

RuBackup

Система резервного копирования и восстановления данных

Резервное копирование и восстановление виртуальных машин KVM



RuBackup

Версия 1.9

2022 г.

Содержание

Введение.....	3
Подготовка хоста KVM для выполнения резервного копирования средствами RuBackup.....	5
Подготовка виртуальной машины KVM для выполнения резервного копирования средствами RuBackup.....	8
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 виртуальной машины.....	28
Операции над VM, восстановленной без развертывания.....	33

Введение

Система резервного копирования RuBackup позволяет выполнять клиентам полное, инкрементальное и дифференциальное резервное копирование виртуальных машин KVM без их остановки.

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 на хост, где установлен KVM, требуется установить клиента RuBackup и модуль `kvm` для клиента RuBackup. В виртуальные машины, для которых предполагается выполнение резервного копирования средствами RuBackup, должен быть установлен `qemu-guest-agent` и в их конфигурацию должен быть добавлен `Channel Device org.qemu.guest_agent.0`.

Резервное копирование выполняется по заранее заданным правилам в глобальном расписании RuBackup. Клиенту доступно срочное резервное копирование виртуальных машин KVM, но в этом случае выполняется полное резервное копирование выбранного ресурса.

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

Полное 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»).

RuBackup может выполнять резервное копирование виртуальных машин KVM с дисками следующих типов: *file*, *block*, *network* (в том случае, когда диски виртуальной машины располагаются в хранилище Ceph в виде *rados block device*).

Резервное копирование поддерживается для *raw*, *lvm*, *qcow2*. Количество дисков в виртуальной машине может быть больше одного, в этом случае резервное копирование выполняется для всех дисков.

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.

Подготовка хоста KVM для выполнения резервного копирования средствами RuBackup

Для возможности резервного копирования и восстановления виртуальных машин KVM при помощи СРК RuBackup на сервер следует установить следующие пакеты:

- `rubackup-client.deb` – клиент резервного копирования,
- `rubackup-kvm.deb` – модуль резервного копирования данных KVM.

Установка клиента RuBackup

Для осуществления резервного копирования и восстановления виртуальных машин KVM при помощи RuBackup на сервер 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 виртуальных машин KVM клиент RuBackup должен работать от имени суперпользователя (`root` в Linux и Unix).

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».

Установка пакетов модулей резервного копирования

Установка пакета модулей резервного копирования RuBackup производится из учётной записи с административными правами на узле KVM после установки на него клиента RuBackup.

Для установки пакета модулей используйте следующий вызов:

В зависимости от типа операционной системы:

```
# sudo dpkg -i ./rubackup-kvm.deb
```

или

```
# sudo rpm -I ./rubackup-ceph-kvm.rpm
```

Настройка каталога для временных файлов

Для создания резервных копий 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 */kvm-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. «Руководство системного администратора RuBackup».

Настройка AppArmor

В некоторых случаях *apparmor* может блокировать выполнение резервного копирования виртуальных машин. На это может указывать следующая ошибка:

```
(error: internal error: unable to execute QEMU command 'transaction': Could not create file: Permission denied)
```

и сообщения о блокировании операций AppArmor в журнале системы.

Для того, чтобы данную ошибку устранить, необходимо выполнить:

```
sudo apt-get install apparmor-utils
```

```
sudo aa-complain /usr/sbin/libvirtd
```

```
sudo aa-complain /etc/apparmor.d/libvirt/libvirt-7d2b303d-8c14-4a1d-9cbd-9020460b2f4e (подобные файлы)
```

Какой именно файл блокируется можно определить командой:

```
sudo cat /var/log/syslog | grep "apparmor" | grep "DENIED" | grep libvirt
```

В том случае, когда у виртуальной машины несколько дисков, при создании снимка может возникнуть блокировка, инициированная *apparmor*.

Чтобы избежать подобных ситуаций, необходимо внести информацию о каталоге для создания резервных копий и хранения временных файлов в шаблон */etc/apparmor.d/libvirt/TEMPLATE.qemu*:

```
profile LIBVIRT_TEMPLATE flags=(attach_disconnected) {
  #include <abstractions/libvirt-qemu>
  /kvm-backup/** rw,
}
```

Удаление клиента RuBackup

При необходимости вы можете удалить с сервера клиент RuBackup и установленные модули резервного копирования. Удаление клиента RuBackup возможно из учётной записи с административными правами.

Для удаления сервиса *rubackup-client* используйте команды:

```
$ sudo systemctl disable rubackup_client
$ sudo systemctl daemon-reload
```

Для удаления клиента RuBackup и модуля **rubackup-kvm** используйте следующие команды.

```
$ sudo apt remove rubackup-kvm
$ sudo apt remove rubackup-client
```

При необходимости удалить клиент RuBackup из конфигурации CPK, это может сделать системный администратор RuBackup при помощи оконного Менеджера Администратора RBM.

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

1. Выполнить команду:

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

2. Перезагрузить операционную систему:

```
$ init 6
```

Подготовка виртуальной машины KVM для выполнения резервного копирования средствами RuBackup

Для подготовки виртуальной машины KVM необходимо выполнить следующие действия:

1. Установить для виртуальной машины оборудование *chanell device org.qemu.guest_agent.0*. Это можно сделать при помощи *virt-manager* (рисунок 1):

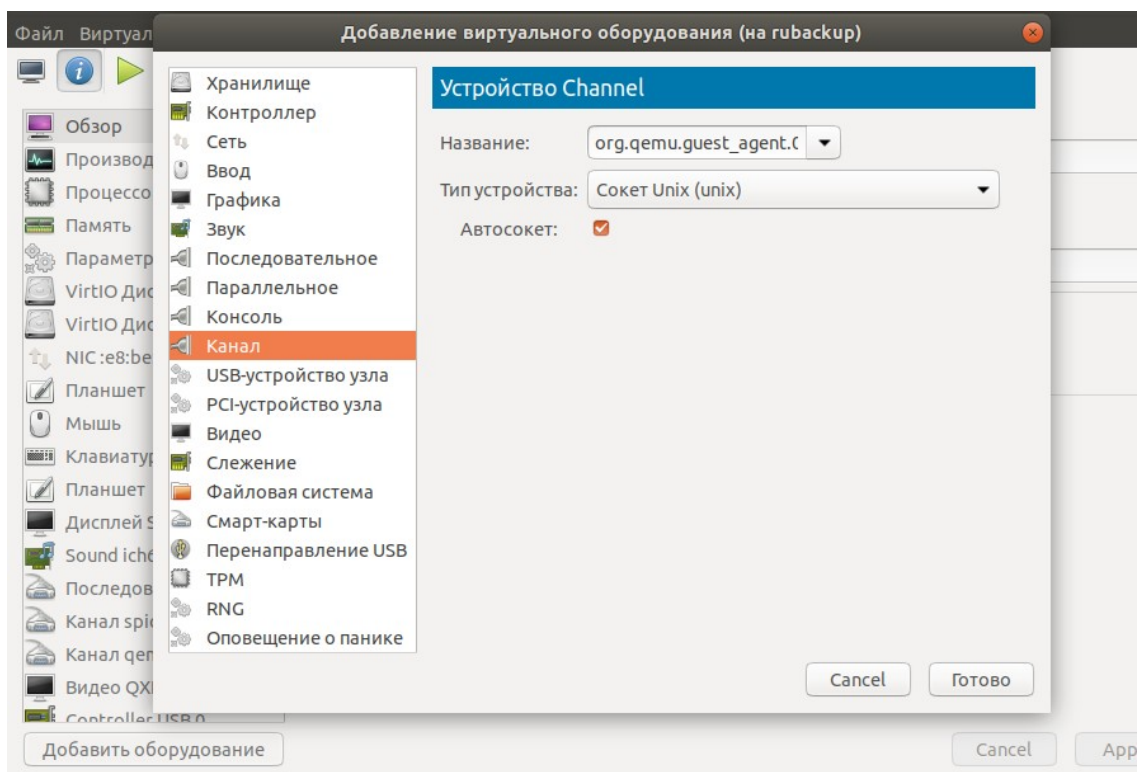


Рисунок 1

2. В операционной системе Linux виртуальной машины необходимо установить пакет *qemu-guest-agent*.

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


или

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

3. Для операционной системы Windows с диска *virtio-win* необходимо установить пакет *qemu-ga* из папки *guest-agent*, которая находится в корне диска.
4. В операционной системе виртуальной машины необходимо установить гостевые расширения из диска Virtio-Win. Для этого:
 - 1) Добавьте ISO-образ с гостевыми расширениями в операционную систему виртуальной машины как виртуальный CD-ROM.
 - 2) В виртуальной машине откройте подключенный виртуальный CD-ROM.
 - 3) Запустите файл *virtio-win-gt-x64*.
 - 4) Используя мастер установки, установите QEMU Guest Agent и SPICE agent.
 - 5) Перезагрузите виртуальную машину.

Диск Virtio-Win доступен для скачивания по ссылке:

<https://fedorapeople.org/groups/virt/virtio-win/direct->

[downloads/latest-virtio/virtio-win.iso](https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/latest-virtio/virtio-win.iso)

5. Для Astra Linux Смоленск необходимо использовать диск разработки и добавить соответствующий iso image в операционную систему как виртуальный CDROM. После этого:

```
# sudo apt-cdrom add
```

```
# sudo apt update
```

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

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 файле */opt/rubackup/etc/rubackup_restriction.list.kvm*.

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).

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 пр.

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&
```

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):

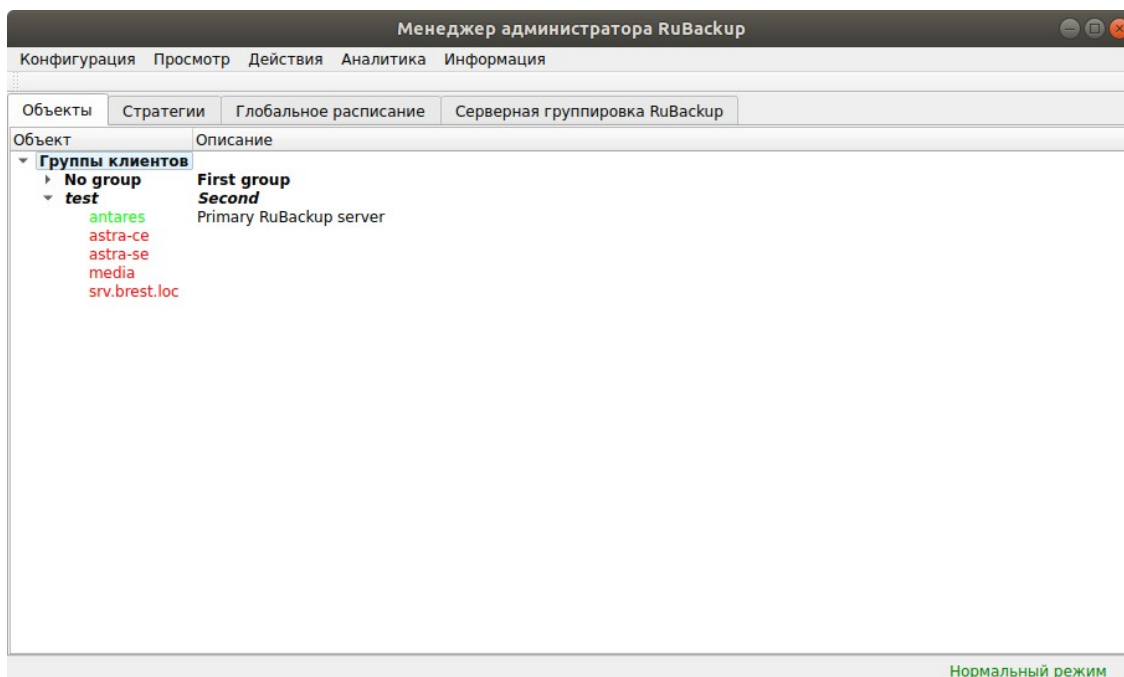


Рисунок 2

Для резервного копирования виртуальных машин KVM на хосте должен быть установлен клиент RuBackup и соответствующий модуль, обеспечивающий резервное копирование. 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 (рисунок 3). Все новые клиенты должны быть авторизованы в системе резервного копирования:

Для авторизации неавторизованного клиента в RBM выполните следующие действия:

- 1 Откройте меню **Действия** → **Клиенты** → **Авторизовать клиентов** (рисунок 3):

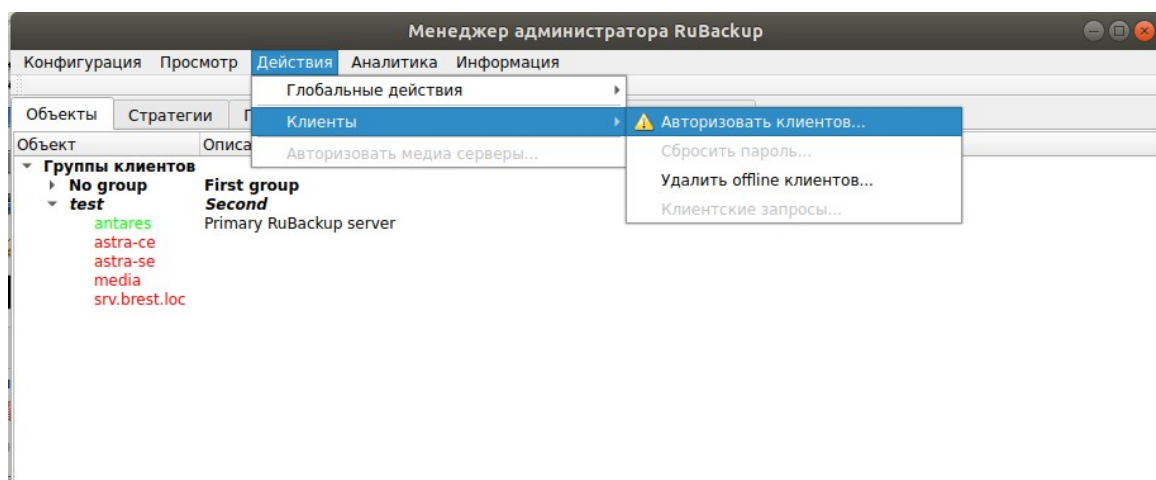


Рисунок 3

- 2 Выберите нужного неавторизованного клиента и нажмите **Авторизовать** (рисунок 4):

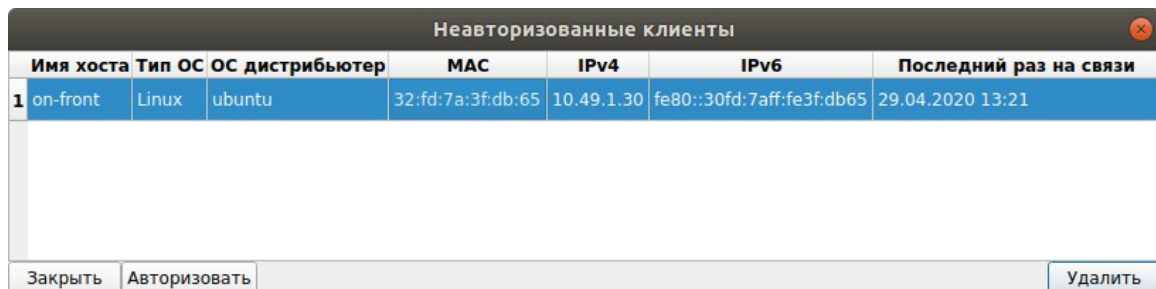


Рисунок 4

После авторизации новый клиент будет виден в главном окне RBM (рисунок 5):

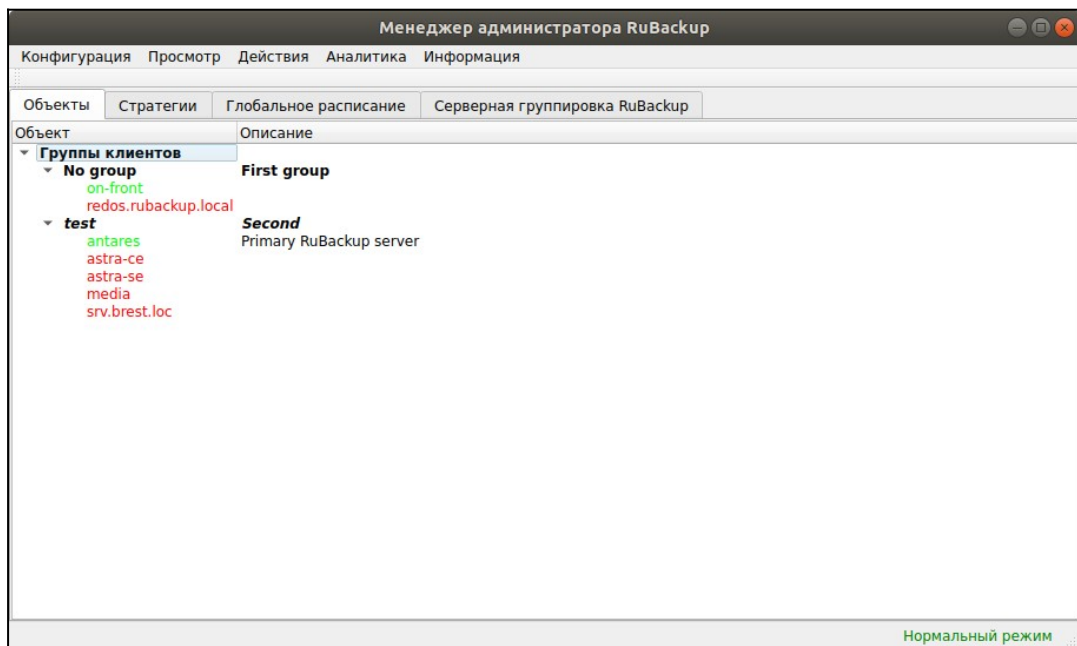


Рисунок 5

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*)

При помощи менеджера администратора RuBackup можно создать в глобальном расписании одно или несколько правил резервного копирования виртуальных машин гипервизора KVM.

Для этого необходимо выполнить следующие действия:

1. Выбрать клиентский хост, на котором установлен KVM и добавить правило резервного копирования (рисунок 6).

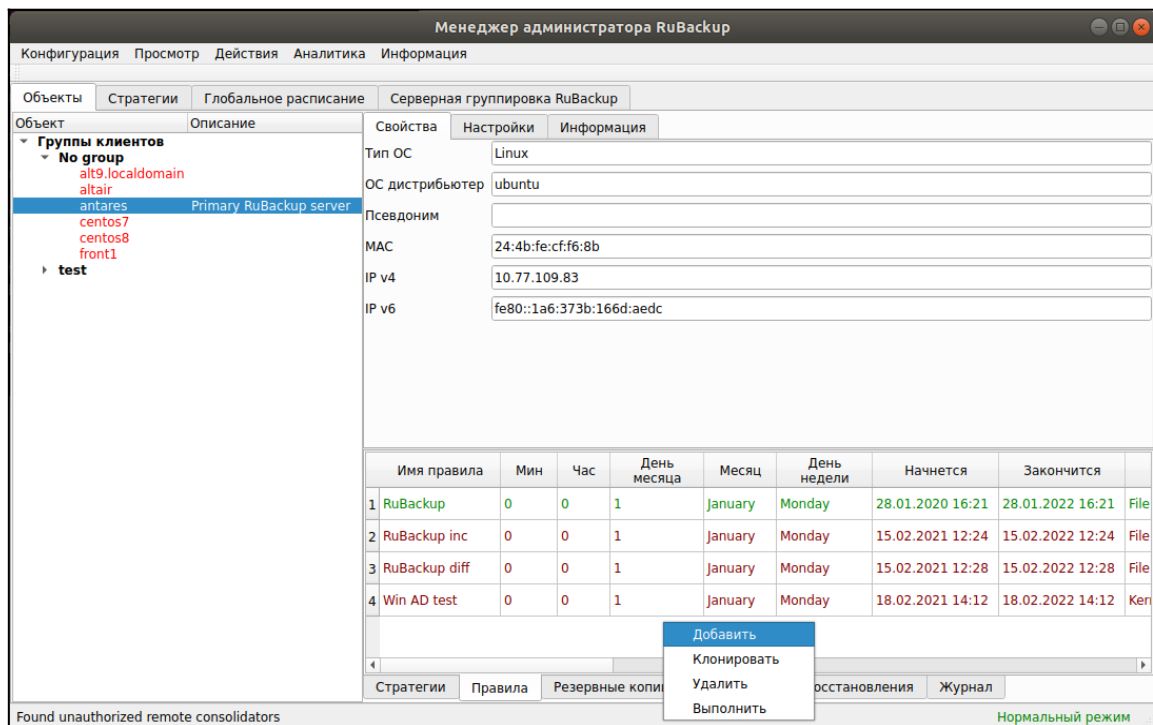


Рисунок 6

2. Выбрать тип ресурса «Kernel-based Virtual Machine (KVM)» (рисунок 7):

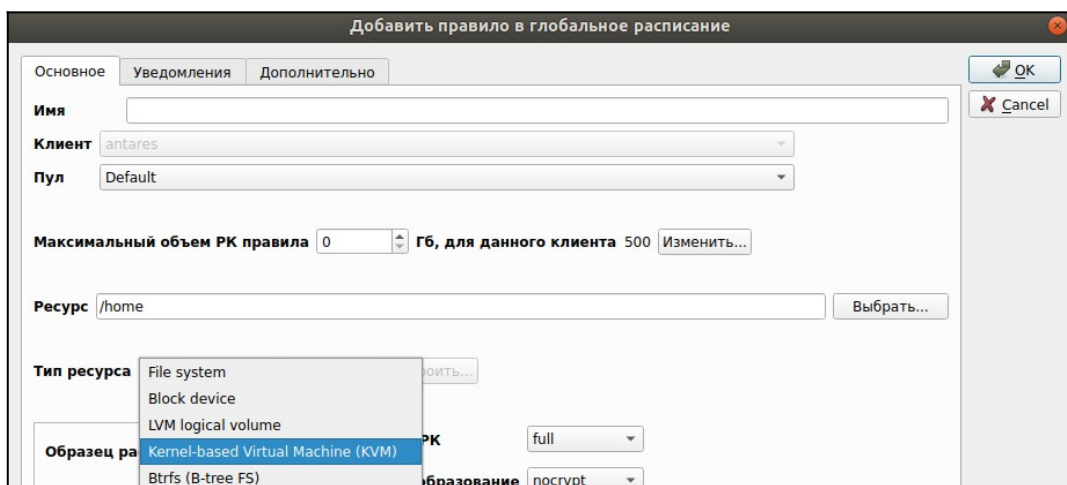


Рисунок 7

3. Выбрать ресурс, для которого будет выполняться правило (рисунок 8):

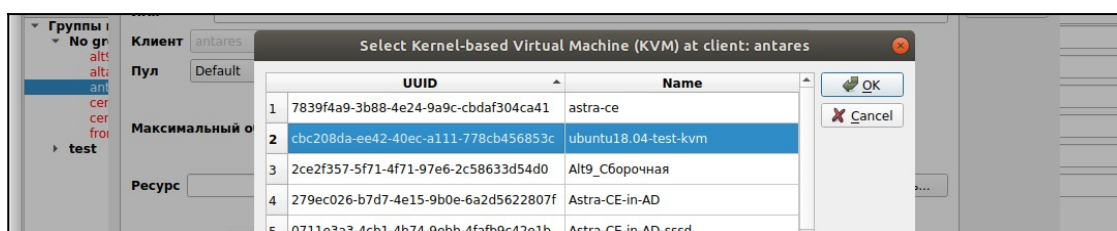
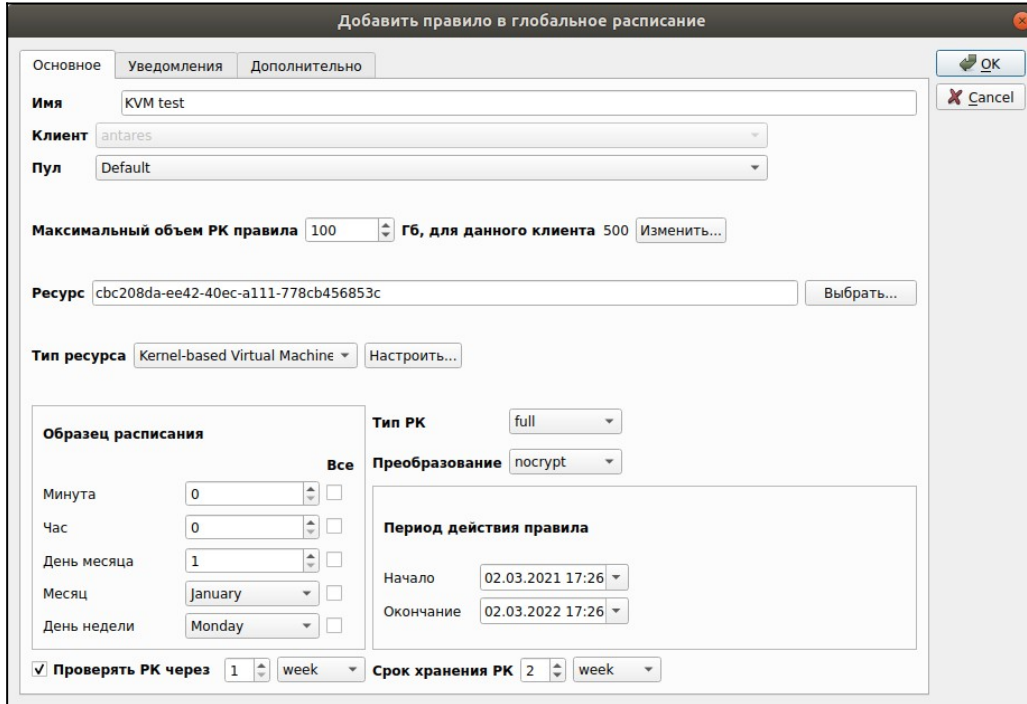


Рисунок 8

4. Установить прочие настройки: тип резервного копирования (Full), максимальный объем для резервных копий данного правила (100 Гб), срок хранения (2 недели), через какой промежуток времени требуется выполнить проверку резервной копии или не проверять её вовсе (рисунок 9).



Добавить правило в глобальное расписание

Основное | Уведомления | Дополнительно

Имя: KVM test

Клиент: antares

Пул: Default

Максимальный объем РК правила: 100 Гб, для данного клиента 500 Изменить...

Ресурс: cbc208da-ee42-40ec-a111-778cb456853c

Тип ресурса: Kernel-based Virtual Machine

Образец расписания:

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

Тип РК: full

Преобразование: nocrypt

Период действия правила:

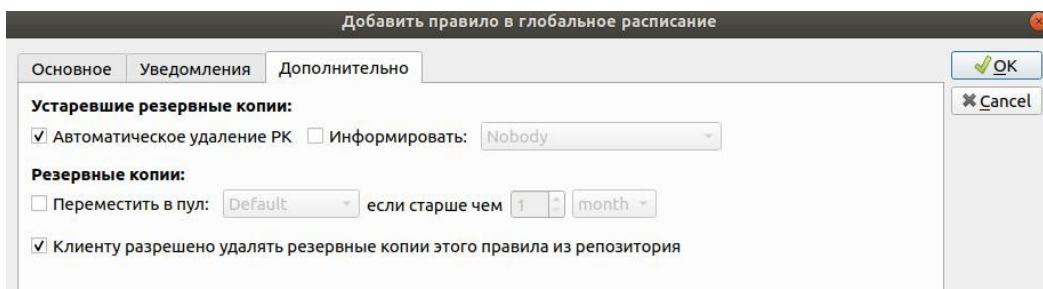
Начало	02.03.2021 17:26
Окончание	02.03.2022 17:26

Проверять РК через 1 week

Срок хранения РК: 2 week

Рисунок 9

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 10).



Добавить правило в глобальное расписание

Основное | Уведомления | Дополнительно

Устаревшие резервные копии:

<input checked="" type="checkbox"/> Автоматическое удаление РК	<input type="checkbox"/> Информировать: Nobody
--	--

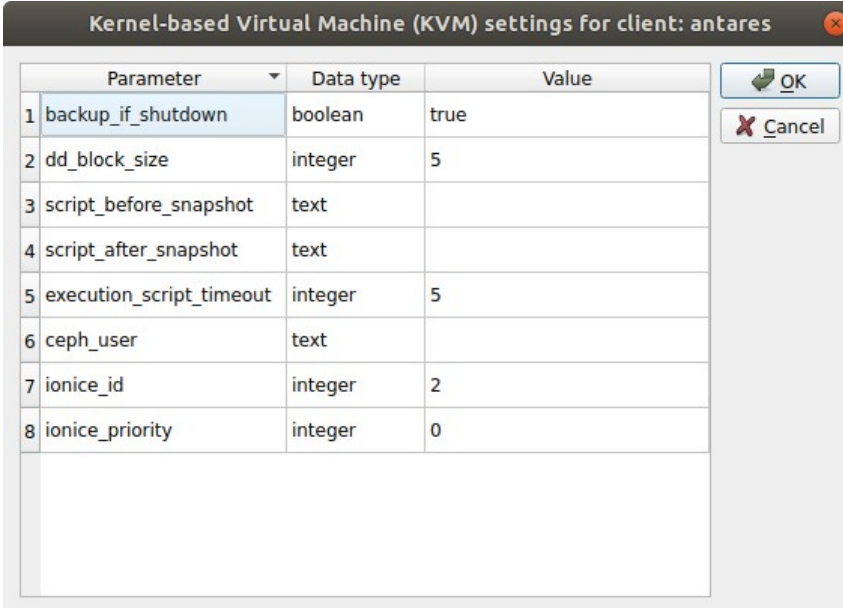
Резервные копии:

<input type="checkbox"/> Переместить в пул: Default	если старше чем 1 month
---	-------------------------

Клиенту разрешено удалять резервные копии этого правила из репозитория

Рисунок 10

Правила для выполнения резервных копий виртуальных машин могут иметь следующие дополнительные настройки (рисунок 11).



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 ceph_user	text	
7 ionice_id	integer	2
8 ionice_priority	integer	0

Рисунок 11

Описание параметров и их значения представлены в таблице 2.

Таблица 2 – Описание параметров настройки

Параметр	Описание	Значение по умолчанию	Допустимые значения
backup_if_shutdown	Выполнять ли резервное копирование, если VM выключена	true	true, false
dd_block_size	Размер блока (в МБ) для операций DD	5	>1
script_before_snapshot	Скрипт внутри VM, который будет выполнен перед операцией мгновенного снимка		
script_after_snapshot	Скрипт внутри VM, 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
ceph_user	Ceph user		
ionice_id	Класс ввода-вывода для текущего процесса. Значение 1 - real time, 2 - best effort, 3 - idle	2	1...3
ionice_priority	Приоритет ввода-вывода. Меньшее значение означает больший приоритет. Применяется к классам 1-2 (параметр ionice_id)	0	0...7

Если дополнительными настройками не заданы скрипты, которые должны быть выполнены в виртуальной машине перед и после создания моментального снимка (снэпшота), но в виртуальной машине существует исполняемый скрипт `/opt/rubackup/scripts/rubackup-kvm.sh`, то перед созданием снимка он будет выполнен с параметром `before`, а после создания снимка он будет выполнен с параметром `after`. Значение таймаута в этом случае равняется 5 сек.

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) Для виртуальных машин KVM в дополнительных настройках правила резервного копирования возможно задать выполнение скрипта непосредственно перед созданием снимка виртуальной машины KVM и непосредственно сразу после создания снимка виртуальной машины KVM;

5) Выполнить преобразование резервной копии на клиенте.

6) Периодически выполнять проверку целостности резервной копии.

7) 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.

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

9) 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.

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 пользователю 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@srv.brest.loc  
  
# /opt/rubackup/bin/rbc&
```

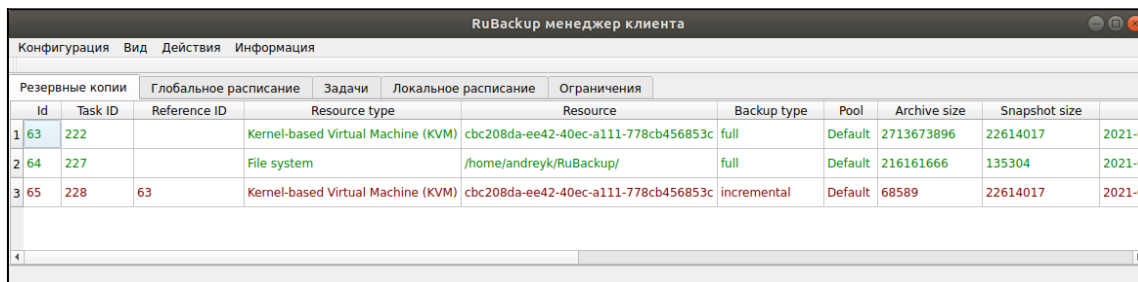
Пользователь, запускающий RBC, должен входить в группу rubackup.

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 12). 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	63	222	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c		full	Default	2713673896	22614017	2021-0	
2	64	227	File system	/home/andreyk/RuBackup/		full	Default	216161666	135304	2021-0	
3	65	228	63	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c		incremental	Default	68589	22614017	2021-0

Рисунок 12

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 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, /gubackup-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).

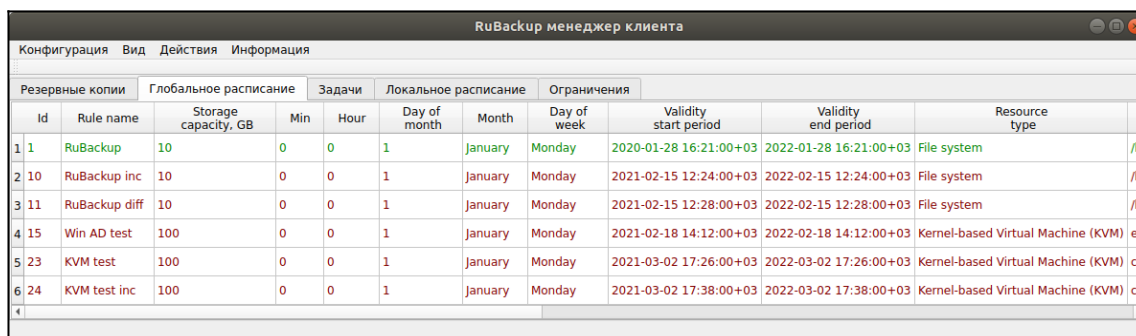
Проверить резервную копию.

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 13).



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 13

Во вкладке «Глобальное расписание» пользователю доступны следующие действия:

Запросить новое правило.

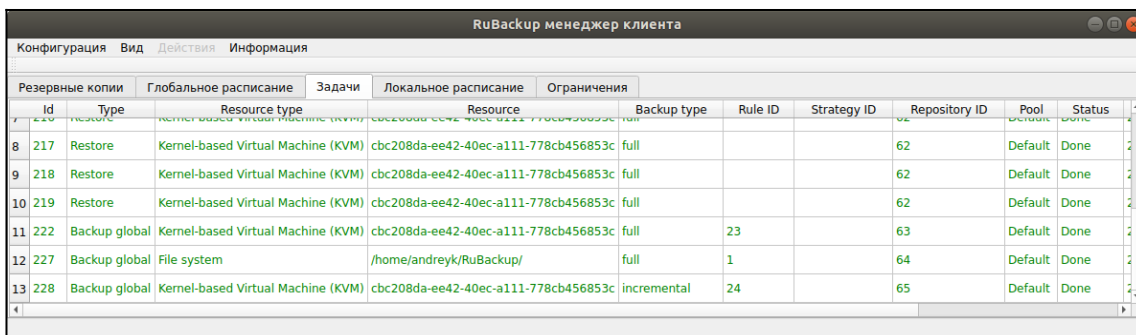
Это действие вызывает диалог подготовки нового правила в глобальном расписании RuBackup для данного клиента. Запрос на добавление правила требует одобрения администратора RuBackup, одобрение может быть сделано в оконном менеджере администратора RuBackup.

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 14). 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 14

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_archives

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

```
andreyk@antares:~$ rb_archives
```

Id	Ref ID	Resource	Resource type	Backup type	Created	Crypto	Signed	Status
63		cbc208da-ee42-40ec-a111-778cb456853c	Kernel-based Virtual Machine (KVM)	full	2021-03-02 17:37:23+03	nocrypt	True	Trusted
64		/home/andreyk/RuBackup/	File system	full	2021-03-02 17:37:46+03	nocrypt	True	Trusted
65	63	cbc208da-ee42-40ec-a111-778cb456853c	Kernel-based Virtual Machine (KVM)	incremental	2021-03-02 17:39:07+03	nocrypt	True	Not Verified

rb_schedule

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

```
andreyk@antares:~$ rb_schedule
```

Id	Name	Resource type	Resource	Backup type	Status
1	RuBackup	File system	/home/andreyk/RuBackup/	full	run
10	RuBackup inc	File system	/home/andreyk/RuBackup/	incremental	wait
11	RuBackup diff	File system	/home/andreyk/RuBackup/	differential	wait
15	Win AD test	Kernel-based Virtual Machine (KVM)	e5e0f18f-2f68-4fca-b98f-a24b8d9f4902	full	wait
23	KVM test	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	full	wait
24	KVM test inc	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	incremental	wait

rb_tasks

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

```
andreyk@antares:~$ rb_tasks
```

Id	Task type	Resource	Backup type	Status	Created
207	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 17:52:39+03
209	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 18:11:09+03
212	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 18:26:15+03
213	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 18:27:15+03
214	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 18:49:17+03
215	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 18:51:22+03
216	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 18:53:48+03
217	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 18:56:16+03
218	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 19:07:40+03
219	Restore	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-01 19:08:22+03
222	Backup global	cbc208da-ee42-40ec-a111-778cb456853c	full	Done	2021-03-02 17:36:46+03
227	Backup global	/home/andreyk/RuBackup/	full	Done	2021-03-02 17:37:43+03
228	Backup global	cbc208da-ee42-40ec-a111-778cb456853c	incremental	Done	2021-03-02 17:38:56+03

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

Восстановление резервной копии виртуальной машины

Непосредственное восстановление виртуальных машин в KVM при помощи RuBackup возможно для таких виртуальных машин, диски которых располагаются в файловой системе и используют формат *qcow2*. Если для дисков виртуальной машины используются блочные устройства или устройства *Scsi*, то резервная копия будет восстановлена в каталоге в виде набора файлов виртуальной машины (конфигурационный *xml*-файл и образы дисков виртуальной машины), которые можно импортировать в среду виртуализации вручную.

Клиент может осуществить восстановление данных резервной копии в окне Менеджера Клиента RuBackup (RBC), либо при помощи утилиты командной строки *rb_archives*.

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

Восстановление резервной копии в RBC

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

1. Выделить нужную резервную копию и в контекстном меню выбрать «Восстановить» (рисунок 15):

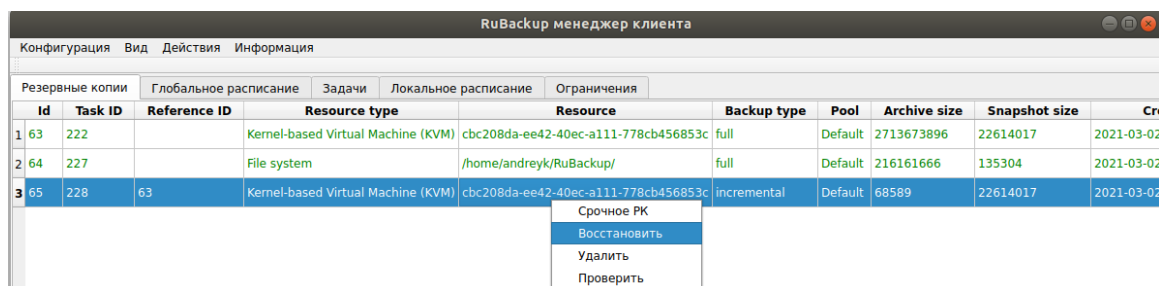


Рисунок 15

2. Ввести пароль клиента и далее RBC выведет информационное сообщение о дальнейших действиях (рисунок 16):

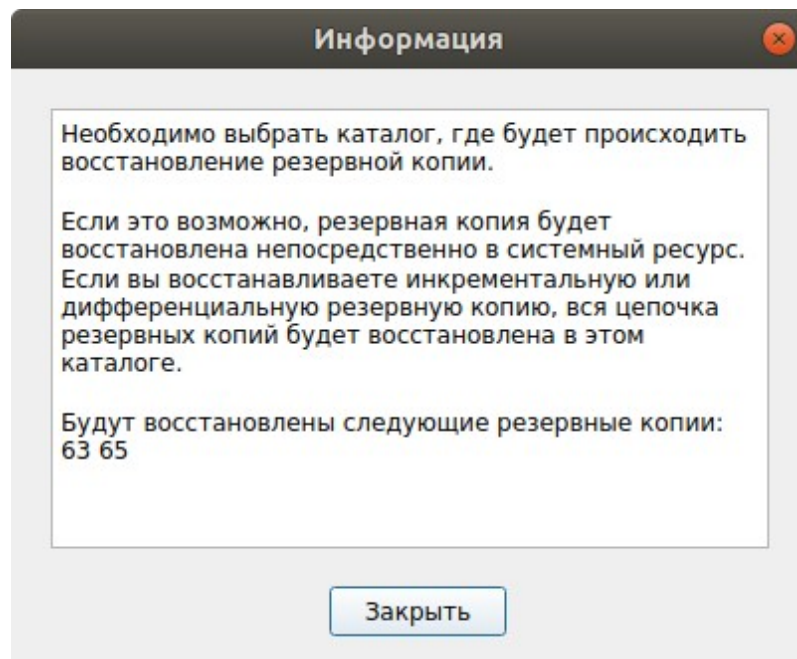


Рисунок 16

3. Указать место восстановления резервной копии (рисунок 17):

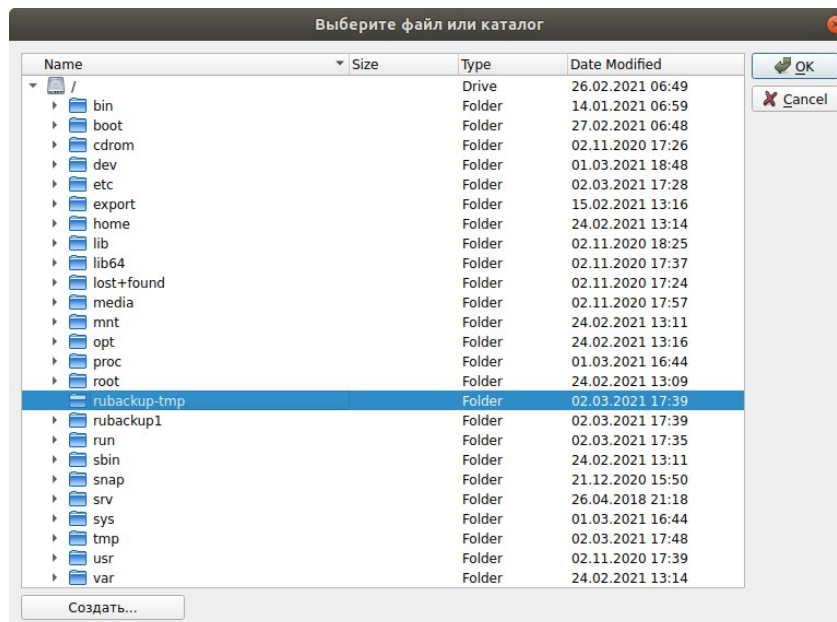


Рисунок 17

4. Далее появится информационное сообщение о создании задачи на восстановление (рисунок 18):

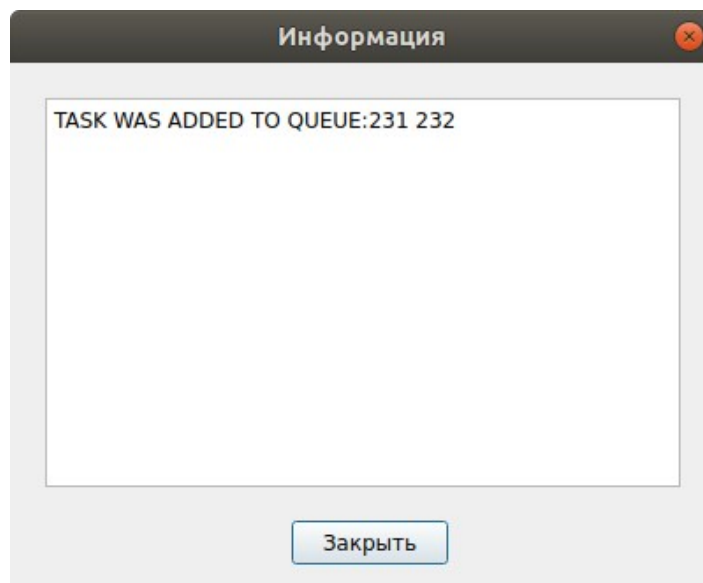
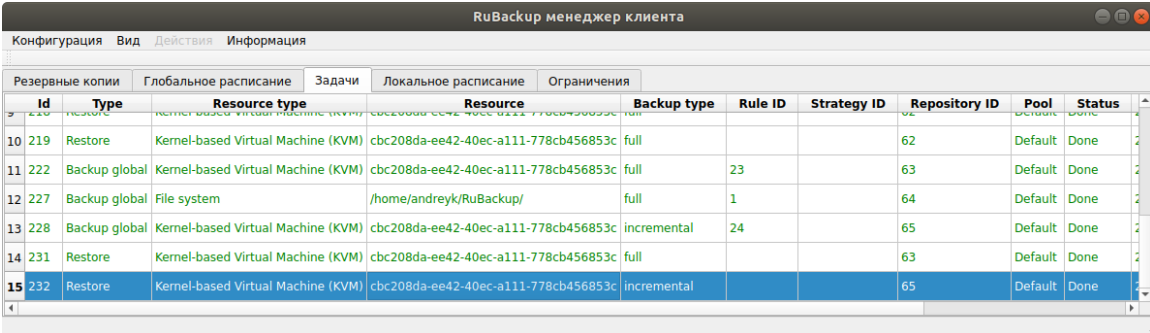


Рисунок 18

5. Проконтролировать результат процесса восстановления можно после автоматического переключения RBC на вкладку «Задачи» (рисунок 19):



	Id	Type	Resource type	Resource	Backup type	Rule ID	Strategy ID	Repository ID	Pool	Status
9	219	Restore	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	full			62	Default	Done
10	222	Backup global	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	full	23		63	Default	Done
11	227	Backup global	File system	/home/andreyk/RuBackup/	full	1		64	Default	Done
12	228	Backup global	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	incremental	24		65	Default	Done
13	231	Restore	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	full			63	Default	Done
14	232	Restore	Kernel-based Virtual Machine (KVM)	cbc208da-ee42-40ec-a111-778cb456853c	incremental			65	Default	Done

Рисунок 19

После выполнения восстановления в KVM появится новая виртуальная машина, полностью идентичная той, которая была в системе в момент резервного копирования.

Восстановление при помощи утилиты rb_archives

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

```
# rb_archives
```

```
andreyk@antares:/rubackup-tmp$ rb_archives
-----|-----|-----|-----|-----|-----|
Id | Ref ID | Resource | Resource type | Backup type | Created |
-----|-----|-----|-----|-----|-----|
63 | | cbc208da-ee42-40ec-a111-778cb456853c | Kernel-based Virtual Machine (KVM) | full | 2021-03-02 17:37:23+03 |
64 | | /home/andreyk/RuBackup/ | File system | full | 2021-03-02 17:37:46+03 |
65 | 63 | cbc208da-ee42-40ec-a111-778cb456853c | Kernel-based Virtual Machine (KVM) | incremental | 2021-03-02 17:39:07+03 |
```

```
# rb_archives -x 63
```

```
Password:
```

```
----> Restore archive chain: 63 < ----
```

```
Record ID: 63 has status: Trusted
```

```
TASK WAS ADDED TO QUEUE:233
```

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

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

```
Tue Mar  2 17:58:29 2021: Try new name: ubuntu18.04-test-kvm-0
Tue Mar  2 17:58:29 2021: The name: ubuntu18.04-test-kvm-0 is free.
Use it
Tue Mar  2 17:58:29 2021: Found disk in XML file. Type: file device:
disk
Tue Mar  2 17:58:29 2021: Source file:
/var/lib/libvirt/images/ubuntu18.04_clean-clone-3-clone-clone-1.qcow2
Tue Mar  2 17:58:29 2021: File is exists:
/var/lib/libvirt/images/ubuntu18.04_clean-clone-3-clone-clone-1.qcow2
Tue Mar  2 17:58:29 2021: Try new filename:
/var/lib/libvirt/images/ubuntu18.04_clean-clone-3-clone-clone-1-0.qcow2
Tue Mar  2 17:58:29 2021: ----->> Direct restore
<<-----
Tue Mar  2 17:58:37 2021: New domain was defined from XML file:
/rubackup-tmp/cbc208da-ee42-40ec-a111-778cb456853c/cbc208da-ee42-40ec-
a111-778cb456853c.xml
Tue Mar  2 17:58:37 2021: Task was done. ID: 234
Tue Mar  2 17:58:37 2021: Task ID: 234. New status: Done
```

После выполнения восстановления в KVM появится новая виртуальная машина, полностью идентичная той, которая была в системе в момент резервного копирования. В том случае, если виртуальная машина с таким же именем уже присутствует в KVM, то новая виртуальная машина будет восстановлена с определенным постфиксом в ее имени, например -0;

В том случае, если необходимо восстановить файлы виртуальной машины без развертывания ее в KVM, то можно воспользоваться опцией -X:

```
rb_archives -X 63
Password:
----> Restore archive chain: 63 < ----
Record ID: 63 has status: Trusted
TASK WAS ADDED TO QUEUE:235
```

В этом случае файлы виртуальной машины будут восстановлены в текущий каталог, из которого была выполнена команда `rb_archives`:

```
sudo ls -l cbc208da-ee42-40ec-a111-778cb456853c/
итого 5913168
-rw----- 1 root root          5475 map  2 17:36 cbc208da-ee42-40ec-a111-
778cb456853c.xml
-rw----- 1 root root           116 map  2 17:36 target_list
-rw----- 1 root root 6055067648 map  2 17:36 ubuntu18.04_clean-clone-
3-clone-clone-1.qcow2
```


Операции над ВМ, восстановленной без развертывания

При восстановлении резервной копии без развертывания она будет восстановлена в выбранный пользователем каталог. При использовании утилиты `rb_archives` (см. опцию `-X`) она будет восстановлена в локальный каталог, либо же в тот, который был задан опцией `-d`.

В выбранном пользователем пути будет создан каталог с именем восстанавливаемой виртуальной машины со следующим содержимым:

- 1) конфигурационный файл виртуальной машины в формате `xml`;
- 2) файлы дисков виртуальной машины.

С целью немедленной проверки восстановленной резервной копии для различных типов файлов необходимо выполнить описанные ниже действия.

Для примера рассмотрим восстановленную виртуальную машину KVM с именем `small`. Файлы резервной копии ВМ были восстановлены в каталог `/kvm/small`.

XML-файл конфигурации ВМ `small.xml` содержит следующее описание базовой конфигурации виртуальной машины:

```
<domain type='kvm'>
<name>small</name>
<uuid>3b42f58f-9fe5-4012-b7d0-2f29a208526e</uuid>
<memory unit='KiB'>2097152</memory>
<currentMemory unit='KiB'>2097152</currentMemory>
<vcpu placement='static'>1</vcpu>
<os>
<type arch='x86_62' machine='pc-i440fx-bionic'>hvm</type>
</os>
```

Внесите следующие изменения в XML-файл конфигурации:

- 1) Удалите строку с UUID (выделена жирным в примере выше).

2) Измените имя домена между тегами <name> и </name>. Например, на <name>small-restored</name>.

Следуйте действиям ниже в зависимости от формата файлов дисков восстановленной ВМ.

А) Файлы дисков виртуальной машины в формате qcow2

Для примера рассмотрим XML-файл конфигурации ВМ, содержащий следующее описание диска в формате qcow2:

```
<disk type='file' device='disk'>
<driver name='qemu' type='qcow2' />
<source file='/var/lib/libvirt/images/small.qcow2' />
<target dev='vda' bus='virtio' />
<boot order='1' />
<address type='pci' domain='0x0000' bus='0x00' slot='0x07'
function='0x0' />
</disk>
```

1. Изменить пути доступа к файлам виртуальной машины в xml файле конфигурации:

```
<source file='/var/lib/libvirt/images/small.qcow2' />
```

на

```
<source file='/kvm/small/small.qcow2' />
```

```
<disk type='file' device='disk'>
<driver name='qemu' type='qcow2' />
<source file='/kvm/small/small.qcow2' />
<target dev='vda' bus='virtio' />
<boot order='1' />
<address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0' />
</disk>
```

При этом необходимо, чтобы данный каталог был разрешён для хранения данных KVM.

2. Проверить запуск виртуальной машины:

```
# virsh create small.xml
```

Для создания виртуальной машины используйте:

```
# virsh define small.xml
```

Б) Файлы дисков виртуальной машины в формате raw

В данном случае есть два пути (предположим, что файлы находятся в /kvm/small):

1. Восстановить файлы дисков виртуальной машины в подходящее raw устройство с помощью команды dd, например:

```
# dd if=/kvm/small/sde1 of=/dev/sde1 bs=5M
```

Далее изменить пути доступа к raw устройствам в XML-файле конфигурации виртуальной машины.

2. Другой путь – это конвертировать восстановленные файлы raw устройств в qcow2 формат при помощи команды qemu-img convert, например:

```
qemu-img convert -f qcow2 -O raw /kvm/small/sde1  
/kvm/small/image.qcow2
```

Далее изменить пути доступа к raw устройствам в XML-файле конфигурации виртуальной машины..

3. Запустить виртуальную машину:

```
# virsh create small.xml
```

После проверки функционирования восстановленной виртуальной машины системный администратор должен принять решение о том, куда именно должны быть размещены файлы восстановленной виртуальной машины в рабочую конфигурацию KVM.

В) Файлы дисков виртуальной машины в raw формате находились в хранилище Ceph в rados block device

В данном случае необходимо внести изменения в xml файл:

1. Для возможности запуска виртуальной машины с локальным образом изменить секцию, заключенную в теги <disk> ... </disk> следующим образом:

```
<disk type='file' device='disk'>  
  
<driver name='qemu' type='raw' />  
  
<source file='path_to_restored_image' />  
  
<backingStore />
```

2. Удалить секцию <auth> ... </auth>

3. Удалить секцию

```
<source protocol='rbd' .... </source>
```

4. Запустить виртуальную машину для проверки:

```
# virsh create small.xml
```

Для создания виртуальной машины используйте команду:

```
# virsh define small.xml
```

После проверки функционирования восстановленной виртуальной машины системный администратор должен принять решение о том, куда именно должны быть размещены файлы восстановленной виртуальной машины в рабочую конфигурацию KVM.