Я давно и успешно использую Openvpn для объединения офисов, подключения удаленных сотрудников, да просто для своих личных нужд. Считаю его удобным и функциональным решением задачи объединения сетей. И всегда у меня вставал вопрос на тему, как бы упростить управление сертификатами. Раньше я заходил в консоль, генерировал сертификат, забирал его с сервера тем или иным способом, создавал файл конфига, передавал пользователю. В какой-то момент я решил, что мне это надоело и начал поиск более удобного решения этого вопроса.

Содержание:

К моему удивлению, никаких готовых бесплатных решений на эту тему нет. Мне пришлось долго разбираться, прежде чем нашел более ли менее приемлемый вариант. В конечном итоге я остановился на такой связке - сервер **freebsd 10**, установленный **webmin** и модуль **OpenVPN-admin**. Модуль не особо популярный, сайт разработчика на непонятном языке, документации толком нет. Но я разобрался и настроил. В итоге получил возможность через web-интерфейс:

1. Редактировать настройки сервера.
2. Генерировать сертификаты пользователей.
3. Задавать пользовательские настройки.
4. Выгружать в едином архиве все необходимые сертификаты с уже готовым файлом настроек пользователя.

Приступим к настройке.

Установка openvpn

Итак, у нас имеется:

```
FreeBSD websrv.local 10.1-RELEASE FreeBSD 10.1-RELEASE #0 r274401
```

Перво-наперво обновляем порты:

```
# portsnap fetch update
```

И устанавливаем openvpn:

```
# cd /usr/ports/security/openvpn
 # make install clean
```

Добавляем openvpn в автозагрузку:

```
# echo 'openvpn_enable="YES"' >> /etc/rc.conf
```

Больше пока ничего не делаем.

Установка и настройка webmin

Ставим webmin:

```
# cd /usr/ports/sysutils/webmin
 # make install clean
```

Добавляем webmin в автозагрузку:

```
# echo 'webmin_enable="YES"' >> /etc/rc.conf
```

После установки запускаем настройку. Я сразу же меняю стандартный порт webmin на что-то экзотическое. Это пусть и не сильно, но повышает безопасность сервера.

```
# /usr/local/lib/webmin/setup.sh
```

Я все настройки оставляю по-умолчанию, только, как уже сказал, меняю порт, например на 11111 и в конце на вопрос:

```
Use SSL (y/n): y
```

отвечаю положительно.

После настройки, запускаем webmin:

```
# /usr/local/etc/rc.d/webmin start
```

Чтобы зайти в панель управления, вводим в браузере адрес:

```
https://ip-adress:11111/
```

Настройка Webmin OpenVPN admin

Теперь скачиваем модуль для webmin

http://www.openit.it/index.php/en/downloads?task=view.download&cid=27

На всякий случай скопирую себе, если ссылка умрет:

//serveradmin.ru/files/openvpn-2.6.wbm.gz

Добавим модуль в webmin. Идем в раздел Webmin -> Webmin Configuration -> Webmin Modules, выбираем From uploaded file, указываем скачанный файл и жмем install module

Получаем сообщение о том, что все в порядке:

Сам модуль располагается в разделе Servers -> OpenVPN + CA. Отправляемся туда. Первым делом вы увидите сообщение:

OpenVPN executable not found

The OpenVPN package can be automatically installed by Webmin. Click here to have it downloaded and installed using .Or reconfigure paths using Module Configuration.

Модуль не обнаружил установленный openvpn. Поможем ему это сделать. Создадим папку:

```
# mkdir /usr/local/etc/openvpn
```

Снова отправляемся в управление модулем. Нас встречает уже другое сообщение:

openssl.cnf batch file not found

The OpenSSL package can be automatically installed by Webmin. Click here to have it downloaded and installed using .Or reconfigure paths using Module Configuration.

Соглашаемся с предложением установить OpenSSL. Жмем **Click here** и ждем окончания установки.

Теперь все готово. Можно генерировать корневой сертификат сервера. Идем в Servers -> OpenVPN + CA, заполняем поля формы и жмем Save:

Мы сформировали корневые сертификаты сервера. Проверить их можно в разделе модуля Certification Authority List. Сверху в списке будет единственный наш сервер.

Теперь создадим сервер openvpn. Идем в раздел **VPN List** и нажимаем **New Vpn server**. Получаем ошибку:

```
No server keys configured
```

Все верно, прежде чем создать рабочий сервер, необходимо сделать для него серверный ключ. Снова идем в раздел Certification Authority List, в строке с нашим сервером, справа, в самом конце нажимаем на ссылку Keys list. Открывается интерфейс создания ключей. Создадим сразу ключ сервера и первого клиента.

Для сервера выбираем в выпадающем списке Server, остальные поля оставляем как есть. Если задать пароль, то при подключении клиента будет выводиться запрос на ввод пароля. Если это не нужно, то поле пароль оставляем пустым:

Создали два сертификата. Теперь идем создавать сервер. В разделе **VPN List** нажимаем **New Vpn server**. Тут нам открывается длинная форма для ввода настроек сервера. Я не буду подробно останавливаться на настройках. Это стандартные параметры файла конфигураций openvpn, в интернете много описаний на эту тему. Я приведу скриншоты своих настроек. Обращаю лишь внимание на пункт **Encrypt packets with cipher algorithm (option cipher)** Я там выбрал **AES-128-CBC 128 bit default key (fixed)** Это не случайный выбор. С другими значениями у меня были ошибки при подключении клиентов, сейчас я уже не помню какие, но методом проб и ошибок я пришел к этому значению. С ним все нормально работает.

Сохраняем настройки. В списке серверов появился наш сервер:

Дальше добавим клиента. Нажимаем в строке нашего сервера на ссылку **Client list** и жмем кнопку **New Vpn client**. Выбираем единственный созданный нами сертификат client1 и заполняем форму с настройками клиента. Там все можно оставить по-умолчанию, кроме одного параметра **remote (Remote IP)** - тут указываем внешний адрес нашего сервера. В разделе **ccd file content** указываются дополнительные параметры клиента. Содержимое этого поля будет перенесено в созданный файл настроек пользователя. Сохраняем пользователя. Появился один пользователь в списке:

Если нажать на ссылку Export, скачается архив со всеми необходимыми сертификатами и файлом конфигурации пользователя. Это очень удобно. Этот архив можно в готовом виде передавать клиенту. В нем будет все, что необходимо для подключения. Сохраним его для тестового подключения, к которому у нас все готово. Идем в начальную страницу модуля и жмем кнопку Start VPN. В ответ получаю ошибку:

```
Command Execution Error /usr/local/etc/rc.d/openvpn start.
```

Чтобы разобраться в чем проблема, смотрим лог messages. Можно через консоль, можно тут же в webmin. Видим там строки:

```
websrv root: /usr/local/etc/rc.d/openvpn: WARNING: /usr/local/etc/openvpn/openvpn.conf is not readable.
websrv root: /usr/local/etc/rc.d/openvpn: WARNING: failed precmd routine for openvpn
```

Все ясно. Сервер не стартовал, потому что ищет настройки в файле /usr/local/etc/openvpn/openvpn.conf, а у нас его нет. Вместо него модуль openvpn admin создал файл websrv.local.conf Я не стал разбираться в чем тут дело. Это актуально, если у нас несколько серверов и нужно запускать отдельно каждый из них. А так как у нас он всего один и нам этого достаточно, то я решил просто явно указать через **rc.conf** файл конфигурации. Делается это так:

```
# echo 'openvpn_configfile="/usr/local/etc/openvpn/websrv.local.conf"' >> /etc/rc.conf
```

Пробуем снова запустить сервер через openvpn admin. Сервер успешно стартовал. Поднялся интерфейс **tun0** с заданным адресом:

```
tun0: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> metric 0 mtu 1500
options=80000<LINKSTATE>
inet6 fe80::20c:29ff:fe19:9976%tun0 prefixlen 64 scopeid 0x3
inet 10.0.0.1 --> 10.0.0.2 netmask 0xffffffff
nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
Opened by PID 66606
```

В каталоге /usr/local/etc/openvpn/websrv.local созданы 3 папки: bin, ssd, logs. В ssd располагается файл настроек нашего пользователя client1. Его можно править как непосредственно тут, так и через webmin. В папке с логами лежат логи и файл ipp.txt, в котором сохраняются ip адреса, выданные клиентам, с тем, чтобы в случае повторного подключения выдать тот же адрес.

Теперь берем архив с настройками клиента и подключаемся к серверу:

Все в порядке. Через webmin можно отзывать сертификаты и удалять клиентов. Функционал небольшой, но, в принципе, достаточный в повседневной жизни. Теперь не обязательно заходить в консоль и готовить сертификаты с настройками для клиента. Все это можно сделать через web сайт и сразу же

отдать файлы пользователю.

Помогла статья? Подписывайся на telegram канал автора

Анонсы всех статей, плюс много другой полезной и интересной информации, которая не попадает на сайт.

Дополнительные материалы по Freebsd

Рекомендую полезные материалы по Freebsd:

• Установка
• Настройка
• Обновление
• Шлюз
• Прокси сервер
• Веб сервер NGINX
• Веб сервер Apache

Описание установки Freebsd 11 на одиночный диск, либо на софтовый raid1, сделанный средствами zfs, которые поддерживает стандартный установщик.
Базовая настройка Freebsd, которую можно выполнить после установки сервера общего назначения. Представлены некоторые рекомендации по повышению удобства пользования и безопасности.
Описание и нюансы обновления системы Freebsd с помощью утилиты freebsd-update. Показано пошагово на конкретном примере обновления.
Настройка Freebsd шлюза для обеспечения выхода в интернет. Используется ipfw и ядерный нат, dnsmasq в качестве dhcp и dns сервера. Мониторинг сетевой активности с помощью iftop.
Подробная настройка на Freebsd прокси сервера squid + sams2 - панели управления для удобного администрирования.
Настройка максимально быстрого web сервера на базе Freebsd и nginx + php-fpm. Существенный прирост производительности по сравнению с классическим apache.
Настройка web сервера на Freebsd в связке с apache, nginx, php и mysql. Пошаговая установка и настройка каждого компонента.