

**RuBackup**

Система резервного копирования и восстановления данных

# Резервное копирование и восстановление OpenStack



**RuBackup**

Версия 2.1

05.04.2024 г.

# Содержание

Введение.....	3
Установка клиента RuBackup.....	5
Настройка SSH-доступа.....	13
Удаление клиента RuBackup.....	14
Мастер-ключ.....	15
Защитное преобразование резервных копий.....	16
Алгоритмы защитного преобразования.....	17
Использование менеджера администратора RuBackup (RBM).....	18
Запуск RBM.....	18
Регулярное резервное копирование виртуальной машины.....	22
Срочное резервное копирование.....	29
Централизованное восстановление резервных копий.....	31
Восстановление со стороны клиента.....	36

## Введение

Система резервного копирования RuBackup позволяет выполнять резервное копирование и восстановление виртуальных машин платформы виртуализации OpenStack. Доступно полное, инкрементальное и дифференциальное резервное копирование. Также возможно выполнять резервное копирование с использованием дедупликации и хранить резервные копии в дедуплицированном хранилище.

**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.

Резервное копирование виртуальных машин OpenStack выполняется безагентным способом. Это означает, что в виртуальную машину, для которой предполагается создание резервной копии, не устанавливается агент RuBackup (однако требуется установка гостевых расширений операционной системы, например qemu-guest-agent). Резервное копирование виртуальной машины выполняется целиком, для всех дисков виртуальной машины. В ходе резервного копирования во всех случаях из резервной копии удаляются дублирующие блоки (всегда выполняется локальная дедупликация).

В случае передачи резервной копии в хранилище дедуплицированных резервных копий всегда происходит передача только тех уникальных блоков (для того же типа источника данных), которых еще нет в хранилище.

Для выполнения резервного копирования виртуальных машин среды виртуализации OpenStack необходимо установить клиент резервного копирования RuBackup по одной из следующих схем:

- на одну из виртуальных машин в данной среде виртуализации, для которой настроен доступ к гипервизору (гипервизорам);
- на несколько виртуальных машин в данной среде виртуализации, если это обусловлено необходимостью динамически распределять нагрузку в ходе резервного копирования или обеспечить возможность вывода той или иной виртуальной машины из эксплуатации без изменений в расписании

резервного копирования (в данной схеме необходимо включить эти гипервизоры в кластерную группу клиентов системы резервного копирования);

При выполнении резервного копирования применяется технология создания моментальных снимков данных для дисков виртуальной машины, что позволяет не останавливать работу на время резервного копирования.

Перед созданием снимка и сразу после его создания RuBackup может выполнить скрипт внутри виртуальной машины для того, чтобы иметь возможность привести данные приложений внутри виртуальной машины в консистентное состояние.

Также внутри виртуальной машины может быть создан скрипт, располагающийся в файле `/opt/rubackup/scripts/openstack.sh`. В том случае, если внутри виртуальной машины существует такой файл с атрибутами на исполнение, то перед созданием моментального снимка он будет выполнен с аргументом `before`, а сразу после создания моментального снимка он будет выполнен с аргументом `after`.

## Установка клиента RuBackup

Для возможности резервного копирования виртуальных машин среды виртуализации OpenStack необходимо установить клиент RuBackup на одну из виртуальных машин в среде виртуализации OpenStack, находящихся под управлением операционной системы Astra Linux 1.7, Ubuntu 18.04 или 20.04, Debian 10 и для которой настроен доступ к гипервизору (гипервизорам). Сюда же необходимо установить модуль `rb_module_openstack` из пакета `rubackup-openstack.deb` (см. дистрибутив для ОС Debian 10).

Установка пакетов клиента RuBackup производится из-под учетной записи с административными правами при помощи следующих команд (имена пакетов могут отличаться в зависимости от используемой операционной системы):

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

```
# sudo dpkg -i rubackup-openstack.deb
```

Подробно процедура установки клиента описана в документе «Руководство по установке и обновлению серверов резервного копирования и Linux-клиентов RuBackup».

В ходе инсталляции пакета в системе будет создан файл настроек доступа системы резервного копирования к API OpenStack `/opt/rubackup/etc/rb_module_openstack.conf`:

```
# Symbol "#" at the beginning of the line treats as a comment
# "#" in the middle of the line treats as a parameter value
# So please do not use comments in one line with parameter

# Mandatory parameters
# Get config URLs at https://<OPENSTACK_WEBUI_IP>/dashboard/project/api_access/
# or https://msk.cloud.vk.com/app/<PROJECT>/project/endpoints
identity_url http://<OPENSTACK_WEBUI_IP>:5000/v3/
compute_url http://<OPENSTACK_WEBUI_IP>:8774/v2.1/<PROJECT_ID>/
volume_url http://<OPENSTACK_WEBUI_IP>:8776/v3/<PROJECT_ID>/
project_id <PROJECT_ID>
# User name on behalf of which the API requests will proceed
username <user name>
# Password to be used with 'username' to authenticate in API
password <user password>
# Domain name to be used with 'username' and 'password' to authenticate in API
domain <domain name>
# REST operations timeout, seconds
# minimum 1, maximum 300, default 5
timeout 20
# ID of VM in Openstack platform where current module is deployed - can be obtained from instance
info in WEB GUI
```

```
rubackup-vm-id <vm id>
##
## Optional parameters:
# Admin user account info of OPENSTACK is required to run scripts inside the target VM
admin_name <admin name>
admin_password <admin password>
#
## Name of admin's project, optional
## If this value is not set, project_id value will be used instead as admin's project
admin_project_name NONE
## Name of admin's project domain, optional
admin_project_domain_name NONE
#
# If certificate info is not specified the module will connect to API w/o certificate verification
enable_ssl no
ca_info <path to cert>
# Turn on debug of REST requests
curl_verbose no
##
## Transport to execute remote scrips: before_backup, after_backup
# possible values: virsh, ssh
# default value: virsh
script_transport virsh
##
## User name for ssh transport
ssh_user rubackup_service_user
## Connection timeout for ssh transport, seconds
# minimum 1, maximum 300, default 5
ssh_connection_timeout 30
## ssh key file for ssh transport, full path only!
ssh_key_file /root/my_keys/my_key_file
# Project's region, optional
region NONE
```

Измените в этом файле настройки для подключения к API:

- identity\_url: URL до сервиса Identity
- compute\_url: URL до сервиса Compute
- volume\_url: URL до сервиса Volume

Конкретные значения для данных параметров можно взять со страницы [https://<OpenStack\\_WEBUI\\_IP>/dashboard/project/api\\_access/](https://<OpenStack_WEBUI_IP>/dashboard/project/api_access/) - WEB UI предоставленной инсталляции сервиса OpenStack (рисунок 1).

## API Access

Displaying 6 items

Service	Service Endpoint
Compute	https://10.177.32.119:8774/v2.1
Identity	https://10.177.32.119:5000
Image	https://10.177.32.119:9292
Network	https://10.177.32.119:9696
Placement	https://10.177.32.119:8780
Volumev3	https://10.177.32.119:8776/v3/2e2969139f7f402aba83b46d60f5a7db

Displaying 6 items

Рисунок 1

- `project_id`: ID проекта платформы OpenStack
- `username`: имя пользователя платформы OpenStack
- `password`: пароль пользователя платформы OpenStack
- `domain`: имя домена платформы OpenStack
- `timeout`: максимально допустимое время выполнения REST-запросов к платформе OpenStack в секундах. Значение по умолчанию - 5 секунд.
- `rubackup-vm-id`: id виртуальной машины, на которой установлен клиент RuBackup с установленным модулем Openstack

Узнать id виртуальной машины можно, в том числе запустив модуль `rb_module_openstack` с аргументом `-l`. Команда примет следующий вид:

```
# /opt/rubackup/modules/rb_module_openstack -l
```

Также можно воспользоваться утилитой `rb_archives`:

```
# rb_archives -t openstack
```

- `admin_name`: имя администратора платформы OpenStack. Необходимо для выполнения скриптов
- `admin_password`: пароль пользователя платформы OpenStack
- `enable_ssl`: включить проверку ssl-сертификатов

- `ca_info`: путь до сертификата

**Внимание!** При настройке клиента рекомендуется включить функцию централизованного восстановления в тех случаях, когда предполагается восстановление виртуальной машины из средства управления RBM.

При старте клиента RuBackup в журнальном файле `/opt/rubackup/log/RuBackup.log` на клиенте появится следующая запись:

```
Try to check module: 'OPENSTACK' ...
Execute OS command: /opt/rubackup/modules/rb_module_openstack -t 2>&1
Module version: 2.1
[2023-06-16 06:25:19] Info: Identity API version: v3.14
[2023-06-16 06:25:19] Info: Compute version: 2.95
[2023-06-16 06:25:19] Info: Volume version: 3.70
... module 'OPENSTACK' was checked successfully. Module supports archiving
```

В ручном режиме проверить правильность настроек можно при помощи следующей команды:

```
# /opt/rubackup/modules/rb_module_openstack -t
```



## Обновление конфигурационного файла

При необходимости вы можете обновить модуль резервного копирования Openstack. При этом обновится конфигурационный файл модуля.

В конфигурационном файле модуля присутствуют следующие необязательные параметры:

```
##
## Transport to execute remote scrips: before_backup, after_backup
# possible values: virsh, ssh
# default value: virsh
#script_transport virsh
##
## User name for ssh transport
#ssh_user rubackup_service_user
## Connection timeout for ssh transport, seconds
# minimum 1, maximum 300, default 5
#ssh_connection_timeout 30
## ssh key file for ssh transport, full path only!
#ssh_key_file /root/my_keys/my_key_file
```

Пользователь при создании правила на бэкап может задать путь к скрипту внутри виртуальной машины. Скрипт, путь к которому указан в качестве значения параметра **script\_before\_snapshot**, будет выполнен на виртуальной машине, для которой выполняется бэкап, перед созданием снэпшотов дисков этой виртуальной машины.

Скрипт, путь к которому указан в качестве значения параметра **script\_after\_snapshot**, будет выполнен на виртуальной машине, для которой выполняется бэкап, после создания снэпшотов дисков этой виртуальной машины.

Выполняемые действия для этих скриптов определяют пользователи.

Значение скрипт-транспорта **virsh** ставится по умолчанию. При этом, если необходима функция запуска скриптов на целевой виртуальной машине, выполните следующие требования::

- 
- :

1. Настройте конфигурационный файл настроек доступа к API РУСТЭК **rb\_module\_rustack.conf** (подробнее в разделе «Установка Клиента RuBackup»), кроме основной пользовательской учетной записи, конфигурационный файл должен содержать учетную запись администратора.

2. Разверните виртуальную машину, для которой предполагается создание резервных копий, используйте гипервизор типа QEMU.

3. Для виртуальной машины, на которой развернут клиент RuBackup и модуль **rb\_module\_openstack**:

- установите пакет **libvirt-clients**;
- скопируйте SSH-ключ на хосты гипервизоров платформы виртуализации. Это необходимо для беспарольного SSH-подключения к виртуальной машины, на которой развернут клиент RuBackup на узлы гипервизоров платформы виртуализации;
- на виртуальной машине, для которой предполагается создание резервных копий, установите пакет **qemu-guest-agent**.

Если выбрать значение **ssh**, то для запуска скриптов на виртуальной машине, для которой создаётся резервная копия, вместо обращения к гипервизору будет использоваться подключение по SSH с узла, где установлен клиент и этот модуль. В этом случае пользователю необходимо задать новые параметры в конфигурационном файле, где:

- **ssh\_user** — пользователь от имени которого будет инициировано подключение к виртуальной машине через SSH для запуска на ней скрипта. Предполагается, что этому пользователю предоставлены внутри целевой виртуальной машины права на исполнение скриптов **script\_before\_snapshot**, **script\_after\_snapshot** или **/opt/rubackup/scripts/rustack.sh**.
- **ssh\_connection\_timeout** — максимальное время ожидания в секундах, в течении которого модуль ожидает успешное подключение по SSH к виртуальной машине. Если соединение не будет установлено в заданное время, задача на создание резервной копии завершится с ошибкой.
- **ssh\_key\_file** — полный путь к файлу SSH-ключа для пользователя, указанного опцией **ssh\_user**, который позволяет этому пользователю подключаться к виртуальным машинам (для которых предполагается создание резервных копий) в проекте OpenStack без ввода пароля.

Эта версия модуля содержит конфигурационный файл, параметры которого могут отличаться от текущей версии, поэтому при обновлении модуля на новую версию также обновляется и его конфигурационный файл. Для переноса значений параметров настроек из старого конфигурационного файла в новый предусмотрен механизм слияния конфигурационных файлов.

Может существовать 3 версии конфигурационного файла:

- **/opt/rubackup/etc/rb\_module\_openstack.conf** — текущий конфигурационный файл модуля. После слияния будет переименован в **rb\_module\_openstack\_old.conf**.
- **/opt/rubackup/etc/rb\_module\_openstack\_old.conf** — старый конфигурационный файл который был загружен в предыдущее обновление или при установке модуля.

- `/opt/rubackup/etc/rb_module_openstack_upgrade.conf` — конфигурационный файл обновления. Должен быть создан вручную.

Механизм слияния конфигурационных файлов запускается автоматически при обновлении пакета `deb` или `rpm`.

### Автоматическое обновление конфигурационного файла

Автоматическое обновление конфигурационного файла выполняется при обновлении пакетов `deb` или `rpm` и не требует действий от пользователя.

Порядок автоматического обновления:

1. Текущий конфигурационный файл `rb_module_openstack.conf` переименовывается в `rb_module_openstack_old.conf`.

2. Создается файл `/opt/rubackup/etc/rb_module_openstack.conf`, который далее будет использован в качестве текущего.

3. В созданный файл `rb_module_openstack.conf` добавляются параметры конфигурационного файла, которые поставляются в пакете `deb` или `rpm`. При этом все параметры закомментированы (выставлен символ `#` перед каждой строкой).

4. Происходит слияние старого конфигурационного файла, конфигурационного файла обновления, и нового конфигурационного файла, который поставляется в пакете, при этом:

- Значение каждого параметра берется из конфигурационного файла обновления.
- Если в конфигурационном файле обновления параметра нет, то значение берется из старого конфигурационного файла.
- Если в старом конфигурационном файле значение параметра отсутствует, то такое значение:
  - Добавляется, если это обязательный параметр. Добавляется без значения.
  - Не добавляется, если настройка не обязательная.
- Если у обязательного параметра нет значения, то при установке пакета возникнет ошибка. Информацию об ошибке можно посмотреть в логе установки:

```
[2024-03-18 12:11:52] Info: UpgradeConfig options.configs_list: /media/nik/Special/resource/test/ol
[2024-03-18 12:11:52] Error: Variable 'host' is mandatory and has not value. Module cannot be used
[2024-03-18 12:11:52] Error: Variable 'port' is mandatory and has not value. Module cannot be used
```

В результате автоматического обновления будет обновлен конфигурационный файл `rb_module_openstack.conf`. Модуль Openstack будет готов к работе.

При слиянии конфигурационных файлов будут удалены все комментарии из старого конфигурационного файла.

Если при обновлении конфигурационного файла возникли ошибки, то пользователю необходимо проверить корректность `/opt/rubackup/etc/rb_module_openstack.conf` и при необходимости заполнить параметры вручную.

## Настройка SSH-доступа

Для работы параметров «script\_before\_snapshot» и «script\_after\_snapshot» необходимо обеспечить беспарольный доступ для пользователя root с клиента RuBackup на узлы с гипервизором.

```
# ssh-keygen -t rsa
```

```
# cat /root/.ssh/id_rsa.pub
```

Этот публичный ключ нужно добавить в файл ~/.ssh/authorized\_keys на узлах с гипервизором для пользователя root.

## Удаление клиента RuBackup

Порядок удаления клиента RuBackup изложен в документе «Руководство по установке серверов резервного копирования и Linux-клиентов 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 потере 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
00000010 6284 54as 83a3 2053 4818 e183 1528 a343
00000020
```

# 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 картриджей ленточной библиотеки или на площадке провайдера облачного хранилища для ваших резервных копий).

Защитное преобразование осуществляется входящей в состав RuBackup утилитой `rbcrypt`. Ключ для защитного преобразования резервных копий располагается на хосте клиента в файле `/opt/rubackup/keys/master-key`. Защитное преобразование данных при помощи `rbcrypt` 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 `rbcrypt`. При таком режиме работы нет необходимости указывать алгоритм преобразования в правиле резервного копирования, иначе архив будет повторно преобразован с использованием мастер-ключа.



## Алгоритмы защитного преобразования

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

Таблица 1 – Алгоритмы защитного преобразования, доступные в утилите rbscrypt.

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	

# Использование менеджера 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 различными параметрами RuBackup.

## Запуск RBM

Для запуска RBM выполните команду:

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

При запуске RBM Вам потребуется пройти аутентификацию (рисунок 2). Уточните *login/password* для вашей работы у главного администратора СРК. Если вы главный администратор, используйте для авторизации суперпользователя *rubackup* и тот пароль, который вы задали ему при установке.

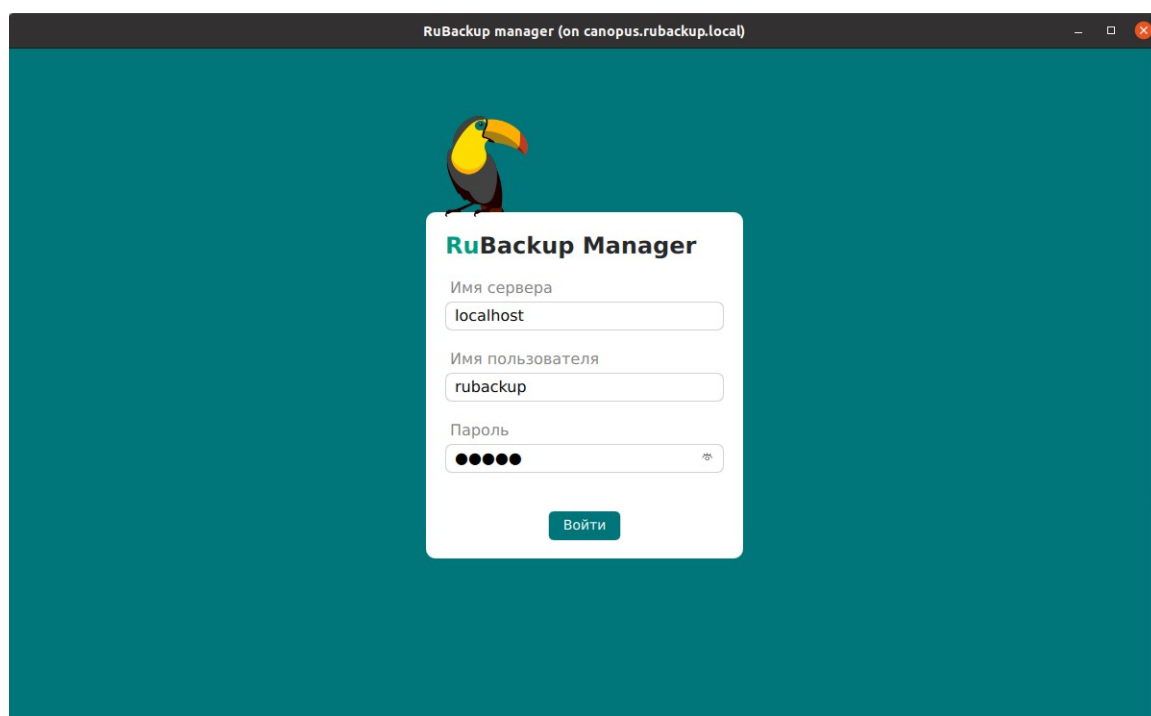
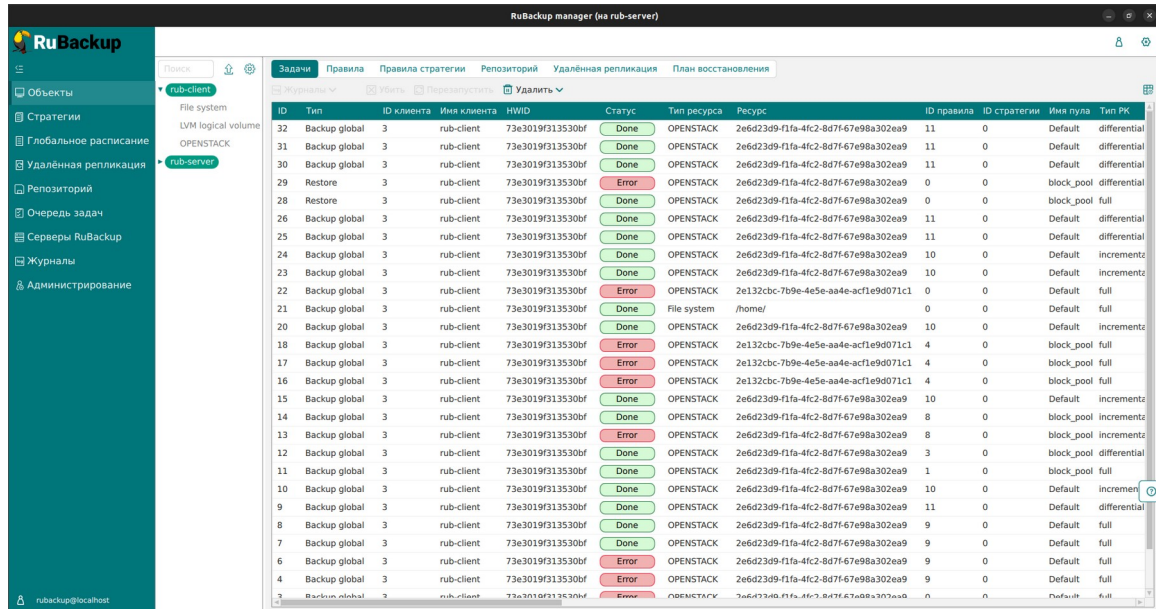


Рисунок 2

На вкладке **Объекты** представлен список клиентов системы резервного копирования. Клиенты отображаются по имени узла, на котором они запущены. Если навести указатель мыши на имя какого-либо из клиентов, будет отображен его HWID. Если развернуть запись для какого-либо из клиентов, в

ниспадющем списке будут отображены типы ресурсов, для которых данный клиент может создавать резервные копии (рисунок 3). Клиенты, которые в данный момент находятся в состоянии online, будут отмечены зеленым цветом. Клиенты в состоянии offline – красным (рисунок 3).



ID	Тип	ID клиента	Имя клиента	HWID	Статус	Тип ресурса	Ресурс	ID правила	ID стратегии	Имя папки	Тип ПК
32	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	11	0	Default	differential
31	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	11	0	Default	differential
30	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	11	0	Default	differential
29	Restore	3	rub-client	73e3019f313530bf	Error	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	0	0	block_pool	differential
28	Restore	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	0	0	block_pool	full
26	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	11	0	Default	differential
25	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	11	0	Default	differential
24	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	10	0	Default	incrementz
23	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	10	0	Default	incrementz
22	Backup global	3	rub-client	73e3019f313530bf	Error	OPENSTACK	2e132cbc-7b9e-4e5e-aa4e-act1e9d071c1	0	0	Default	full
21	Backup global	3	rub-client	73e3019f313530bf	Done	File system	/home/	0	0	Default	full
20	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	10	0	Default	incrementz
18	Backup global	3	rub-client	73e3019f313530bf	Error	OPENSTACK	2e132cbc-7b9e-4e5e-aa4e-act1e9d071c1	4	0	block_pool	full
17	Backup global	3	rub-client	73e3019f313530bf	Error	OPENSTACK	2e132cbc-7b9e-4e5e-aa4e-act1e9d071c1	4	0	block_pool	full
16	Backup global	3	rub-client	73e3019f313530bf	Error	OPENSTACK	2e132cbc-7b9e-4e5e-aa4e-act1e9d071c1	4	0	block_pool	full
15	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	10	0	Default	incrementz
14	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	8	0	block_pool	incrementz
13	Backup global	3	rub-client	73e3019f313530bf	Error	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	8	0	block_pool	incrementz
12	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	3	0	block_pool	differential
11	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	1	0	block_pool	full
10	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	10	0	Default	incrementz
9	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	11	0	Default	differential
8	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	9	0	Default	full
7	Backup global	3	rub-client	73e3019f313530bf	Done	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	9	0	Default	full
6	Backup global	3	rub-client	73e3019f313530bf	Error	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	9	0	Default	full
4	Backup global	3	rub-client	73e3019f313530bf	Error	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	9	0	Default	full
3	Backup global	3	rub-client	73e3019f313530bf	Error	OPENSTACK	2e6d23d9-f1fa-4fc2-8d7f-67e98a302ea9	0	0	Default	full

Рисунок 3

Для резервного копирования клиент должен быть авторизован администратором RuBackup.

Если клиент RuBackup установлен, но не авторизован, в нижней части окна RBM появится сообщение о том, что найдены неавторизованные клиенты. Все новые клиенты, средствами которых планируется создавать резервные копии, должны быть авторизованы в системе резервного копирования RuBackup.

Для авторизации неавторизованного клиента в RBM необходимо выполнить следующие действия:

1. Нажмите на вкладку «Администрирование» и выберите иконку «Клиенты» (рисунок 4).

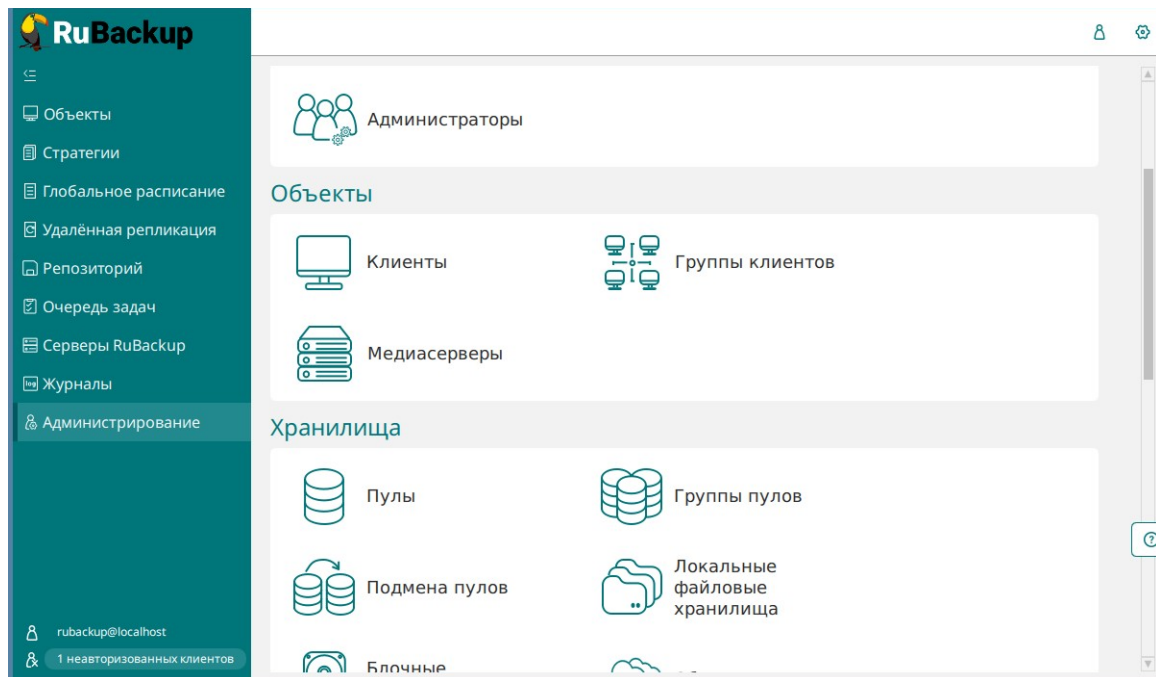


Рисунок 4

2. На верхней панели перейдите на вкладку «Неавторизованные клиенты» (рисунок 5):

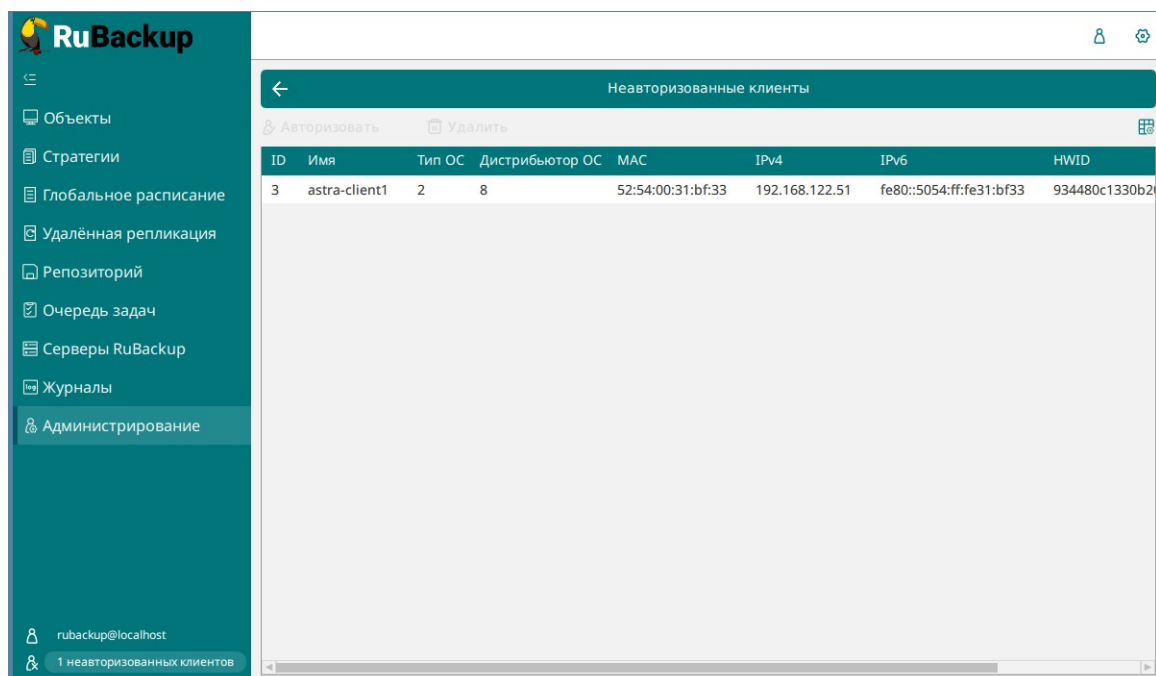


Рисунок 5

3. Нажмите на требуемый неавторизованный клиент правой кнопкой мыши и выберите «**Авторизовать**» (рисунок 6):

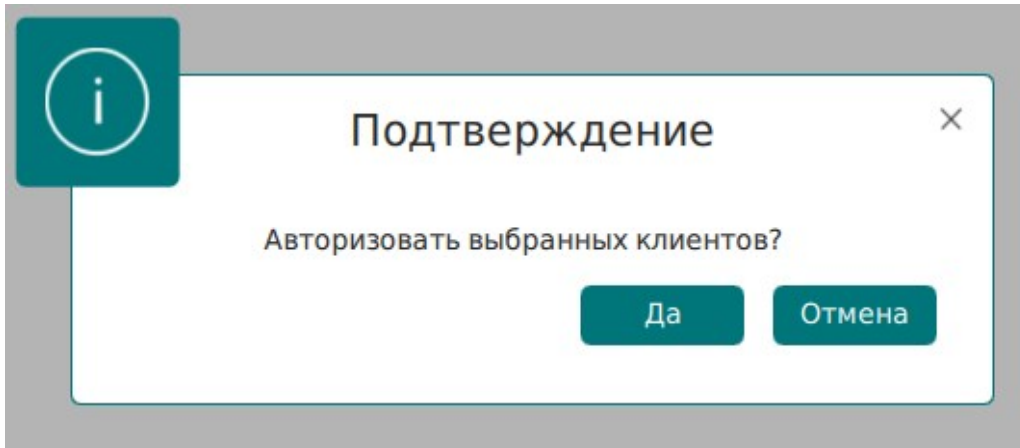


Рисунок 6

После авторизации клиент будет виден во вкладке «**Объекты**» (рисунок 7):

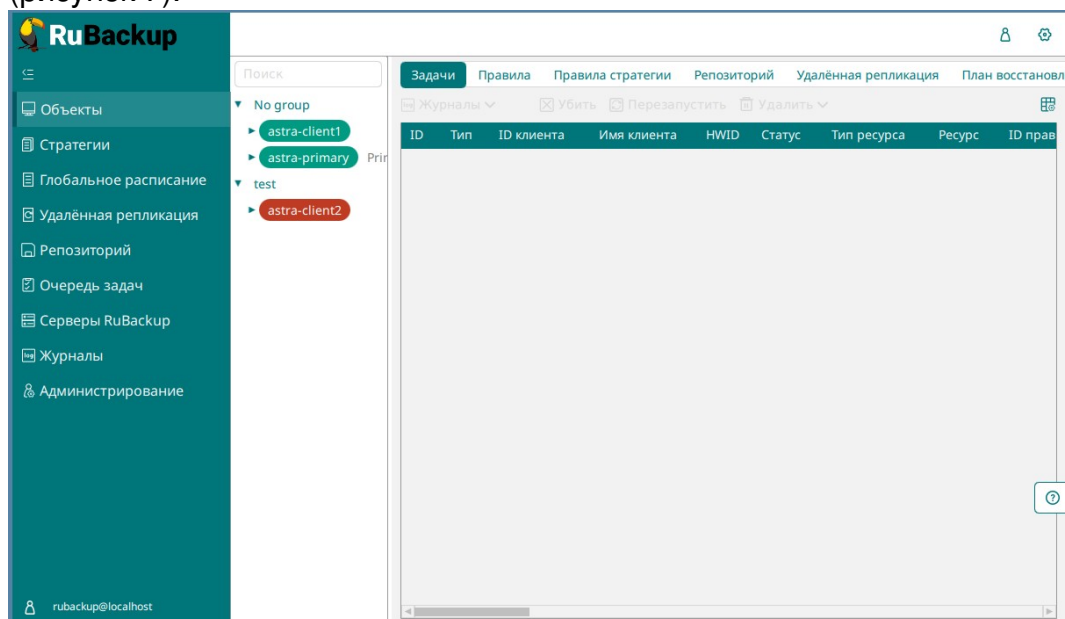


Рисунок 7

## Регулярное резервное копирование виртуальной машины

Чтобы выполнять регулярное резервное копирование виртуальной машины, необходимо создать правило в глобальном расписании (в случае операций с несколькими ресурсами и/или типами ресурсов можно также использовать стратегии резервного копирования). Для этого выполните следующие действия:

1. Находясь в разделе «**Объекты**», выберите вкладку «**Правила**» и нажмите на иконку «**+**» (рисунок 8):

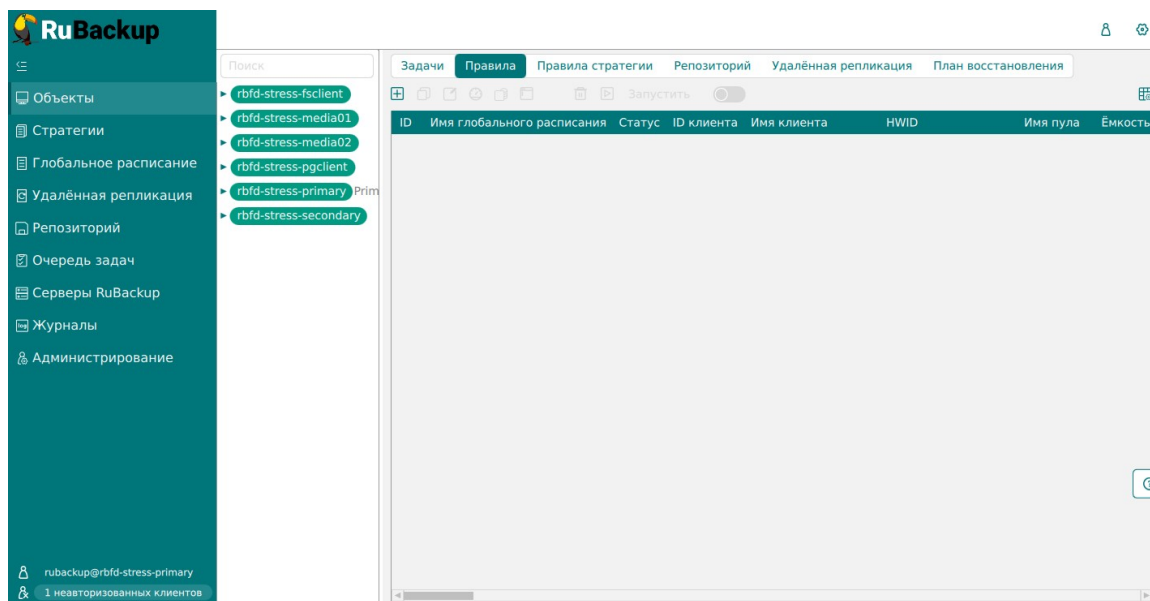


Рисунок 8

2. Выберите клиент, вместе с которым установлен модуль RuBackup, предназначенный для резервного копирования виртуальных машин **OpenStack** (модуль «**rb\_module\_openstack**», тип ресурса «**OPENSTACK**») (Рисунок 9):

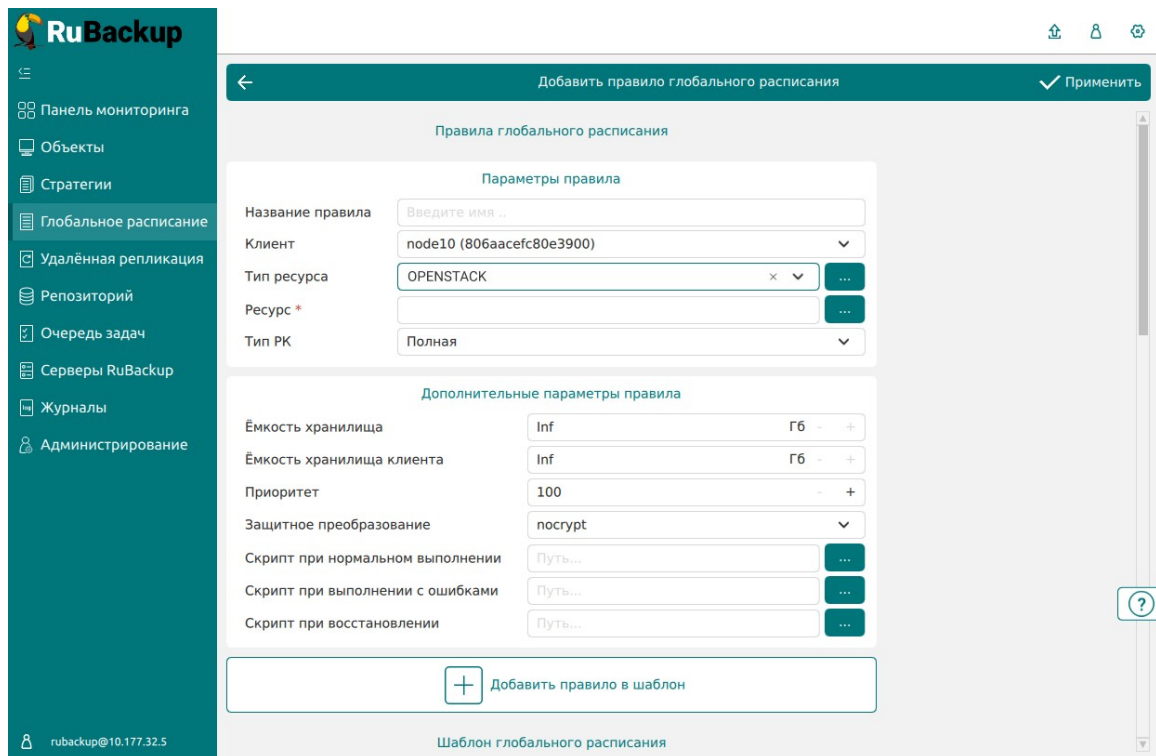


Рисунок 9

3. Выберите «Тип ресурса» -> «OPENSTACK» (Рисунок 10):

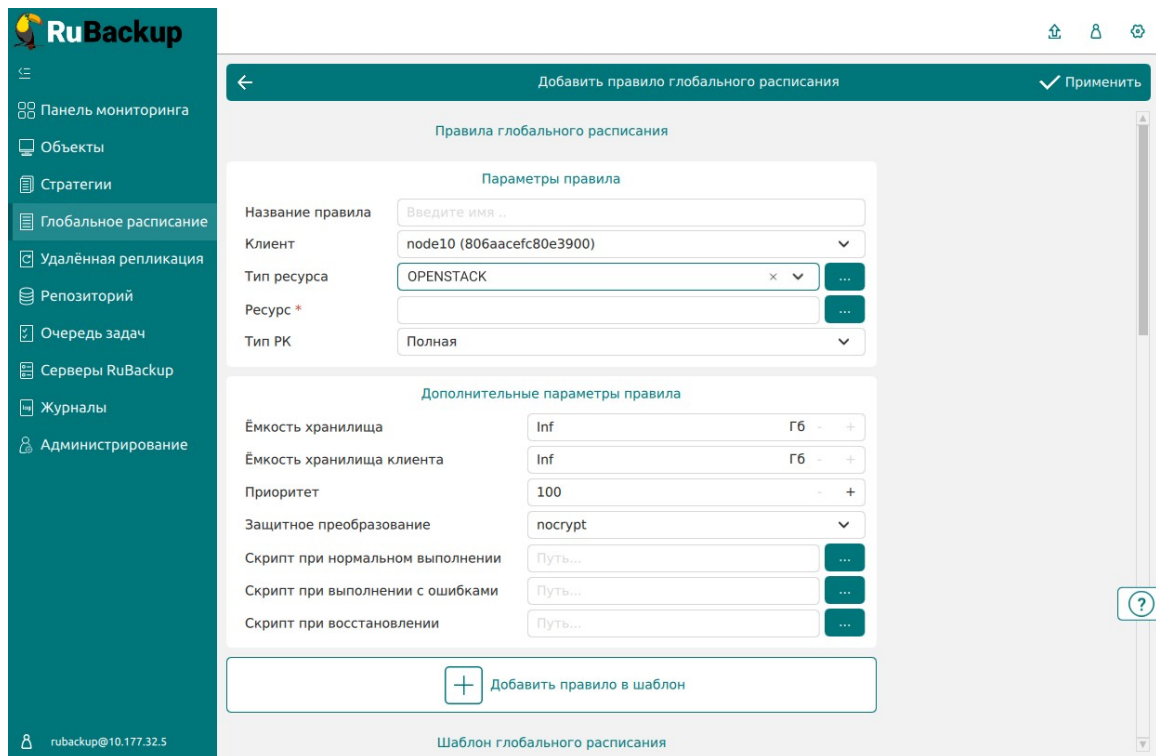


Рисунок 10



4. Нажмите на иконку «...» рядом с надписью «Ресурс» и выберите виртуальную машину, для которой требуется создать резервную копию (Рисунок 11):

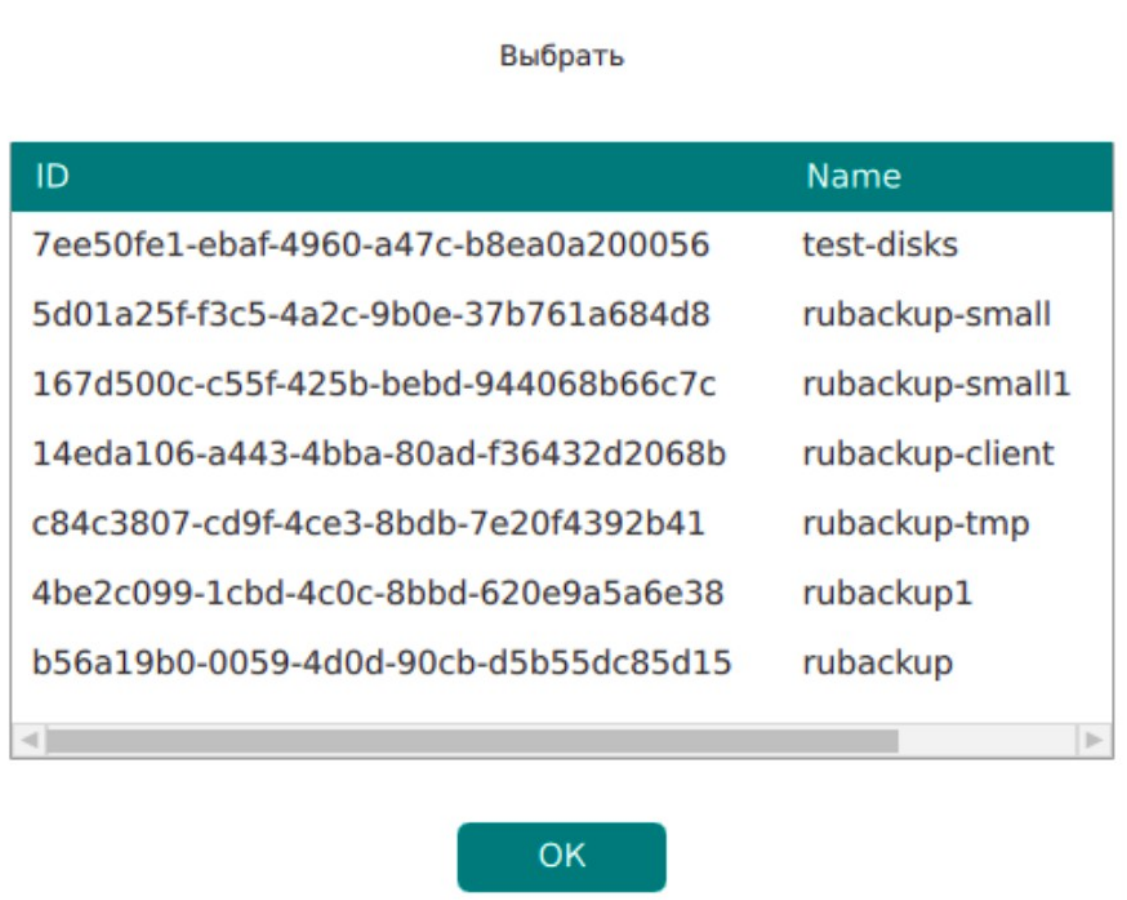


Рисунок 11

5. Установите остальные настройки правила: название правила, тип резервной копии (полная, инкрементальная или дифференциальная), ёмкость хранилища и ёмкость хранилища клиента, приоритет выполнения правила, алгоритм защитного преобразования, скрипт при нормальном выполнении, скрипт при выполнении с ошибками, скрипт, используемый при восстановлении резервной копии (Рисунок 12):



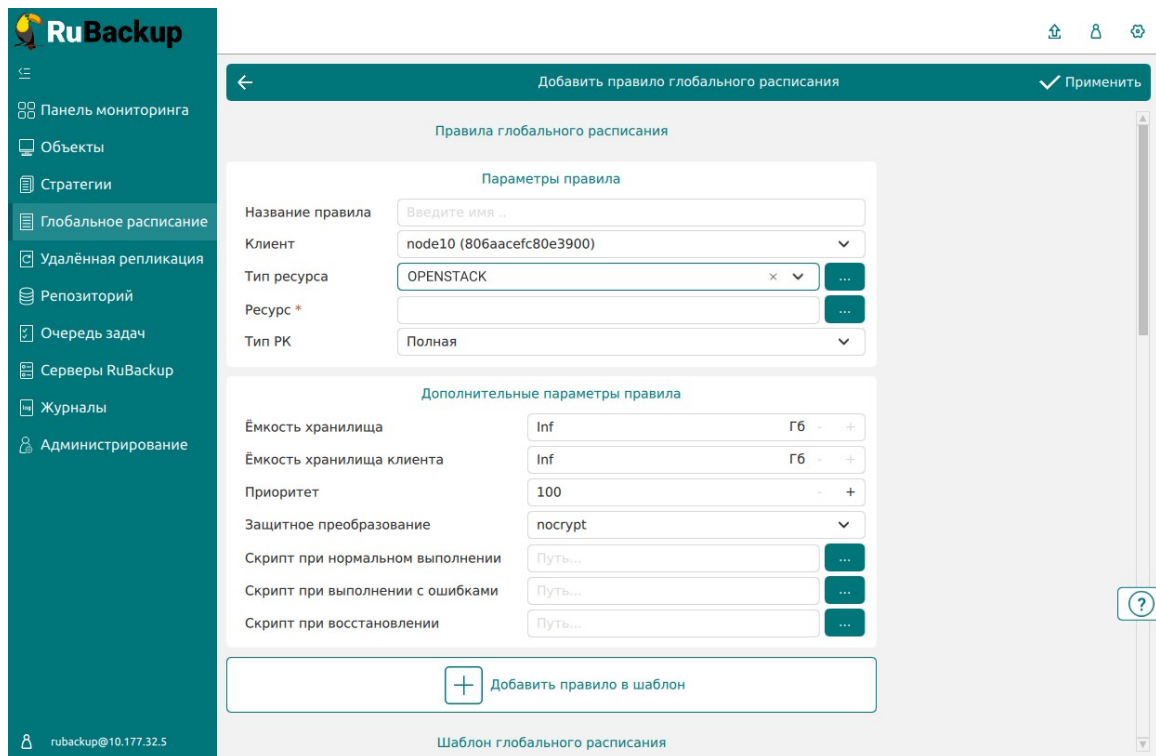


Рисунок 12

6. Нажав на иконку «...» рядом с выбранным типом ресурса «**OPENSTACK**», установите дополнительные настройки правила резервного копирования (Рисунок 13, Таблица 2).

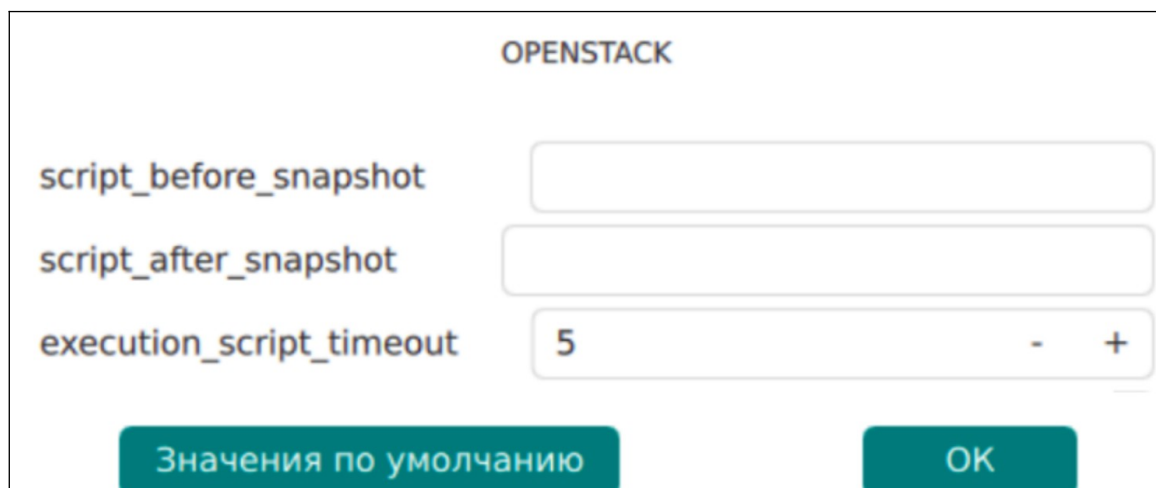


Рисунок 13

Таблица 2 – Дополнительные параметры правила резервного копирования виртуальных машин OpenStack

Параметр	Описание – Алгоритмы защитного преобразования, доступные в утилите <b>rbcrypt</b> .	Значение по умолчанию	Допустимые значения
script_before_snapshot	Полный путь к скрипту внутри виртуальной машины, который будет выполнен перед созданием снимка для данной виртуальной машины.	/opt/rubackup/scripts/openstack.sh	
script_after_snapshot	Полный путь к скрипту внутри виртуальной машины, который будет выполнен после создания снимка для данной виртуальной машины.	/opt/rubackup/scripts/openstack.sh	
execution_script_timeout	Время в секундах, в течение которого модуль RuBackup будет ожидать выполнения скриптов внутри виртуальной машины до и после создания снимка.	5	1 - 600

**Примечание:** для работы параметров «script\_before\_snapshot» и «script\_after\_snapshot» настройте SSH-доступ (см. раздел «Настройка SSH-доступа»).

- После выбора настроек правила резервного копирования нажмите на кнопку «**Добавить правило в шаблон**», если хотите создать сразу несколько правил - правило для выбранного типа ресурса (OPENSTACK) и выбранного ресурса (виртуальной машины) появится в списке правил под кнопкой (Рисунок 14). Таким образом создайте столько правил, сколько требуется. Для создания одного правила нажимать на кнопку не нужно.

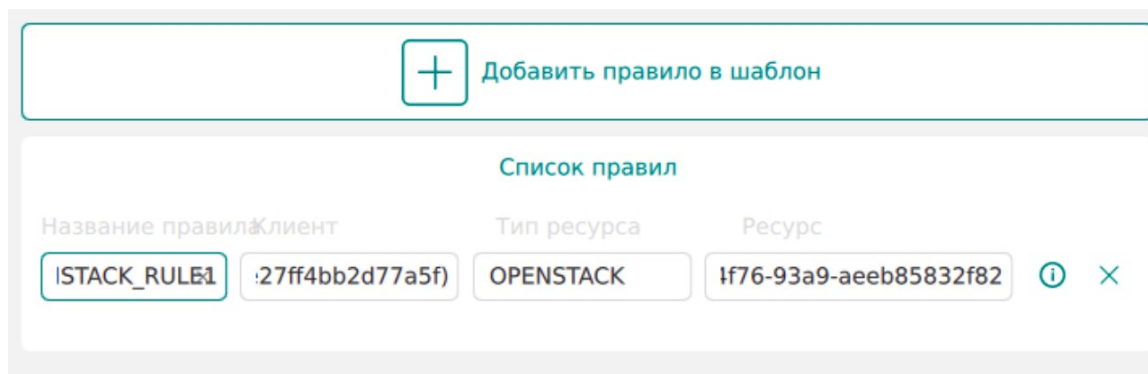


Рисунок 14

8. Заполните раздел «Шаблон глобального расписания» (подробнее см. в документе «Руководство системного администратора RuBackup»).
9. Нажмите на кнопку «**Применить**» в правом-верхнем углу для завершения настройки и создания правила/правил.

Вновь созданное правило будет иметь статус **run**. Если необходимо создать правило, которое пока не должно порождать задач резервного копирования, нужно убрать отметку «**Включить после создания**».

При необходимости, администратор может приостановить работу правила или немедленно запустить его (т. е. инициировать немедленное создание задачи при статусе правила wait).

Правило глобального расписания имеет срок жизни, определяемый при его создании, а так же предусматривает следующие основные 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 защитное преобразование резервной копии на клиенте.
- 5) Периодически выполнять проверку целостности резервной копии.
- 6) Хранить резервные копии определённый срок, а после его окончания удалить их из хранилища резервных копий и из записей репозитория либо уведомить 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 очереди задач. Отслеживать исполнение правил может как администратор с помощью RBM или утилит командной строки, так и клиент при помощи Менеджера клиента RuBackup (RBC) или утилиты командной строки `rb_tasks`.

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

## Срочное резервное копирование

В случае необходимости срочного резервного копирования, выберите клиент и нажмите кнопку «Срочное РК» (рисунок 15):

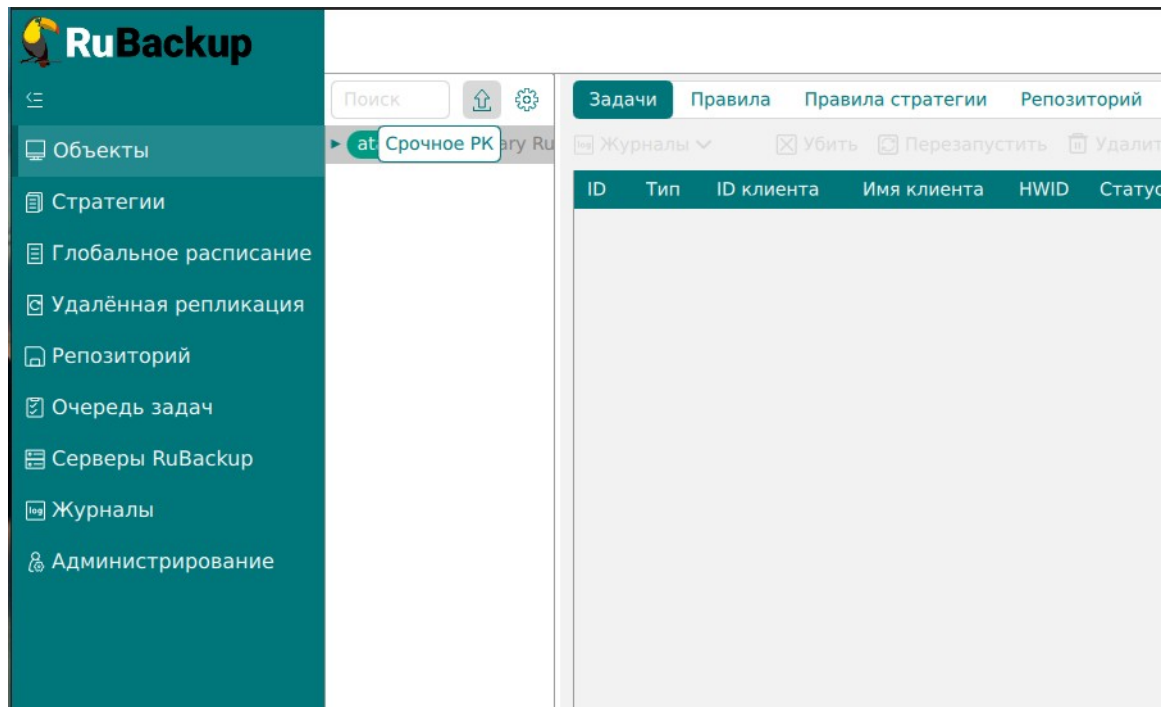


Рисунок 15

Откроется окно (рисунок 16). Выберите тип ресурса для срочного резервного копирования «**OPENSTACK**»:

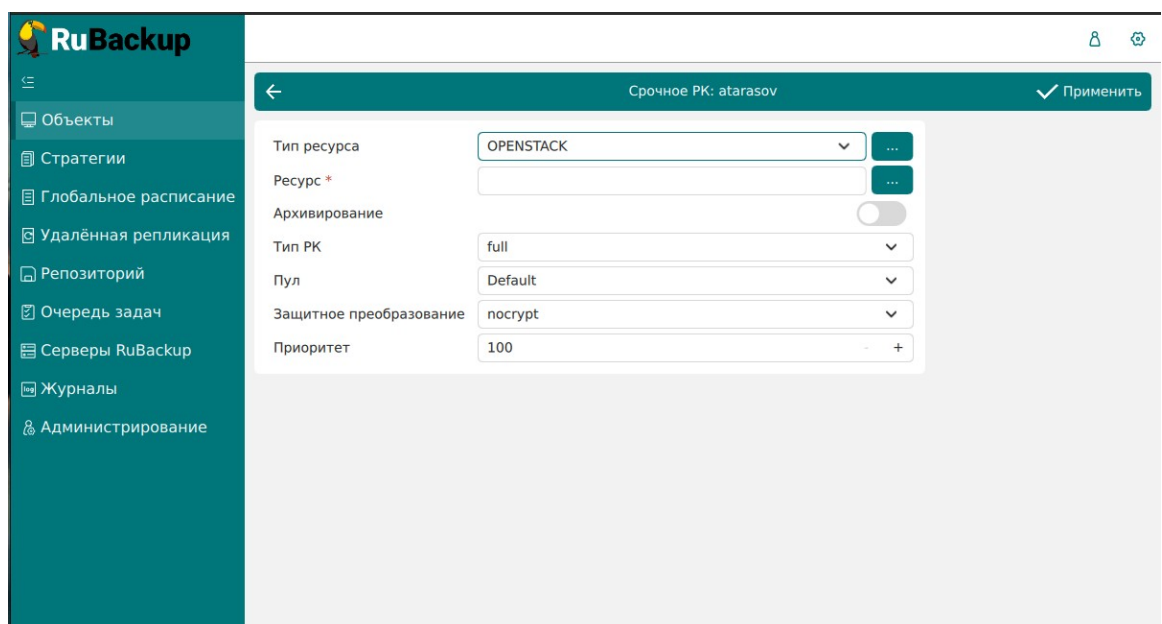


Рисунок 16

Нажмите на иконку «...» рядом с надписью «Ресурс» и выберите виртуальную машину, для которой требуется создать резервную копию (рисунок 17).

ID	Name	S
72d1ddee-a0a0-486e-892b-98e8d90cd1fd	rubackup-tmp	S
3a1de04d-a3a2-4080-8138-fb684220109c	atarasov	A
71a7fe98-41d8-4f76-93a9-aeeb85832f82	rubackup	S

OK

Рисунок 17

После выбора виртуальной машины заполните остальные настройки срочного резервного копирования: архивирование, тип резервной копии (полная, инкрементальная или дифференциальная), пул, алгоритм защитного преобразования, приоритет.

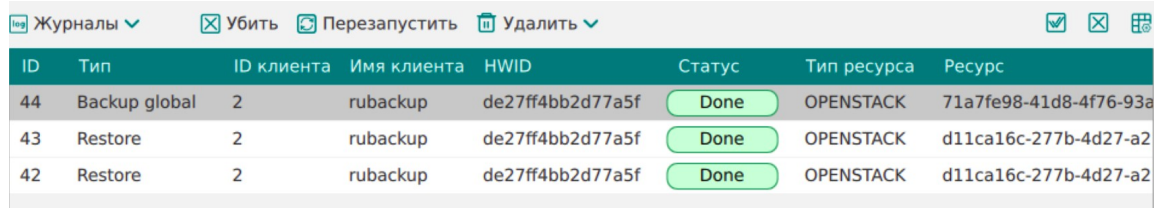
После заполнения настроек нажмите кнопку «Применить» в правом верхнем углу экрана.

Проверить ход выполнения резервного копирования можно, перейдя на вкладку «**Очередь задач**» (рисунок 18).

ID	Тип	ID клиента	Имя клиента	HWID	Статус	Тип ресурса	Ресурс
44	Backup global	2	rubackup	de27ff4bb2d77a5f	0%	OPENSTACK	71a7fe98-41d8-4f76-93a
43	Restore	2	rubackup	de27ff4bb2d77a5f	Done	OPENSTACK	d11ca16c-277b-4d27-a2
42	Restore	2	rubackup	de27ff4bb2d77a5f	Done	OPENSTACK	d11ca16c-277b-4d27-a2

Рисунок 18

При успешном завершении резервного копирования соответствующая задача перейдет в статус «**Done**» (рисунок 19):



ID	Тип	ID клиента	Имя клиента	HWID	Статус	Тип ресурса	Ресурс
44	Backup global	2	rubackup	de27ff4bb2d77a5f	Done	OPENSTACK	71a7fe98-41d8-4f76-93a
43	Restore	2	rubackup	de27ff4bb2d77a5f	Done	OPENSTACK	d11ca16c-277b-4d27-a2
42	Restore	2	rubackup	de27ff4bb2d77a5f	Done	OPENSTACK	d11ca16c-277b-4d27-a2

Рисунок 19

## Централизованное восстановление резервных копий

Система резервного копирования RuBackup предусматривает возможность восстановления резервных копий как со стороны клиента системы, так и со стороны администратора СРК. В тех случаях, когда централизованное восстановление резервных копий нежелательно, например, когда восстановление данных является зоной ответственности владельца клиентской системы, эта функциональность может быть отключена на клиенте (см. «Руководство системного администратора RuBackup»).

В тех случаях, когда централизованное восстановление на клиенте доступно, его можно инициировать, перейдя во вкладку «**Репозиторий**» на левой панели RBM. Для этого найдите в списке требуемую резервную копию, нажмите на нее правой кнопкой мыши и выберите в контекстном меню «**Восстановить**» (рисунок 20):

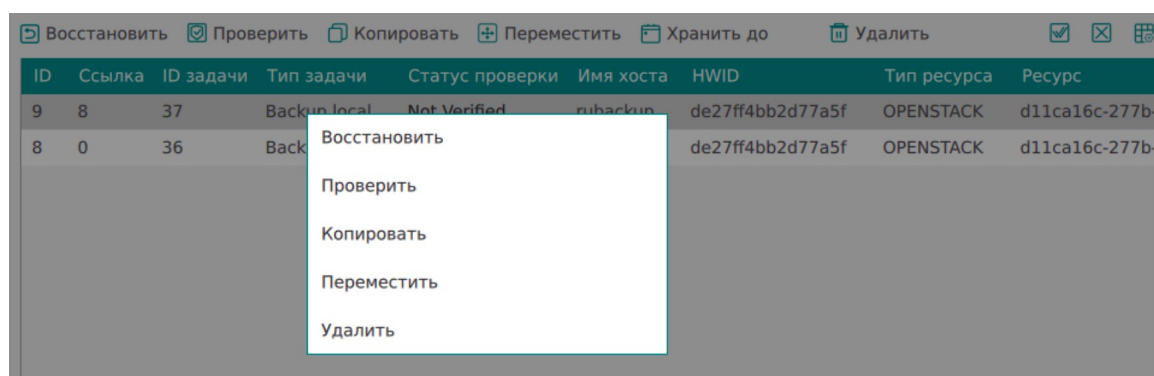


Рисунок 20

В окне централизованного восстановления (рисунок 21) можно увидеть и заполнить основные параметры резервной копии, задать путь назначения для восстановления резервной копии и параметры восстановления модуля OpenStack:

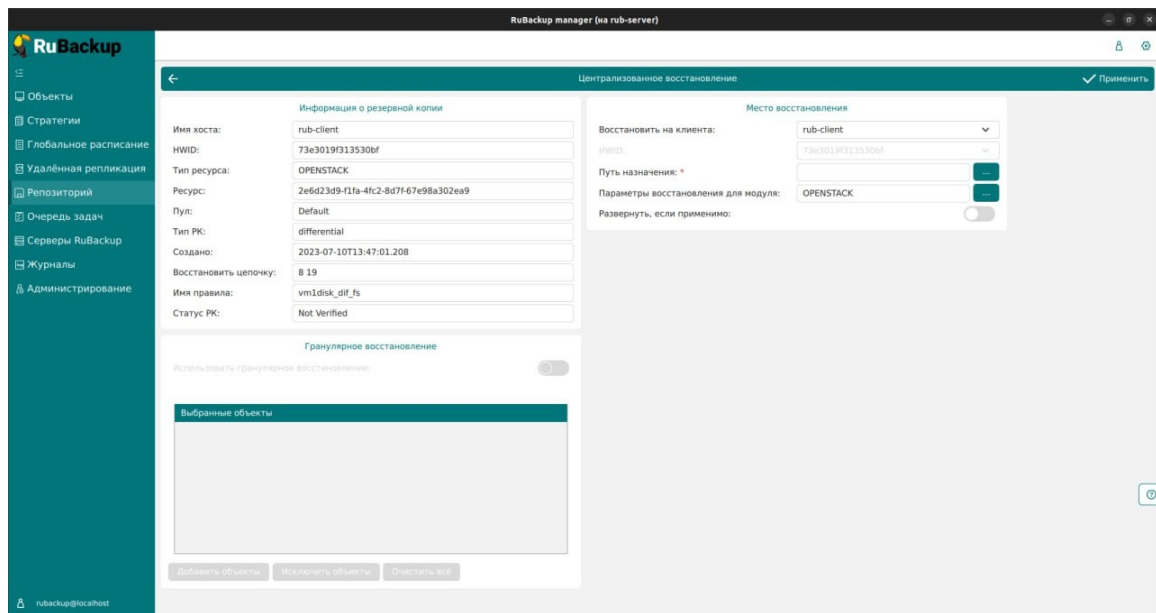



Рисунок 21

Путь назначения обозначает временную директорию, в которую будет выполнена распаковка архива резервной копии. Если выполняется восстановление резервной копии с развертыванием виртуальной машины, то после развертывания содержимое данной директории будет очищено. Убедитесь в наличии свободного места в выбранной директории распаковки (потребуется свободное место как минимум в объеме, равном суммарному объему дисков виртуальной машины, для которой сделана резервная копия).

Для настройки параметров восстановления модуля нажмите на иконку «...» рядом с полем «**Параметры восстановления для модуля: OpenStack**». Откроется окно (рисунок 22). Для отображения описания интересующего параметра нажмите на пиктограмму  рядом с именем параметра. Подробнее информация о параметрах восстановления модуля указана в Таблице 3.

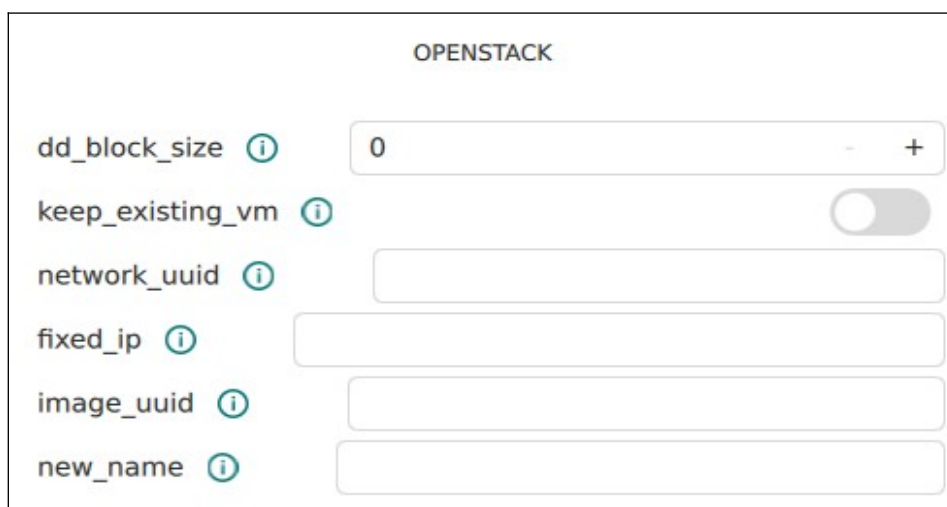


Рисунок 22



Таблица 3– Параметры восстановления для модуля OpenStack

Параметр	Описание	Значение по умолчанию	Допустимые значения
dd_block_size	Указывает размер блока для утилиты dd при операциях восстановления в мегабайтах	5	1-100
keep_existing_vm	Указывает, сохранять ли виртуальную машину с таким же именем при операции восстановления или создавать новую	true	true, false
network_uuid	Указывает идентификатор виртуальной сети в среде виртуализации OpenStack, где будет находиться восстановленная виртуальная машина		
fixed_ip	Указывает IP-адрес виртуальной машины в сети в среде визуализации OpenStack, где будет находиться восстановленная виртуальная машина		
image_uuid	Указывает идентификатор имиджа в среде визуализации OpenStack , с которым будет ассоциироваться восстановленная виртуальная машина.		
new_name	Присваивает новое имя восстанавливаемой виртуальной машине. Если указать параметр, виртуальная машина восстановится с новым именем. Если параметр не указывать, то виртуальная машина восстановится с именем, присвоенным ей при развёртывании.		

**Примечание:**

- Параметр `fixed_ip` требует обязательного указания параметра `network_uuid`.

В случае восстановления виртуальной машины из резервной копии будет выполнена проверка наличия в среде визуализации виртуальной машины с таким же именем. Если такой виртуальной машины нет, то будет выполнено восстановление с оригинальным именем. Если виртуальная машина с таким именем уже есть, то к имени виртуальной машины будет добавлен цифровой постфикс (рисунок 23).

<input type="checkbox"/>	Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State
<input type="checkbox"/>	rubackup-tmp_2	20.04		4-4-12	-	Active	nova	None	Running
<input type="checkbox"/>	rubackup-tmp	20.04		4-4-12	-	Active	nova	None	Running
<input type="checkbox"/>	rubackup	20.04		4-4-12	-	Active	nova	None	Running

Рисунок 23

В том случае, если необходимо восстановить резервную копию в локальный каталог на клиенте без развертывания виртуальной машины в среде виртуализации, необходимо снять отметку **“Развернуть, если применимо”** (рисунок 24):

**Место восстановления**

Восстановить на клиента:

HWID:

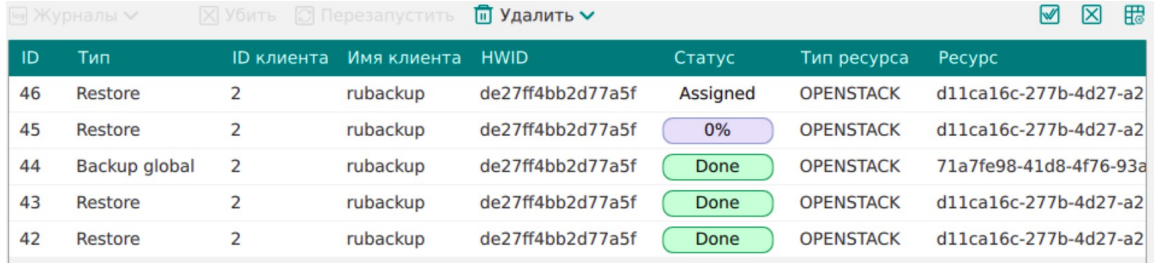
Путь назначения: \*

Параметры восстановления для модуля:

Развернуть, если применимо:

Рисунок 24

Проверить ход выполнения восстановления резервной копии можно в окне «**Очередь задач**» (рисунок 25). При успешном завершении восстановления резервной копии или цепочки резервных копий соответствующие задачи на восстановление перейдут в статус «**Done**» (рисунок 17)



ID	Тип	ID клиента	Имя клиента	HWID	Статус	Тип ресурса	Ресурс
46	Restore	2	rubackup	de27ff4bb2d77a5f	Assigned	OPENSTACK	d11ca16c-277b-4d27-a2
45	Restore	2	rubackup	de27ff4bb2d77a5f	0%	OPENSTACK	d11ca16c-277b-4d27-a2
44	Backup global	2	rubackup	de27ff4bb2d77a5f	Done	OPENSTACK	71a7fe98-41d8-4f76-93a
43	Restore	2	rubackup	de27ff4bb2d77a5f	Done	OPENSTACK	d11ca16c-277b-4d27-a2
42	Restore	2	rubackup	de27ff4bb2d77a5f	Done	OPENSTACK	d11ca16c-277b-4d27-a2

Рисунок 25

## Восстановление со стороны клиента

В случае необходимости восстановления резервной копии со стороны клиента вы можете воспользоваться утилитой командной строки `rb_archives`:

Просмотр списка доступных резервных копий:

```
root@rubackup:/home/ubuntu# rb_archives
```

Id	Ref ID	Resource	Resource type	Backup type	Created	Crypto	Signed	Status
8		d11ca16c-277b-4d27-a214-937166f1b463	OPENSTACK	full	2023-06-06 05:13:11+00	nocrypt	True	Not Verified
9	8	d11ca16c-277b-4d27-a214-937166f1b463	OPENSTACK	incremental	2023-06-06 05:16:52+00	nocrypt	True	Not Verified

Запрос на восстановление резервной копии:

```
root@rubackup-ubuntu:~# rb_archives -X 2
Password:
The archive will be restored in the directory: /rubackup-tmp
----> Restore archive chain: 1 2 < ----
Record ID: 1 has status: Not Verified
Continue (y/n)? yes
Record ID: 2 has status: Not Verified
Continue (y/n)? yes
TASK WAS ADDED TO QUEUE:4 5
```

В том случае если резервная копия должна быть развернута, т. е. необходимо восстановить виртуальную машину в среду виртуализации, нужно использовать опцию `-x`. В том случае когда требуется восстановить резервную копию в локальном каталоге клиента без развертывания, нужно использовать опцию `-X`.