



Есть множество способов выполнить резервное копирование отдельной информации или целых серверов. Я хочу рассказать о самом простом способе полного бэкапа сервера и переноса его на другое железо, если будет такая необходимость. Делается все это очень просто, без лишних телодвижений с помощью бесплатного Veeam Agent for Linux FREE.

Если у вас есть желание научиться строить и поддерживать высокодоступные и надежные системы, рекомендую познакомиться с **онлайн-курсом «Администратор Linux»** в OTUS. Курс не для новичков, для поступления нужно пройти .

Содержание:

- 1 Введение
- 2 Установка Veeam Agent for Linux
- 3 Настройка полного бэкапа сервера
- 4 Перенос или восстановление linux сервера
- 5 Перенос виртуальной машины с KVM на Hyper-V
- 6 Заключение

Введение

Ранее я уже неоднократно рассматривал вопрос резервного копирования данных или целых серверов linux. Конкретно в этих статьях:

- Бэкап сервера с помощью Duplicity.
- Создание резервной копии на Яндекс.Диске.
- Настройка Rsync для бэкапа данных.

Забэкапить сразу весь сервер можно, например, с помощью Duplicity. Но вот восстановить его на другом железе будет не так просто. Помимо данных



нужно будет, как минимум, позаботиться о разметке диска, установке загрузчика. На это необходимо затратить некоторые усилия и немного разбираться в теме initramfs и grub. Сам я не очень разбираюсь в нюансах работы этих инструментов и очень не люблю с ними возиться.

Некоторое время назад появился отличный бесплатный продукт для бэкапа всего сервера целиком. Речь идет о Veeam Agent for Linux FREE. С его помощью можно сделать полный backup сервера, положить его куда-нибудь по **smb** или **nfs**, потом загрузиться с live cd и восстановить из бэкапа на другом железе.

Сразу расскажу о некоторых нюансах работы бесплатной версии, с которыми столкнулся в процессе эксплуатации замечательного продукта от veeam.

1. Бэкап можно сделать либо всего сервера сразу, либо отдельного диска, либо отдельных папок и файлов. При выборе бэкапа всего диска или сервера, нельзя задать исключения для отдельных папок или файлов. Это очень неудобно, но увы и ах, таков функционал. Исключения можно сделать только если вы делаете бэкап на уровне папок.
2. Бэкап можно положить локально на соседний раздел, если делаете резервную копию раздела, локально в папку — если делаете бэкап файлов и папок. Если бэкапите всю систему целиком, то удаленно по smb и nfs. К сожалению, по ftp или sftp программа не работает.

В качестве хранилища для архивов может выступать репозиторий Veeam Backup & Replication. Но я не рассматриваю этот вариант, так как в данном случае использую только бесплатное решение.

Мне очень хотелось настроить резервную копию всего сервера на Яндекс.Диск, но, к сожалению, у меня это не получилось из-за технических ограничений. Яндекс.Диск подключается к системе через **webdav**. Для того, чтобы сделать резервную копию всей системы, нужно бэкапить либо всю систему сразу, либо образ диска. Если у вас небольшой веб сервер, то скорее всего на нем только один раздел. На этом же разделе хранится кэш, который использует webdav для передачи файлов. Без кэша он работать не умеет.

Думаю вы уже поняли, в чем проблема сделать полный backup сервера с помощью Veeam Agent for Linux на Яндекс.Диск по webdav. Вы не сможете добавить в исключения папку с кэшем от webdav. В итоге, во время бэкапа с помощью veeam будет расти папка с кэшем webdav, которая, в свою очередь, будет бэкапиться. В итоге, свободное место на диске закончится, бэкап прервется.

Я подробно описал ситуацию с Яндекс.Диском, потому что пространство на нем не дорого стоит. Я часто его использую в повседневной жизни, настраиваю бэкапы, храню данные и т.д. В общем, мне он нравится по ряду причин. Для того, чтобы бэкапить весь сервер целиком, вам придется найти место для архивных копий с доступом по **smb** или **nfs**. Таких предложений не очень много на рынке. Практически не из чего выбирать, я специально искал.

Остановился вот на этом варианте — KeyDisk. После оплаты, вам дают адрес сервера, логин и пароль. Вы можете сразу же подключаться по smb к хранилищу. Можно прям в windows через два обратных слеша зайти или подмонтировать хранилище к linux серверу.





KeyDisk стоит примерно 350р. в месяц за 100 гигабайт. Не очень дешево, конечно, в сравнении с облачными сервисами, но все равно не дорого. Похожих предложений с доступом по smb я лично вообще не нашел в принципе. Этот объем позволит вам забэкапить небольшой веб сервер с глубиной архива в несколько недель или месяцев, в зависимости от того, сколько данных у вас на нем хранится.

Дальше я подробно на конкретном примере расскажу как все настроить и восстановить или перенести сервер целиком, если понадобится. Причем переносить буду вообще на другое железо. Но обо всем по порядку.

Установка Veeam Agent for Linux

Для установки Veeam Agent for Linux необходимо подключить репозиторий veeam под нужную вам систему. Это можно сделать либо руками, либо скачать файл с репозиторием в виде **rpm** или **deb** пакета. Сделать это можно на странице с описанием продукта.



Для того, чтобы получить доступ к разделу с загрузками, придется зарегистрироваться. Выбираете тип системы и скачиваете репу.



Чуть ниже рекомендую сразу же скачать **Veeam Linux Recovery Media**. Он нам понадобится, когда мы будем переносить сервер на другое железо или восстанавливать из бэкапа.

Копируем файл с репозиторием на сервер и устанавливаем его. На момент написания статьи, файл можно было скачать по прямой ссылке.

```
# cd /root
# wget https://download2.veeam.com/veeam-release-el7-1.0-1.x86_64.rpm
# rpm -Uvh veeam-release-el7-1.0-1.x86_64.rpm
```



Обновляем репозитории и устанавливаем veeam.

```
# yum update  
# yum install veeam
```

Все, Veeam Agent for Linux установлен и готов к работе.

Настройка полного бэкапа сервера

Сделать бэкап с помощью Veeam Agent for Linux очень просто. Вариантов настроек не так много, можете сами все проверить и посмотреть. Я для примера рассмотрю вариант с созданием полного бэкапа всей системы и перенос ее на другое железо. Создаем задачу для резервного копирования сервера на наше хранилище по smb.

```
# veeam
```

Нам сразу же предлагают указать файл с лицензией. Так как у нас лицензии нет, то отказываемся. Нас встречает главное окно программы.



Нажимаем **C (configure)** для настройки задания на backup. Задаем любое имя задания, затем указываем, что будем делать полный бэкап сервера.



В качестве приемника для архива системы, указываем **Shared Folder**.



Далее нужно ввести параметры доступа к хранилищу бэкапов. Я использую свои от системы KeyDisk.





В пункте **Restore Points** указывается глубина архива. Это число копий, которые будут храниться на сервере. Если делать бэкап каждый день и указать число 14, то будут храниться резервные копии системы за последние 14 дней. Если делать будете через день, то за 28 дней и т.д.

Можно создавать несколько заданий с различной глубиной архива. Например, каждый день с глубиной 7 копий, раз в неделю с глубиной 4, и раз в месяц с глубиной в 12. Таким образом у вас всегда будут последние 7 бэкапов системы на этой неделе. Потом по одному бэкапу в неделю за последний месяц и 12 бэкапов по месяцам в течении последнего года.

Если получите ошибку:

```
Current system does not support cifs. Please install cifs client package.
```



Установите пакет **cifs**. В CentOS вот так:

```
# yum install cifs-utils
```

И так в Debian/Ubuntu:

```
# apt install cifs-utils
```

Запускайте заново veeam и продолжайте. После настройки Destination, предлагается указать скрипты для выполнения перед и после бэкапа. Нам сейчас это не надо. Далее настраиваем расписание и запускаем задание на архивацию в конце настройки.



Запустилась архивация. Можно следить за ее прогрессом.



После завершения архивации системы, можно проверить содержимое сетевого хранилища, зайдя на него прямо из винды.



На этом настройку полного бэкапа сервера мы завершили. Резервная копия системы лежит в надежном месте. Попробуем теперь с нее восстановиться.

Перенос или восстановление linux сервера

Представим теперь ситуацию, что наш веб, или какой-нибудь другой сервер умер, и нам надо восстановить систему в другом месте. Выполним полное восстановление всего сервера с помощью созданной ранее резервной копии. Для этого нам понадобится **Veeam Linux Recovery Media**, который мы скачали ранее.

Для восстановления системы нужно соблюсти два обязательных условия:

1. Готовим новый сервер с диском, который должен быть не меньше диска исходного сервера. Это обязательное условие, иначе восстановление системы даже не начнется. Veeam скажет, что размер диска недостаточный и не предложит больше никаких вариантов восстановления.
2. Оперативной памяти для системы должно быть не меньше 1024 Мб. Если меньше, то загрузка с диска не будет выполнена. Система скажет, что она не может развернуть корневой раздел.

Загружаемся с диска. В разделе **Configure network** убеждаемся, что сеть настроена, получен ip адрес, который имеет доступ к интернету. Далее выбираем **Restore volumes** -> **Add shared folder**. Заполняем параметры доступа к хранилищу архивов.



Выбираем там директорию с нашим архивом системы, которую будем восстанавливать. Далее будет показан список задач в левом столбце и список резервных копий в правом.



В моем случае там только одна копия. Выбираю ее. Дальше мы видим слева список дисков нашего сервера, справа диски резервной копии.



У меня слева чистый диск, справа тоже один диск, на который установлен загрузчик и есть один раздел с корнем системы. Выбираем справа наш диск (не



раздел с корнем!!!) и жмем **Restore whole disk to**.



В качестве приемника выбираем пустой диск на новом сервере.



Нажимаем **S (Start restore)**. Визард покажет список действий, которые будут выполнены и попросит их подтвердить, нажатием на Enter.



Делаем это и наблюдаем за процессом восстановления сервера centos из бэкапа.



Дожидаемся окончания переноса сервера, выбираем перезагрузку и извлекаем загрузочный CD. Грузимся с жесткого диска.

Дальше может быть много различных вариантов. Если вы переносите сервер на тот же гипервизор, то проблем скорее всего не будет, и все заведется сразу. Если же гипервизор другой, то могут быть варианты, в зависимости от ситуации.

Перенос виртуальной машины с KVM на Hyper-V

В моем случае я переносу сервер с KVM на Hyper-V. После загрузки системы я получаю такую картину.



Сервер начинает бесконечно висеть в подобном состоянии с такими характерными ошибками:

```
Warning: dracut-initqueue timeout starting timeout scripts
```



```
a start job is running for dev-disk-by .....
```

Начинаю разбираться в чем может быть дело. Конечно, тут решение проблемы будет зависеть от конкретной ситуации. А успешность решения от квалификации сисадмина. Я уже немного повозился с подобными переносами и примерно представляю, в чем тут может быть проблема. Частично я эту тему затрагивал, когда делал перенос виртуальных машин с XenServer на Hyper-V. Но там была другая проблема, связанная с кастомным ядром от Xen.

В нашей ситуации с переносом виртуальной машины с KVM на Hyper-V проблема в другом. У нас поменялось имя диска. Нам нужно изменить это имя в **fstab** и в конфиге **grub**. До кучи я еще собрал заново initramfs, но не уверен на 100%, что в данном случае это нужно было делать. Я сделал на всякий случай сразу все за один заход.

Итак, загружаемся с установочного диска CentOS 7 и выбираем режим **Rescue a CentOS system**. Подробно об этом рассказывал в упомянутой ранее статье с переносом от хеп. Выбираем первый режим запуска.



Дальше работаем в консоли. Смотрим, как называется наш диск.

```
# fdisk -l
```



У меня это **sda**, а на прошлом сервере он назывался **vda**. Нам нужно внести эти изменения в 2 файла:

1. /etc/fstab
2. /boot/grub2/grub.cfg

Диск восстановления в самом начале мог сам смонтировать системный раздел в директорию */mnt/sysimage*. Если он этого не делает по какой-то причине, то сделайте это сами:

```
# mount /dev/sda1 /mnt/sysimage
```




Теперь нам надо сделать chroot в систему, предварительно смонтировав туда информацию о текущей системе. Выполняем команды:

```
# mount --bind /proc /mnt/sysimage/proc
# mount --bind /dev /mnt/sysimage/dev
# mount --bind /sys /mnt/sysimage/sys
# mount --bind /run /mnt/sysimage/run
# chroot /mnt/sysimage
```

Мы загрузились в окружение нашего сервера. Тут можете использовать установленный у вас на сервере текстовый редактор. С его помощью изменяете имена дисков в файлах `/etc/fstab` и `/boot/grub2/grub.cfg`. Можете просто автозаменой поменять имена.

Теперь соберем новый **initramfs**. Идем в директорию `/boot` и смотрим там последнюю версию ядра.

```
# cd /boot
# ls -l | grep initramfs
```



В данном случае просто смотрим самые высокие цифры. Соберем новый `initramfs` в соответствии с версией ядра.

```
# dracut initramfs-3.10.0-514.26.2.el7.x86_64.img 3.10.0-514.26.2.el7.x86_64
```

В завершении установим измененный загрузчик на наш диск:

```
# grub2-install /dev/sda
```

Перезагружаем сервер. После этих изменений, у меня благополучно все загрузилось. Перенос виртуальной машины с KVM на Hyper-V выполнен полностью. Причем, у нас не было доступа к образу системы. Хотя подобная ошибка скорее всего все равно возникла бы, даже если бы мы конвертировали и переносили готовый образ.



Заключение

Изначально планировал написать небольшую заметку на тему использования Veeam для бэкапа сервера. Но в процессе получилось разобрать еще и перенос сервера с одного гипервизора на другой. Еще раз повторюсь, кому показалось это слишком сложным. Если вы будете бэкапить и восстанавливать сервер в рамках одного и того же гипервизора, то описанных выше проблем у вас не будет. Все пройдет гладко.

При переносе с железа на виртуальную машину или наоборот, тоже скорее всего возникнут какие-нибудь проблемы. Не существует софта или готового решения, которое бы позволило все это выполнить в автоматическом режиме. С проблемами загрузки придется разбираться по ходу дела. Но две основные проблемы я разобрал:

1. Неподходящие версии ядер. После переноса нужно будет переустановить или обновить ядро.
2. Разные имена дисков или меток разделов. Нужно будет их привести в соответствие с новым железом.

Это наиболее популярные проблемы. С другими мне не приходилось сталкиваться. Хотя не сказать, что мне часто приходилось переносить сервера, но некоторый опыт есть. Думаю, эта статья будет многим полезна, так как подобный перенос не очень раскрыт в статьях в интернете. По крайней мере мне не попадались хорошие гайды на эту тему. Разбираюсь обычно сам с помощью гугления по англоязычному сегменту.

Делитесь своим опытом и оставляйте замечания к статье или указывайте на ошибке в комментариях.

Онлайн курс "DevOps практики и инструменты"

Если у вас есть желание научиться строить и поддерживать высокодоступные и надежные системы, научиться непрерывной поставке ПО, мониторингу и логированию web приложений, рекомендую познакомиться с **онлайн-курсом «DevOps практики и инструменты»** в OTUS. Курс не для новичков, для поступления нужны базовые знания по сетям и установке Linux на виртуалку. Обучение длится 5 месяцев, после чего успешные выпускники курса смогут пройти собеседования у партнеров. Проверьте себя на вступительном тесте и смотрите программу детальнее по .



Помогла статья? Есть возможность отблагодарить автора