Оглавление

[**Про VPN (бесплатные и платные)** 1](#_Toc489259055)

[**Ищем и арендуем VPS за Bitcoin** 6](#_Toc489259056)

[**Первичная настройка VPS** 9](#_Toc489259057)

[**Поднимаем OpenVPN вручную** 15](#_Toc489259058)

[**Установка OpenVPN скриптом** 30](#_Toc489259059)

[**Клиенты OpenVPN под разные ОС.** 33](#_Toc489259060)

[**Антидетект OpenVpn** 35](#_Toc489259061)

# **Про VPN (бесплатные и платные)**

Intro. Зачем нужен VPN

Что такое VPN и для чего он нужен.

VPN расшифровывается как Virtual Private Network (Виртуальная Частная Сеть). Это технология, которая создает виртуальный защищенный канал (или туннель) между вашим устройством (компьютер/смартфон/планшет) и Интернетом.

VPN - это некий сервис, который обычно работает на удаленной машине (сервере).

Зачем нужен VPN

VPN обычно используют разные предприятия и организации, что бы объеденить разные её филиалы или удаленных пользователей, которые находятся физически далеко друг от друга, в одну сеть. Но нам интересно не это.

VPN позволяет выходить в сеть от пользователя с другим IP адресом, возможно, другой страны, что дает возможность обходить разного рода блокировки (на рутрекер сможете зайди или в ваш любимый ВК, если заблокируют и его) или воспользоваться ресурсами, которые доступны только для пользователей определенных стран (допустим существует сайт, который доступен только для пользователей Германии).

Также, если ваш провайдер не предоставляет вам статический IP адрес, VPN может быть использован для создания такого выделенного IP.

Поскольку VPN создает зашифрованный криптостойким алгоритмом туннель, это позволяет послать нахуй сразу несколько категорий потенциально опасных для нас и нашего дела существ - провайдеров, ментов (гос-во), хакеров и конечный ресурс (прим. сайт в интернете).

Для нас VPN является отличным средством анонимизации и защиты нашего траффика от провайдера и самых разных SleepWalker'ов.

Использование VPN, если так можно сказать, почти обязательно для каждого более-менее вменяемого юзера. Причем речь не только об около-криминале, VPN неплохо бы использовать и в вашей повседневной жизни. Что уж говорить об остальных категориях пользователей, например, которое посещают Runion.

VPN будет скрывать от вашего провайдера ресурсы, которые вы посещаете. А от ресурсов, которые вы посещаете, VPN будет скрывать ваш реальный IP адрес.

VPN часто используется как средство для сокрытия Tor-траффика от вашего провайдера, ведь запуская VPN, а затем Tor Браузер, ваш трафик имеет цепь VPN -> Tor, что означает, что факт использования Tor скрыт от провайдера, в ваш настоящий IP адрес не палится перед первой (входной) нодой сети Tor.

Описано в сжатой форме. За остальным курите интернеты.

Популярные VPN протоколы.

Существует множество разных протоколов VPN.

Здесь я постарался сравнить самые популярные и используемые из них.

Начнем с хорошего.

OpenVPN - это твой бро. Это протокол с открытым исходным кодом, использующий библиоеку OpenSSL - поддерживает большое количество шифров, очень безопасный, достаточно гибкий в настройке, может работать на любом порту, быстрый по скорости. Для работы необходимо стороннее ПО (клиенты OpenVPN под разные ОС, об этом ниже). Клиентский файл .ovpn может использоваться на любой ОС, в том числе и мобильных. Везде рекомендую вам именно OpenVPN.

IKEv2 - Тоже очень неплохой протокол с очень высокой степенью защиты, быстрый по скорости. Один из немногих протоколов, поддерживающий устройства Blackberry, но сам по себе поддерживает достаточно мало платформ. Этот протокол встретишь не так часто в VPN сервисаХ. в отличии от OpenVPN или L2TP. Исходный код не открыт, а это потенциально небезопасно.

PPTP - Этот протокол достаточно старый, один из самых первых. Быстрый по скорости и легкий в настройке, поддерживается почти всеми ОС, однако очень уязвимый и совершенно небезопасный. Несколько раз взламывался спецслужбами. Этот протокол использовать крайне не рекомендуется за исключением случаев, где вам не нужна защита. Это юзать не нужно ни в коем случае!

L2TP/IPsec - Уже лучше, чем PPTP. L2TP также поддерживается многими ОС, в том числе и мобильными и тоже достаточно прост в настройке. IPsec обеспечивает более стойкое шифрование, чем PPTP. Однако, L2TP/IPsec медленнее, чем PPTP или OpenVPN. Не обходит Фаерволлы.

SSTP - Протокол от Microsoft. SSTP с хорошим, стойким шифрованием, доступен только для операционных систем Windows. Как и OpenVPN, позволяет обходить Firewall. Проще в настройке, чем OpenVPN. Без открытого исходного кода, поэтому OpenVPN лучше.

Вывод:

Если есть возможность, использовать лучше только OpenVPN из-за открытого исходного кода, безопасности и относительной скорости. IKEv2 хороший протокол, безопасный, быстрый, стабильный, но с закрытым исходным кодом. L2TP/IPsec можно использовать вместо PPTP, а PPTP использовать вовсе не рекомендую. SSTP доступен только для Винды, тоже не очень рекомендую.

Если интересует больше подробностей - гуглите и читайте.

Бесплатные VPN vs Свой VPN vs Покупной VPN.

Я часто вижу вопросы на эту тему, она популярна.

Посему решил разобрать этот вопрос в данной статье.

Итак, что же лучше?

Бесплатные VPN, предоставляемые разными сервисами, свой VPN, поднятый самостоятельно на собственном сервере или платные VPN-сервисы?

Если вам нужен VPN срочно, быстро, здесь и сейчас и у вас нет оформленной подписки на каком-то VPN-сервисе, а своего сервера нет и подавно - тогда можете воспользоваться бесплатными VPN. Но затея, конечно, сомнительная.

Я знаю только два сервиса, предоставляющих бесплатный VPN:

1) vpngate.net

2) vpnbook.com

Все просто. Идем на сайты, скачиваем конфигурационный файл OpenVPN или получаем инструкцию по подключению, если вы используете другой VPN протокол.

Конфигурационный файл OpenVPN (с расширением .ovpn) нужно "скормить" вашему OpenVPN клиенту (о клиентах я написал ниже).

В чем приимущество бесплатных VPN?

Очевидно, что сама по себе бесплатность является плюсом. Также то, что такими серверами пользуются большое количество людей, прибавляет анонимности. На этом его преимущества заканчиваются.

Данные VPN очень нестабильные. Часто отваливаются. Очень медленные. Использовать их я бы рекомендовал, только если нужно срочно для чего-то и нет других возможностей. Но лучше что бы у вас не было таких ситуаций. Не исключено, что на таких серверах ведутся логи. Да и что на этих серверах ЕЩЕ происходит - никому не известно.

Для долгосрочного использования и по важным делам использовать крайне не рекомендую.

Можно воспользоваться собственным VPN.

Нужно лишь купить сервер в интернете, зайти на него и поднять там OpenVPN по инструкции.

По сравнению с платными VPN-сервисами, свой собственный VPN выходит еще дешевле. Арендовать VPS (Virtual Private Server) можно чуть ли не за 2-3$/месяц в интернете. Что, по-моему, достаточно дешево - нужно лишь поискать.

Конечно же, свой VPN имеет и другие приимущества. Например, у вас будет root-доступ к серверу, поэтому вы можете отключить логирование и 100% быть уверенными в том, что логирование не ведется. В отличии от не своего VPN (бесплатные или платные), где у вас нет root-доступа, а это значит, что вы не можете никак быть уверенны в том, что логирование отключено, как бы вас не убеждали в этом эти сервисы. Вы сам себе хозяин на сервере. Это ваш VPN.

В отличии от бесплатных VPN, свой сервер будет работать гараздо быстрее и стабильнее. Это очень весомый плюс.

Также, поскольку у вас будет свой собственный сервер, туда можно прикрутить что-нибудь помимо OpenVPN, какой-нибудь другой сервис. Например, веб-сервер и разместить там небольшой сайт. Или еще какую ерундень поставить. Или можно сделать ее Tor-нодой, если провайдер разрешает. Tor relay разрешают чаще чем Tor Exit Node (выходную ноду).

Однако, существует проблема, что на сервере будете находиться вы и только вы один (если вы, конечно, не дадите использовать свой VPN еще пару сотен человек, но тогда и сервер нужен будет мощнее). Такой проблемы нет у бесплатных или платных VPN.

Еще одним минусом является то, что не каждый способен поднять себе VPN. Кто-то может плохо разбираться в IT и ему эти инструкции будут казаться совершенно дикими и непонятными. А кто-то будет совершенно не понимать что он делает вообще, это тоже не хорошо.

В общем, настоятельно рекомендую использовать.

Можно воспользоваться услугами платных VPN сервисов

которые тоже относительно недорогие (в среднем 5-12$/месяц).

Такие сервисы нужно искать, конечно же, в интернете.

Большенство из них принимают к оплате Bitcoin, что несомненно круто.

Что-то конкретное рекомендовать не буду, но есть ряд VPN-сервисов, которые постоянно мелькают:

AirVPN

IPVanish

PIA

TorGuard

IVPN

NordVPN

Proxy.sh

ExpressVPN

BolehVPN

Все вышеперечисленные сервисы принимают к оплате Bitcoin (на момент написания статьи). Некотрые даже принимают другие криптовалюты.

Я детально не изучал каждый из этих сервисов, но они предлагают самые разные решения. Разные VPN протоколы, под разные десктопные и мобильные ОС. У многих есть свой OpenVPN клиент под какую-то операционную систему, но пользоваться их клиентами не очень рекомендуется из соображений безопасности (об этом ниже).

Все просто. Регистрируетесь и платите за месяц (или год, что выходт дешевле), скачиваете конфигурационный факл OpenVPN (.ovpn), скармливаете его вашему OpenVPN клиенту и всё, у вас будет VPN. Если у вас не OpenVPN, то читаете инструкцию на сайте сервиса по подключению с другим протоколом. Благо на этих сайтах у них хорошие инструкции и все предельной просто и понятно.

Из преимуществ можно выделить то, что каждый сервис предлагает сервера во многих разных странах, которые доступны вам. Между этими странами можно переключаться, вручную с .ovpn конфигами или как-то автоматически, если эта фича реализованна в их клиенте. Тогда как на своем собственном сервере вы привязаны к одной стране, одному IP адресу. Не понравилась одна страна - переключились на другую. Скажу от себя, что фича хоть и полезная, но какая-то сомнительная.

Говорить про то, что эти сервера намного стабильнее, надежнее и быстрее по сравнению с "бесплатным" решением, я думаю не нужно. Доверие к платным сервисам тоже как-то больше, чем к беслпатным VPNам, хоть все равно нет гарантий.

По сранению со своим собственным сервером, у вас нет root-доступа, посему контроллировать процессы, которые происходят внутри вы не можете. Это значит, что никаких гарантий того, что логирование отключено, у вас нет.

Хотелось бы верить, что логирование отключено и платные сервисы не пиздят своим клиентам. Я думаю во многих случаях так и есть, но все зависит от страны наверное. Где-то могут и не принуждать включать логирование, это значит, что сервису ничего не будемт мешать сделать это.

По поводу выбора VPN сервиса, есть сайт, где автор сравнивает большое количество разных VPN сервисов по разным критериям. Можете почитать.

Есть версия с простым сравнением.

Есть версия с более детальным сравнением.

Немного про страны и прочее:

Когда вы выбираете себе VPN сервис или VPS, обращайте внимание на то, где зарегистрирована фирма представляющая VPN, а также на то, в каких странах расположены их сервера. Про выбор VPS тоже самое.

Юридический момент в этом деле важен, но здесь я вам не помощник.

Могу сказать лишь то, что использовать VPN расположенный в стране, где вы совершаете преступление, так себе затея. Так что, если вы работаете против России, заказывать сервера в этой стране или пользоваться услугами Российского VPN сервиса КРАЙНЕ нелогично.

Выбираете сервис или VPS - смотрите где расположены сервера, где зарегистрирован сервис или хостер, читайте политику конфиденциальности.

На последок хотелось бы немного коснуться такой темы, как VPN-расширения в браузерах, которые на сегодняшний день очень популярны. Ну это полноценное говно, которое не нужно использовать ни в коем случаи для более-менее серьезных дел, где требуется анонимность. Для домохозяек может подойти для обхода блокировки какого-нибудь сайта.

# **Ищем и арендуем VPS за Bitcoin**

t.me/forkstgJuly 04, 2017

Для того чтобы поднять собственный OpenVPN, нам неожиданно нужен свой сервер (далее VPS). Нужно купить VPS, зайти на него средствами SSH, произвести первучную настройку серва (необзательно) и поднять OpenVPN

Проблемы при анонимной покупке VPS.

.Многие зарубежные компании-хостеры принимают к оплате Bitcoin. Чего не скажешь о российских сервисах, предлагающих арендовать VPS. Хотя и среди Российских можно найти с оплатой данной криптовалютой.

Однако, даже с учетом того, что "белый" хостер принимает к оплате анонимную криптовалюту, многие из них все равно не любят анонимных клиентов. Поэтому при регистрации требуют ввести такие данные как ФИО, страна, город, адрес проживания, e-mail, мобильный телефон (не всегда). Также при регистрации и последующих заходах в личный кабинет логируется ваш IP адрес. Бывают более лояльные хостеры, бывают менее (в кратце об этом ниже).

Так что достаточно просто и анонимно купить VPS получится далеко не всегда.

Казалось бы, что при регистрации можно вбить левые данные, а проблема логирования IP решается проксями или Tor Браузером. НО НЕТ! Дело в том, что у многих таких "белых" хостеров на сайтах внедрены специальные антифрод системы, которые борются с мошенническими транзакциями. Вообще-то, это защита прежде всего от кардеров, но успешно работает и против черезчур анонимных клиентов, ведь какая нахрен мошенническая транзакция может быть с оплатой Bitcoin?

Антифрод система расчитывает риск мошеннической транзации или "странного" клиента на основе очень многих факторов, которые долго и бесполезно перечислять в рамках данной статьи. Внутреннюю кухню описывать нет смысла, это тема отдельной статьи о кардинге. В кратце расскажу о некоторых базовых факторах, которые стоит учитывать:

1) IP адрес. Он должен быть более-менее чистый и не быть в базе антифрод системы как прокся и прочее. Это важнейший фактор. Как вы понимаете, IP адрес Tor во всех списках, и если у хостера стоит самый хуевый антифрод и есть желание борость с анонимными клиентами, то через Tor зарегистрировать не получтся.

Вы можете использовать хорошие прокси или VPN. В общем, разберетесь.

2) Человек должен соответстовать реальности. Вбивать в ФИО 'ksdjfsjd skdfkk skdgs" примерно такое - не нужно.

3) Адрес должен быть тоже реальным. Не нужно писать улица ленина, в городе залупинс в Афганистане. Ок? Также, если у вас IP российский, значит и адрес тоже должен быть российским.

4) Желательно чтобы адрес соответсоввал Zip Code (индексу, кароче). В смысле, он должен принадлежать ему.

5) Телефон. Должен иметь разный вид в зависимости от страны. Если россия, то +7901... и дальше любые цифры.

Зарубежные компании (это относится, кстати, не только к хостерам), очень редко присылают смс-код на номер телефона для подтверждения регистрации или чего-то еще, в отличии от российских компаний.

В общем, все должно быть максимально правдоподобно, будто заказывает реальный, нормальных, не анонимный клиент. Но даже эти меры в каких-то случаях могут не спасти, поскольку антифрод системы бывают разные и бывают настроены по-разному. Но соблюдение данных правил увеличит вероятность успешной аренды. На моем опыте, их соблюдение помогает во многих случаях.

Ищем VPS с арендой за Bitcoin.

C самого начала встает вопрос - А где купить VPS?

Аренду VPS предлагает много хостеров. Нам не нужны большие мощности, поэтому подойдут самые слабые сервера с технологией виртуализации OpenVZ (но лучше KVM или Xen) с малой процессорной мощностью и небольшим количеством оперативной памяти (256-512 mb). Стоить такие сервера могут в среднем 2-5$ в месяц. А если постараться поискать, то можно найти и за 12-18$ за целый год.

Кстати, если вы нашли VPS, который не принимает к оплате Bitcoin или у вас просто их нет, зато есть Qiwi или Яндекс Деньги, то вы можете попытаться заплатить за сервер Виртуальной Кредитной Картой (VCC - Virtual Credit Card). Эти карты можно выпистить на сайте Qiwi и Яндекс Денег соответственно. При этом, естественно, Киви и ЯД должны быть левыми. Но лучше платите Биткоинами.

Прочитать о том, как купить Bitcoin и о работе с электронными коельками в целом, можете прочитать в моей статье - Статья по Эл. Кошелькам. Там же я частично писал о VCC.

Для поднятия OpenVPN нам нужно чтобы наш VPS поддерживал TUN/TAP.

Обычно почти все VPS провайдеры предоставляют эту фичу. Где-то она сразу включена, как только создается VPS, где-то приходится включать через панель управления на сайте (если OpenVZ), где-то нужно просить техническую поддержку что бы включили вам TUN/TAP (но это редко).

Что бы найти VPS, нужно ввести правильный запрос в гугле.

Запросы лучше делать на англ. языке, чтобы выдавал зарубежных хостеров (если нужно). Запросы могут выглядить следующим образом: "buy vps", "buy cheap vps", "cheapest vps", "cheap openvz vps", "cheap vps <страна>" и подобное.

Также существуют различные сайты, на которых можно найти достаточно дешевые VPS.

Вот пример таких сайтов:

1) LowEndBox - сайт, на котором можете посмотреть дешевых VPS провайдеров. Там их дохрена. Собственно, сайт и посвящен дешевым VPS.

2) LowEndTalk - этот сайт существует как дополнение к предыдущему (или наоборот? О\_о). На нем много пользователей, идет активное обсуждение разных VPS провайдеров. Сайт этот очень полезный. На нем можно найти много мелких и не очень предпринимателей, которые подторговывают VPS. Также много полезной инфы, обзоров различных хостеров и прочее. Иногда сам почитываю.

3) LowEndStock - вот здесь представлены разные дешевые хостеры VPS в таблице. Тоже полезно - пользуйтесь.

  3.1) VPS-List.Cryto - сайт, подобный предыдущему.

  3.2) CompareVPS - еще один подобный сайт с таблицей, по-моему мнению, хуже LowEndStock.

4) PoiskVPS.ru - Российский сайт по подбору VPS. Подбирать можно по разым параметрам, например таким как страна размещения сервера или способ оплаты (можно отфильтровать за биткоин, киви или что вам удобно). Полезный сайт. В основном здесь российские хостеры. Цена на VPS начитается от 70 руб./месяц.

5) crypto.net - а на этом сайте, по ссылке, автор собрал и переодически обновляет список VPS провайдеров, которые принимают к оплате Bitcoin. Раньше это было более актуально, но сейчас очень большое количество провайдеров принимают Bitcoin.

В дополнение:

6) FreeVPS.us - данный сайт может быть тоже интересен. Можно встретить каких-то VPS провайдеров дешевых. Можно найти и бесплатных или с бесплатным пробным периодом - этому сайт и посвящен.

7) WebHostingTalk.comp - в дополнение. Большой форум с обсуждением провайдеров VPS/VDS, и хостингов.

На этих сайтах достаточно много информации по дешевым VPS. Заходите, читайте, находите.

Также оставлю список провайдеров от себя:

Собрал немного ссылок.

BuyVM

HostSailor

Vscale.io (RU)

DeepNetSolutions

Verelox

Briehost

QuadHost

Veesp (RU)

Virmach

TakeWYN (UA)

Host1Plus

GalaxyHostPlus

HostDare

YourServer.se

А теперь что-то странное:

DigitalOcean - Диджитал Оушен. Известный гигант среди провайдеров VPS и любитель агрессивного маркетинга. Что с ним не так? Он не только не принимает Bitcoin напрямую к оплате, а также требует дохуя всего, чего не требуют остальные. Эти пидарасы могут запросто попросить скан паспорта или еще чего-то подобного. Может, скан кредитной карты. Создает дискомфорт.

Vultr - это пидарасы по той же причине, что и DO. Они принимают Биткоин, но только если вы привяжите кредитную карту. Создает дискомфорт.

Это не значит что их не нужно юзать. Просто это мое мнение. Если сильно понравилось - дерзайте.

# **Первичная настройка VPS**

t.me/forkstgJuly 04, 2017

После того, как все данные для подключения были высланы вам на почту или вы нашли их в личном кабинете, то для коннекта к серву вы обладаете всеми данными, а именно:

IP адресс.

Логин: root.

Пароль от root: либо вы указывали при создании сервера, либо был сгенерирован.

Для того чтобы подключиться по SSH на Windows, нужно воспользоваться сторонними программами.

Скачивайте на выбор: Putty или Kitty.

Работать с этими программами достаточно просто. Гуглите.

На ОС Linux и MacOS все проще.

В терминале вводим:

ssh user@ip

где:

ip - IP адрес VPS.

user - username пользователя. При первом подключении / в нашем случае - root.

После нажатия enter, при самом первом коннекте, вам выскочит сообщение:

Are you sure you want to continue connecting (yes/no)?

вручную вводим yes, потом вам предложит ввести пароль от root, который у вас также есть. Если все хорошо, то вы подключились к VPS и вам будет передано управление командной строкой.

Скопировать и вставить в терминале можно так:

Если Linux, то ctrl+shift+v вставить в терминале (ctrl+shift+c скопировать из терминала),

Если Windows и вы используете Putty/Kitty, то вставить в консоль можно по нажатию на ПКМ.

Проверяем TUN/TAP на VPS.

Для начала, елси поднимать мы будем именно OpenVPN, который как я писал выше, требует TUN/TAP, то нужно проверить, включен он или нет.

Для этого в терминале вводим следующую команду:

cat /dev/net/tun

И если вам выдает:

root@ppVPS:~# cat /dev/net/tun cat: /dev/net/tun: File descriptor in bad state

То это значит, что он включен и можно работать дальше.

А если вывод такой:

root@ppVPS:~# cat /dev/net/tun cat: /dev/net/tun: No such file or directory

То выключен. Значит, нам надо его включить. Делается это в панее упарвления VPS либо в личном кабиенетена сайте провайдера. Либо, по запросу в тех. поддержку.

Я захожу в VPS Control Panel (об этом выше под спойлером написано), введя логин и пароль, ищу нужную кнопку, в моем случае это VPS Configuration, затем ставлю галочку на Enable TUN/TAP

После этого сервер перезагрузится.

Подключитесь к нему по-новой в введите:

cat /dev/net/tun

Должно вывести:

cat: /dev/net/tun: File descriptor in bad state

Если так, то все нормально. Если нет, то вы что-то сделали не так.

Далее, перед поднятием непосредственно OpenVPN, было бы неплохо произвести какую-то первичную настройку на сервере: обновить пакеты - поставить нужные пакеты, отключить ненужные сервисы, создать пользователя, настроить sshd\_config (поменять порт, запретить коннет от root).

Поскольку я показываю на примере дистрибутива Debian, вводимые команды будут для этого дистибутива и для дистрибутивов основанных на Debian (Ubuntu).

Итак, приступим.

Обновляем пакеты:

apt-get update && apt-get dist-upgrade -y

Ставим нужные пакеты:

apt-get install -y nano sudo htop curl perl python wget git openvpn openssl easy-rsa iptables ca-certificates ufw

где:

nano - консольный редактор. Обычно стоит изначально, но не всегда.

htop - консольный диспетчер задач.

openvpn - пакет openvpn. Нужно для поднятия OpenVPN.

easy-rsa - скрипты для легкой генерации ключей. Нужно для поднятия OpenVPN.

git - понадобится, если будете поднимать OpenVPN скриптом (далее в статье).

Еще можно поставить следующие пакеты, но не обязательно:

apt-get install build-essential make automake autoconf pkg-config

Далее, нам нужно создать пользователя:

useradd -m -s /bin/bash FORKSCLUB

где:

useradd - команда, создающая пользователя.

-m - создает домашнюю папку пользователя (по пути /home/<имя пользователя>/), по-умолчанию имеет такое же название, как и имя пользователя

-s - указывает, какой shell использовать

FORKSCLUB - имя пользователя.

После создания юзера, вам ОБЯЗАТЕЛЬНО нужно создать ему пароль, потому что мы будем использовать этого юзера для коннекта по ssh, поэтому вводим команду:

passwd forkstg

Предложит дважды ввести пароль. При вводе не будет отображаться. Как вставлять в терминале я писал вышел.

Далее, нужно отключить лишние сервисы.

Вводим команду:

netstat -tulpn

Для того что бы посмотреть, какие сервисы запущены.

На данном этапе в идеале должен быть запущен только sshd. Поскольку у меня установлена Debian-minimal, то у меня так и есть и команда

netstat -tulpn

Выдаёт мне только

Active Internet connections (only servers) Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name tcp 0 0 0.0.0.0:22 0.0.0.0:\* LISTEN 204/sshd tcp6 0 0 :::22 :::\* LISTEN 204/sshd

Но обычно может быть запущен какой-нибудь веб-сервер, mail-server, еще что-то...

Не знаю, что запущено у вас, поэтому по-очереди вводите следующие команды:

/etc/init.d/apache2 stop update-rc.d apache2 disable /etc/init.d/postfix stop update-rc.d postfix disable /etc/init.d/nginx stop update-rc.d nginx disable /etc/init.d/exim4 stop update-rc.d exim4 disable /etc/init.d/rpcbind stop update-rc.d rpcbind disable /etc/init.d/nfs-common stop update-rc.d nfs-common disable /etc/init.d/rsyslog stop update-rc.d rsyslog disable

Это должно отключить то, что обычно запущено уже при создании сервера.

Далее, можно настроить sshd.

Из соображений безопасности поменяем порт со стандартного на нестандартный и запретим коннект от root.

Редактируем файл /etc/ssh/sshd\_config при помощи nano:

nano /etc/ssh/sshd\_config

Как пользоваться редактором nano: Вкратце расскажу основные горячие клавиши при работе с консольным редактором nano.

nano <путь до файла> - открываем файл в nano.

Ctrl+O - сохранить. Записать изменения в файл. Жмем Ctrl+O, потом Enter.

Ctrl+X - выйти из nano.

Ctrl+K - вырезать целую строку. Будет скопировано в буфер обмена.

Ctrl+U - вставить вырезанную строку, которая находится в буфере обмена.

Ctrl+W - поиск по файлу. Поиск в редаткоре, вводим че нужно -> жмем интер.

Разве трудно? Это вы еще Emacs не видели.

Если хотим скопировать часть текста, в самом редакторе выделяем мышкой нужный текст и жмем комбинацию Ctrl+Shift+C. Это скопировать.

Для того того чтобы вставить текст, который сейчас находится в буфере обмена, жмем комбинацию Ctrl+Shift+V.

Не забывайте вводить sudo nano если редактируете файл, к которому у вас нет прав на запись.

Ищем строку Port (будет где-то наверху), и меняем ее значение на другое, отличное от 22, в диапазоне от 1024 до 64000.

Port 1488

Я выбрал число Трепета.

Далее, ищем строку PermitRootLogin и меняем ее значение с yes на no:

PermitRootLogin no

Ищем строку PermitEmptyPasswords и проверяем, чтобы там было no:

PermitEmptyPasswords no

Можно еще, но не обязательно, запретить коннект всем пользователям по SSH кроме того, которого мы создали выше.

Для этого где-нибудь в файле, например внизу, пишем следующее:

AllowUsers FORKSCLUB

После AllowUsers имя юзера.

Внимательно проверяем, запоминаем порт, сохраняем, закрываем.

Важно все сделать нормально, потому что-то если вы где-то накосячите в этом файле, например напишите один порт, а потом его забудите, или запретите логин от рута, а другого пользователя у вас нет, то вы просто не сможете подключиться к серверу.

Обычно данная проблема решается в личном кабинете, там можно законектиться по SSH, далее залогиниться под рутом и изменить в файле все что нужно. В крайнем случае выполните переустановку.

После всех манипуляций в файле /etc/ssh/sshd\_config, перезапускаем:

systemctl restart sshd

или

service sshd restart

Изменения применятся.

Следующий ваш коннект по SSH будет выглядить следующим образом:

ssh -p 1488 user@ip

где:

-p 1488 - порт SSH, который вы указывали в /etc/ssh/sshd\_config

user - пользователь, отличный от root. Под рутом не получится подключиться.

Если у вас винда, то в PUTTY/KITTY тоже меняете порт.

Теперь вы будете коннектиться по SSH на серв через этого юзера. У него нет почти никаких прав, он даже не суперпользователь.

Поэтому, для того чтобы производить какие-то манипуляции на сервера, нужно залогиниться под рутом. Для этого под обычным юзером вводим:

su -

И вводим пароль от root. Все - можете делать что хотите. Чтобы выйти из-под рута, вводите exit.

За статью спасибо пользователю pp

# **Поднимаем OpenVPN вручную**

t.me/forkstgJuly 04, 2017

Итак, собственно, поднимаем OpenVPN-server.

Начнем с установки каких-то пакетов. Некоторые из них вы уже могли ставить, если производили первичную настройку из раздела выше.

Обновим систему:

apt-get update && apt-get dist-upgrade -y

apt-get install -y sudo nano htop curl perl python wget git openssl ca-certificates iptables

И три основных пакета:

apt-get install -y openvpn easy-rsa ufw

После установки пакетов, разархивируем пример конфигурационного файла в папку /etc/openvpn/:

gunzip -c /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz > /etc/openvpn/server.conf

Файл /etc/openvpn/server.conf - основной серверный конфигурационный файл OpenVPN.

Теперь немного отредактируем его:

nano /etc/openvpn/server.conf

Файл длиный, с большим количесвом комментариев (все, что идет в строчке после символа # - комментарий).

Строки, которые начинаются с символа ; в данном файле - тоже комментарии.

Частичный разбор конфига будет ниже в статье.

Редактируем. В самом верху сразу будет port:

port 1194

Я поменяю порт на какой-нибудь совсем не стандартный:

port 16122

(Диапазон от 0 до 65000), я выбрал рандомно. Не займите порт, который слушает другой сервис, например в моем случае sshd случает 1488.

Идем дальше.

Выбираем протокол - пара строк ниже:

proto udp

И видим, что она никак не закоментированная. Потому что по-умолчанию OpenVPN будет работать по UDP.

Если вам нужен TCP, выше есть строчка proto tcp - раскоментируйте ее, удалите символ ; в нчале строки,

а proto udp закоментируйте, добавив символ #. Оба нельзя, или то или другое. Я буду показывать на примере UDP, поэтому оставляю как есть.

Опускаемся ниже в конфиге и находим строку:

dh dh1024.pem

Меняем на dh2048.pem:

dh dh2048.pem

Идем ниже и находим закоментированную строку:

;push "redirect-gateway def1 bypass-dhcp"

Раскоментируем:

push "redirect-gateway def1 bypass-dhcp"

Еще чуть ниже находим две раскоментированные строки рядом:

push "dhcp-option DNS 208.67.222.222" push "dhcp-option DNS 208.67.220.220"

Это то, какой DNS будет использоваться. По-умолчанию прописаны сервера OpenDNS, но можем прописать публичные сервера Google:

push "dhcp-option DNS 8.8.8.8" push "dhcp-option DNS 8.8.4.4"

Можете оставить и OpenDNS.

Идем дальше и натыкаемся на строку

;tls-auth ta.key 0 # This file is secret

Просто раскоментируем:

tls-auth ta.key 0 # This file is secret

Листаем ниже и находим строку:

;cipher AES-128-CBC # AES

Это шифрование. Раскоментируем строку и переделаем на AES-256-CBC:

cipher AES-256-CBC

Еще чуть ниже в файле находим две закоментированные строки:

;user nobody ;group nogroup

Раскоментим их:

user nobody group nogroup

Далее, ниже находим строку, которая начинается со слова status и переделываем ее в следующий вид:

#status /dev/null 2>&1

Закоментированная.

Еще чуть ниже, находим строку, которая начинается со слова log, и придаем ей такой вид:

log /dev/null 2>&1

Не закоментированная. Это логи. Отключить логирование OpenVPN нельзя в привычном понимании, поскольку если закоментировать строку, то по-умолчанию будет писаться в syslog файл (/var/log/syslog).

строку

;log-append

не трогаем

Еще чуть ниже, verb 3 меняем на:

verb 0

Сохраняем, закрываем. Мы еще вернемся к этому файлу позже.

Теперь включаем перенаправление пакетов, вводим команду:

echo 1 > /proc/sys/net/ipv4/ip\_forward

Затем:

cat /proc/sys/net/ipv4/ip\_forward

Если возвращает единицу, значит все ОК.

Теперь надо чтобы изменения были и при перезагрузке сервера, поэтому редактирум файл /etc/sysctl.conf:

nano /etc/sysctl.conf

Ищем строку:

#net.ipv4.ip\_forward=1

Раскоментируем ее:

net.ipv4.ip\_forward=1

Если такой строки в файле нет вообще, добавьте вручную.

Настроим Firewall.

Ранее мы установили пакет ufw.

Теперь нам нужно добавить некоторые правила.

Во-первых, нам нужно дать разрешить траффик на SSH.

Я ввожу:

ufw allow 1488

Поскольку у меня sshd слушает порт 1488,

если вы не меняли и у вас sshd слушает 22 порт, то вводите:

ufw allow ssh

Далее, нам нужно разрешить траффик на порт, который будет слушать OpenVPN, в моем случае это port 16122, потому что я менял его в конфиг. файле (выше):

ufw allow 16122/udp

Поскольку OpenVPN будет слушать UDP порт, я в конце добавил /udp. Если у вас OpenVPN настроен не на UDP, а на TCP, без слеша на конце (или /tcp).

Если вы ничего не меняли в конфиге, то у вас слушает порт 1194 UDP, а значит:

ufw allow 1194/udp

Теперь нужно изменить Forward Policy, редактируем файл /etc/default/ufw:

nano /etc/default/ufw

Ищем строку:

DEFAULT\_FORWARD\_POLICY="DROP"

И меняем ее значение на:

DEFAULT\_FORWARD\_POLICY="ACCEPT"

Теперь нам нужно разобраться с сетевыми интерфейсами, вводим команду:

ip addr

Команда выведет ваши интерфейсы.

Там будет интерфейс lo и еще какой-нибудь, например eth0 или venet0.

В моем случае интерфейс называется venet0 (ибо OpenVZ).

У меня такой вывод:

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00 inet 127.0.0.1/8 scope host lo 2: venet0: <BROADCAST,POINTOPOINT,NOARP,UP,LOWER\_UP> mtu 1500 qdisc noqueue state UNKNOWN link/void inet 127.0.0.2/32 scope host venet0 inet МОЙ IP/32 brd МОЙ IP scope global venet0:0

Вам нужен интерфейс, который показывает ваш внешний IP.

Из кода выше понятно, что venet0 показывает внешний IP. (inet МОЙ IP)

Далее, что нужно редактировать файл /etc/ufw/before.rules:

nano /etc/ufw/before.rules

И куда-нибудь наверх, вставляем следующее:

# START \*nat :POSTROUTING ACCEPT [0:0] -A POSTROUTING -s 10.8.0.0/8 -o <ВАШ ИНТЕРФЕЙС> -j MASQUERADE COMMIT

где вместо <ВАШ ИНТЕРФЕЙС> пишем интерфейс, в моем случае это venet0 и получилось так:

# START \*nat :POSTROUTING ACCEPT [0:0] -A POSTROUTING -s 10.8.0.0/8 -o venet0 -j MASQUERADE COMMIT # END

После всего этого включаем ufw:

ufw enable

Cмотрим на наши правила:

ufw status

Выводом покажет правила, которые вы создавали выше, у меня так:

root@ppVPS:~# ufw status Status: active To Action From -- ------ ---- 1488 ALLOW Anywhere 16122/udp ALLOW Anywhere 1488 ALLOW Anywhere (v6) 16122/udp ALLOW Anywhere (v6)

где 1488 - для SSH и 16122 - UDP опенвпн.

Отключить ufw можно так:

ufw disable

Генерируем ключи для сервера.

Ранее мы ставили пакет easy-rsa.

Теперь копируем папку со скриптами easy-rsa в папку openvpn:

cp -r /usr/share/easy-rsa/ /etc/openvpn

внутри папки easy-rsa создаем папку keys:

mkdir /etc/openvpn/easy-rsa/keys

Теперь сгенерируем Diffie-Hellman длиной 2048 и поместим его в /etc/openvpn:

openssl dhparam -out /etc/openvpn/dh2048.pem 2048

Еще сгенерируем файл ta.key в папку /etc/openvpn, для tls-auth:

openvpn --genkey --secret /etc/openvpn/ta.key

Далее...

Переходим в папку easy-rsa:

cd /etc/openvpn/easy-rsa

Тут лежит файл vars с какими-то переменными, редактируем его:

nano vars

Крутим вниз пока не наткнемся на блок:

export KEY\_COUNTRY="US" export KEY\_PROVINCE="CA" export KEY\_CITY="SanFrancisco" export KEY\_ORG="Fort-Funston" export KEY\_EMAIL="[me@myhost.mydomain](mailto:me@myhost.mydomain)" export KEY\_OU="MyOrganizationalUnit" # X509 Subject Field export KEY\_NAME="EasyRSA"

В первых 6 строках можете менять значения, можете оставить как есть. Я оставлю как есть. Пофиг. Главное нельзя оставлять их пустыми.

Но конкретно здесь нас интересует строка

export KEY\_NAME="EasyRSA"

Здесь значение меняем на server, что бы было вот так:

export KEY\_NAME="server"

Можно поменять на любое другое, но не ебем мозг и меняем как я, потому что иначе вам придется менять значения в server.conf.

Если дали название отличное от server - запомните его.

Вот так. Сохраняем, закрываем файл.

Далее вводим следующее:

source vars

Далее, вводим:

./clean-all && ./build-ca

Будет предлогать вводить какие-то значения, но мы их меняли (или не меняли) в файле vars. Просто везде жмем Enter и все. При этом, дойдя до пункта Name, нажимая Enter будет то имя, которое вы указывали ранее в KEY\_NAME у меня это server.

Следующим делом вводим:

./build-key-server <ИМЯ В KEY\_NAME>

где:

<ИМЯ В KEY\_NAME> - это то имя, которое вы давали в файле vars. Если ничего не давали, то по-умолчанию там было Easy-RSA. Я давал имя server, поэтому:

./build-key-server server

Точно таким же образом, везде жмем Enter. Даже там, где предложит ввести 'A challenge password', жмем enter.

Потом дважды вводим символ y, где попросит:

Sign the certificate? [y/n]:y 1 out of 1 certificate requests certified, commit? [y/n]y

В конце вы увидите:

Write out database with 1 new entries Data Base Updated

Теперь нам нужно переместить все сгенерированные ключи и сертификаты в папку /etc/openvpn.

Всё добро находится в папке /etc/oepnvpn/easy-rsa/keys. Нас интересуют 3 файла (исплючая dh2048.pem и ta.key, которые мы сгенерировали ранее) - ca.crt, server.crt, server.key,

где файлы server.crt и server.key - имеют название, которое вы указывали в KEY\_NAME, я указывал server, помним да.

Перемещаем их следующим образом:

cp /etc/openvpn/easy-rsa/keys/{ca.crt,server.crt,server.key} /etc/openvpn/

Идем обратно в папку openvpn:

cd /etc/openvpn

Вводим к консоли ls и смотрим, какие файлы лежат. На данный момент должны быть следующие файлы: ca.crt, dh2048.pem (который мы генерировали ранее), server.conf (основной конфиг), server.crt, server.key и ta.key Еще может быть файл update-resolv-conf.

root@ppVPS:/etc/openvpn# ls ca.crt dh2048.pem easy-rsa server.conf server.crt server.key update-resolv-conf

Теперь запускаем openvpn:

systemctl start openvpn

или

/etc/init.d/openvpn start

Проверяем, запустился ли:

systemctl status openvpn

или

/etc/init.d/openvpn status

Должно гореть зелененьким (лол), вы увидите надпись:

Active: active (exited) since

Проверяем, слушает ли порт:

netstat -tulpn | grep vpn

Должно вывести один сервис, который слушает на порту, который вы указывали.

У меня вывод такой:

udp 0 0 0.0.0.0:16122 0.0.0.0:\* 484/openvpn

Все проверили, молодцы. Теперь остановим:

systemctl stop openvpn

или

/etc/init.d/openvpn stop

Делаем клиента, генерируем ключи.

Для начала нам нужно скопировать пример конфиг. файла для клиента.

Скопируем его в папочку /etc/openvpn/easy-rsa/keys и переименуем (даем расширение .ovpn):

cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf /etc/openvpn/easy-rsa/keys/client.ovpn

Редактируем файл клиента:

nano /etc/openvpn/easy-rsa/keys/client.ovpn

Листаем файл вниз, натыкаемся на строку

remote my-server-1 1194

где:

my-server-1 - IP адрес вашего сервера. Тот, что пришел вам на почту или смотрите по команде ip addr | grep inet и ищите.

1194 - Порт, на котором слушает OpenVPN на вашем сервере, в моем случа это 16122.

Меняйте эту строку, я меняю:

remote 111.222.111.222 16122

Листаем ниже, находим строку:

ns-cert-type server

стираем ее и меняем на:

remote-cert-tls server

Еще чуть ниже строка:

;cipher x

Меняем шифрование на то, которое вы указывали в server.conf. Я указывал AES-256-CBC:

cipher AES-256-CBC

Если вы в server.conf ничего не указывали, то эту строку не трогайте.

Все. Сохраняем и закрываем.

Теперь нужно сгенерировать ключи и серты для клиента.

Перейдем в папку easy-rsa:

cd /etc/openvpn/easy-rsa

Генерим:

./build-key client1

где:

client1 - название клиента. Может быть любым. Я сделал client1.

Таким же образом, как и с сервом, везде жмем Enter. И соглашаемся, нажимая y, там где это нужно.

Клиентский конфиг состоит из:

.ovpn файла. В моем случае это client.ovpn.

файла ca.crt - который общий для сервера и все клиентов.

файл ta.key - тоже общий для сервера и клиента. Это для tls-auth, что не обязательно...

файлов client1.crt и client1.key - которые индивидуальные для каждого клиента.

Дело в том, что все эти файлы нужны .ovpn конфигу, без них он не будет работать. Но не будем же мы их везде вместе таскать! Поэтому есть возможность все эти файлы строить в .ovpn конфиг, чтобы был один единый файл и все.

Нужно проделать кое-какие манипуляции, поэтому для удобства создаем папку client1:

mkdir /etc/openvpn/easy-rsa/keys/client1

Нужно скопировать туда конфиг.файл .ovpn и все серты и ключи,

по аналогии с сервером, файлы client1.crt и client1.key имеют названия специфические, в зависимости от того, какое вы указывали имя клиента, выполняя команду выше. Я назвал client1 поэтому у меня они называются именно так.

Копируем:

cp /etc/openvpn/easy-rsa/keys/{ca.crt,client1.crt,client1.key} /etc/openvpn/easy-rsa/keys/client1

За одно копируем client.ovpn (называться может как угодно, хоть xuy.ovpn):

cp /etc/openvpn/easy-rsa/keys/client.ovpn /etc/openvpn/easy-rsa/keys/client1

И не забываем про ta.key, которые лежит в папке openvpn:

cp /etc/openvpn/ta.key /etc/openvpn/easy-rsa/keys/client1/

Переходим в папку client1:

cd /etc/openvpn/easy-rsa/keys/client1

На данный момент там находится 5 файлов:

ca.crt  client.ovpn (конфиг файл) client1.crt client1.key ta.key

Теперь нужно быть внимательнее! В особенности с именами файлов.

Внедряем содержимое нужных файлов в наш конфиг .ovpn.

ПООЧЕРЕДИ, в том порядке, в котором это у меня. выполняем следующие команды.

Для ca.crt:

echo '<ca>' >> client.ovpn cat ca.crt >> client.ovpn echo '</ca>' >> client.ovpn

где client.ovpn название конфиг файла.

Теперь для client1.crt:

echo '<cert>' >> client.ovpn cat client1.crt >> client.ovpn echo '</cert>' >> client.ovpn

где client1.crt - серт. файл, которые генерировали выше.

Теперь для client1.key:

echo '<key>' >> client.ovpn cat client1.key >> client.ovpn echo '</key>' >> client.ovpn

Теперь для ta.key, нужно для tls-auth:

echo 'key-direction 1' >> client.ovpn echo '<tls-auth>' >> client.ovpn cat ta.key >> client.ovpn echo '</tls-auth>' >> client.ovpn

Таким образом, все ключи и сертификаты теперь в одном файле .ovpn, и он может работать сам по себе.

Теперь нужно еще немного отредактировать файл client.ovpn:

nano client.ovpn

Листаем, пока не наткнемся на блок:

ca ca.crt cert client.crt key client.key

Нам это не нужно, поэтому закоментируем их:

#ca ca.crt #cert client.crt #key client.key

Собственно, конфиг клиента готов. Он будет работать, если вы запустите его на OpenVPN клиенте вашего устройства.

Можем запускать OpenVPN:

systemctl start openvpn

или

/etc/init.d/openvpn start

ПОСЛЕ ЛЮБЫХ манупуляций в файле server.conf, для того что бы применились изменения, нужно перезапускать OpenVPN:

systemctl restart openvpn

или

/etc/init.d/openvpn restart

Но я хочу еще проделать какие-то модификации на server.conf и client.ovpn

Для начала я хотел бы привести оба конфига в более читабельный и простой вид, а именно, удалить все комментарии и пустые строки.

Сделать это легко.

Но для начала, на всякий случай, сделаем backup:

server.conf

cp /etc/openvpn/server.conf /etc/openvpn/server.conf.bak

и client.ovpn:

cp /etc/openvpn/easy-rsa/keys/client1/client.ovpn /etc/openvpn/easy-rsa/keys/client1/client.ovpn.bak

теперь одной командой удаляем все комментарии (# и ; в начале строки) и пустые строки:

server.conf

sed -i '/^[#;]\|^$/ d' /etc/openvpn/server.conf

client.ovpn

sed -i '/^[#;]\|^$/ d' /etc/openvpn/easy-rsa/keys/client1/client.ovpn

И оба наших конфига примут более читабельный вид.

Открываем server.conf

nano /etc/openvpn/server.conf

И куда-нибудь в середину прихерачим следующее:

sndbuf 0 rcvbuf 0 topology subnet auth SHA512

И на клиенте.

nano /etc/openvpn/easy-rsa/keys/client1/client.ovpn

Куда-нибудь в конфиг, до начала сертификатов и ключей:

sndbuf 0 rcvbuf 0 keepalive 10 120 auth SHA512

Таким образом, мой конфиг server.conf выглядит так:

port 16122 proto udp dev tun sndbuf 0 rcvbuf 0 ca ca.crt cert server.crt key server.key # This file should be kept secret dh dh2048.pem tls-auth ta.key 0 topology subnet server 10.8.0.0 255.255.255.0 ifconfig-pool-persist ipp.txt push "redirect-gateway def1 bypass-dhcp" push "dhcp-option DNS 8.8.8.8" push "dhcp-option DNS 8.8.4.4" keepalive 10 120 cipher AES-256-CBC auth SHA512 comp-lzo persist-key persist-tun log /dev/null 2>&1 verb 0

А client.ovpn так (БЕЗ ключей и сертификатов):

client dev tun proto udp sndbuf 0 rcvbuf 0 remote 111.222.111.222 14500 resolv-retry infinite nobind persist-key persist-tun remote-cert-tls server cipher AES-256-CBC auth SHA512 keepalive 10 120 comp-lzo verb 3

Можете использовать эти конфиги.

Какие конфиги использовать?

proto udp

Какой протокол юзать, TCP или UDP?

UDP обычно работает быстрее.

TCP более надежный, стабильный, лучше обходит фаерволл - например можно повестить на порт 443.

UDP, также можно повесить на порт, который обычно не блокируют - 53.

Если вас не блокируют, то лучше я думаю UDP. Сам юзаю UDP.

На сервере и клиенте должен быть указан один протокол.

Port 16122

На какой порт повесить? Лучше на нестандартный, как у меня. Или замаскировать под какой-то сервис - 443/ssl для tcp и 53/dns для udp

cipher AES-256-CBC

Какое шифрование использовать?

Используйте AES-256-CBC или AES-128-CBC. Если явно не задать, то по-умолчанию будет исопльзоваться Blowfish (cipher BF-CBC).

Этот параметр должен быть и на сервере и на клиенте. Если на сервере есть, на клиенте нет или они отличаются, может не законнектить.

auth SHA512

Если не задавать, будет использоваться auth SHA1. Используйте auth SHA512 или auth SHA256.

Этот параметр должен быть и на сервере и на клиенте. Если на сервере есть, на клиенте нет или они отличаются, может не законнектить.

#status /dev/null 2>&1

По умолчанию будет записываться в файл какой-то, какие клиенты сейчас онлайн. Чтобы ничего не писало, нужно закомментировать строку. Я также ЗАЧЕМ то написал /dev/null 2>&1 ... (??)

log /dev/null 2>&1

Это то самое логирование OpenVPN которое для безопасности лучше выключить.

Дело в том, что выключить его нельзя, а если закоментировать, то логи будут писаться в /var/log/syslog

Поэтому я раскоментирова и сказал писать в /dev/null.

push "dhcp-option DNS 8.8.8.8" push "dhcp-option DNS 8.8.4.4"

Какие DNS юзать?

Я хз, юзайте паблик DNS от Google (8.8.8.8 и 8.8.4.4.) или от OpenDNS (208.67.222.222 и 208.67.220.220) .

Боремся с утечкой DNS

Существует утечка DNS.

Поэтому, если вы будете юзать этот конфиг на Windows, в конфиг добавляем следующую строку:

setenv opt block-outside-dns

Да, вот так просто.

Под MacOS с Tunnelblick все норм.

А под Linux

Есть файл update-resolv-conf, который находится в папке /etc/openvpn на клиентской машине.

Файл может иметь название или update-resolv-conf или update-resolv-conf.sh.

Поэтому на клиентском Линукс вводим в терминале:

ls /etc/openvpn | grep update

И узнаем название файла.

Далее, в конфиг .ovpn нужно вставить следующие строки:

script-security 2 up /etc/openvpn/update-resolv-conf down /etc/openvpn/update-resolv-conf

где update-resolv-conf - название файла, которое вы получили выше.

Перекидываем конфиг .ovpn с серва на клиент

Если вы создавали пользователя и запрещали ssh коннект от рута, то первым делом нужно поместить .ovpn конфиг в домашнюю папку созданного пользователя. У меня это был FORKSCLUB

cp /etc/openvpn/easy-rsa/keys/client1/client.ovpn /home/FORKSCLUB

Изменить права у файла:

chown FORKSCLUB:FORKSCLUB /home/FORKSCLUB/client.ovpn

Linux.

Будем юзать sftp.

sftp -P 1488 FORKSCLUB@ip

где 1488 - порт на котором слушает sshd.

как законнектились, мы уже будем в домашней папке FORKSCLUB. Можем посмотреть файлы - вводим ls.

копируем конфиг:

get client.ovpn ~/

Windows, linux, MacOS:

Скачиваем программу для работы по FTP - FileZilla (гуглите)

И подключаемся к серву по sftp. Нужено будет указать port sftp (у меня 1488), пользователя (FORKSCLUB, ибо руту зарпетили).

Копируем файл куда хотим и радуемся. Как пользоваться программой разберетесь сами.

# **Установка OpenVPN скриптом**

t.me/forkstgJuly 04, 2017

Поднять OpenVPN вручную вам не под силу? Не отчаивайтесь, у меня для вас хорошие новости!

О Чудо!

На Github существует Open Source скрипт, который позволяет быстро и легко поднять OpenVPN сервер на вашем VPS.

Скрипт называтся openvpn-install, и вот ссылка на Github: nyr/openvpn-install

И этого скрипта есть куча форков, которые вы можете изучить.

Посмотреть исходный код можно здесь.

Приступим.

Работаем под пользователем root или суперпользователем.

Для начала, обновим систему:

apt-get update && apt-get dist-upgrade -y

Если вы уже пытались поднять OpenVPN вручную, но у вас ничего не получилось, то нужно сначала его полностью снести.

Удаляем openvpn:

apt-get remove --purge openvpn

удаляем папку:

rm -rf /etc/openvpn/

выключаем ufw:

ufw disable

и установим нужные пакеты:

apt-get install -y sudo nano curl perl python wget git iptables openvpn openssl ca-certificates

Скачиваем скрипт в домашнюю папку root:

git clone <https://github.com/Nyr/openvpn-install.git> ~/nyr-openvpn

Переходим туда:

cd ~/nyr-openvpn/

Запускаем скрипт:

bash openvpn-install.sh

И вы попадете в так называемый Инсталятор. Где нужно будет отвечать на вопросы или что-то вписывать. Для перехода на следующий этап установки жмите Enter.

Первый делом, предложит ввести внешний IP адрес, но вероятнее всего скрипт сам его определит:

IP address: 111.222.111.222

Следующий этап, выбрать протокол. Я жму 1 и выбираю UDP.

Далее, выбираете порт. Я поставлю, например 14000.

После этого предложит выбрать DNS сервера. Выбор будет из Google, OpenDNS или текущих, которые на VPS и еще каких-то. Я выбираю OpenDNS, жму 3.

После этого предложит ввести имя клиента. Я введу " FORKSCLUB".

Далее начнется скачивание-утановка пакетов, генерация ключей, сертификатов и конфигурационных файлов.

когда все закончится, он поместит конфиг. в домашнюю папку юзера, от имени которого запсукался скрипт.

То есть, сейчас конфиг лежит по адресу:

/root/

Собственно, на этом конфигурация OpenVPN скриптом завершается.

Можно уже сейчас взять новый конфиг и использовать.

Просто, да?

Интересно, чего он там такого сгенерировал вам, посмотрим на конфиг pp-runion.ovpn:

nano ~/FORKSCLUB.ovpn

>>: (без ключей и сертификатов)

client dev tun proto udp sndbuf 0 rcvbuf 0 remote 111.222.111.222 14000 resolv-retry infinite nobind persist-key persist-tun remote-cert-tls server cipher AES-256-CBC comp-lzo setenv opt block-outside-dns key-direction 1 verb 3 <ca>

Все вроде бы нормально, да лучше auth SHA512 явно указать.

Поэтому добавим

auth SHA512

Вот так:

client dev tun proto udp sndbuf 0 rcvbuf 0 remote 111.222.111.222 14000 resolv-retry infinite nobind persist-key persist-tun remote-cert-tls server cipher AES-256-CBC auth SHA512 comp-lzo setenv opt block-outside-dns key-direction 1 verb 3

Обратите внимание, что скрипт уже запихнул фичу для предотвращения утечки DNS под клиент Windows:

setenv opt block-outside-dns

Если у вас Linux или MacOS - эту строку лучше закомментировать.

Теперь посмотрим на серверный конфиг server.conf:

nano /etc/openvpn/server.conf

>>:

port 14000 proto udp dev tun sndbuf 0 rcvbuf 0 ca ca.crt cert server.crt key server.key dh dh.pem tls-auth ta.key 0 topology subnet server 10.8.0.0 255.255.255.0 ifconfig-pool-persist ipp.txt push "redirect-gateway def1 bypass-dhcp" push "dhcp-option DNS 208.67.222.222" push "dhcp-option DNS 208.67.220.220" keepalive 10 120 cipher AES-256-CBC comp-lzo user nobody group nogroup persist-key persist-tun status openvpn-status.log verb 3 crl-verify crl.pem

Здесь также добавляем auth SHA512 (а то работать не будет, если на клиенте указан а на серве нет), делаем verb 0.

Также нужно проделать следующее.

Нужно закомментировать строку

status openvpn-status.log

И добавить log:

log /dev/null 2>&1

Где-то выше писал, зачем он нужен.

Получилось так:

port 14000 proto udp dev tun sndbuf 0 rcvbuf 0 ca ca.crt cert server.crt key server.key dh dh.pem tls-auth ta.key 0 topology subnet server 10.8.0.0 255.255.255.0 ifconfig-pool-persist ipp.txt push "redirect-gateway def1 bypass-dhcp" push "dhcp-option DNS 208.67.222.222" push "dhcp-option DNS 208.67.220.220" keepalive 10 120 cipher AES-256-CBC auth SHA512 comp-lzo user nobody group nogroup persist-key persist-tun #status openvpn-status.log log /dev/null 2>&1 verb 0 crl-verify crl.pem

Берем клиентский файл-конфиг и помещаем на ваше устройство. Об этом я писал выше.

При повторном запуске скрипта из-под рута:

bash /root/nyr-openvpn/openvpn-install.sh

Скрипт предложит создать нового клиент, удалить существующего (не сможет больше коннектиться) или удалить OpenVPN.

# **Клиенты OpenVPN под разные ОС.**

t.me/forkstgJuly 04, 2017

Поскольку OpenVPN не встроен по-умолчанию ни в одну популярную ОС, то требуется установить сторонее ПО - клиент OpenVPN.

Windows.

Под Windows есть официальный клиент, называется OpenVPN GUI.

Скачать его можно на офф. сайте OpenVPN

Скачиваете, устанавливаете. Появится ярлык на раб. столе, после запуска - иконка в трее.

Далее, помещаете ваш .ovpn файл в папку config, в установленной папке OpenVPN.

В общем, если устанавливать OpenVPN GUI по-умолчанию в папку Program Files, то адрес такой:

C:\Program Files\OpenVPN\config

В эту папку кидаем конфиги .ovpn.

Потом из трея можно запускать VPN. Все достаточно просто.

Linux.

На Линукс в репозиориях вашего дистрибутива пакет скорее всего будет называется "openvpn" (еще может openvpn-client), и ставится следующим образом:

sudo apt-get install openvpn

Если у вас Debian-подобные дистрибутивы (Debian, Ubuntu, Mint).

Если у вас дистрибутив другой линейки - сами разберетесь как ставить.

Работает это следующим образом. У вас есть .ovpn файл, далее вы в терминале вводите команду:

sudo openvpn --config <путь до файла .ovpn>

где <путь до файла .ovpn>, собственно, путь до файла .ovpn.

Или можете скормить файл network-manager ' у.

MacOS.

Под MacOS из клиентов OpenVPN я знаю только Tunnelblick.

Скачать его можно здесь.

Устанавливается просто, требует права суперпользователя.

По-умолчанию будет отпрыать файлы с расширением .ovpn.

Будет иконка в трее, там конектимся.

Можно настроить, что бы коннектился к OpenVPN сразу при загрузке системы.

Когда соеденение обрывается с VPN, интернет пропадает. В этом случае нужно вручную отсоеденяться от VPN.

Android.

Официальный клиент OpenVPN Connect под Android доступен для скачивания на Google Play

iOS.

Для iOS клиент скачиваем из AppStore.

Клиенты, предоставляемые провайдером VPN.

Некоторые провайдеры предоставляеют свой собственный OpenVPN клиент под разные ОС и платформы.

Использовать их или нет, решать вам. Но это потенциально не очень безопасно.

# **Антидетект OpenVpn**

t.me/forkstgJuly 04, 2017

VPN может детектироваться конечным ресурсом (например сайт) какими-то хитро-выебанными способами.

Поэтому, если вам НЕ ПОХУЙ (а мне похуй, например), то читайте дальше.

Для начала можете читануть статью на хабре: Определяем пользователей VPN

Заходим на browserleaks.com/ip

И смотрим строку "TCP/IP OS Fingerprinting"

У меня определился OpenVPN.

ЧТО ДЕЛАТЬ?

Если у вас, как у меня, конфиг UDP, то можно проделать следующее.

Мне помогает, если прописать

mssfix 0

На клиенте .ovpn и сервере server.conf