MPD 5.7 настройка VPN-сервера в FreeBSD

Настраиваем MPD5 в качестве VPN-сервера для подключений Windows-клиентов к офисной сети по протоколу PPTP.

MPD5 также может использоваться в качестве VPN-клиента для доступа к удаленной сети, или подключения к интернет провайдеру.

Для объединения нескольких сетей через интернет обратите внимание на OpenVPN. Ключевые отличия OpenVPN: туннель по UDP или TCP протоколу, TLS-шифрование, возможность задать клиенту маршруты к обслуживаемым сетям.

Содержание

Установка MPD5

Устанавливаем пакет:

```
pkg install mpd5
```

Для установки из коллекции портов, обновляем коллекцию:

```
portsnap fetch && portsnap update || portsnap extract
```

Затем, устанавливаем порт:

```
cd /usr/ports/net/mpd5
make install clean
```

Настройка MPD5

Переходим в папку с файлами конфигурации:

```
cd /usr/local/etc/mpd5
```

В папке имеются следующие файлы:

- mpd.conf.sample - шаблон файла конфигурации,
- mpd.script.sample - скрипты для модемов,
- mpd.secret.sample - пример списка пользователей и паролей.

Копируем файл конфигурации из шаблона:

```
cp mpd.conf.sample mpd.conf
```

Открываем в редакторе:

```
ee mpd.conf
```

Идентификаторы секций в файле конфигурации задаются с начала строки и заканчиваются двоеточием. Значения параметров в начале строки обязательно отделяются табуляцией, пробелы также допустимы.

Удаляем все секции кроме startup, default и pptp_server. Для удаления строк в ее используем Ctrl+K. Также конфиг можно создать копипастом приведенного ниже листинга.

Задаем параметры. Строки, требующие корректировки, выделены жирным шрифтом:

```
#Секция startup загружается при запуске MPD5
startup:
  #Протоколировать IP-адреса, с которых выполняются подключения
  log +PHYS2
  #Параметры http и telnet доступа для мониторинга и оперативного управления.
  #Задать логин, пароль и роль администратора
  set user пользователь пароль admin
  #Задать пользователя, доступны роли operator и user, по умолчанию подразумевается user
  #set user foo1 bar1
  #Открыть локальный telnet доступ, порт 5005
```

```
  set console self 127.0.0.1 5005
  set console open
  #Открыть веб-админку на всех интерфейсах, порт 5006
  set web self 0.0.0.0 5006
  set web open

#Если в команде запуска MPD не задана конфигурация, обрабатывается секция default.
default:
  #Загрузить секцию pptp_server
  load pptp_server

pptp_server:
  #Пул динамических IP-адресов, начальный и конечный адрес
  set ippool add pool1 192.168.1.50 192.168.1.99

  #Создать динамический пучок (bundle) с именем B.
  #Сетевые интерфейсы будут создаваться динамически при подключении клиента
  create bundle template B

  #Если IP-адрес клиента принадлежит локальной подсети, присвоить ему MAC-адрес
  set iface enable proxy-arp

  #Отключать клиента при отсутствии трафика в течение заданного количества секунд
  #set iface idle 1800

  #Корректировать размер пакета TCP-соединений через туннель,
  #в случае если он превышает заданный MTU (TCP Maximum Segment Size Fix)
  set iface enable tcpmssfix

  #Разрешить сжатие заголовков TCP
  set ipcp yes vjcomp

  #IP-адрес для сервера и клиентов
  #Адрес сервера: 192.168.1.1
  #Для клиентов используем пул: "pool1", объявленный ранее
  set ipcp ranges 192.168.1.1/32 ippool pool1

  #Задать DNS-серверы для клиентов
  set ipcp dns 192.168.1.3 192.168.2.3
  #Задать WINS-серверы для клиентов
  #set ipcp nbns 192.168.1.4

  #Использовать шифрование Microsoft
  #Протокол сжатия Microsoft в базовой поставке не поддерживается, требуется пересборка ядра со стор
  set bundle enable compression
  set ccp yes mppc
  #Разрешить 40-битное и 128-битное шифрование Microsoft
  #set mppc yes e40
  set mppc yes e128
  #Разрешить безстатусный (stateless) режим шифрования.
  #Повышает устойчивость к потерям, ценой повышенной загрузки на процессор.
  set mppc yes stateless
```

```
#Создать динамическую PPTP линию
create link template L pptp

#Связать с пучком B
set link action bundle B

#Использовать фрагментацию, если размер пакета превышает MTU
set link enable multilink

#Разрешить сжатие некоторых полей в заголовке, экономит от 1 до 3 байт на пакет
set link yes acfcomp protocomp

#Использовать CHAP авторизацию, протоколы: Microsoft CHAP v2, Microsoft CHAP, CHAP MD5.
set link no pap chap eap
set link enable chap

#Отправлять LCP эхо запрос каждые 10 секунд, если ответа не последует в течение 60 секунд, считать
#Не работает вплоть до версии 5.5, LCP-пинг отправляется раз в минуту
#set link keep-alive 10 60

#Максимальный размер пакета
#Стандарт для VPN - 1400
#Если VPN-соединение разрывается при передаче большого объема данных, следует уменьшить значен
set link mtu 1402
set link mru 1400

#Принимать PPTP подключения на всех интерфейсах
set pptp self 0.0.0.0
set link enable incoming
```

Задаем список пользователей нашего VPN-сервера в файле mpd.secret:

```
ee mpd.secret
```

Задаем имена пользователей, пароли, при необходимости, присваиваем фиксированные IP-адреса. Имя пользователя и пароль чувствительны к регистру.

```
#login   password   ip
user1    pass1     192.168.0.193
user2    pass2
```

Ограничиваем доступ к файлу:

```
chmod 600 mpd.secret
```

Включение маршрутизации

Проверяем, включена ли маршрутизация, командой:

```
sysctl net.inet.ip.forwarding
```

Если в результате видим: "`net.inet.ip.forwarding: 1`", значит маршрутизация включена, переходим к следующему разделу.

Если видим: "`net.inet.ip.forwarding: 0`", значит в /etc/rc.conf необходимо добавить параметр: `gateway_enable="YES"`:

```
printf '\ngateway_enable=\"YES\"\n' >>/etc/rc.conf
```

Перезагружаем систему:

```
reboot
```

Проверяем, включена ли маршрутизация:

```
sysctl net.inet.ip.forwarding
```

В результате получаем:

```
net.inet.ip.forwarding: 1
```

Настройка брандмауэра

Для приема PPTP-подключений необходимо открыть входящие TCP-соединения на порт 1723 и разрешить GRE-трафик.
Для IPFW правила будут примерно следующие:

```
allow tcp from any to me dst-port 1723 setup keep-state
allow gre from any to me
allow gre from me to any
```

Подключенным клиентам также необходимо открыть доступ к локальной сети. В простейшем варианте, для пула 192.168.1.50 - 192.168.1.99, открываем полный доступ для всех локальных подключений, независимо от интерфейсов:

```
allow all from 192.168.1.0/24 to 192.168.1.0/24
```

Настройка логов MPD5

Для записи логов MPD5 использует syslog. Идентификатор источника сообщений задается ключом командной строки "--syslog-ident", значение по умолчанию - "mpd".

Редактируем syslog.conf:

```
ee /etc/syslog.conf
```

Добавляем следующие строки в конец файла:

```
!mpd
*.*                                     /var/log/mpd.log
!*
```

Таким образом, все сообщения от источника "mpd" будут направлены в mpd.log.

Задаем параметры ротации логов:

```
ee /etc/newsyslog.conf
```

Ежедневная ротация в полночь с сохранением 7-ми логов в сжатом формате:

```
/var/log/mpd.log                        600  7    *    @T00  JC
```

Альтернативный вариант, с ежемесячной ротацией и хранением логов за последние три месяца:

```
/var/log/mpd.log                        600  3    *    $M1D0 JC
```

Подробнее о файле конфигурации читаем в справке: man newsyslog.conf.

Создаем лог-файл:

```
touch /var/log/mpd.log
```

Перезагружаем конфигурацию syslog:

```
service syslogd reload
```

Запуск MPD5

Разрешаем запуск MPD, добавляем в /etc/rc.conf параметр: `mpd_enable="YES"`:

```
printf '\nmpd_enable=\"YES\"\n' >>/etc/rc.conf
```

Запускаем службу:

```
service mpd5 start
```

Проверяем, запущена ли служба и параметры запуска:

```
ps -ax | grep mpd5
```

В результате получаем:

```
1274  -  Ss    0:00.00 /usr/local/sbin/mpd5 -p /var/run/mpd5.pid -b
```

Проверяем, слушается ли порт:

```
netstat -an | grep 1723
```

Результат должен быть следующим:

```
tcp4       0      0 *.1723                 *.*                    LISTEN
```

Проверяем сообщения в лог-файле:

```
cat /var/log/mpd.log
```

В случае успешного старта видим следующий текст:

```
Multi-link PPP daemon for FreeBSD

process 1274 started, version 5.7 (root@10i386-default-job-10 21:06 12-Jun-2014)
CONSOLE: listening on 127.0.0.1 5005
web: listening on 0.0.0.0 5006
PPTP: waiting for connection on 0.0.0.0 1723
```

Заходим браузером в веб-админку: http:/адрес.сервера:5006, проверяем параметры соединений.

Создаем подключение в Windows, либо настраиваем на удаленной стороне MPD-клиент и пробуем подключиться. При настройке клиентского подключения в Windows, чтобы предотвратить туннелирование всего интернет трафика в удаленную сеть, в дополнительных настройках протокола TCP/IP, необходимо отключить флажок "Использовать основной шлюз в удаленной сети". В этом случае будет туннелироваться только трафик удаленной подсети в соответствии с ее классом.

В случае проблем используем tcpdump.

Мониторим физический канал:

```
tcpdump -v -ni интерфейс tcp port 1723 or proto gre
```

Мониторим туннель:

```
tcpdump -ni ng0
```

Возможные проблемы

Ошибка: "Can't create socket node: No such file or directory. Netgraph initialization failed", может возникнуть после установки новой версии системы. Проблема возникает из-за линковки программы со старыми системными библиотеками. В этом случае необходимо переустановить или пересобрать MPD.

Анализ логов MPD5

Чтобы быть в курсе как используется наш VPN-сервер, организуем ежедневную отправку на почту отчета по сессиям. Для получения административных почтовых уведомлений по протоколу POP3 потребуется установить Dovecot или Qpopper.

Готового решения на все случаи жизни мне найти не удалось. Рассмотрим примеры скриптов, которые можно взять за основу для построения своего анализатора.

Простейший отчет по сессиям

Сканируем лог за прошедший день командой grep, выбираем сообщения о подключении, отключении клиентов и имена пользователей. Ротация логов должна быть ежедневной.

Результат работы выдает следующий:

```
Jun 14 18:10:20 bsd-10 mpd: pptp0: attached to connection with 92.68.55.2 2539
Jun 14 18:10:23 bsd-10 mpd: [L-1]   MESG: MSRASV5.20
Jun 14 18:10:23 bsd-10 mpd: [L-1]   MESG: MSRAS-0-DESKTOP
Jun 14 18:10:23 bsd-10 mpd: [L-1]   Name: "user1"
Jun 14 18:15:19 bsd-10 mpd: pptp0-0: call cleared by peer
Jun 14 18:15:19 bsd-10 mpd: pptp0-0: killing channel
```

Добавим в /usr/local/etc/periodic/daily скрипт с именем 800.mpd:

```
ee /usr/local/etc/periodic/daily/800.mpd
```

Со следующим содержимым:

```
#!/bin/sh
echo
echo PPTP connections
```

```
bzgrep -E '(mpd: pptp)|(MESG:)|(Name)' /var/log/mpd.log.0.bz2
```

Задаем права доступа:

```
chmod 755 /usr/local/etc/periodic/daily/800.mpd
```

Mpdstat

Единственный готовый скрипт, который мне удалось накопать на просторах интернета.

Сайт проекта: http://code.google.com/p/mpdstat/.

Perl-скрипт интегрируется в periodic, выдает следующий результат:

```
user1
        duration: 0d 00:04:56   link: [L-1]
        from: Jun 14 18:10:23   to: Jun 14 18:15:19

Active Session Report

User Report
       user1: total: 0d 03:53:41          5times
```

Устанавливаем Perl:

```
pkg install perl5
```

Создаем папку для загрузки в пользовательском профиле:

```
mkdir -p ~/src
```

Загружаем Mpdstat:

```
svnlite checkout http://mpdstat.googlecode.com/svn/trunk/ ~/src/mpdstat
```

Переходим в папку с программой:

```
cd ~/src/mpdstat
```

Выполняем установку:

```
make install
```

В случае ошибки установки, переименовываем файл 900.mpdstatus в 900.mpdstat и запускаем установку повторно:

```
mv 900.mpdstatus 900.mpdstat
make install
```

Открываем /etc/periodic.conf, если файл отсутствует в вашей системе, создаем его:

```
ee /etc/periodic.conf
```

Разрешаем исполнения скрипта и задаем путь к лог файлам:

```
daily_status_security_mpdstatus_enable="YES"
daily_status_security_logdir="/var/log"
```

Выполняем тестовый запуск:

```
periodic security
```

Скрипт ищет лог за прошедший день, если он не найден, обрабатывает mpd.log.

Парсер на PHP

Я использую свой парсер на PHP, генерирующий следующий вывод:

```
L1> Jun 14 18:10:20    92.68.55.2 00:04:59 user1    static.kpn.net MSRASV5.20
L1> Jun 14 18:15:51    92.68.55.2 00:04:01 user1    static.kpn.net MSRASV5.20
L1> Jun 14 18:20:12    92.68.55.2 01:47:45 user1    static.kpn.net MSRASV5.20
L1> Jun 14 21:53:35    92.68.55.2 01:56:10 user1    static.kpn.net MSRASV5.20
L1> Jun 14 23:50:37    92.68.55.2 00:00:55 user1    static.kpn.net MSRASV5.20
```

Скрипт отображает IP-адреса, с которых устанавливаются соединения, и выполняет реверсные DNS-запросы. Параметр "log +PHYS2" должен быть включен в mpd.conf.

Устанавливаем PHP:

```
pkg install php5 php5-bz2
```

Создаем папку программы в профиле пользователя и переходим в нее:

```
mkdir /root/Programs
cd /root/Programs
```

Загружаем архив:

```
fetch http://itadept.ru//files/freebsd-mpd5-server/MpdLogParser-0.1.tar.bz2
```

Распаковываем:

```
tar -xf MpdLogParser-0.1.tar.bz2
```

Переходим в папку с программой:

```
cd MpdLogParser
```

Создаем файл конфигурации:

```
cp MpdLogParser_Config.php.sample MpdLogParser_Config.php
```

Редактируем конфиг:

```
ee MpdLogParser_Config.php
```

Задаем параметры:

```
/* Часовой пояс из списка: http://php.net/manual/ru/timezones.php */

date_default_timezone_set("Europe/Moscow");

/* Путь к лог-файлу */
define('LOG_PATH',"/var/log/mpd.log.0.bz2");

/* IP-адреса, исключаемые из списка */
$IgnoreIP=array(
//    "1.2.3.4",
//    "5.6.7.8",
);
```

Выполняем пробный запуск, лог-файл, заданный в конфиге (mpd.log.0.bz2) должен существовать, автопоиск не предусмотрен:

```
./MpdLogParser.php
```

Добавим в /usr/local/etc/periodic/daily скрипт с именем 800.mpd:

```
ee /usr/local/etc/periodic/daily/800.mpd
```

Со следующим содержимым:

```
#!/bin/sh
echo
echo PPTP connections
/usr/local/bin/php /root/Programs/MpdLogParser/MpdLogParser.php
```

Задаем права доступа:

```
chmod 755 /usr/local/etc/periodic/daily/800.mpd
```

Выполняем тестовый запуск:

```
periodic daily
```

Вывод команды будет отправлен на почту пользователя root.

Настройка завершена.

Revision #1
Created 4 July 2025 10:50:17 by buzz
Updated 4 July 2025 10:51:02 by buzz