VPN через IPsec

Table of Contents

Безопасность интернет-протокола (IPsec) — это набор протоколов, работающих поверх уровня интернет-протокола (IP). Он позволяет двум или более узлам обмениваться данными безопасным образом, аутентифицируя и шифруя каждый IP-пакет сеанса связи. Сетевой стек IPsec в FreeBSD основан на реализации http://www.kame.net/ и поддерживает сеансы как IPv4, так и IPv6.

IPsec состоит из следующих подпротоколов:

- *Протокол Encapsulated Security Payload (ESP)*: этот протокол защищает данные IP-пакета от вмешательства третьих сторон, шифруя содержимое с использованием алгоритмов симметричной криптографии, таких как Blowfish и 3DES.
- *Authentication Header (AH)*: этот протокол защищает заголовок IP-пакета от вмешательства третьих сторон и подмены путем вычисления криптографической контрольной суммы и хеширования полей заголовка IP-пакета с использованием безопасной хеш-функции. Затем следует дополнительный заголовок, содержащий хеш, что позволяет аутентифицировать информацию в пакете.
- *Протокол сжатия передаваемых данных IP (IPComp — IP Payload Compression Protocol)*: этот протокол пытается повысить производительность связи за счёт сжатия передаваемых данных IP, чтобы уменьшить объём отправляемых данных.

Эти протоколы могут использоваться как вместе, так и по отдельности, в зависимости от окружения.

IPsec поддерживает два режима работы. Первый режим, *Transport Mode* (Транспортный режим), защищает соединение между двумя хостами. Второй режим, *Tunnel Mode* (Туннельный режим), используется для построения виртуальных туннелей, обычно известных как Virtual Private Networks (VPN, Виртуальные частные сети). Подробную информацию о подсистеме IPsec в FreeBSD можно найти в ipsec(4).

В статье демонстрируется процесс настройки IPsecVPN между домашней сетью и корпоративной сетью.

В примере сценария:

- Оба сайта подключены к Интернету через шлюз, на котором работает FreeBSD.
- Шлюз в каждой сети имеет как минимум один внешний IP-адрес. В этом примере внешний IP-адрес корпоративной LAN — `172.16.5.4`, а внешний IP-адрес домашней LAN — `192.168.1.12`.

- Внутренние адреса двух сетей могут быть как публичными, так и частными IP-адресами. Однако их адресные пространства не должны пересекаться. В этом примере внутренний IP-адрес корпоративной LAN — `10.246.38.1`, а внутренний IP-адрес домашней LAN — `10.0.0.5`.

```
          corporate                              home
10.246.38.1/24 -- 172.16.5.4 <--> 192.168.1.12 -- 10.0.0.5/24
```

1. Настройка VPN на FreeBSD

Для начала необходимо установить пакет security/ipsec-tools из Коллекции портов. Это программное обеспечение предоставляет ряд приложений, поддерживающих настройку.

Следующее требование — создать два псевдоустройства gif(4), которые будут использоваться для туннелирования пакетов и обеспечения правильной связи между обеими сетями. От имени `root` выполните следующую команду на каждом шлюзе:

```
corp-gw# ifconfig gif0 create
corp-gw# ifconfig gif0 10.246.38.1 10.0.0.5
corp-gw# ifconfig gif0 tunnel 172.16.5.4 192.168.1.12
```

```
home-gw# ifconfig gif0 create
home-gw# ifconfig gif0 10.0.0.5 10.246.38.1
home-gw# ifconfig gif0 tunnel 192.168.1.12 172.16.5.4
```

Проверьте настройку на каждом шлюзе, используя `ifconfig gif0`. Вот вывод с домашнего шлюза:

```
gif0: flags=8051 mtu 1280
tunnel inet 172.16.5.4 --> 192.168.1.12
inet6 fe80::2e0:81ff:fe02:5881%gif0 prefixlen 64 scopeid 0x6
inet 10.246.38.1 --> 10.0.0.5 netmask 0xffffff00
```

Вот вывод с корпоративного шлюза:

```
gif0: flags=8051 mtu 1280
tunnel inet 192.168.1.12 --> 172.16.5.4
inet 10.0.0.5 --> 10.246.38.1 netmask 0xffffff00
inet6 fe80::250:bfff:fe3a:c1f%gif0 prefixlen 64 scopeid 0x4
```

После завершения оба внутренних IP-адреса должны быть доступны с использованием ping(8):

```
home-gw# ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
64 bytes from 10.0.0.5: icmp_seq=0 ttl=64 time=42.786 ms
64 bytes from 10.0.0.5: icmp_seq=1 ttl=64 time=19.255 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=64 time=20.440 ms
64 bytes from 10.0.0.5: icmp_seq=3 ttl=64 time=21.036 ms
--- 10.0.0.5 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 19.255/25.879/42.786/9.782 ms

corp-gw# ping 10.246.38.1
PING 10.246.38.1 (10.246.38.1): 56 data bytes
64 bytes from 10.246.38.1: icmp_seq=0 ttl=64 time=28.106 ms
64 bytes from 10.246.38.1: icmp_seq=1 ttl=64 time=42.917 ms
64 bytes from 10.246.38.1: icmp_seq=2 ttl=64 time=127.525 ms
64 bytes from 10.246.38.1: icmp_seq=3 ttl=64 time=119.896 ms
64 bytes from 10.246.38.1: icmp_seq=4 ttl=64 time=154.524 ms
--- 10.246.38.1 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 28.106/94.594/154.524/49.814 ms
```

Как и ожидалось, обе стороны имеют возможность отправлять и получать ICMP-пакеты с приватно настроенных адресов. Далее, оба шлюза должны быть настроены для маршрутизации пакетов, чтобы корректно передавать трафик из сетей за каждым шлюзом. Следующие команды позволят достичь этой цели:

```
corp-gw# route add 10.0.0.0 10.0.0.5 255.255.255.0
corp-gw# route add net 10.0.0.0: gateway 10.0.0.5
home-gw# route add 10.246.38.0 10.246.38.1 255.255.255.0
home-gw# route add host 10.246.38.0: gateway 10.246.38.1
```

Внутренние машины должны быть доступны с каждого шлюза, а также с машин за шлюзами. Снова используйте ping(8) для проверки:

```
corp-gw# ping -c 3 10.0.0.8
PING 10.0.0.8 (10.0.0.8): 56 data bytes
64 bytes from 10.0.0.8: icmp_seq=0 ttl=63 time=92.391 ms
64 bytes from 10.0.0.8: icmp_seq=1 ttl=63 time=21.870 ms
64 bytes from 10.0.0.8: icmp_seq=2 ttl=63 time=198.022 ms
--- 10.0.0.8 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 21.870/101.846/198.022/74.001 ms

home-gw# ping -c 3 10.246.38.107
PING 10.246.38.1 (10.246.38.107): 56 data bytes
64 bytes from 10.246.38.107: icmp_seq=0 ttl=64 time=53.491 ms
64 bytes from 10.246.38.107: icmp_seq=1 ttl=64 time=23.395 ms
64 bytes from 10.246.38.107: icmp_seq=2 ttl=64 time=23.865 ms
```

```
--- 10.246.38.107 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 21.145/31.721/53.491/12.179 ms
```

На этом этапе трафик передаётся между сетями, инкапсулированный в туннель gif, но без какого-либо шифрования. Далее используйте IPSec для шифрования трафика с использованием предварительно согласованных ключей (PSK). За исключением IP-адресов, файл /usr/local/etc/racoon/racoon.conf на обоих шлюзах будет идентичным и выглядеть примерно так:

```
path    pre_shared_key  "/usr/local/etc/racoon/psk.txt"; #location of pre-shared key
file
log     debug;  #log verbosity setting: set to 'notify' when testing and debugging is
complete

padding # options are not to be changed
{
        maximum_length  20;
        randomize       off;
        strict_check    off;
        exclusive_tail  off;
}

timer   # timing options. change as needed
{
        counter         5;
        interval        20 sec;
        persend         1;
#       natt_keepalive  15 sec;
        phase1          30 sec;
        phase2          15 sec;
}

listen  # address [port] that racoon will listen on
{
        isakmp          172.16.5.4 [500];
        isakmp_natt     172.16.5.4 [4500];
}

remote  192.168.1.12 [500]
{
        exchange_mode   main,aggressive;
        doi             ipsec_doi;
        situation       identity_only;
        my_identifier   address 172.16.5.4;
        peers_identifier        address 192.168.1.12;
        lifetime        time 8 hour;
        passive         off;
        proposal_check  obey;
#       nat_traversal   off;
```

```
        generate_policy off;

                        proposal {
                                encryption_algorithm    blowfish;
                                hash_algorithm          md5;
                                authentication_method   pre_shared_key;
                                lifetime time           30 sec;
                                dh_group                1;
                        }
}

sainfo  (address 10.246.38.0/24 any address 10.0.0.0/24 any)    # address
$network/$netmask $type address $network/$netmask $type ( $type being any or esp)
{                               # $network must be the two internal networks you are
joining.
        pfs_group       1;
        lifetime        time    36000 sec;
        encryption_algorithm    blowfish,3des;
        authentication_algorithm        hmac_md5,hmac_sha1;
        compression_algorithm   deflate;
}
```

Для описания каждой доступной опции обратитесь к справочной странице racoon.conf.

База данных политики безопасности (SPD) должна быть настроена таким образом, чтобы FreeBSD и racoon могли шифровать и расшифровывать сетевой трафик между узлами.

Это может быть реализовано с помощью shell-скрипта, подобного приведённому ниже, на корпоративном шлюзе. Этот файл будет использоваться при инициализации системы и должен быть сохранён как /usr/local/etc/racoon/setkey.conf.

```
flush;
spdflush;
# To the home network
spdadd 10.246.38.0/24 10.0.0.0/24 any -P out ipsec esp/tunnel/172.16.5.4-
192.168.1.12/use;
spdadd 10.0.0.0/24 10.246.38.0/24 any -P in ipsec esp/tunnel/192.168.1.12-
172.16.5.4/use;
```

После настройки racoon может быть запущен на обоих шлюзах с помощью следующей команды:

```
# /usr/local/sbin/racoon -F -f /usr/local/etc/racoon/racoon.conf -l
/var/log/racoon.log
```

Вывод должен быть похож на следующий:

```
corp-gw# /usr/local/sbin/racoon -F -f /usr/local/etc/racoon/racoon.conf
Foreground mode.
2006-01-30 01:35:47: INFO: begin Identity Protection mode.
2006-01-30 01:35:48: INFO: received Vendor ID: KAME/racoon
2006-01-30 01:35:55: INFO: received Vendor ID: KAME/racoon
2006-01-30 01:36:04: INFO: ISAKMP-SA established 172.16.5.4[500]-192.168.1.12[500]
spi:623b9b3bd2492452:7deab82d54ff704a
2006-01-30 01:36:05: INFO: initiate new phase 2 negotiation:
172.16.5.4[0]192.168.1.12[0]
2006-01-30 01:36:09: INFO: IPsec-SA established: ESP/Tunnel 192.168.1.12[0]-
>172.16.5.4[0] spi=28496098(0x1b2d0e2)
2006-01-30 01:36:09: INFO: IPsec-SA established: ESP/Tunnel 172.16.5.4[0]-
>192.168.1.12[0] spi=47784998(0x2d92426)
2006-01-30 01:36:13: INFO: respond new phase 2 negotiation:
172.16.5.4[0]192.168.1.12[0]
2006-01-30 01:36:18: INFO: IPsec-SA established: ESP/Tunnel 192.168.1.12[0]-
>172.16.5.4[0] spi=124397467(0x76a279b)
2006-01-30 01:36:18: INFO: IPsec-SA established: ESP/Tunnel 172.16.5.4[0]-
>192.168.1.12[0] spi=175852902(0xa7b4d66)
```

Чтобы убедиться, что туннель работает правильно, переключитесь на другую консоль и используйте tcpdump(1) для просмотра сетевого трафика с помощью следующей команды. Замените `em0` на требуемую сетевую интерфейсную карту:

```
corp-gw# tcpdump -i em0 host 172.16.5.4 and dst 192.168.1.12
```

На консоли должны появиться данные, аналогичные следующим. Если этого не произошло, возникла проблема, и потребуется отладка возвращённых данных.

```
01:47:32.021683 IP corporatenetwork.com > 192.168.1.12.privatenetwork.com:
ESP(spi=0x02acbf9f,seq=0xa)
01:47:33.022442 IP corporatenetwork.com > 192.168.1.12.privatenetwork.com:
ESP(spi=0x02acbf9f,seq=0xb)
01:47:34.024218 IP corporatenetwork.com > 192.168.1.12.privatenetwork.com:
ESP(spi=0x02acbf9f,seq=0xc)
```

На этом этапе обе сети должны быть доступны и казаться частью одной сети. Скорее всего, обе сети защищены межсетевым экраном. Чтобы разрешить передачу трафика между ними, необходимо добавить правила для пропуска пакетов. Для межсетевого экрана ipfw(8) добавьте следующие строки в файл конфигурации межсетевого экрана:

```
ipfw add 00201 allow log esp from any to any
ipfw add 00202 allow log ah from any to any
ipfw add 00203 allow log ipencap from any to any
ipfw add 00204 allow log udp from any 500 to any
```

Номера правил могут потребовать изменения в зависимости от текущей конфигурации хоста.

Для пользователей pf(4) или ipf(8) должны сработать следующие правила:

```
pass in quick proto esp from any to any
pass in quick proto ah from any to any
pass in quick proto ipencap from any to any
pass in quick proto udp from any port = 500 to any port = 500
pass in quick on gif0 from any to any
pass out quick proto esp from any to any
pass out quick proto ah from any to any
pass out quick proto ipencap from any to any
pass out quick proto udp from any port = 500 to any port = 500
pass out quick on gif0 from any to any
```

Наконец, чтобы включить поддержку VPN при загрузке системы, добавьте следующие строки в /etc/rc.conf:

```
ipsec_enable="YES"
ipsec_program="/usr/local/sbin/setkey"
ipsec_file="/usr/local/etc/racoon/setkey.conf" # allows setting up spd policies on
boot
racoon_enable="yes"
```