

Развёртывание и настройка oVirt 4.0. Часть 12.

Резервное копирование виртуальных машин

 blog.it-kb.ru/2016/10/07/install-ovirt-4-0-part-12-backup-and-restore-vm-in-ovirt-cluster-with-ovirtbackup-scripts

Автор: Алексей Максимов

07.10.2016



Продолжая тему резервного копирования, в этой части мы рассмотрим практический пример автоматизации регулярного резервного копирования виртуальных машин в кластере **oVirt 4.0.4**, а также посмотрим на процедуру восстановления виртуальной машины из резервной копии.

Варианты резервного копирования виртуальных машин

Исходя из той информации, что мне удалось собрать о возможностях резервного копирования VM в oVirt, на данный момент имеется два основных варианта – это либо использовать коммерческие программные продукты сторонних производителей, либо использовать автоматизацию на базе скриптов с применением возможностей oVirt API. При этом, насколько я понимаю, платные решения строятся всё на том же API.

Так как ранее в комментариях был задан вопрос о коммерческих решениях, сделаю небольшое отступление в эту сторону. Информацию о коммерческих решениях мне удалось найти в двух местах:

- На публично доступной странице Red Hat Products & Services - Ecosystem - Certified Software, где среди сертифицированных Red Hat продуктов, что-то более или менее подходящее по смыслу - это **Acronis Backup & Recovery 11.5 Virtual Edition for RHEV**, но это только для **3** ветки **RHEV**.
- В документе базы знаний Red Hat с ограниченным доступом (хотя непонятно, зачем было ограничивать доступ к информации такого рода) - How to backup and restore a VM on RHEV

Наберусь наглости и позволю себе процитировать небольшой отрывок из второго документа:

For backing up VMs online:

- From RHEV 3.3 onwards, you can also back up VMs online using the the backup and restore API with a third-party backup software. See The Backup and Restore API in the RHEV REST API Guide for more information. For upstream docs, see <http://www.ovirt.org/develop/release-management/features/storage/backup-restore-api-integration/>.
- Third-party online backup solutions certified by Red Hat include Acronis, Commvault, SEP, and Veritas (previously part of Symantec Corporation) . See <https://access.redhat.com/ecosystem/search/#/vendor/> for more information. Note that third-party products are not supported by Red Hat. Contact the third-party support for more details.
- For Red Hat Virtualization (RHV) 4.0, Commvault and SEP are certified third-party online backup solutions.

То есть, для **RHV/oVirt 4** на данный момент подходящим сертифицированным решением будут продукты компаний SEP и Commvault.

Если же говорить о решении, не требующем финансовых затрат на сторонне ПО, то на **GitHub** доступен набор скриптов wefixit-AT - oVirtBackup. Применение этих скриптов мы и рассмотрим в данной заметке. Данный набор скриптов позволяет выполнять резервное копирование виртуальных машин без остановки этих виртуальных машин и без прерывания работы сервисов внутри гостевых ОС на этих виртуальных машинах.

Общий порядок работы скриптов при резервном копировании каждой отдельно взятой VM следующий:

- Для целевой виртуальной машины создаётся онлайн-снимок;
- Созданный снимок клонируется в новую виртуальную машину на том же хранилище Data Domain, где расположена целевая VM;
- После создания клона, из целевой VM удаляется созданный ранее снимок;
- Выполняется экспорт созданной виртуальной машины - клона в специальное хранилище в виде NFS шары, подключенное к oVirt в качестве Export Domain;
- С основного хранилища Data Domain удаляется созданный ранее клон VM.

Далее мы рассмотрим необходимые действия для настройки скриптов резервного копирования.

Подготовительные мероприятия

Скрипты в своей работе используют oVirt Python-SDK, поэтому для начала убедимся в том, что нужный пакет есть в системе на сервере oVirt Engine (насколько я понимаю, этот пакет устанавливается в составе oVirt Engine):

```
# yum info ovirt-engine-sdk-python
```

Installed Packages

```
Name       : ovirt-engine-sdk-python
Arch        : noarch
Version     : 3.6.9.1
Release     : 1.el7.centos
Size        : 6.8 M
Repo        : installed
From repo   : ovirt-4.0
Summary     : oVirt Engine Software Development Kit (Python)
URL         : http://ovirt.org
License     : ASL 2.0
Description : This package contains The oVirt-Engine Software Development Kit.
              : With this package, custom software can be built for oVirt-Engine.
```

<

```
p align="center">***
```

Настроим **NFS** сервер для хранения резервных копий виртуальных машин. Создадим отдельную NFS-шару, к которой ограничим доступ - разрешим доступ на чтение и запись в шару только для NFS-клиента с сервера oVirt Engine.

На NFS-сервере создаём каталог для сохранения резервных копий ВМ и меняем для него владельца и группу (36:36 это UID/GID vds:kvm):

```
# mkdir -p /mnt/mdadm-vv1/nfs/ovirt-vm-backup
# chown -R 36:36 /mnt/mdadm-vv1/nfs/ovirt-vm-backup
# chmod 755 /mnt/mdadm-vv1/nfs/ovirt-vm-backup
```

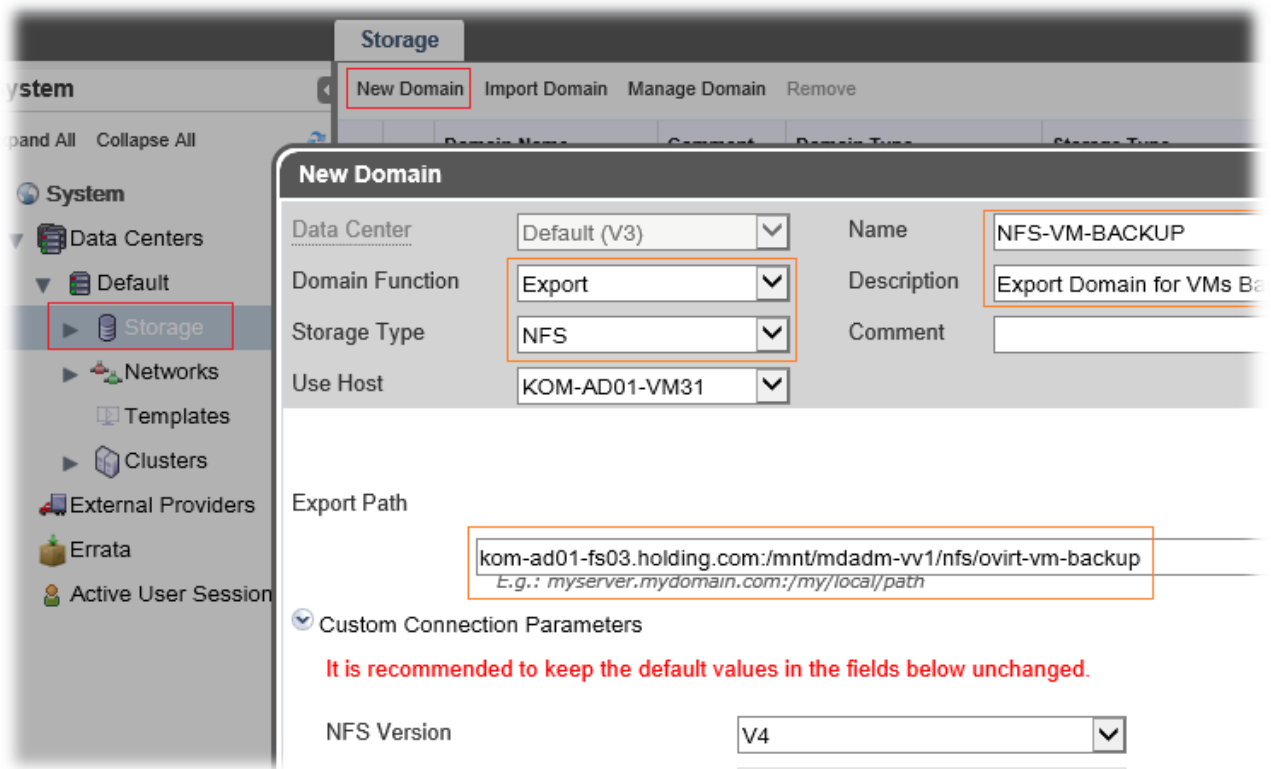
Добавляем в конфигурационный файл /etc/exports запись о NFS-шаре, доступ к которой разрешаем для всех допустимых NFS-клиентов (хостов виртуализации в кластере oVirt) с назначением для всех подключающихся NFS-клиентов UID/GID равными также 36:36:

```
/mnt/mdadm-vv1/nfs/ovirt-vm-backup KOM-AD01-
VM3*.holding.com(rw,sync,no_subtree_check,all_squash,anonuid=36,anongid=36)
```

Заставляем службу **nfs-server** пересчитать конфигурацию и проверяем результат:

```
# exportfs -r
# exportfs
...
/mnt/mdadm-vv1/nfs/ovirt-vm-backup
      KOM-AD01-VM3*.holding.com
...
```

Переходим на веб-портал администрирования oVirt в раздел **Storage** и добавляем **Export Domain**, указав путь в ранее созданной NFS-шаре на файловом сервере



Теперь можно переходить к настройке скриптов резервного копирования.

Настройка скриптов резервного копирования

Загружаем и распаковываем скрипты на сервер oVirt Engine:

```
# wget https://codeload.github.com/wefixit-AT/oVirtBackup/zip/master -O
/usr/local/sbin/oVirtBackup.zip
# unzip /usr/local/sbin/oVirtBackup.zip -d /usr/local/sbin/
# rm -f /usr/local/sbin/oVirtBackup.zip
# mv /usr/local/sbin/oVirtBackup-master /usr/local/sbin/ovirt-vm-backup
```

Таким образом скрипты резервного копирования будут размещаться в каталоге /usr/local/sbin/ovirt-vm-backup.

Создаём конфигурационный файл, скопировав его из шаблона config_example.cfg:

```
# cp /usr/local/sbin/ovirt-vm-backup/config_example.cfg /usr/local/sbin/ovirt-vm-
backup/config.cfg
# nano /usr/local/sbin/ovirt-vm-backup/config.cfg
```

Настраиваем параметры конфигурационного файла. В шаблоне все параметры имеют небольшие комментарии, поэтому несложно понять какой параметр за что отвечает. Привожу пример уже настроенного конфигурационного файла без комментариев:

```
[config]
vm_names: ["KOM-AD01-APP31"]
vm_middle=_BACKUP
snapshot_description=Snapshot for backup script
server=https://kom-ad01-ovirt1.ad.ies-holding.com/ovirt-engine/api
username=admin@internal
password=P@ssw0rd
export_domain=NFS-VM-BACKUP
timeout=5
cluster_name=Default
backup_keep_count=3
dry_run=False
vm_name_max_length=64
use_short_suffix=False
storage_domain=3PAR-VOLUME2
storage_space_threshold=0.1
```

В параметре **vm_names** для начала указываем одну какую-нибудь не особо критичную виртуальную машину. Красным цветом я подкрасил те параметры, которые были в моём случае изменены.

Так как в данном файле мы сохраняем учётные данные локального администратора oVirt, то желательно ограничить доступ к этому файлу:

```
# chmod 600 /usr/local/sbin/ovirt-vm-backup/config.cfg
```

Посмотрим, какие параметры имеет текущая версия главного скрипта, который вызывается для запуска задач резервного копирования:

```
# /usr/local/sbin/ovirt-vm-backup-master/backup.py -h
```

```
Usage: backup.py -c [-a] [-d] [-h]
    -c      Path to the config file
    -a      Backup all VM's and override the list of VM's in the config file
    -d      Debug flag
    -h      Display this help and exit
```

Как видим, скрипт может выполняться с явным указанием того или иного конфигурационного файла. И это хорошо, так как добавляет дополнительную гибкость в случае необходимости настройки резервного копирования разных виртуальных машин по разным правилам.

Выполним проверочный запуск скрипта с ранее настроенным нами конфигурационным файлом, включив дополнительно флаг отладки:

```
# /usr/local/sbin/ovirt-vm-backup/backup.py -c /usr/local/sbin/ovirt-vm-backup/config.cfg -d
```

```
Oct 05 17:44:04: Start backup for: KOM-AD01-APP31
Oct 05 17:44:05: Snapshot creation started ...
Oct 05 17:44:08: Snapshot operation(creation) in progress ...
Oct 05 17:44:19: Snapshot created
Oct 05 17:44:29: Clone into VM started ...
Oct 05 17:44:33: Cloning in progress ...
Oct 05 17:45:08: Cloning finished
Oct 05 17:45:09: Found snapshots(1):
Oct 05 17:45:09: Snapshots description: Snapshot for backup script, Created on:
2016-10-05 17:44:08.258000+03:00
Oct 05 17:45:09: Snapshot deletion started ...
Oct 05 17:47:14: Snapshot operation(deletion) in progress ...
Oct 05 17:47:19: Snapshots deleted
Oct 05 17:47:19: Export started ...
Oct 05 17:47:22: Exporting in progress ...
Oct 05 17:49:40: Exporting finished
Oct 05 17:49:40: Delete cloned VM started ...
Oct 05 17:49:43: Cloned VM deleted
Oct 05 17:49:43: Duration: 5:39 minutes
Oct 05 17:49:43: VM exported as KOM-AD01-APP31_BACKUP_20161005_174404
Oct 05 17:49:43: Backup done for: KOM-AD01-APP31
Oct 05 17:49:43: All backups done
```

Как видно, все ранее описанные этапы работы скрипта прошли успешно. В ходе работы скрипта в веб-консоли oVirt можно наблюдать появление снимота для целевой виртуальной машины, из которого в последствии создаётся временная клонированная VM для последующей выгрузки в Export Domain

Name	Memory	CPU	Network	Status
HostedEngine	26%	19%	0%	Up
KOM-AD01-APP31	8%	0%	0%	Up
KOM-AD01-APP31_BACKUP_20161005_174404	0%	0%	0%	Image Locked

Date	Status	Memory	Description
Current	OK	<input type="checkbox"/>	Active VM
Oct 5, 2016 5:44:08...	Locked	<input type="checkbox"/>	Snapshot for backup script

В завершающей стадии временная виртуальная машина и снимот, из которого она была сделана удаляются. На протяжении всего процесса резервного копирования гостевая ОС виртуальной машины продолжает работать.

Убедившись в том, что резервное копирование одной виртуальной машины отработало успешно, возвращаемся в конфигурационный файл `config.cfg` и дополняем список всеми машинами, которые мы хотим на регулярной основе подвергать резервному копированию.

```
...
vm_names: ["KOM-AD01-APP31", "KOM-AD01-APP32", "KOM-AD01-WEB01"]
...
```

При этом, как я уже заметил ранее, если для некоторых машин требуется какая-то иная конфигурация резервного копирования, например, сохранение в какой-то другой выделенный Export Domain или увеличенный/сокращённый срок хранения копий, то для таких ВМ мы просто настраиваем дополнительные конфигурационные файлы по аналогии с `config.cfg`.

Теперь нам нужно настроить периодическое выполнение скрипта резервного копирования. Создадим для этого файл задачи `cron`:

```
# nano -Y sh /etc/cron.d/ovirt-vm-backup
```

Наполним его содержимым таким образом, чтобы наш скрипт выполнялся, например, раз в сутки по ночам, сохраняя при этом весь отладочный вывод в лог-файл:

```
# Regular cron job for the oVirt VMs full backup
10 00 * * * root /usr/local/sbin/ovirt-vm-backup/backup.py -c
/usr/local/sbin/ovirt-vm-backup/config.cfg -d >> /var/log/ovirt-vm-backup/ovirt-
vm-backup.log
```

Настроим логирование. Создадим каталог для лог-файлов:

```
# mkdir /var/log/ovirt-vm-backup
```

А также создадим файл правил ротации логов для утилиты **logrotate**:

```
# nano -Y sh /etc/logrotate.d/ovirt-vm-backup
```

Наполним файл примерно следующим содержимым:

```
/var/log/ovirt-vm-backup/*.log {
    daily
    rotate 30
    compress
    delaycompress
    missingok
    notifempty
}
```

Теперь ежедневно будет создаваться новый лог-файл со сжатием старых логов, при этом будет сохраняться не более 30 предыдущих сжатых файлов.

Восстановление виртуальной машины из резервной копии

Теперь рассмотрим практический пример процедуры восстановления отдельно взятой виртуальной машины. Для восстановления будем использовать ту же виртуальную машину, которая ранее фигурировала в процессе создания резервной копии – KOM-AD01-APP31.

На веб-портале администрирования oVirt на закладке **Storage** выберем **Export Domain**, который мы ранее делали для сохранения резервных копий виртуальных машин и переключимся на вкладку **VM Import**

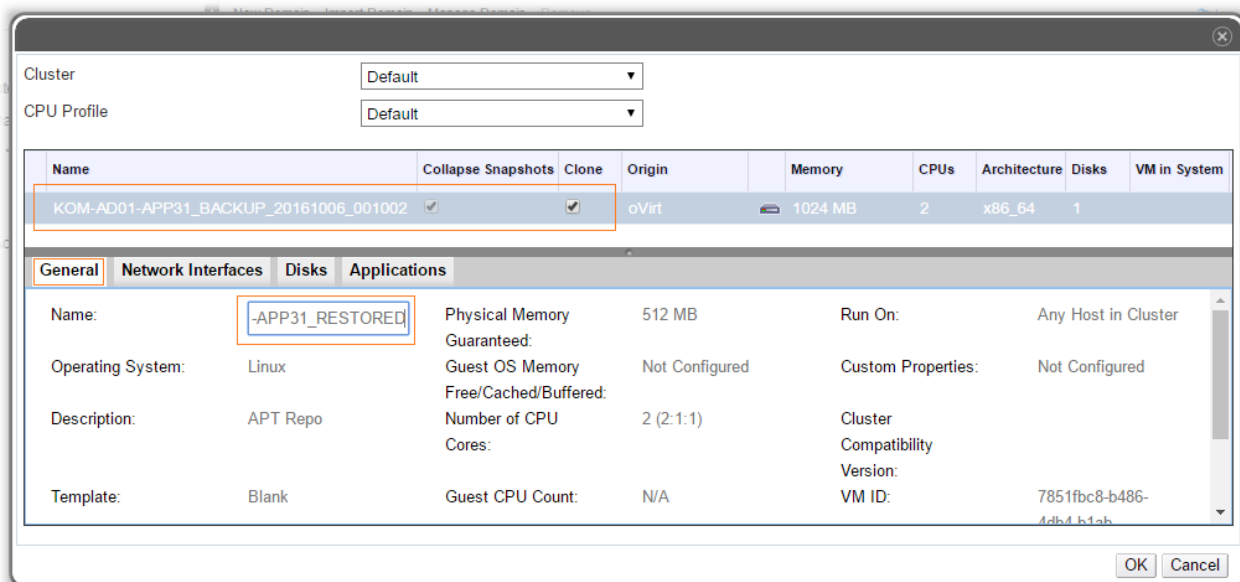
The screenshot shows the oVirt web portal interface. At the top, there is a navigation bar with tabs: Dashboard, Data Centers, Clusters, Hosts, Networks, Storage (highlighted), Disks, and Virtual Machines. Below this, there is a sub-navigation bar with links: New Domain, Import Domain, Manage Domain, and Remove. The main content area displays a table of storage domains. The table has columns: Domain Name, Domain Type, Storage Type, Format, Cross Data, Total Space, Free Space, and Description. The rows are: 3PAR-VOLUME1 (Data, Fibre Channel, V3, Active, 89 GB, 44 GB), 3PAR-VOLUME2 (Data (Master), Fibre Channel, V3, Active, 1999 GB, 1962 GB), NFS-ISO-DOMAIN (ISO, NFS, V1, Active, 3666 GB, 3463 GB, ISO Domain), and NFS-VM-BACKUP (Export, NFS, V1, Active, 3666 GB, 3463 GB, Export Domain). The NFS-VM-BACKUP row is highlighted with a red box. Below the table, there is another navigation bar with tabs: General, Data Center, VM Import (highlighted), Template Import, and Permissions. Under the VM Import tab, there is a sub-navigation bar with links: Import (highlighted) and Remove. The main content area displays a table of virtual machine imports. The table has columns: Name, Template, Origin, Memory, CPU, Architecture, Disk, Creation Date, and Export Date. The rows are: KOM-AD01-APP31_BACKUP_20161005_174404, KOM-AD01-APP31_BACKUP_20161005_181501, and KOM-AD01-APP31_BACKUP_20161006_001002. The last row is highlighted with a red box.

Domain Name	Domain Type	Storage Type	Format	Cross Data	Total Space	Free Space	Description
3PAR-VOLUME1	Data	Fibre Channel	V3	Active	89 GB	44 GB	
3PAR-VOLUME2	Data (Master)	Fibre Channel	V3	Active	1999 GB	1962 GB	
NFS-ISO-DOMAIN	ISO	NFS	V1	Active	3666 GB	3463 GB	ISO Domain
NFS-VM-BACKUP	Export	NFS	V1	Active	3666 GB	3463 GB	Export Domain

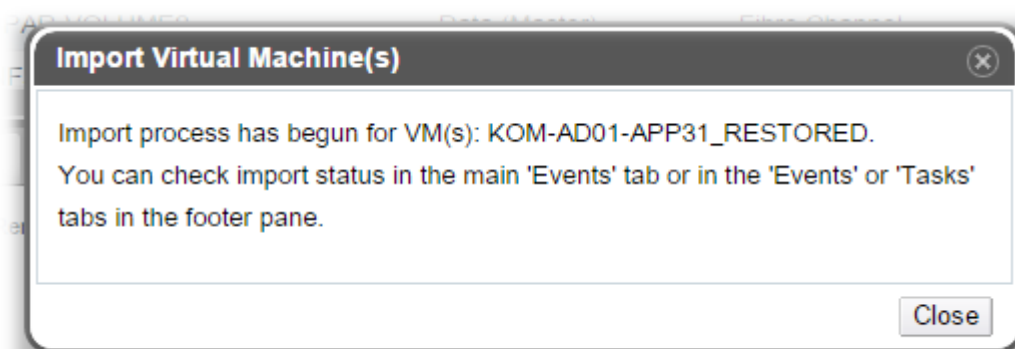
Name	Template	Origin	Memory	CPU	Architecture	Disk	Creation Date	Export Date
KOM-AD01-APP31_BACKUP_20161005_174404	Blank	oVirt	1024 MB	2	x86_64	1	Oct 5, 2016...	Oct 5, 2016...
KOM-AD01-APP31_BACKUP_20161005_181501	Blank	oVirt	1024 MB	2	x86_64	1	Oct 5, 2016...	Oct 5, 2016...
KOM-AD01-APP31_BACKUP_20161006_001002	Blank	oVirt	1024 MB	2	x86_64	1	Oct 6, 2016...	Oct 6, 2016...

Здесь мы увидим информацию о всех копиях виртуальных машин, которые сделаны нашим скриптом резервного копирования. Каждая копия при этом будет иметь в имени отметку времени создания. Здесь можно выбрать ту копию, которую мы хотим восстановить и нажать кнопку **Import**.

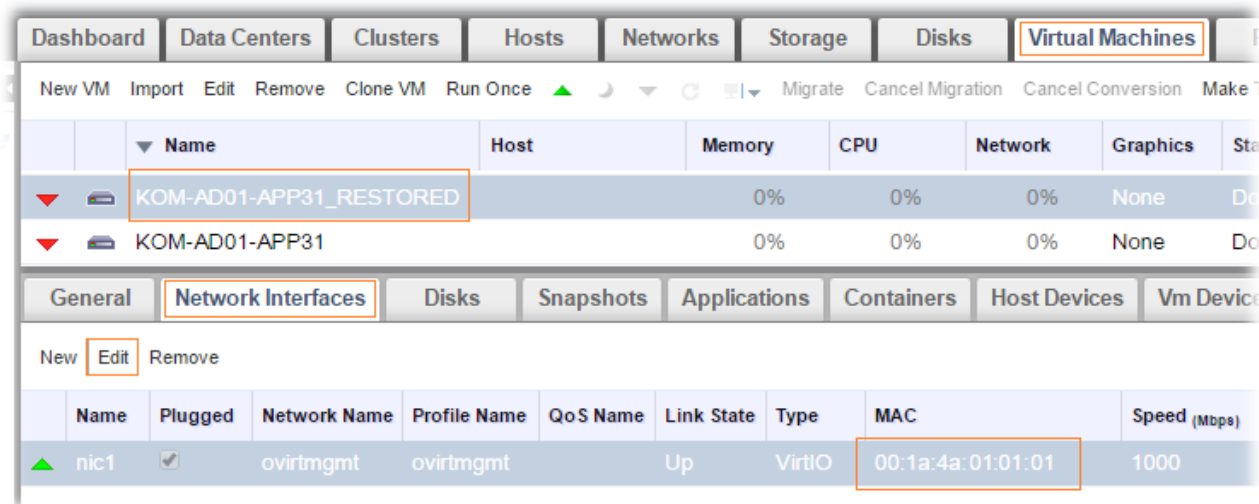
В открывшейся форме импорта, выбрав закладку **General** мы сможем при желании изменить имя создаваемой при восстановлении виртуальной машины. При этом имя импортируемой VM не должно совпадать с именами существующих в кластере виртуальных машин, иначе oVirt попросту откажет нам в импорте.



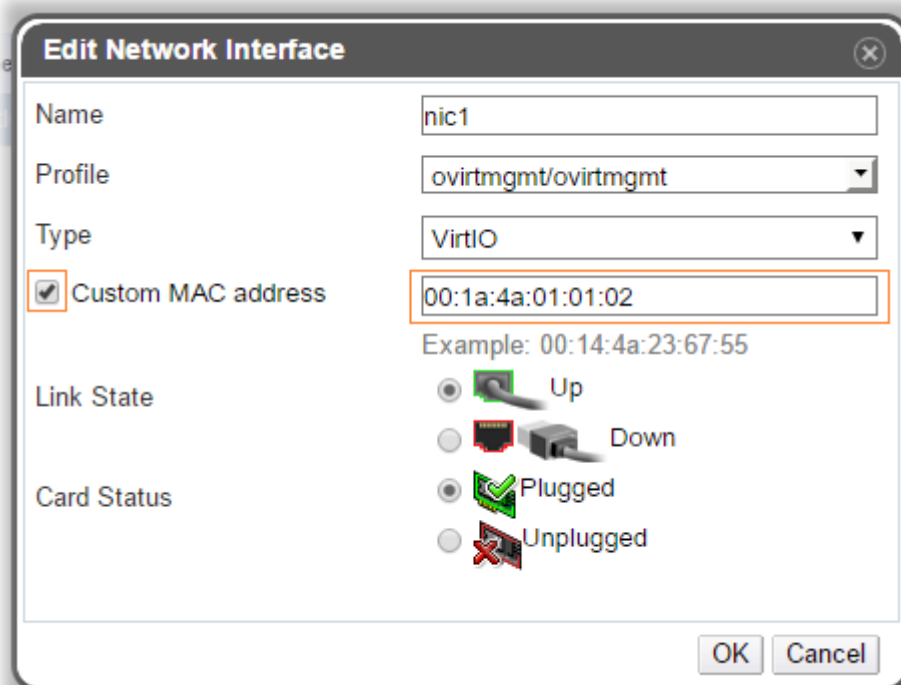
Параметры, перечисленные на других закладках, на данный момент носят исключительно информативный характер и не имеют возможности корректировки. После нажатия кнопки **OK** в форме импорта виртуальной машины мы получим сообщение о том, что окончание процедуры импорта можно отследить на закладке **Events**



В итоге в перечне виртуальных машин появится восстановленная нами копия с указанным ранее именем в выключенном состоянии. И тут нужно обратить внимание на одну немаловажную вещь. Если вы сравните **MAC**-адреса виртуальных сетевых интерфейсов исходной и восстановленной из копии виртуальных машин, то сможете заметить то, что они различаются. При импорте виртуальной машины oVirt автоматически назначает новые MAC-и из своего пула, чтобы свести к минимуму возможность конфликтов с другими VM.

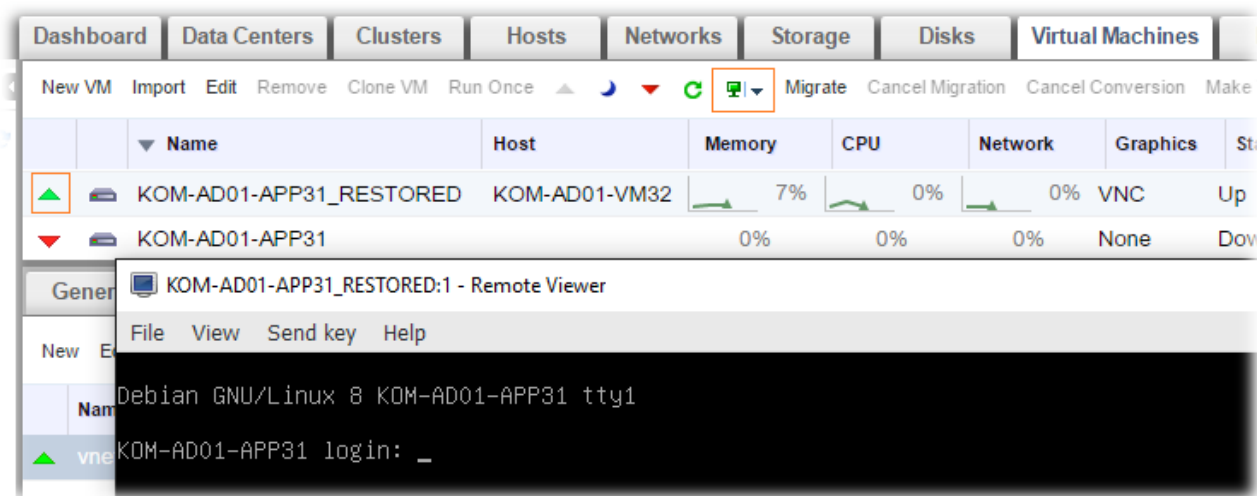


В некоторых случаях это может привести к тому, что если сразу включить восстановленную виртуальную машину, то в гостевой ОС может потребоваться дополнительная настройка сети, так как виртуальный сетевой адаптер с новым MAC-адресом будет распознан как новое устройство. Если мы хотим оставить для восстановленной VM новый MAC адрес, то тогда возможно будет полезен ранее описанный способ изменения настроек сетевых интерфейсов в гостевой ОС, чтобы потом не пришлось менять настройки TCP/IP. Если же мы хотим использовать для восстановленной VM тот же MAC-адрес, который был у оригинальной VM, то перед запуском VM с помощью кнопки **Edit** можно будет задать нужное значение.



При этом нужно помнить про то, что старый MAC, который мы хотим назначить восстановленной VM, не должен быть использован ни на какой другой VM в oVirt, то есть если старая VM всё ещё не удалена, то ей придётся задать предварительно какой-нибудь другой MAC, чтобы освободить нужный нам MAC-адрес. В противном случае oVirt нам попросту не даст произвести замену адреса.

Теперь мы можем запустить нашу восстановленную виртуальную машину и убедиться в том, что она работает с прежними сетевыми настройками и как и раньше доступна по сети.



Всё что нам останется сделать, это проверить работоспособность всех значимых для нас сервисов внутри гостевой VM и удалить старую VM, если она ещё не была удалена ранее.

Таким образом, один раз настроив резервное копирование виртуальных машин на стороне сервера oVirt Engine, мы получаем вполне жизнеспособный функционал последующего управления резервными копиями непосредственно из веб-консоли администрирования oVirt. И это хорошо.

Единственным, существенным на мой взгляд, недостатком в данном способе резервного копирования VM является невозможность инкрементального копирования, что, при использовании VM с дисками большого объёма может привести к необходимости наличия существенного объёма дисковых ресурсов под резервные копии, хранимые в Export Domain. Однако не стоит забывать и про то, что выше рассмотренный скрипт резервного копирования можно вызывать с разными настройками для разных виртуальных машин, например машины малого объёма хранить большее количество дней, а машины большого объёма – меньшее количество дней. При этом также можно использовать разные хранилища Export Domain с разным объёмом и разными физическими дисками.