

Построение VPN туннеля между двумя подсетями, защищаемыми шлюзами безопасности «Bel VPN Gate»

Описание стенда

Сценарий иллюстрирует построение защищенного соединения между двумя подсетями SN1 и SN2, которые защищаются шлюзами безопасности «Bel VPN Gate». Для защиты будет построен VPN туннель между устройствами GW1 и GW2. Устройства IPHost1 и IPHost2 смогут общаться между собой по защищенному каналу (VPN). Все остальные соединения разрешены, но защищаться не будут.

В рамках данного сценария для аутентификации партнеры будут использовать сертификаты.

Параметры защищенного соединения:

- IKE параметры:
 - Аутентификация – на сертификатах открытого ключа ЭЦП по СТБ 34.101.45-2013;
 - Алгоритм шифрования – СТБ 34.101.31-2011 (6.4);
 - Алгоритм вычисления хеш-функции – СТБ 34.101.31-2011 (6.9);
 - Протокол согласования ключей – протокол Диффи-Хеллмана на эллиптических кривых (СТБ 34.101.66-2014).
- IPsec параметры:
 - Туннельный режим, протокол ESP:
 - Алгоритм шифрования – СТБ 34.101.31-2011 (6.4);
 - Алгоритм контроля целостности – СТБ 34.101.31-2011 (6.6).

Схема стенда (Рисунок 1):

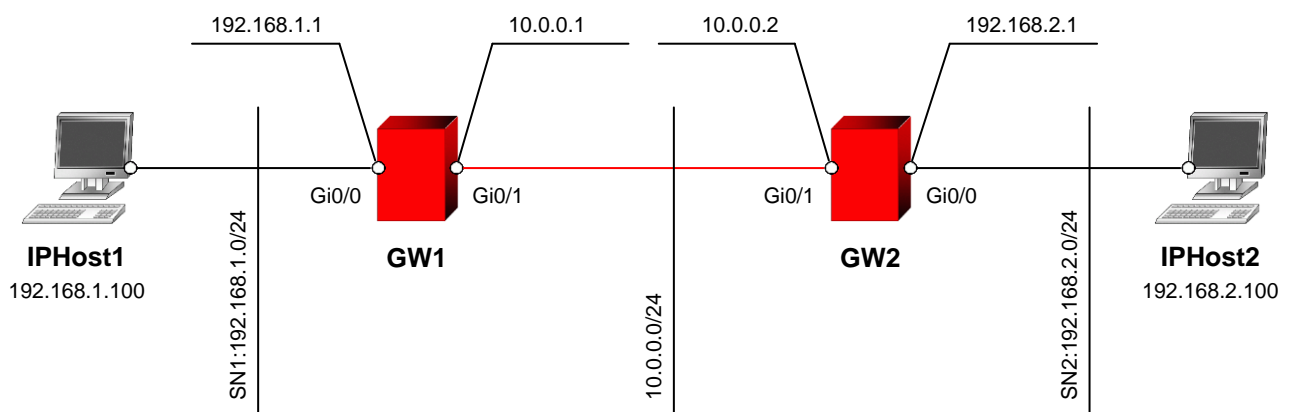


Рисунок 1

Настройка стенда

Настройка шлюза безопасности GW1

Настройку начните со шлюза безопасности GW1. Все настройки производятся через локальную консоль или удаленно (SSH с правами суперпользователя) по доверенному каналу связи.

Инициализация шлюза описывается в документации на ПАК «Bel VPN Gate 4.1» – [Initialization_gate_Gate_41](#) («Инициализация», раздел «Инициализация шлюза безопасности Bel VPN Gate 4.1 при первом старте»).

В данном сценарии для аутентификации используются сертификаты. Для корректной работы необходимо зарегистрировать сертификат CA (УЦ) и локальный сертификат.

В данном сценарии список отозванных сертификатов (CRL) не используется и будет отключен. Информацию об использовании CRL можно найти в документации на ПАК «Bel VPN Gate 4.1» – [Console_command_reference_Gate_41](#) («Руководство администратора Cisco-like команды», раздел «Команды для работы с сертификатами»).

Настройка интерфейсов

IP-адреса для интерфейсов рекомендуется настроить через cisco-like консоль.

1. Для входа в консоль запустите `cs_console`:

```
root@belvpngate:~# cs_console
belvpngate>enable
Password:
```

Пароль по умолчанию: `csp`.

2. Перейдите в режим настройки:

```
belvpngate#conf t
```

```
Enter configuration commands, one per line. End with CNTL/Z.
```

3. В настройках интерфейсов задайте IP-адреса:

```
belvpngate(config)#interface GigabitEthernet 0/0
belvpngate(config-if)#ip address 192.168.1.1 255.255.255.0
belvpngate(config-if)#no shutdown
belvpngate(config-if)#exit
belvpngate(config)#interface GigabitEthernet 0/1
belvpngate(config-if)#ip address 10.0.0.1 255.255.255.0
belvpngate(config-if)#no shutdown
belvpngate(config-if)#exit
```

4. Задайте адрес шлюза по умолчанию:

```
belvpngate(config)#ip route 0.0.0.0 0.0.0.0 10.0.0.2
```

5. Выйдите из cisco-like интерфейса:

```
belvpngate(config)#end
belvpngate#exit
```

Формирование запроса и регистрация сертификата

Для регистрации CA сертификата (сертификата УЦ) необходимо выполнить следующие действия:

1. Установите правильное системное время.

```
root@belvpngate:~# date MMDDHHmmYYYY
```

MM – месяц;

DD – день;

HH – часы;

mm – минуты;

YYYY – год

Пример установки даты:

```
root@belvpngate:~# date 041013152013
Wed Apr 10 13:15:00 UTC 2013
```

Данная запись соответствует 10 апреля 2013 года 13:15.

2. Создайте папку /opt/certs:

```
root@belvpngate:~# mkdir /opt/certs
```

3. Создайте контейнер на ключевом носителе:

```
root@belvpngate:~/opt/Avset/bin/cryptocont n -n=контейнер -p=пароль
```

контейнер – название создаваемого контейнера, для создания на НКИ ДОЛЖНО содержать в начале названия префикс **“av:”**;

пароль – пароль (PIN) для доступа к носителю ключевой информации AvPass/AvBign.

Пример создания криптоконтейнера на НКИ:

```
root@belvpngate:~/opt/Avest/bin/cryptocont n -n=av:container -p=12345678
```

4. Сформируйте запрос на сертификат.

```
root@belvpngate:~/opt/Avset/bin/cryptcont r -n=контейнер -p=пароль -cn=CommonName -c=BY -o=OrgName -t=OrgUnitName -f=путь_к_файлу
```

контейнер – название контейнера, созданного на предыдущем шаге;

пароль – пароль (PIN) для доступа к носителю ключевой информации;

CommonName – идентификатор устройства;

OrgName – наименование организации;

OrgUnitName – наименование подразделения;

путь_к_файлу – путь к файлу с создаваемым запросом, рекомендуется указывать расширение **“.req”**.

Пример создания запроса:

```
root@belvpngate:~/opt/Avest/bin/cryptocont r -n=av:container -p=12345678 -cn=GW1 -c=BY -o=S-Terra -t=Research -f=/opt/certs/GW1.req
```

5. Передайте полученный запрос сертификата на УЦ и получите файл сертификата (в с расширением **r7b** или **cer**).

Если вы получили файл сертификата в формате r7b, выполните экспорт в отдельные cer файлы.

6. Доставьте файлы сертификатов на Шлюз безопасности в предварительно созданный на нем каталог /opt/certs. Для доставки можно воспользоваться утилитой pscp.exe из пакета Putty, применив команду:

```
pscp исходный_файл root@адрес_шлюза:/путь_к_файлу
```

исходный_файл – путь к файлу сертификата;

- адрес_шлюза** – сетевой адрес Шлюза;
- путь_к_файлу** – полный путь для сохранения файла на Шлюзе.

Пример передачи файла на Шлюз безопасности:

```
pscp D:\ca.cer root@192.168.1.1:/opt/certs
...
Store key in cache? (y/n)
root@192.168.1.1's password:
```

Важно: Среда передачи в этом случае должна быть доверенной. Описание создания доверенной среды через недоверенные каналы связи смотрите в документации на ПАК «Bel VPN Gate 4.1» [Settings_gate Gate 41](#) («Общие настройки», раздел «Построение VPN туннеля между шлюзом безопасности Bel VPN Gate 4.1 и рабочим местом администратора для удаленной настройки шлюза»).

7. Выполните импорт сертификата УЦ в базу Шлюза используя утилиту `cert_mgr`:

```
root@belvpngate:~# cert_mgr import -f путь_к_файлу -t
```

путь_к_файлу – полный путь к файлу сертификата УЦ

Пример импорта:

```
root@belvpngate:~# cert_mgr import -f /opt/cert/UC.cer -t
1 OK C=BY,L=Minsk,O=S-Terra,OU=Research,CN=UC
```

8. Выполните импорт локального (личного) сертификата в базу Шлюза:

```
root@belvpngate:~# cert_mgr import -f путь_к_файлу -kc контейнер -kcp пароль
```

путь_к_файлу – полный путь к файлу сертификата УЦ;

контейнер – название контейнера, созданного ранее;

пароль – пароль для доступа к ключевому носителю информации.

Пример импорта:

```
root@belvpngate:~# cert_mgr import -f /opt/cert/GW1.cer -kc av:container -kcp 12345678
1 OK CN=GW1,C=BY,O=S-Terra,OU=Research
```

9. Выведите список сертификатов, находящихся в базе Шлюза, командой `cert_mgr show` и проверьте наличие записей **trusted** и **local**:

```
root@belvpngate:~# cert_mgr show
```

Пример вывода:

```
root@belvpngate:~# cert_mgr show
Found 2 certificates. No CRLs found.
1 Status: trusted C=BY,L=Minsk,O=S-Terra,OU=Research,CN=UC
2 Status: local CN=GW1,C=BY,O=S-Terra,OU=Research
```

10. Убедитесь что все сертификаты активны – статус сертификата должен быть **active**:

```
root@belvpngate:~# cert_mgr check
```

Пример:

```
root@belvpngate:~# cert_mgr check
1 State: Active C=BY,L=Minsk,O=S-Terra,OU=Research,CN=UC
2 State: Active CN=GW1,C=BY,O=S-Terra,OU=Research
```

Создание политики безопасности

После регистрации сертификатов необходимо создать политику безопасности для GW1. Создавать политику рекомендуется в интерфейсе командной строки. Для входа в консоль запустите `cs_console`:

```
root@belvpngate:~# cs_console
belvpngate>en
```

```
Password:
```

Пароль по умолчанию: csp.

Важно: пароль по умолчанию необходимо сменить.

1. Перейдите в режим настройки:

```
belvpngate#conf t
```

```
Enter configuration commands, one per line. End with CNTL/Z.
```

2. Смените пароль по умолчанию:

```
belvpngate(config)#username cscons password <пароль>
```

3. Смените название шлюза:

```
belvpngate(config)#hostname GW1
```

4. Задайте тип идентификации:

```
GW1(config)#crypto isakmp identity dn
```

5. Задайте параметры для IKE:

```
GW1(config)#crypto isakmp policy 1
GW1(config-isakmp)#hash belt
GW1(config-isakmp)#encryption belt
GW1(config-isakmp)#authentication belt-sig
GW1(config-isakmp)#group beltdh
GW1(config-isakmp)#exit
```

6. Создайте набор преобразований для IPsec:

```
GW1(config)#crypto ipsec transform-set TSET esp-belt esp-belt-mac
GW1(cfg-crypto-trans)#mode tunnel
GW1(cfg-crypto-trans)#exit
```

7. Опишите трафик, который планируется защищать. Для этого создайте расширенный список доступа:

```
GW1(config)#ip access-list extended LIST
GW1(config-ext-nacl)#permit ip 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255
GW1(config-ext-nacl)#exit
```

8. Создайте крипто-карту:

```
GW1(config)#crypto map CMAP 1 ipsec-isakmp
```

```
% NOTE: This new crypto map will remain disabled until a peer
and a valid access list have been configured.
```

```
GW1(config-crypto-map)#match address LIST
GW1(config-crypto-map)#set transform-set TSET
GW1(config-crypto-map)#set pfs beltdh
GW1(config-crypto-map)#set peer 10.0.0.2
GW1(config-crypto-map)#exit
```

9. Привяжите крипто-карту к интерфейсу, на котором будет туннель:

```
GW1(config)#interface GigabitEthernet 0/1
GW1(config-if)#crypto map CMAP
GW1(config-if)#exit
```

10. Отключите обработку списка отозванных сертификатов (CRL):

```
GW1(config)#crypto pki trustpoint s-terra_technological_trustpoint
```

```
GW1 (ca-trustpoint) #revocation-check none
GW1 (ca-trustpoint) #exit
```

11. Настройка устройства GW1 в cisco-like консоли завершена. При выходе из конфигурационного режима происходит загрузка конфигурации:

```
GW1 (config) #end
GW1 #exit
```

В **Приложении** представлен текст [cisco-like конфигурации](#) для шлюза GW1.

Настройка шлюза GW2

Настройка шлюза безопасности GW2 происходит аналогично настройке устройства GW1, с заменой IP-адресов в соответствующих разделах конфигурации.

В **Приложении** представлен текст [cisco-like конфигурации](#) для шлюза GW2.

Настройка устройства IPHost1

На устройстве IPHost1 задайте IP-адрес, а в качестве шлюза по умолчанию укажите IP-адрес внутреннего интерфейса шлюза безопасности GW1 – 192.168.1.1.

Настройка устройства IPHost2

На устройстве IPHost2 задайте IP-адрес, а в качестве шлюза по умолчанию укажите IP-адрес внутреннего интерфейса шлюза безопасности GW2 – 192.168.2.1.

Проверка работоспособности стенда

После того, как настройка всех устройств завершена, иницируйте создание защищенного соединения.

На устройстве IPHost1 выполните команду ping:

```
ping 192.168.2.100
```

```
PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.  
64 bytes from 192.168.2.100: icmp_req=1 ttl=62 time=917 ms  
64 bytes from 192.168.2.100: icmp_req=2 ttl=62 time=3.76 ms  
64 bytes from 192.168.2.100: icmp_req=3 ttl=62 time=3.38 ms  
64 bytes from 192.168.2.100: icmp_req=4 ttl=62 time=3.88 ms  
^C  
--- 192.168.2.100 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3004ms  
rtt min/avg/max/mdev = 3.381/232.189/917.719/395.790 ms
```

В результате выполнения этой команды между устройствами GW1 и GW2 будет установлен VPN туннель.

Убедиться в этом можно, выполнив на устройстве GW1 команду:

```
root@GW1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded  
  
ISAKMP connections:  
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd  
1 3 (10.0.0.1,500)-(10.0.0.2,500) active 1976 1904  
  
IPsec connections:  
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd  
1 3 (192.168.1.0-192.168.1.255,*)-(192.168.2.0-192.168.2.255,*) * ESP tunn 352 352
```

Согласно созданной политике безопасности весь трафик между сетями SN1 и SN2 будет зашифрован. Прохождение остального трафика будет разрешено, но не будет защищаться шифрованием.

Приложение

Текст cisco-like конфигурации для шлюза GW1

```
!  
version 12.4  
no service password-encryption  
!  
crypto ipsec df-bit copy  
crypto isakmp identity dn  
username ccons privilege 15 password 0 csp  
aaa new-model  
!  
!  
hostname GW1  
enable password csp  
!  
!  
logging trap debugging  
!  
!  
crypto isakmp policy 1  
  encr belt  
  hash belt  
  authentication belt-sig  
  group beltdh  
!  
crypto ipsec transform-set TSET esp-belt esp-belt-mac  
!  
ip access-list extended LIST  
  permit ip 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255  
!  
!  
crypto map CMAP 1 ipsec-isakmp  
  match address LIST  
  set transform-set TSET  
  set pfs beltdh  
  set peer 10.0.0.2  
!  
interface GigabitEthernet0/0  
  ip address 192.168.1.1 255.255.255.0  
!  
interface GigabitEthernet0/1  
  ip address 10.0.0.1 255.255.255.0  
  crypto map CMAP  
!  
interface GigabitEthernet0/2  
  no ip address  
  shutdown  
!  
interface GigabitEthernet0/3  
  no ip address  
  shutdown  
!  
!  
ip route 0.0.0.0 0.0.0.0 10.0.0.2  
!  
crypto pki trustpoint s-terra_technological_trustpoint  
  revocation-check none
```



```
crypto pki certificate chain s-terra_technological_trustpoint
certificate 4E4B0B11EFDB389E4E86244CDAA1B275
30820216308201C5A00302010202104E4B0B11EFDB389E4E86244CDAA1B27530
0806062A85030202033064310B3009060355040613025255310F300D06035504
...
06010401823715010403020100300806062A85030202030341004A63F022F03D
009B097DD81A81CFC792664AAC9E6908587195AE17A5D526DE196CB0D5B7E713
E9D07F4DC61F04CDBC87579FC44CE66D524CF742F2784805733F
quit
!
end
```

Текст cisco-like конфигурации для шлюза GW2

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity dn
username cscons privilege 15 password 0 csp
aaa new-model
!
!
hostname GW2
enable password csp
!
!
logging trap debugging
!
!
crypto isakmp policy 1
  encr belt
  hash belt
  authentication belt-sig
  group beltdh
!
crypto ipsec transform-set TSET esp-belt esp-belt-mac
!
ip access-list extended LIST
  permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
!
!
crypto map CMAP 1 ipsec-isakmp
  match address LIST
  set transform-set TSET
  set pfs beltdh
  set peer 10.0.0.1
!
interface GigabitEthernet0/0
  ip address 192.168.2.1 255.255.255.0
!
interface GigabitEthernet0/1
  ip address 10.0.0.2 255.255.255.0
  crypto map CMAP
!
interface GigabitEthernet0/2
  no ip address
  shutdown
!
```

```
interface GigabitEthernet0/3
  no ip address
  shutdown
  !
  !
  ip route 0.0.0.0 0.0.0.0 10.0.0.1
  !
  crypto pki trustpoint s-terra_technological_trustpoint
    revocation-check none
  crypto pki certificate chain s-terra_technological_trustpoint
  certificate 4E4B0B11EFDB389E4E86244CDAA1B275
30820216308201C5A00302010202104E4B0B11EFDB389E4E86244CDAA1B27530
0806062A85030202033064310B3009060355040613025255310F300D06035504
0713064D6F73636F7731143012060355040A130B532D54657272612043535031
...
06010401823715010403020100300806062A85030202030341004A63F022F03D
009B097DD81A81CFC792664AAC9E6908587195AE17A5D526DE196CB0D5B7E713
E9D07F4DC61F04CDBC87579FC44CE66D524CF742F2784805733F
quit
!
end
```