаление РК  Информировать: Nobody

**Резервные копии:**

Переместить в пул: Default если старше чем 1 month

Клиенту разрешено удалять резервные копии этого правила из репозитория

Рисунок 14

Таблица 3 – Описание параметров настройки шаблонов

Параметр	Описание	Значение по умолчанию	Допустимые значения
dd_block_size	Размер блока (в МБ) для операций DD	5	>1
backup_only_config	Выполнять резервное копирование только конфигурации шаблона без ассоциированных с ним образов	false	true, false
new_name	Имя, с которым создавать шаблон при восстановлении из резервной копии. В том случае, если этот параметр пуст, то шаблон будет создан с прежним именем. Если такое имя уже есть в системе, то к нему будет добавлено число		
set_images_persistent	Установить для всех образов шаблона параметр PERSISTENT=yes после восстановления	false	true, false

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.

# Использование клиентского менеджера 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 (для примера использован хост front OpenNebula on-front) следует выполнить команды:

```
# ssh X root@on-node  
  
# /opt/rubackup/bin/rbc&
```

```
root@on-front:~#  
root@on-front:~# rbc&  
[1] 12849  
root@on-front:~# Logfile is /opt/rubackup/log/RuBackup.log
```

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 (рисунок 15). Без ввода пароля получить резервную копию для клиента из хранилища невозможно.

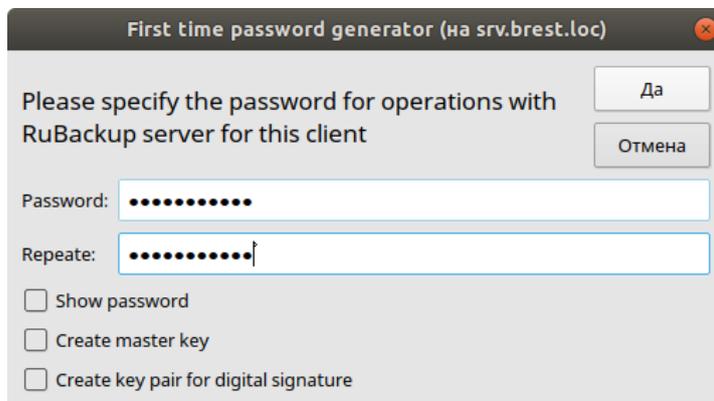


Рисунок 15

В случае успешного выполнения появится окно (рисунок 16):

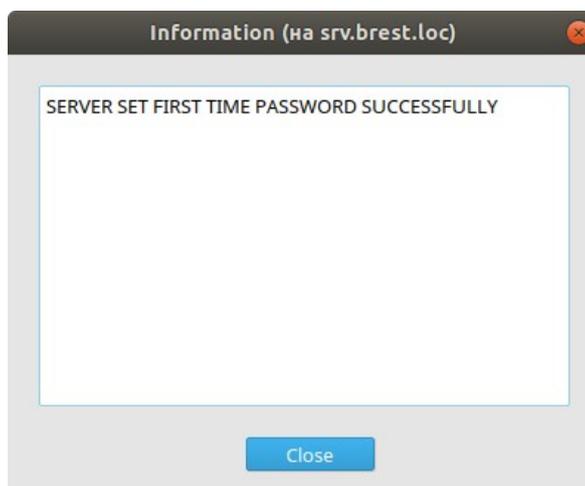


Рисунок 16

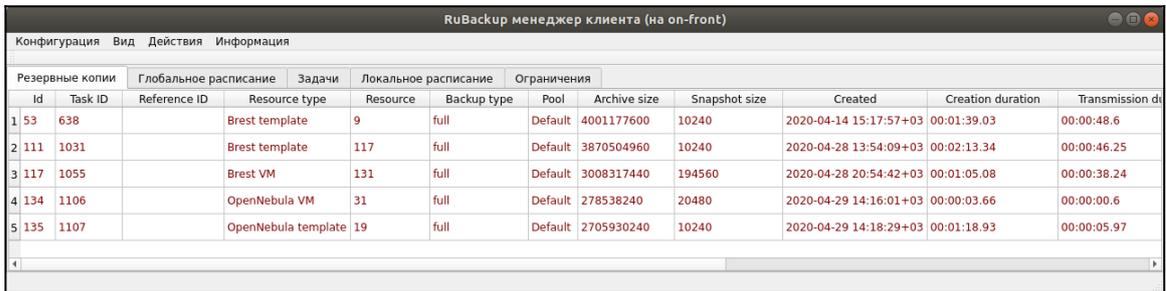
Хэш пароля восстановления хранится в базе данных RuBackup сервера. При необходимости можно изменить пароль при помощи клиентского менеджера (меню «**Конфигурация**» → «**Изменить пароль**»).

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 17). Дифференциальные резервные копии ссылаются на полные резервные копии, инкрементальные резервные копии ссылаются на полные резервные копии или предыдущие инкрементальные, так что при

необходимости восстановить всю цепочку резервных копий данных можно одной командой.



RuBackup менеджер клиента (на on-front)											
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	Creation duration	Transmission d
1	53	638	Brest template	9	full	Default	4001177600	10240	2020-04-14 15:17:57+03	00:01:39.03	00:00:48.6
2	111	1031	Brest template	117	full	Default	3870504960	10240	2020-04-28 13:54:09+03	00:02:13.34	00:00:46.25
3	117	1055	Brest VM	131	full	Default	3008317440	194560	2020-04-28 20:54:42+03	00:01:05.08	00:00:38.24
4	134	1106	OpenNebula VM	31	full	Default	278538240	20480	2020-04-29 14:16:01+03	00:00:03.66	00:00:00.6
5	135	1107	OpenNebula template	19	full	Default	2705930240	10240	2020-04-29 14:18:29+03	00:01:18.93	00:00:05.97

Рисунок 17

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 локальной файловой системе клиента.

При восстановлении резервной копии или цепочки резервных копий клиент должен выбрать место для восстановления файлов резервной копии. Рекомендуется использовать либо временный каталог для операций с резервными копиями (например, /rubackup-tmp) или SAFE\_DIRS для datastore OpenNebula (по умолчанию /var/tmp).

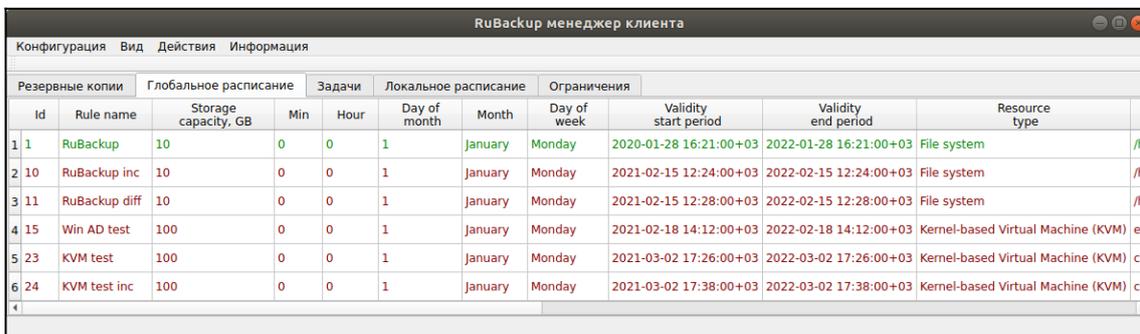
RBC не ожидает окончания восстановления всех резервных копий. Клиент должен проконтролировать на вкладке «Задачи» успешное завершение созданных задач на восстановление данных завершились успешно (статус задач «Done»). Для успешного выполнения этого действия требуется наличие достаточного свободного места в каталоге, предназначенном для создания и временного хранения резервных копий (см. параметр 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 будет 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 18).



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		
1	RuBackup	10	0	0	1	January	Monday	2020-01-28 16:21:00+03	2022-01-28 16:21:00+03	File system /hc		
2	RuBackup inc	10	0	0	1	January	Monday	2021-02-15 12:24:00+03	2022-02-15 12:24:00+03	File system /hc		
3	RuBackup diff	10	0	0	1	January	Monday	2021-02-15 12:28:00+03	2022-02-15 12:28:00+03	File system /hc		
4	Win AD test	100	0	0	1	January	Monday	2021-02-18 14:12:00+03	2022-02-18 14:12:00+03	Kernel-based Virtual Machine (KVM) e5		
5	KVM test	100	0	0	1	January	Monday	2021-03-02 17:26:00+03	2022-03-02 17:26:00+03	Kernel-based Virtual Machine (KVM) cb		
6	KVM test inc	100	0	0	1	January	Monday	2021-03-02 17:38:00+03	2022-03-02 17:38:00+03	Kernel-based Virtual Machine (KVM) cb		

Рисунок 18

Во вкладке «Глобальное расписание» пользователю доступны следующие действия:

### Запросить новое правило.

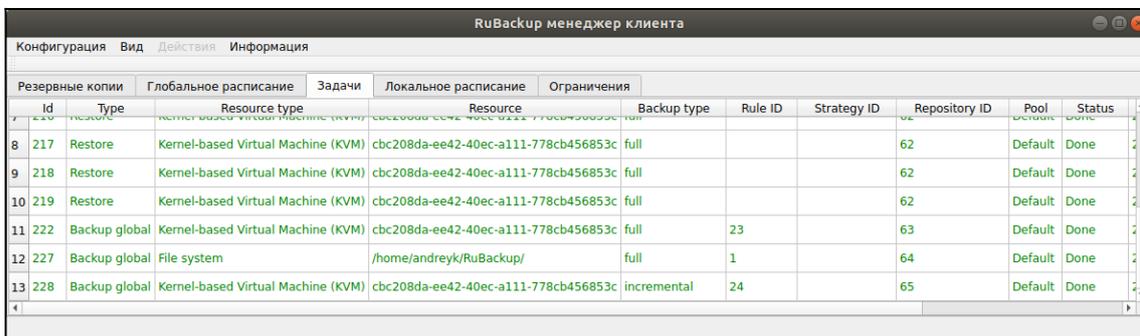
Это действие вызывает диалог подготовки нового правила в глобальном расписании RuBackup для данного клиента. Запрос на добавление правила требует одобрения администратора RuBackup, одобрение может быть сделано в оконном менеджере администратора RuBackup.

### 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 19). 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»).



Id	Type	Resource type	Resource	Backup type	Rule ID	Strategy ID	Repository ID	Pool	Status
8	217	Restore	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	full		62	Default	Done
9	218	Restore	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	full		62	Default	Done
10	219	Restore	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	full		62	Default	Done
11	222	Backup global	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	full	23	63	Default	Done
12	227	Backup global	File system	/home/andreyk/RuBackup/	full	1	64	Default	Done
13	228	Backup global	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	incremental	24	65	Default	Done

Рисунок 19

Примечание – Информация о выполнении служебных задач в данной вкладке не отображается. Служебными являются задачи проверки, удаления, перемещения резервных копий, а также их копирования в другой пул.

## 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@on-front:~# rb_archives
```

Id	Ref ID	Resource	Resource type	Backup type	Created	Crypto	Signed	Status
53	9	Brest template	Brest template	full	2020-04-14 15:17:57+03	nocrypt	True	Not Verified
111	117	Brest template	Brest template	full	2020-04-28 13:54:09+03	nocrypt	True	Not Verified
117	131	Brest VM	Brest VM	full	2020-04-28 20:54:42+03	nocrypt	True	Not Verified
134	31	OpenNebula VM	OpenNebula VM	full	2020-04-29 14:16:01+03	nocrypt	True	Not Verified
135	19	OpenNebula template	OpenNebula template	full	2020-04-29 14:18:29+03	nocrypt	True	Not Verified

#### rb\_schedule

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

```
root@on-front:~# rb_schedule
```

Id	Name	Resource type	Resource	Backup type	Status
40	OpenNebula VM test	OpenNebula VM	31	full	run
41	OpenNebula template	OpenNebula template	19	full	wait

```
root@on-front:~#
root@on-front:~#
```

#### rb\_tasks

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

```
root@on-front:~# rb_tasks
```

Id	Task type	Resource	Backup type	Status	Created
1106	Backup global	31	full	Done	2020-04-29 14:15:56+03
1107	Backup global	19	full	Done	2020-04-29 14:16:59+03

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

# Восстановление резервной копии виртуальной машины

Для восстановления резервной копии виртуальной машины необходимо определить идентификатор резервной копии, которую необходимо восстановить, например, при помощи команды `rb_archives`:

```

Id | Ref ID | Resource | Resource type | Backup type | Created | Crypt
-----+-----+-----+-----+-----+-----+-----
53 | | 9 | Brest template | full | 2020-04-14 15:17:57+03 | noscr
111 | | 117 | Brest template | full | 2020-04-28 13:54:09+03 | noscr
117 | | 131 | Brest VM | full | 2020-04-28 20:54:42+03 | noscr
134 | | 31 | OpenNebula VM | full | 2020-04-29 14:16:01+03 | noscr
135 | | 19 | OpenNebula template | full | 2020-04-29 14:18:29+03 | noscr

```

В приведенном примере в системе резервного копирования присутствуют три резервные копии. Виртуальная машина с идентификатором 31 может быть восстановлена из полной резервной копии с идентификатором 134. Для этого необходимо выполнить команду

```
# rb_archives -x 134
```

В случае успешно принятой задачи команда вернет «ок», а восстановление будет происходить в фоновом режиме.

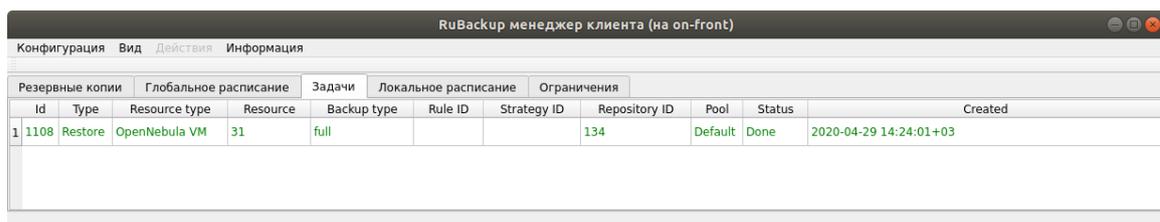
Проконтролировать процесс восстановления можно при помощи `rb_task`:

```

root@on-front:~# rb_tasks
Id | Task type | Resource | Backup type | Status | Created
-----+-----+-----+-----+-----+-----
1107 | Backup global | 19 | full | Done | 2020-04-29 14:16:59+03
1108 | Restore | 31 | full | Done | 2020-04-29 14:24:01+03
root@on-front:~#

```

или при помощи RBC (рисунок 20):



The screenshot shows the RuBackup client manager interface. The main window title is "RuBackup менеджер клиента (на on-front)". The interface includes a menu bar with "Конфигурация", "Вид", "Действия", and "Информация". Below the menu is a tabbed interface with tabs for "Резервные копии", "Глобальное расписание", "Задачи", "Локальное расписание", and "Ограничения". The "Задачи" tab is active, displaying a table with the following data:

Id	Type	Resource type	Resource	Backup type	Rule ID	Strategy ID	Repository ID	Pool	Status	Created
1	1108	Restore	OpenNebula VM	31	full		134	Default	Done	2020-04-29 14:24:01+03

Рисунок 20

Кроме того, можно детально проконтролировать происходящее при помощи журнала:

```

root@on-front:~# tail /opt/rubackup/log/RuBackup.log
Wed Apr 29 11:18:31 2020: Remove obsoleted file: /rubackup-tmp/on-front_TaskID_1107_RuleID_41_D2020_4_29H11_16_59_BackupType_1_ResourceType_19.snap.tar
Wed Apr 29 11:18:31 2020: Remove obsoleted signature file: /rubackup-tmp/on-front_TaskID_1107_RuleID_41_D2020_4_29H11_16_59_BackupType_1_ResourceType_19.tar.signature
Wed Apr 29 11:18:31 2020: Remove obsoleted signature file: /rubackup-tmp/on-front_TaskID_1107_RuleID_41_D2020_4_29H11_16_59_BackupType_1_ResourceType_19.snap.tar.signature
Wed Apr 29 11:18:31 2020: Task was done. ID: 1107
Wed Apr 29 11:24:01 2020: [RBC] Request to restore next archive(s) ID from repository: 134 to: /root
Wed Apr 29 11:24:02 2020: RuBackup server commands: Run task ID: 1108 Resource type: 20 Module: OpenNebula VM Resource: 31 Media server: antares
Wed Apr 29 11:24:02 2020: Create a file: /root/on-front_TaskID_1106_RuleID_40_D2020_4_29H11_15_56_BackupType_1_ResourceType_20.tar
Wed Apr 29 11:24:03 2020: md5sum of transferred file is ok: 4f82dae0ead6cd7a5cc9dba351523ec1
Wed Apr 29 11:24:03 2020: Transfer file is succeeded: /root/on-front_TaskID_1106_RuleID_40_D2020_4_29H11_15_56_BackupType_1_ResourceType_20.tar
Wed Apr 29 11:24:34 2020: Task was done. ID: 1108

```

В случае восстановления инкрементальной резервной копии будет сформирована цепочка восстановления: вначале будет восстановлена полная резервная копия и на нее будут наложены изменения из инкрементальных резервных копий.

После выполнения восстановления в OpenNebula появится новая виртуальная машина (ubuntu test-2), полностью идентичная той, которая была в системе в момент резервного копирования (рисунок 21):

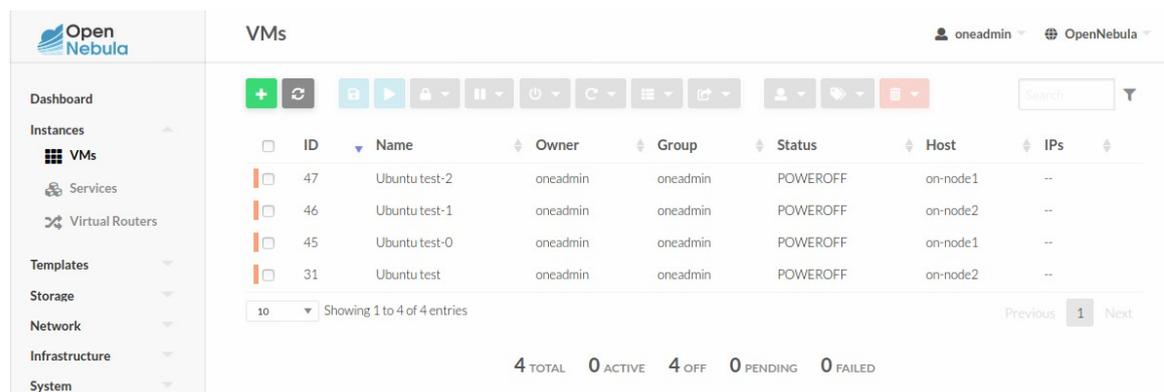


Рисунок 21

После восстановления можно запустить и проверить виртуальную машину (рисунок 22):

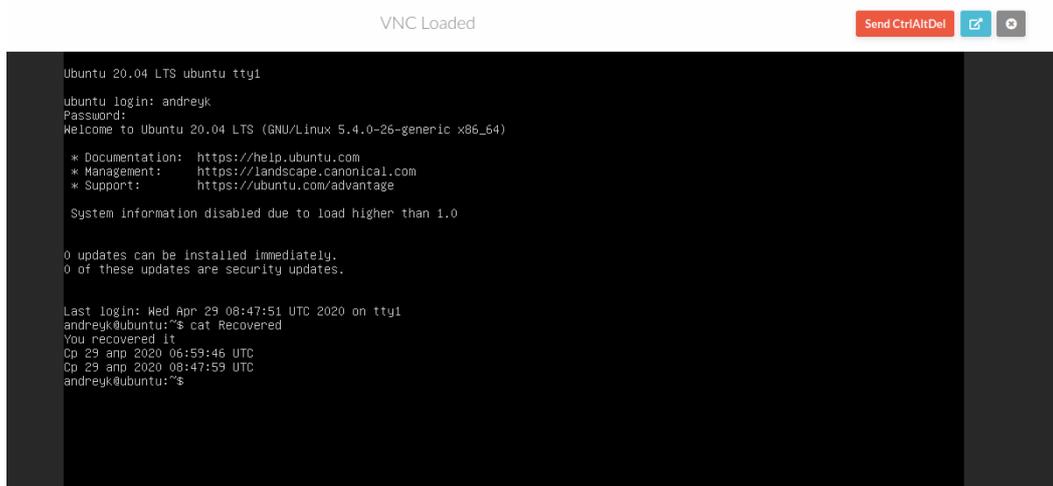


Рисунок 22

## Поддерживаемые конфигурации

Модули RuBackup для резервного копирования и восстановления шаблонов и виртуальных машин комплекса OpenNebula поддерживаются для следующих конфигураций:

OpenNebula 5.10

libvirt 4.0.0

Операционные системы комплекса OpenNebula:

Ubuntu 18.04, Ubuntu 20.04

Гостевые операционные системы:

любые поддерживаемые OpenNebula

Гипервизоры:

KVM

Расположение виртуальных машин:

qcow2, кластерная файловая система поверх iscsi устройств

В том случае, если вам для целей тестирования необходимо развернуть комплекс OpenNebula, это можно сделать с использованием вложенной виртуализации. Для того, чтобы получить полную инструкцию по развертыванию и настройке платформы OpenNebula ее и резервного копирования при помощи RuBackup, обращайтесь по адресу [support@rubackup.ru](mailto:support@rubackup.ru).