

# Глава 6. Виртуальная сеть

## Оглавление

[Виртуальное сетевое оборудование](#)

[Введение в сетевые режимы](#)

[Network Address Translation \(NAT\)](#)

[Настройка проброса портов в NAT](#)

[PXE загрузка в NAT](#)

[Ограничения NAT](#)

[Сетевой мост](#)

[Внутренняя сеть](#)

[Виртуальный адаптер хоста](#)

[UDP Tunnel networking \(UDP туннель\)](#)

[VDE сеть](#)

Как было упомянуто в [разделе "Настройка сети"](#), VirtualBox предоставляет до восьми виртуальных PCI Ethernet интерфейсов(карт) для каждой машины. Для каждого интерфейса вы можете указать

1. оборудование которое будет эмулироваться
2. типподключения — режим эмуляции и взаимодействия виртуальных и физических сетевых устройств хоста

Четыре сетевых карты могут быть настроены в графическом интерфейсе - в окне настройки свойств виртуальной машины на вкладке «Сеть». Вы можете настроить все восемь сетевых интерфейсов в с помощью команды VBoxManage modifyvm; см. [Раздел, "VBoxManage modifyvm"](#) .

Эта глава детально объясняет различные параметры сетевых настроек.

## Виртуальное сетевое оборудование

Для каждого сетевого интерфейса, вы можете выбрать оборудование которое будет подключено к виртуальной машине. VirtualBox может эмулировать работу следующего сетевого оборудования:

- AMD PCNet PCI II (Am79C970A);
- AMD PCNet FAST III (Am79C973, the default);
- Intel PRO/1000 MT Desktop (82540EM);
- Intel PRO/1000 T Server (82543GC);
- Intel PRO/1000 MT Server (82545EM);
- Паравиртуальная сетевая карта (virtio-net).

Сетевая карта PCNet FAST III устанавливается по умолчанию т.к. она поддерживается практически всеми операционными системами, а также загрузчиком GNU GRUB. В случае выбора карт семейства карт Intel PRO/1000 , учтите что некоторых гостевые операционные системы не поставляются с драйверами для карт PCNet, например такие как Windows Vista.

Intel PRO/1000 MT Desktop поддерживается в Windows Vista и старших версиях. Серверный вариант T карты Intel PRO/1000 опознается гостевой

Windows XP без дополнительной установки драйверов. MT Server облегчает импорт OVF файлов с других платформ.

Адаптер "Паравиртуальная сеть (virtio-net)" является специальным устройством. Если вы выберете его, то VirtualBox не будет виртуализировать обычное сетевое оборудование (которое поддерживается популярными операционными системами). Вместо этого, VirtualBox ожидает, что гостевой системой будет обеспечен специальный программный интерфейс для виртуальной среды, избегая таким образом сложной эмуляции оборудования, что увеличивает производительность. Начиная с версии 3.1, VirtualBox поставляется с поддержкой стандартного сетевого драйвера "virtio", который является частью открытого проекта KVM.

Сетевой драйвер "virtio" доступен для следующих гостевых ОС:

- Linux с ядром начиная с версии 2.6.25 может быть настроен для предоставления поддержки virtio; некоторые дистрибутивы предоставляют его поддержку для старых ядер.
- Для Windows 2000, XP и Vista, virtio драйвера можно загрузить и установить со web страницы проекта KVM.<sup>[28]</sup>

VirtualBox имеет ограниченную поддержку для так называемых jumbo frames , т.е. сетевых пакетов размером более чем 1500 байт, при использовании сетевых карт Intel и в режиме сетевого моста. Другими словами, jumbo frames не поддерживаются с сетевыми AMD устройствами; в данном случае , jumbo пакеты не будут посылаться и отправляться. Гостевые операционные системы пытающиеся использовать эту возможность столкнутся с потерей пакетов, что может привести к непредсказуемому поведению программ в гостевой системе. В сконфигурированных гостевых системах по умолчанию это не вызывает проблем, т.к. для использования jumbo frames настраивается дополнительно.

## Введение в сетевые режимы

Для каждого сетевого интерфейса возможно указать в каком из следующих режимов они будут функционировать:

Не подключен

В этом режиме, VirtualBox сообщает гостю что сетевой адаптер есть, но он не подключен -- так как если бы Ethernet кабель не присоединен к карте. Таким образом возможно симулировать "выдергивание" кабеля из виртуальной сетевой карты и обрыв соединения, что может быть полезно для информирования гостевой ОС об отсутствии сетевого соединения, но возможности его настройки.

Network Address Translation (NAT)

Если вам нужно просматривать Web, загружать файлы и пользоваться почтой e-mail в гостевой системе, то данная сетевая конфигурация по умолчанию подойдет вам и вы можете пропустить чтение этого раздела. Примите к сведению, что существуют некоторые ограничения при использовании сетевых папок Windows (см. <a3> раздел "Ограничения NAT" </a3>).

## Сетевой мост

Этот режим нужен для более продвинутых сетевых возможностей, например моделирование сетей и работающих серверов в гостевой системе. При его включении, VirtualBox подключается к одной из установленных сетевых карт и напрямую работает с сетевыми пакетами, не используя сетевой стек ОС хоста.

## Внутренняя сеть

Этот режим можно использовать для создания виртуальной сети которая доступна из виртуальной машины, но не для приложений запущенных на хосте или других внешних сетевых устройств.

## Виртуальный адаптер хоста

Данный режим можно использовать для создания сетей из хоста и нескольких виртуальных машин, без использования физического сетевого интерфейса хоста. На хосте создается виртуальный сетевой интерфейс (подобный петлевому интерфейсу) , обеспечивающий соединения между хост системой и виртуальными машинами.

## Универсальный драйвер

Редко используемый режим универсального сетевого интерфейса (Rarely used modes share the same generic network interface), позволяет выбирать пользователю драйвер, который может быть включен в VirtualBox или поставляться с пакетом расширений (extension pack).

В данное время доступны два подтипа:

### UDP тунель

Можно использовать для простого и прозрачного прямого соединения виртуальных машин работающих на разных хостах , через существующую сетевую инфраструктуру.

### VDE (Virtual Distributed Ethernet) сеть

Позволяет использовать Virtual Distributed Ethernet коммутатор на Linux или FreeBSD хосте. В данный момент, для его использования требуется сборка VirtualBox из исходных кодов, т.к. в дистрибутив Oracle он не включен.

В следующих разделах более подробно описываются эти сетевые режимы.

## Network Address Translation (NAT)

Режим Network Address Translation (NAT) предоставляет наиболее простой способ доступа к внешней среде из виртуальной машины. Обычно, для него не требуется никаких настроек хоста и гостевой системы. Поэтому он является сетевым режимом, настраиваемым по умолчанию.

Виртуальная машина с сетевым интерфейсом в режиме NAT подключается к сети, также как реальный компьютер подключается к Internet через маршрутизатор. "Маршрутизатором" в данном случае выступает сетевой модуль VirtualBox, который обрабатывает сетевой трафик виртуальной машины. Недостаток режима NAT, как и в случае локальной сети за маршрутизатором, в том что виртуальная машина недоступна для внешней сети (internet); вы не можете обрабатывать сетевые запросы, пока не настроите переброс портов (описывается ниже).

Сетевые пакеты посылаемые гостевой ОС получает модуль VirtualBox NAT, который обрабатывает данные стека TCP/IP и пересылает их операционной системе хоста. Он определяет какие данные посылать приложениям хоста, а какие другим компьютерам той же сети, что и хост, используя сетевой интерфейс хоста. VirtualBox также перехватывает и пересылает ответные пакеты гостевой машине.

Виртуальная машина получает сетевой адрес и другие его настройки в локальной сети от сервера DHCP встроенного в VirtualBox. Таким образом IP адрес присваивается виртуальной машине отличный от сетевого адреса сети хоста. Возможно настроить использование нескольких сетевых карт в виртуальной машине в режиме NAT, тогда по умолчанию первая карта будет принадлежать локальной сети 10.0.2.0, вторая 10.0.3.0 и т.д. Если по каким то причинам, вам необходимо изменить диапазон выдаваемых гостям IP, то обратитесь к [разделу "Тонкая настройка VirtualBox NAT"](#).

## Настройка проброса портов в NAT

Так как виртуальная машина подключается к своей собственной сети VirtualBox и не доступна для хоста, то сетевые службы в гостевой системе не доступны для хоста и других компьютеров в сети хоста. Однако, подобно физическому маршрутизатору, VirtualBox позволяет сделать выбранные сервисы доступными с использованием механизма проброса портов. Данный механизм позволяет прослушивать некоторые порты хоста и пересылать все сетевые пакеты приходящие на этот порты гостевой системе.

Для программ на хосте или других физических (или виртуальных) машин в сети, это выглядит так как будто бы служба работает на хост системе. Это означает также, что вы не можете запускать на хосте такую же службу на порту с тем же номером. Тем не менее, обслуживание сетевых запросов в виртуальной машине дает некоторые преимущества, например, уязвимость или крах виртуальной сетевой службы не ведет к компрометированию хост системы, так как сервис работает в другой операционной системе.

Вы можете настроить проброс портов, используя команду VBoxManage ; за подробностями обращайтесь к [разделу "VBoxManage modifyvm"](#).

Вам необходимо знать какие порты используются в госте сетевой службой и определить какие порты использовать на хосте (не обязательно использовать номера тех же портов, что и в гостевой системе). Вы можете использовать любые порты на хосте, которые не используются им. Например, для настройки службы ssh в госте, используйте следующую команду:

```
VBoxManage modifyvm "VM name" --natpf1 "guestssh,tcp,,2222,,22"
```

Для примера выше, весь TCP трафик поступающий на порт 2222 любого интерфейса хоста будет перенаправлен на порт 22 гостя. Имя протокола `tcp` является обязательным параметром для определения того, какой протокол будет использоваться для проброса (можно также использовать `udp`). Имя `guestssh` выбрано произвольно для этого конкретного примера. Число после `--natpf` определяет сетевую карту, так же как и в других командах `VBoxManage`.

Для удаления правила перенаправления трафика порта, используйте следующую команду:

```
VBoxManage modifyvm "VM name" --natpf1 delete "guestssh"
```

Если по каким то причинам гость использует статический IP адрес, который не выдан встроенным DHCP сервером, то обязательно нужно указать IP адрес гостя на котором привязано правило проброса:

```
VBoxManage modifyvm "VM name" --natpf1 "guestssh,tcp,,2222,10.0.2.19,22"
```

Данный пример идентичен предыдущему, за исключением того что механизму обеспечивающему NAT, сообщают что гостевая система должна искать на адресе 10.0.2.19.

Для проброса всего входящего трафика с нужного интерфейса хоста, укажите IP адрес интерфейса, например:

```
VBoxManage modifyvm "VM name" --natpf1 "guestssh,tcp,127.0.0.1,2222,,22"
```

В данном случае весь входящий TCP трафик на локальном интерфейсе хоста (127.0.0.1) с порта 2222 будет пересылаться на 22 порт гостя.

Нельзя настраивать проброс портов соединений для запущенной виртуальной машины. Однако, это возможно сделать для ВМ, которая сохранена или выключена.

## РХЕ загрузка в NAT

Механизм загрузки РХЕ теперь поддерживается в режиме NAT. Сервер NAT DHCP предоставляет загрузочный файл `vmname.pxe` если существует каталог TFTP в каталоге где находится файл `VirtualBox.xml`. Пользователю должен обеспечить существование корректного файла `vmname.pxe`.

## Ограничения NAT

Существует четыре ограничения в режиме NAT о которых необходимо знать:

Ограничения протокола ICMP:

Многие, часто используют сетевые утилиты отладки ( `ping` или `tracert`) используют протокол ICMP для отправки и получения сообщений. Хотя поддержка ICMP протокола была значительно улучшена в `VirtualBox 2.1` ( `ping` теперь работает), но при работе с некоторыми утилитами возможны проблемы.

Широковещательные пакеты UDP:

Гостевые системы осуществляют ненадежное получение широковещательных пакетов, что сделано для улучшения производительности, они получают широковещательные пакеты только в определенный промежуток времени, после того как гость отправляет пакет UDP. Как следствие, протокол разрешения имен NetBios не всегда работает корректно (но WINS работает). В данном случае используйте обходной путь - вы можете использовать непосредственно IP адреса для доступа к сетевым ресурсам \\server\share.

Не поддерживаются протоколы, такие как GRE:

Протоколы отличные от TCP и UDP не поддерживаются. Это означает что нельзя использовать VPN ( PPTP от Microsoft). Существуют другие реализации VPN которые используют TCP и UDP.

Проброс портов хоста с номерами меньше 1024 невозможен:

На Unix системах (Linux, Solaris, MacOS X) нельзя использовать порты с номерами меньше 1024 в приложениях которые запущены не с правами root . В результате, если вы попытаетесь настроить переброс таких портов, то VM не запустится.

Эти ограничения обычно не влияют на обычное использование сети. Но наличие их в NAT режиме может приводить к проблемам в сетевой работе. Приведем пример - NFS, часто сервера настроены так, что отказывают в соединениях от непривилегированных портов (т.е. портов с номерами меньше 1024).

## Сетевой мост

В данном режиме VirtualBox использует драйвер устройства на хост системе , который обрабатывает данные проходящие через физический сетевой интерфейс. Этот драйвер обычно называют "net filter". Он позволяет перехватывать VirtualBox пакеты из физической сети и изменять данные в них, а также создавать новые программные сетевые интерфейсы. Когда гость использует такой программный интерфейс, то выглядит это так как будто бы гостевая система подключается к физической сети, хост система может посылать и принимать данные от гостевой. Это означает, что вы можете использовать физический интерфейс хоста в качестве маршрутизатора или шлюза между гостевой системой и вашей физической сетью.

Для этого требуется драйвер от VirtualBox на хост системе. Режим сетевого моста был полностью переписан в версиях VirtualBox 2.0 и 2.1(в зависимости от операционной системы хоста). С точки зрения пользователя, основное различие в реализации в том, что более не требуется сложная процедура настройки режима в поддерживаемых операционных системах. [\[29\]](#)

### Замечание

Несмотря на то, что в TAP больше нет необходимости для организации режима «сетевого моста», вы все еще можете использовать интерфейсы TAP для настройки «продвинутых» решений, так как вы можете соединить VM с

любым интерфейсом хоста - который так же может быть интерфейсом TAP.

Для подключения режима сетевого моста, вам необходимо в диалоговом окне свойств виртуальной машины на вкладке "Сеть" выбрать "Сетевой мост" в выпадающем списке поля "Тип подключения". Выберите необходимый интерфейс хоста из списка ниже, который содержит список физических сетевых интерфейсов вашей системы. В типичном MacBook, например, вы можете выбрать между "en1: AirPort" (беспроводный интерфейс) и "en0: Ethernet", которые представляет собой проводной сетевой интерфейс.

В зависимости от операционной системы хоста, имеются ограничения о которых вы должны знать:

- В Macintosh , ограничена функциональность AirPort в режиме сетевого моста. В настоящее время, VirtualBox поддерживает только IPv4 для AirPort. Для других протоколов, таких как IPv6 и IPX, вы должны выбрать проводные интерфейсы.
- В Linux , ограничена функциональность беспроводных интерфейсов в режиме сетевого моста. VirtualBox поддерживает только IPv4 для беспроводных соединений. Для других протоколов, таких как IPv6 и IPX, вы должны выбрать проводные интерфейсы.

Также, установленное значение MTU менее чем 1500 для проводных интерфейсов предоставляемых устройствами драйверами sky2 в Marvell Yukon II EC Ultra Ethernet NIC может приводить к потере сетевых пакетов.

- В Solaris , нет поддержки беспроводных интерфейсов. Фильтрация трафика гостя с использованием IPFilter также не полностью поддерживается, что связано с техническими ограничениями сетевой подсистемы Solaris. Эти проблемы будут решены в будущих выпусках Solaris 11.

В версии VirtualBox 2.0.4 и выше, возможно использовать Crossbow Virtual Network Interfaces (VNICs) в режиме сетевого моста, но со следующими условиями:

- VNIC не может быть разделен между несколькими гостевыми интерфейсами, т.е. каждый гостевой интерфейс должен иметь собственный VNIC.
- VNIC и использующему его гостевому интерфейсу должны быть назначены одинаковые MAC адреса.

Когда используется VLAN интерфейсы с VirtualBox, то они должны быть именованы в соответствии со схемой PPA-hack (PPA-hack naming scheme) (например, "e1000g513001"), иначе гость будет получать пакеты в некорректном формате.

## Внутренняя сеть

Режим Внутренняя сеть похож на режим Сетевой мост в котором ВМ может связываться с внешним миром. Однако, "внешний мир" ограничен другими

виртуальными машинами на том же хосте и которые подключены к той же внутренней сети.

Хотя с технической стороны, все что можно сделать используя режим внутренняя сеть, можно также сделать в режиме сетевого моста, но в режиме внутренней сети есть преимущества в безопасности. В режиме моста, весь трафик проходит через физический интерфейс хоста. Поэтому возможно подключение анализаторов пакетов (например, Wireshark) к интерфейсу хоста и сбор всего трафика проходящего через него. Если по каким либо причинам вам необходимо чтобы две и более виртуальные машины имели защищенное соединение, скрывая свои данные от хост системы и других пользователей, то режим сетевой мост не является лучшим выбором.

Внутренние сети создаются автоматически, т.е. не существует централизованной настройки. Каждая внутренняя сеть идентифицируется своим именем. При существовании более чем одной виртуальной сетевой карты с одним идентификатором внутренней сети, драйвер VirtualBox автоматически поддерживает соединение этих интерфейсов в одну сеть, выступая в роли сетевого коммутатора. VirtualBox поддерживает полностью Ethernet коммутацию, широковещательную и многоадресную рассылку сетевых пакетов и режим promiscuous mode.

Для включения режима внутренней сети для сетевой карты виртуальной машины, установите для нее тип подключения в "Внутренняя сеть". Имеется два способа сделать это:

- Вы можете использовать диалоговое окно "Свойства" в графическом пользовательском интерфейсе VirtualBox. На вкладке "Сеть", выберите "Внутренняя сеть" из выпадающего списка тип подключения. Далее выберете имя существующей внутренней сети из выпадающего списка ниже или введите новое имя.
- Вы можете использовать командную строку  
`VBoxManage modifyvm "VM name" --nic<x> intnet`

Дополнительно, вы можете указать имя сети в команде

```
VBoxManage modifyvm "VM name" --intnet<x> "network name"
```

Если вы не укажете имя сети, сетевая карта будет подключена к внутренней сети по умолчанию intnet.

Для сетевых карт (виртуальных) гостевых операционных систем в режиме внутренней сети используется статическая настройка IP адресов, вы вероятно захотите использовать DHCP сервер встроенный в VirtualBox для выдачи IP адресов во внутренней сети. Подробности в [разделе "VBoxManage dhcpserver"](#).

В качестве меры безопасности, в режиме внутренней сети Linux предоставляет доступ к этой сети только виртуальным машинам запущенным пользователем который создал для них эту внутреннюю сеть.

## Виртуальный адаптер хоста

Режим Виртуальный адаптер хоста был добавлен в версии VirtualBox 2.2. Данный режим можно рассматривать как гибридный режим сетевого моста и внутренней сети: как и в режиме моста виртуальные машины могут

соединяться друг с другом и с хост системой, как будто бы они соединены через физический коммутатор. Как и в режиме внутренней сети, нет необходимости в предоставлении физического сетевого интерфейса и виртуальные машины не могут общаться с внешней сетью хоста, т.к. они никак не связаны с физическим сетевым интерфейсом.

При использовании режима внутренней сети, VirtualBox создает новый программный интерфейс на хосте, который добавляется к списку существующих сетевых интерфейсов хоста. Другими словами, в режиме сетевого моста существующий физический интерфейс используется для подключения виртуальных машин, а в режиме "виртуальный адаптер хоста" создается новый "петлевой" интерфейс хоста. В режиме внутренней сети трафик между виртуальными машинами не "виден", а трафик "петлевого" интерфейса возможно перехватить.

Режим виртуального адаптера хоста удобно использовать для нескольких предварительно настроенных виртуальных систем, которые предназначены для совместного использования. Например, одна виртуальная машина представляет собой web сервер который использует вторую с сервером базы данных. Другой дополнительный, сетевой интерфейс (мост) может соединить web сервер с внешним миром для выдачи данных, но внешнему миру не будет доступа к серверу базы данных.

Для установки интерфейса виртуальной машины в режим виртуального адаптера :

- перейдите на вкладку "Сеть" в окне настроек виртуальной машины и установите для интерфейса режим "Виртуальный адаптер хоста" или
- в командной строке, введите `VBoxManage modifyvm <VM name> --nic<x> hostonly` ; см. [раздел "VBoxManage modifyvm"](#) .

Для данного режима, как и для режима внутренняя сеть, вы можете использовать сервер DHCP встроенный в VirtualBox. Это позволит управлять IP адресами для интерфейсов в режиме виртуальный адаптер хоста, вместо того чтобы вручную назначать IP адреса.

- В графическом интерфейсе VirtualBox, вы можете получить список всех виртуальных сетей хоста, через "Файл" -> "Настройки" -> "Сеть". Кликнете на имени сети и кнопку "Редактировать" справа, для того чтобы изменить свойства устройства и настройки DHCP.
- В качестве альтернативы вы можете использовать команду `VBoxManage dhcpserver`; см. [раздел "VBoxManage dhcpserver"](#).

## UDP Tunnel networking (UDP туннель)

Этот сетевой режим позволяет соединять две виртуальные машины запущенные на разных хостах.

Технически это осуществляется путем инкапсуляции Ethernet кадров, отправляемых или получаемых гостевой сетевой картой в UDP/IP датаграммы и отправки их в любую сеть доступную хосту.

Режим UDP туннеля имеет три параметра:

UDP порт источник

Это порт хост на котором принимает пакеты. Датаграммы поступающие на данный порт с любого адреса перенаправляются на сетевую карту гостевой системы.

Адрес назначения

IP адрес хоста на который передаются данные.

UDP порт назначения

Номер порта на который посылаются данные.

При соединении двух VM на различных хостах, нужно поменять местами их IP адреса (When interconnecting two virtual machines on two different hosts, their IP addresses must be swapped). В случае одного хоста, UDP порты источника и назначения должны быть поменяны местами.

В следующем примере хост 1 имеет IP адрес 10.0.0.1, а хост 2 10.0.0.2. Настройка через командную строку:

```
VBoxManage modifyvm "VM 01 on host 1" --nic<x> generic
VBoxManage modifyvm "VM 01 on host 1" --nicgenericdrv<x> UDPTunnel
VBoxManage modifyvm "VM 01 on host 1" --nicproperty<x> dest=10.0.0.2
VBoxManage modifyvm "VM 01 on host 1" --nicproperty<x> sport=10001
VBoxManage modifyvm "VM 01 on host 1" --nicproperty<x> dport=10002
```

и

```
VBoxManage modifyvm "VM 02 on host 2" --nic<y> generic
VBoxManage modifyvm "VM 02 on host 2" --nicgenericdrv<y> UDPTunnel
VBoxManage modifyvm "VM 02 on host 2" --nicproperty<y> dest=10.0.0.1
VBoxManage modifyvm "VM 02 on host 2" --nicproperty<y> sport=10002
VBoxManage modifyvm "VM 02 on host 2" --nicproperty<y> dport=10001
```

Вы можете соединять две виртуальные машины на одном хосте, установив адрес назначения на обоих VM в 127.0.0.1 . В данном случае это будет работать аналогично режиму "внутренняя сеть", однако на хосте будет доступен трафик между VM, который не виден в случае внутренней сети.

### Замечание

На Unix системах (Linux, Solaris, MacOS X) нельзя использовать порты с номерами меньше 1024 в приложениях которые запущены не с правами root . В результате, если вы попытаетесь настроить проброс таких портов, то VM не запустится.

## VDE сеть

Virtual Distributed Ethernet (VDE<sup>[30]</sup>) является гибкой и безопасной виртуальной сетевой системой. Она реализует L2/L3 коммутаторы, что позволяет эмулировать протоколы spanning-tree, VLANs и WAN. Эта необязательная часть VirtualBox, которая содержится только в исходных кодах.

Базовыми элементами инфраструктуры являются VDE коммутаторы, VDE plugs и VDE wires.

VirtualBox VDE драйвер имеет один параметр:

VDE сеть

Имя сокета коммутатора VDE сети к которому подключается VM.

В следующем примере показано, как подключить виртуальную машину к VDE коммутатору (VDE switch):

1. Созайте VDE switch:

```
vde_switch -s /tmp/switch1
```

2. Настройка через командную строку:

```
VBoxManage modifyvm "VM name" --nic<x> generic
```

```
VBoxManage modifyvm "VM name" --nicgenericdrv<x> VDE
```

Для автоматического подключения к выделенному порту коммутатора, используйте:

```
VBoxManage modifyvm "VM name" --nicproperty<x> network=/tmp/switch1
```

Для подключения к определенному порту <n>, используйте команду:

```
VBoxManage modifyvm "VM name" --nicproperty<x> network=/tmp/switch1[<n>]
```

Последний вариант полезно применять в VLANs.

3. Дополнительно, для связывания порта VDE switch и VLAN: (в командной строке коммутатора CLI)

```
vde$ vlan/create <VLAN>
```

```
vde$ port/setvlan <port> <VLAN>
```

VDE доступен только на хостах с Linux и FreeBSD. It is only available if the VDE software and the VDE plugin library from the VirtualSquare project are installed on the host system. For more information on setting up VDE networks, please see the documentation accompanying the software.<sup>[31]</sup>

---

[28] <http://www.linux-kvm.org/page/WindowsGuestDrivers>.

[29] Для хостов Mac OS X и Solaris, драйвер сетевого фильтра были добавлены в VirtualBox 2.0. With VirtualBox 2.1, net filter drivers were also added for the Windows and Linux hosts, replacing the mechanisms previously present in VirtualBox for those platforms; especially on Linux, the earlier method required creating TAP interfaces and bridges, which was complex and varied from one distribution to the next. None of this is necessary anymore. Bridged network was formerly called "Host Interface Networking" and has been renamed with version 2.2 without any change in functionality.

[30] VDE is a project developed by Renzo Davoli, Associate Professor at the University of Bologna, Italy.

[31] [http://wiki.virtualsquare.org/wiki/index.php/VDE\\_Basic\\_Networking](http://wiki.virtualsquare.org/wiki/index.php/VDE_Basic_Networking).

[home](#) [start](#) [up](#) [prev](#) [next](#)