Организация защищенного взаимодействия между подсетями центрального офиса и филиалов посредством IPsec VPN туннеля с аутентификацией на предварительно распределенных ключах (протокол IPv4)

Настоящий документ содержит описание способа совместного использования Продуктов компании ООО «С-Терра СиЭсПи» и Продуктов третьих производителей.
ООО «С-Терра СиЭсПи» осуществляет сопровождение настоящего сценария в части настроек Продуктов Компании. Упоминание наименований, продуктов, торговых марок третьих организаций исключительно неформально и не является поддержкой, рекомендацией либо рекламой. ООО «С-Терра СиЭсПи» не несет какой-либо ответственности в отношении работоспособности и использования этих Продуктов. Документ имеет статус вспомогательного материала, который может быть использован технологическими партнерами, компаниями-интеграторами, при разработке собственных решений.
Решения, разработанные на базе данного сценария, могут применяться в действующих сетях/системах только после тестовой и/или опытной эксплуатации.

Введение

Данный сценарий описывает настройку криптошлюзов «С-Терра Шлюз» для обеспечения безопасного взаимодействия между центральным офисом и двумя филиалами в условиях, когда криптошлюзы с обеих сторон находятся за NAT.

Защищаемые подсети в LAN сегменте и недоверенные подсети в WAN сегменте – IPv4. Обеспечение безопасного взаимодействия достигается путем шифрования и туннелирования трафика с применением отечественных отраслевых стандартов ГОСТ и протокола IPsec.

Партнеры по IPsec, в настоящем документе – криптошлюзы, для аутентификации будут использовать предварительно распределенные ключи. В качестве криптопровайдера будет использована криптографическая библиотека, разработанная компанией «С-Терра СиЭсПи». Шлюзы безопасности (или криптошлюзы) – «С-Терра Шлюз» версии 5.0.

Предварительные требования

Требования к материально-техническому обеспечению

Для переноса ключей между криптошлюзами требуется USB Flash накопитель.

Требования к квалификации администратора

Администратор должен обладать обширными знаниями в области сетевой информационной безопасности, иметь опыт работы с аналогичным оборудованием/программным обеспечением, знать и понимать следующие технологии и протоколы: PSK, IPsec, NAT, Firewall, routing, switching.

Требования к инфраструктуре

1. Требования к устройствам.
   1.1. «С-Терра Шлюз».

1.1.1. Устройства «С-Терра Шлюз» должны быть инициализированы (подробнее на https://doc.s-terra.ru раздел С-Терра Шлюз -> С-Терра Шлюз 5.0 -> «Подключение ПАК и инициализация С-Терра Шлюз на вычислительных системах архитектуры Intel x86-64»).

В «С-Терра Шлюз» версии 5.0 появилась возможность пройти инициализацию через интерфейс управления (management) по SSH, данная процедура и ее ограничения описаны в документации и на портале https://doc.s-terra.ru.

2. Требования к сетевому взаимодействию.

2.1. Между устройствами стенда должна быть обеспечена IP связность.

Схема взаимодействия

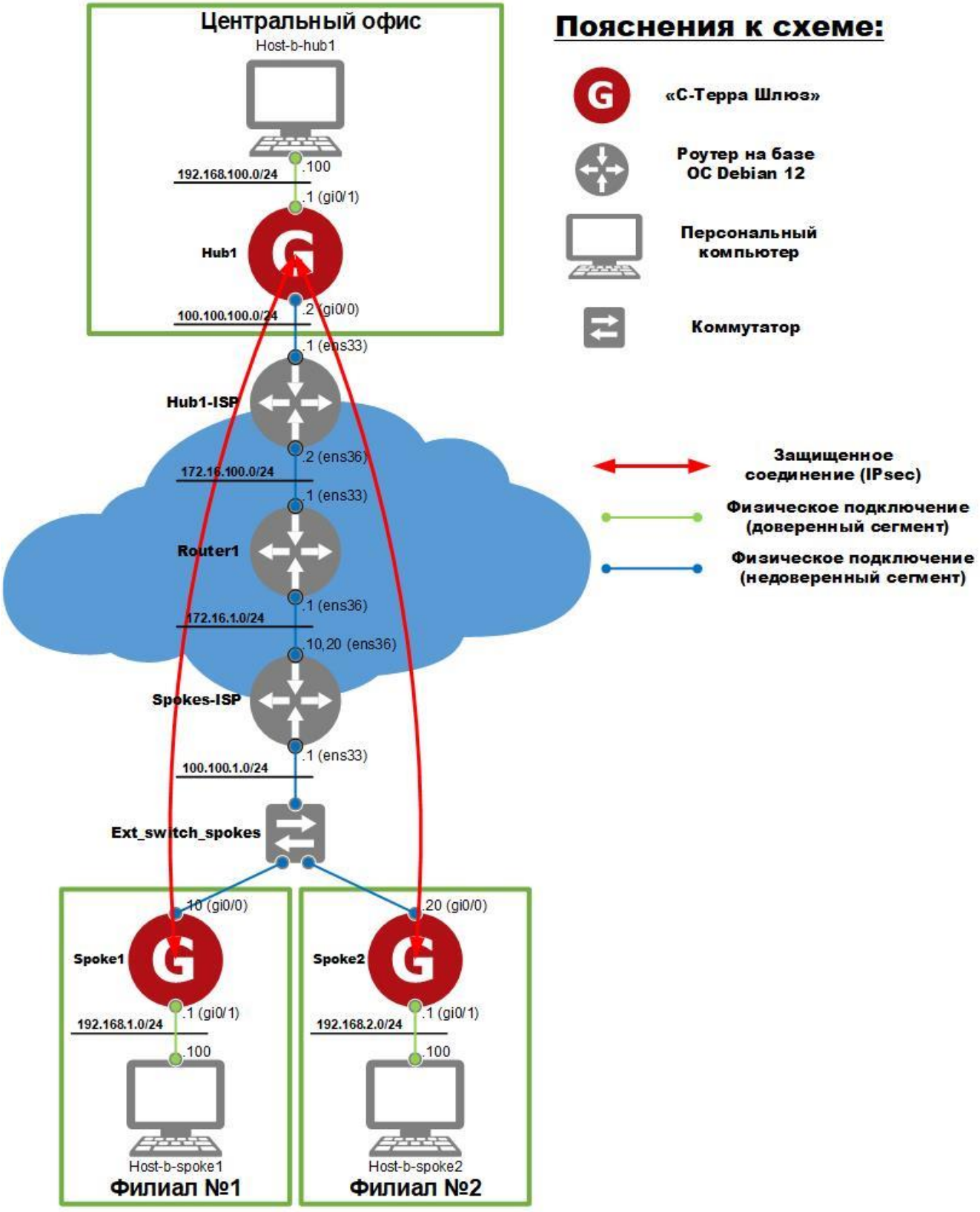


Рисунок 1. Схема взаимодействия

Общая логика работы

1. Размещение устройств.

   1.1. В центральном офисе размещаются: криптошлюз «С-Терра Шлюз» (Hub1) и персональный компьютер (Host-b-hub1):

   1.2. В филиале №1 размещаются: криптошлюз «С-Терра Шлюз» (Spoke1) и персональный компьютер (Host-b-spoke1).

   1.3. В филиале №2 размещаются: криптошлюз «С-Терра Шлюз» (Spoke2) и персональный компьютер (Host-b-spoke2).

   1.4. В неконтролируемом сегменте (синее облако на схеме) размещаются: маршрутизаторы (Router1, Hub1-ISP, Spokes-ISP) и коммутатор (Ext_switch_spokes).

2. Подключение к сети Интернет.

   В данном сценарии для эмуляции сети Интернет используются маршрутизаторы Router1, Hub1-ISP и Spokes-ISP.

   Подключение к сети Интернет на устройствах «С-Терра Шлюз» будет считаться успешным, если по протоколу ICMP (или «ping») будет доступен маршрутизатор Router1.

   2.1. Криптошлюз Hub1 подключаются к сети Интернет с помощью статической маршрутизации (маршрут по умолчанию через маршрутизатор Hub1-ISP).

   2.2. Криптошлюзы Spoke1 и Spoke2 подключаются к сети Интернет с помощью статической маршрутизации (маршрут по умолчанию через маршрутизатор Spokes-ISP).

3. Параметры NAT

   На маршрутизаторах Hub1-ISP и Spokes-ISP настроен статический DNAT (Destination NAT) и статический SNAT (Source NAT). При этом, на обоих маршрутизаторах трансляции адресов с помощью DNAT подвергаются только пакеты, относящиеся к IPsec трафику (ESP, UDP 500, UDP 4500).

   3.1. Настройка трансляции адресов на маршрутизаторе Hub1-ISP

      3.1.1. SNAT производит подмену адреса источника пакетов, исходящих с внешнего интерфейса маршрутизатора Hub1-ISP. Исходный адрес источника пакетов (100.100.100.2) заменяется на адрес внешнего интерфейса маршрутизатора Hub1-ISP (172.16.100.2).

      3.1.2. DNAT производит подмену адреса назначения пакетов, входящих на внешний интерфейс маршрутизатора Hub1-ISP. Исходный адрес назначения пакетов (17.16.100.2) заменяется на адрес внешнего интерфейса криптошлюза Hub1 (100.100.100.2).

   3.2. Настройка трансляции адресов на маршрутизаторе Spokes-ISP

      3.2.1. SNAT производит подмену адреса источника пакетов, исходящих с внешнего интерфейса маршрутизатора Spokes-ISP. Исходный адрес источника пакетов (внешний интерфейс криптошлюза Spoke1 - 100.100.1.10) заменяется на адрес внешнего интерфейса маршрутизатора Spokes-ISP (172.16.1.10). Аналогично исходный адрес источника пакетов (адрес внешнего интерфейса криптошлюза Spoke2 - 100.100.1.20), заменяется на вторичный адрес внешнего интерфейса маршрутизатора Spokes-ISP (172.16.1.20).

      3.2.2. DNAT производит подмену адреса назначения пакетов, входящих на внешний интерфейс маршрутизатора Spokes-ISP. Исходный адрес назначения пакетов (адрес внешнего интерфейса маршрутизатора Spokes-ISP - 172.16.1.10) заменяется на адрес внешнего интерфейса криптошлюза Spoke1 (100.100.1.10). Аналогично исходный адрес назначения пакетов (вторичный адрес внешнего интерфейса маршрутизатора Spokes-ISP - 172.16.1.20) заменяется на адрес внешнего интерфейса криптошлюза Spoke2 (100.100.1.20).

При аутентификации на предопределенных ключах, если количество филиалов больше одного и они находятся за NAT, то построение IPsec соединения возможно только если NAT является статическим, использование динамического DNAT/SNAT недопустимо.

4. Параметры межсетевого экранирования.

   4.1. Настройка межсетевого экрана не осуществляется

5. Параметры шифрования IPsec.

   Весь IPv4 трафик между подсетью центрального офиса (192.168.100.0/24) и филиалами (192.168.1.0/24 и 192.168.2.0/24) защищается с использованием алгоритмов ГОСТ и протокола IPsec в туннельном IPv4 режиме.

   Инициировать защищенное соединение между криптошлюзами центрального офиса и филиалов может как трафик, идущий из филиалов в центр, так и трафик, идущий из центра в любой из филиалов.

   Аутентификация партнеров происходит при помощи предварительно распределенных ключей. Для идентификации партнеров используются адреса криптошлюзов.

   5.1. Параметры протокола IKE:

   - Аутентификация при помощи предварительно распределенных ключей;
   - Алгоритм шифрования – ГОСТ Р 34.12-2015 Магма (ключ 256 бит);
   - Алгоритм вычисления хеш-функции – ГОСТ Р 34.11-2012 ТК26 Стрибог (ключ 256 бит);
   - Алгоритм выработки общего ключа (аналог алгоритма Диффи-Хеллмана) – VKO GOST R 34.10-2012 (ключ 256 бит).

   5.2. Параметры протокола ESP:

   - Комбинированный алгоритм шифрования и имитозащиты (контроль целостности) – ГОСТ Р 34.12-2015 Магма в режиме MGM (ключ 256 бит).

Настройка стенда

Настройка устройства Router1

1. Настройте IP адрес – 172.16.100.1 и маску – 255.255.255.0 на сетевом интерфейсе ens33.
2. Настройте IP адрес – 172.16.1.1 и маску – 255.255.255.0 на сетевом интерфейсе ens36.
3. Разрешите прохождение IP трафика (если заблокировано).

Настройка устройства Hub1-ISP

1. Настройте IP адрес – 172.16.100.2 и маску – 255.255.255.0 на сетевом интерфейсе ens36.
2. Настройте IP адрес – 100.100.100.1 и маску – 255.255.255.0 на сетевом интерфейсе ens33.
3. Задайте маршрут по умолчанию через 172.16.100.1.
4. Настройте статический Source NAT на сетевом интерфейсе ens36. Пример команды для iptables:

```
iptables -t nat -A POSTROUTING -s 100.100.100.2/32 -o ens36 -j SNAT --to-source 172.16.100.2
```

Настройте статический Destination NAT для IPsec трафика (UDP 500, UDP 4500) на сетевом интерфейсе ens36. Пример команды для iptables:

```
iptables -t nat -A PREROUTING –p udp –m multiport --dport 500,4500 -d 172.16.100.2/32 -i ens36 -j DNAT --to-destination 100.100.100.2
```

5. Разрешите прохождение IP трафика (если заблокировано).

Настройка устройства Spokes-ISP

1. Настройте IP адреса – 172.16.1.10, 172.16.1.20 и маску – 255.255.255.0 на сетевом интерфейсе ens36.
2. Настройте IP адрес – 100.100.1.1 и маску – 255.255.255.0 на сетевом интерфейсе ens33.
3. Задайте маршрут по умолчанию через 172.16.1.1.
4. Настройте статический Source NAT на сетевом интерфейсе ens36. Пример команды для iptables:

```
iptables -t nat -A POSTROUTING -s 100.100.1.10 -o ens36 -j SNAT --to-source 172.16.1.10
iptables -t nat -A POSTROUTING -s 100.100.1.20 -o ens36 -j SNAT --to-source 172.16.1.20
```

5. Настройте статический Destination NAT для IPsec трафика (UDP 500, UDP 4500) на сетевом интерфейсе ens36. Пример команды для iptables:

```
iptables -t nat -A PREROUTING -p udp -m multiport --dport 500,4500 -d 172.16.1.10/32
-i ens36 -j DNAT --to-destination 100.100.1.10
iptables -t nat -A PREROUTING -p udp -m multiport --dport 500,4500 -d 172.16.1.20/32
-i ens36 -j DNAT --to-destination 100.100.1.20
```

6. Разрешите прохождение IP трафика (если заблокировано).

Настройка устройства Host-b-hub1

1. Настройте IP адрес – 192.168.100.100 и маску – 255.255.255.0 на сетевом интерфейсе.
2. Задайте маршрут по умолчанию через 192.168.100.1.
3. Разрешите прием и отправку ICMP пакетов.

Настройка устройства Host-b-spoke1

1. Настройте IP адрес – 192.168.1.100 и маску – 255.255.255.0 на сетевом интерфейсе.
2. Задайте маршрут по умолчанию через 192.168.1.1.
3. Разрешите прием и отправку ICMP пакетов.

Настройка устройства Host-b-spoke2

1. Настройте IP адрес – 192.168.2.100 и маску – 255.255.255.0 на сетевом интерфейсе.
2. Задайте маршрут по умолчанию через 192.168.2.1.
3. Разрешите прием и отправку ICMP пакетов.

Настройка криптошлюза Hub1

Настройка будет происходить локально при помощи консольного подключения.

Настройка может осуществляться и удаленно (по SSH), но исключительно по доверенному каналу связи. Доверенным каналом связи может считаться канал в пределах контролируемой зоны в случае отсутствия в нем нарушителя. Доверенным каналом связи также считается канал, защищенный при помощи протокола IPsec.

Начальные настройки

Дата и время на всех криптошлюзах должны быть одинаковы, так как для аутентификации используются контейнеры с предварительно распределенными ключами. Контейнеры фиксируют период действия ключа. Также, одинаковые дата и время на всей инфраструктуре облегчают поиск неисправностей по лог-файлам.

1. Войдите в CLI разграничения доступа. Для этого, после появления сообщения:

```
S-Terra administrative console
```

введите логин и пароль для CLI разграничения доступа:

Пользователь и пароль по умолчанию: `administrator`, `s-terra`. Обязательно смените пароль для пользователя `administrator` при помощи команды `change user password`.

```
login as: administrator
```

```
administrator's password:
administrator@sterragate]
```

CLI разграничения доступа используется для локальной аутентификации пользователя, прохождения инициализации и переключения в другие консоли криптошлюза.

2. Установите правильный тип терминала (для `putty` тип терминала `xterm`) и требуемую ширину (для удобства работы), например:

```
administrator@sterragate] terminal terminal-type xterm
administrator@sterragate] terminal width 150
```

3. Установите нужную временную зону и правильные дату и время на криптошлюзе, используя консоль linux bash. Для этого выполните следующие команды.

   3.1. Войдите в linux bash.

```
administrator@sterragate] system
```

```
Entering system shell...
```

   3.2. Установите нужную временную зону:

```
root@sterragate:~# dpkg-reconfigure tzdata
```

   3.3. Установите правильное время и дату (формат – месяц/день/год часы:минуты):

```
root@sterragate:~# date -s "07/04/2023 12:32"
```

```
Thu Jul  4 12:32:00 MSK 2023
```

4. Установите надежный пароль для пользователя root (под данным пользователем осуществляется доступ по SSH в linux bash), если не сделали это ранее:

```
root@sterragate:~# passwd
```

```
New password:
Retype new password:
passwd: password updated successfully
```

5. Выйдите из linux bash обратно в CLI разграничения доступа:

```
root@sterragate:~# exit
```

```
logout
Leaving system shell...
administrator@sterragate]
```

Начальные настройки завершены.

Создание и импорт предварительно распределенных ключей

Для аутентификации партнеров по IPsec будут использоваться предварительно распределенные ключи, созданные и введённые в соответствии с правилами пользования СКЗИ (появились в версии 5.0).

Создание предварительно распределенного ключа подразумевает создание файла-контейнера, в который кладутся маскированное тело ключа, полученное с датчика случайных чисел и время его валидности. Данные в этом файле-контейнере шифруются на основе введенного пароля, что обеспечивает защиту целостности и конфиденциальности.

Создание файла контейнера и генерация ключа

1. Создайте файл контейнера для соединения со шлюзом Spoke1. Задайте имя файла контейнера, пароль и его период действия. Для создания файла контейнера с именем `cont_file1`, паролем `russia` и периодом действия с 1.07.2024 до 1.10.2025 (15 месяцев) в linux bash введите следующую команду:

```
root@sterragate:~# genpskey create -kc /root/cont_file1 -kcp russia -tfrom "01.07.2024"
-tto "01.10.2025"
```

   где:

   - `-kc` – путь до файла контейнера. В данном случае контейнер будет расположен в директории `/root`
   - `-kcp` – пароль для контейнера

- `-tfrom` и `-tto` – параметры, указывающие период действия контейнера

Срок действия контейнеров для предварительно распределенных ключей не должен превышать 15 мес.

2. Для создания ключа будет запущен процесс выработки случайных данных (гаммы). Нажимайте предлагаемые клавиши. В результате, в командной строке выведется сообщение об успешном создании контейнера:

```
OK! created key container: /root/cont_file1
```

3. По аналогии создайте файл контейнера `cont_file2` и сгенерируйте ключ для соединения со шлюзом Spoke2. Для этого в linux bash введите следующую команду:

```
root@sterragate:~# genpskey create -kc cont_file2 -kcp russia -tfrom "01.07.2024" -tto
"01.10.2025"
```

4. Для просмотра содержания контейнеров введите следующие команды:

```
root@sterragate:~# cat cont_file1
```

```
vers=500
key_cont=EE0369E4192581673A8445335EC1F2712FC180E4967DD0B370A3129F249F47786E5681ABEA99
0EA3058453F877A0313C062E99F8EDC8C54FB816583833CD0551F3B8E172B0F506498DF9CEF1EC8ED1502
7F33F4F4936AF085FBE6AA791E617749B9A712FA29FEDD86329389A89270C0180822F36C392B41A76DBB0
8480E20213
key_body_hash_inv=8D228432D1580C7195EB8145845B7FB0A19E471485B0ACAFE006716F15945528
```

```
root@sterragate:~# cat cont_file2
```

```
vers=500
key_cont=02B9DB7E8245BC41DB13447B64ECD6D7C42F9BD0F8CAF6C6E6E93CA1C285FEA7F47E9C6C25A5
13FECA1A1F123E9F7776F99F45C58B63EB6F60AAF2AA249AAB3FF3D5A583335C955D74F62F6FF5FD8EDB6
431E08F11A8124F9C7FABDDAC2FBC42EA9931B4278FF70223E33995BA38BCD30CF9189B0503C58DDA3285
2DE62BB76F
key_body_hash_inv=92A39BDBEE0801BD506D8514F44E6C461AFE382968A27DA9C23C9411DABF4EAD
```

Импорт ключа через cisco-like консоль

1. Импортируйте полученный ключ из файла контейнера `cont_file1` в базу Продукта. Для этого в cisco-like консоли введите следующую команду:

```
sterragate#crypto key import name key_Hub1_Spoke1 container file:/root/cont_file1
```

```
Password:
```

Для импорта ключа из контейнера потребуется ввести пароль, заданный при создании файла контейнера.

2. По аналогии импортируйте в систему ключ из файла `cont_file2`.

```
sterragate#crypto key import name key_Hub1_Spoke2 container file:/root/cont_file2
```

```
Password:
```

3. Просмотрите импортированные в систему ключи. Для этого в linux bash введите команду `key_mgr show`:

```
root@sterragate:~# key_mgr show
```

```
Found 2 keys.
--[Key]--------------
Name          :  cs_preshared_key_Hub1_Spoke1
State Details:  Active
Valid from    :  01.07.2024
Valid to      :  01.10.2025
Key body size:  32
Content hash :  D82248231D85C01759BE185448B5F70B1AE97441580BCAFA0E6017F651495582
--[Key]--------------
```

```
Name         :  cs_preshared_key_Hub1_Spoke2
State Details:  Active
Valid from   :  01.07.2024
Valid to     :  01.10.2025
Key body size:  32
Content hash :  293AB9BDEE8010DB05D658414FE4C664A1EF8392862AD79A2CC34911ADFBE4DA
```

4. Через cisco-like консоль просмотреть импортированные ключи можно при помощи команды `show crypto isakmp key`:

```
sterragate#show crypto isakmp key
```

```
Name         : key_Hub1_Spoke1
State Details: Active
Valid from   : 01.07.2024
Valid to     : 01.10.2025
Key body size: 32
Content hash : D82248231D85C01759BE185448B5F70B1AE97441580BCAFA0E6017F651495582

Name         : key_Hub1_Spoke2
State Details: Active
Valid from   : 01.07.2024
Valid to     : 01.10.2025
Key body size: 32
Content hash : 293AB9BDEE8010DB05D658414FE4C664A1EF8392862AD79A2CC34911ADFBE4DA
```

Ключи, импортированные в базу продукта через cisco-like консоль, при просмотре ключей через linux bash будут иметь приставку `cs_preshared_`, указывающую на то что ключ был импортирован через cisco-like консоль.

Настройка сетевых параметров

1. Задайте имя устройства:

```
sterragate(config)#hostname Hub1
```

2. Включите внешний (`GigabitEthernet0/0`) и внутренний (`GigabitEthernet0/1`) интерфейсы:

```
Hub1(config)#interface range gigabitEthernet 0/0 - 1
Hub1(config-if-range)#no shutdown
Hub1(config-if-range)#exit
```

3. Настройте IP адреса на внешнем и внутренних интерфейсах в соответствии со схемой стенда:

```
Hub1(config)#interface gigabitEthernet 0/0
Hub1(config-if)#ip address 100.100.100.2 255.255.255.0
Hub1(config-if)#exit
Hub1(config)#interface gigabitEthernet 0/1
Hub1(config-if)#ip address 192.168.100.1 255.255.255.0
Hub1(config-if)#exit
```

4. Задайте маршрут по умолчанию через устройство Hub1-ISP:

```
Hub1(config)#ip route 0.0.0.0 0.0.0.0 100.100.100.1
```

Настройка шифрования

1. Настройте параметры IKE.

1.1. Задайте общие параметры IKE. В роли идентификатора в данном сценарии будут выступать адреса устройств:

```
Hub1(config)#crypto isakmp identity address
Hub1(config)#crypto isakmp keepalive 3 2
Hub1(config)#crypto isakmp keepalive retry-count 5
Hub1(config)#crypto isakmp fragmentation
Hub1(config)#crypto isakmp security-association lifetime delta 50
```

```
Hub1(config)#crypto isakmp initiator-sessions-max 100
Hub1(config)#crypto isakmp responder-sessions-max 100
```

1.2. Создайте политику, описывающую параметры IKE туннеля. В качестве метода аутентификации укажите параметр pre-share:

```
Hub1(config)#crypto isakmp policy 1
Hub1(config-isakmp)#encryption magma
Hub1(config-isakmp)#hash stribog-256
Hub1(config-isakmp)#authentication pre-share
Hub1(config-isakmp)#group vko2
Hub1(config-isakmp)#exit
```

1.3. Создайте связку ключей Hub1_Spoke1 и укажите в ней ранее импортированный ключ key_Hub1_Spoke1:

```
Hub1(config)#crypto keyring Hub1_Spoke1
Hub1(conf-keyring)#pre-shared-key key_Hub1_Spoke1
```

1.4. Создайте связку ключей Hub1_Spoke2 и добавьте в нее ранее импортированный ключ key_Hub1_Spoke2:

```
Hub1(config)#crypto keyring Hub1_Spoke2
Hub1(conf-keyring)#pre-shared-key key_Hub1_Spoke2
Hub1(conf-keyring)#exit
```

1.5. Создайте профиль настроек ISAKMP (алгоритм IKE) под именем PSK_Spoke1:

1.5.1. Укажите в качестве типа идентификатора партнера (identity) адрес криптошлюза Spoke1:

```
Hub1(conf-isa-prof)#match identity address ipv4 100.100.1.10
```

1.5.2. Прикрепите к профилю связку ключей:

```
Hub1(conf-isa-prof)#keyring Hub1_Spoke1
```

1.5.3. Укажите в качестве значения параметра самоидентификации адрес криптошлюза Hub1:

```
Hub1(conf-isa-prof)#self-identity address ipv4 100.100.100.2
```

1.5.4. Прикрепите к профилю созданную ранее политику безопасности:

```
Hub1(conf-isa-prof)#set policy 1
Hub1(conf-isa-prof)#exit
```

При создании профиля настроек ISAKMP параметр set identity mapping будет выставлен по умолчанию. При аутентификации на предварительно распределенных ключах этот параметр будет игнорироваться.

1.6. Создайте профиль настроек ISAKMP (алгоритм IKE) под именем PSK_Spoke2:

```
Hub1(config)# crypto isakmp profile PSK_Spoke2
Hub1(conf-isa-prof)#match identity address ipv4 100.100.1.10
Hub1(conf-isa-prof)#keyring Hub1_Spoke2
Hub1(conf-isa-prof)#self-identity address ipv4 100.100.100.2
Hub1(conf-isa-prof)#set policy 1
Hub1(conf-isa-prof)#exit
```

2. Настройте параметры IPsec.

1.1. Отключите обновление ключа IPsec SA по достижению определенного объема переданного трафика:

```
Hub1(config)#crypto ipsec security-association lifetime kilobytes unlimited
```

1.2. Задайте комбинированный алгоритм шифрования и имитозащиты (набор преобразований) для трафика:

```
Hub1(config)#crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
Hub1(cfg-crypto-trans)#exit
```

1.3. Создайте список доступа (ACL) для трафика, который нужно защищать между подсетью центрального офиса и филиалом №1:

```
Hub1(config)#ip access-list extended IPSEC_ACL_HUB1_AND_SPOKE1
Hub1(config-ext-nacl)#permit ip 192.168.100.0 0.0.0.255 192.168.1.0 0.0.0.255
Hub1(config-ext-nacl)#exit
```

1.4. Создайте список доступа (ACL) для трафика, который нужно защищать между подсетью центрального офиса и филиалом №2:

```
Hub1(config)#ip access-list extended IPSEC_ACL_HUB1_AND_SPOKE2
Hub1(config-ext-nacl)#permit ip 192.168.100.0 0.0.0.255 192.168.2.0 0.0.0.255
Hub1(config-ext-nacl)#exit
```

1.5. Создайте статическую крипто-карту (имя `MAP_PSK`, раздел 1) для построения IPsec туннеля между криптошлюзом центрального офиса и криптошлюзом Spoke1:

```
Hub1(config)# crypto map MAP_PSK 1 ipsec-isakmp
```

1.5.1 Укажите список доступа для защищаемого трафика:

```
Hub1(config-crypto-map)#match address IPSEC_ACL_HUB1_AND_SPOKE1
```

1.5.2 Укажите при помощи какого набора алгоритмов нужно защищать трафик:

```
Hub1(config-crypto-map)#set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
```

1.5.3 Укажите IPv4 адрес партнера по IPsec, для данного раздела криптокарты это внешний IPv4 адрес устройства Spokes-ISP:

```
Hub1(config-crypto-map)# set peer 172.16.1.10
```

1.5.4 Прикрепите ISAKMP профиль `PSK_Spoke1`:

```
Hub1(config-crypto-map)#set isakmp-profile PSK_Spoke1
```

1.5.5 Включите префрагментацию:

```
Hub1(config-crypto-map)#set pre-fragmentation
```

1.5.6 Включите автоматическую и интеллектуальную корректировку значения MSS:

```
Hub1(config-crypto-map)#set security-association mss-fix out
```

1.5.7 Обязательно отключите историю удаленных туннелей:

```
Hub1(config-crypto-map)#set dead-connection history off
Hub1(config-crypto-map)#exit
```

1.6. Создайте статическую крипто-карту (имя `MAP_PSK`, раздел 2) для построения IPsec туннеля между криптошлюзом центрального офиса и криптошлюзом Spoke2:

```
Hub1(config)# crypto map MAP_PSK 2 ipsec-isakmp
Hub1(config-crypto-map)#match address IPSEC_ACL_HUB1_AND_SPOKE2
Hub1(config-crypto-map)#set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
Hub1(config-crypto-map)#set isakmp-profile PSK_Spoke2
Hub1(config-crypto-map)#set security-association mss-fix out
Hub1(config-crypto-map)#set peer 172.16.1.20
Hub1(config-crypto-map)#set pre-fragmentation
Hub1(config-crypto-map)#set dead-connection history off
Hub1(config-crypto-map)#exit
```

1.7. Прикрепите крипто-карту `MAP_PSK` к внешнему интерфейсу `GigabitEthernet0/0`:

```
Hub1(config)#interface GigabitEthernet0/0
Hub1(config-if)#crypto map MAP_PSK
Hub1(config-if)#exit
```

1.8. Примените настройки:

```
Hub1(config)#end
```

Подробное описание всех команд приведено в сценарии «Организация защищенного взаимодействия между подсетями центрального офиса и филиалов посредством IPsec VPN туннеля», который находится в документации на портале https://doc.s-terra.ru.

Настройка криптошлюза Hub1 завершена.

В Приложении представлены тексты конфигураций для криптошлюза Hub1:

- текст конфигурации консоли cisco-like;
- Текст LSP;

Настройка криптошлюза Spoke1

Настройка будет происходить локально при помощи консольного подключения.

Настройка может осуществляться и удаленно (по SSH), но исключительно по доверенному каналу связи. Доверенным каналом связи может считаться канал в пределах контролируемой зоны в случае отсутствия в нем нарушителя. Доверенным каналом связи также считается канал, защищенный при помощи протокола IPsec.

Начальные настройки

Войдите в CLI разграничения доступа (пользователь и пароль по умолчанию: `administrator`, `s-terra`) установите правильный тип терминала и требуемую ширину, смените пароль для пользователя `administrator`.

Перейдите в linux bash консоль, установите пароль для пользователя `root`, правильное время и временную зону.

Подробное описание настроек приведено в разделе Начальные настройки криптошлюза Hub1.

Создание и импорт предварительно распределенных ключей

Перенос ключа между устройствами

Для аутентификации партнеров по IPsec в данном сценарии используются предварительно распределенные ключи, созданные и введённые в соответствии с правилами пользования СКЗИ (появились в версии 5.0).

1. Созданный на шлюзе Hub1 файл-контейнер перенесите при помощи USB Flash накопителя с Hub1 на Spoke1, сохраните в системе.

   1.1. Вставьте USB Flash накопитель в свободный USB порт криптошлюза Hub1 (накопитель будет автоматически примонтирован).

   1.2. Определите имя (идентификатор) USB Flash накопителя на криптошлюзе.

   ```
   administrator@Hub1] dir media:
       1  drwx        16384  Thu Jan  1 03:00:00 1970  USBDISK
   ```

   1.3. Перейдите в linux bash консоль

   ```
   administrator@Hub1]system
   ```

   1.4. Скопируйте файл-контейнер `/root/cont_file1` с устройства Hub1 на USB Flash накопитель:

   ```
   root@Hub1:~# cp /root/cont_file1 /media/USBDISK/cont_file1
   ```

   1.5. Отмонтируйте и извлеките USB Flash накопитель (посмотреть точки монтирования можно при помощи команды `mount`):

   ```
   root@Hub1:~#umount /media/USBDISK
   ```

   1.6. Вставьте USB Flash накопитель в свободный USB порт криптошлюза Spoke1 (накопитель будет автоматически примонтирован).

   1.7. Определите имя (идентификатор) USB Flash накопителя на криптошлюзе:

   ```
   administrator@sterragate] dir media
       1  drwx        16384  Thu Jan  1 03:00:00 1970  USBDISK
   ```

   1.8. Перейдите в linux bash консоль

   ```
   administrator@sterragate]system
   ```

   1.9. Скопируйте файл-контейнер `/media/USBDISK/cont_file1` с USB Flash накопителя на устройство Spoke1 в папку `/root`:

   ```
   root@sterragate:~# cp /media/USBDISK/cont_file1 /root/cont_file1
   ```

1.10. Отмонтируйте и извлеките USB Flash накопитель (посмотреть точки монтирования можно при помощи команды `mount`):

```
root@sterragate:~#umount /media/USBDISK
```

Импорт ключа через cisco-like консоль

1. Импортируйте ключ из файла контейнера `file_cont1`.

```
sterragate#crypto key import name key_Hub1_Spoke1 container file:/root/cont_file1
Password:
```

Для импорта ключа из контейнера потребуется ввести пароль, заданный при создании файла контейнера.

2. Через cisco-like консоль просмотреть импортированные ключи можно при помощи команды `show crypto isakmp key`:

```
sterragate#show crypto isakmp key
Name          : key_Hub1_Spoke1
State Details: Active
Valid from    : 01.07.2024
Valid to      : 01.10.2025
Key body size: 32
Content hash : D82248231D85C01759BE185448B5F70B1AE97441580BCAFA0E6017F651495582
```

Настройка сетевых параметров и шифрования криптошлюза Spoke1 происходит аналогично настройке Hub1.

В Приложении представлены тексты конфигураций для криптошлюза Spoke1:

- текст конфигурации консоли cisco-like;
- текст LSP;

Настройка криптошлюза Spoke2

Настройка криптошлюза Spoke2 происходит аналогично настройке Spoke1.

В Приложении представлены тексты конфигураций для криптошлюза Spoke2:

- текст консоли cisco-like;
- текст LSP;

Проверка работоспособности стенда

Проверка IP связности

1. Проверьте, что с криптошлюзов Hub1, Spoke1 и Spoke2 доступен по протоколу ICMP маршрутизатор Router1. Для этого выполните команду `ping` из cisco-like консоли криптошлюзов:

```
Hub1#ping 172.16.100.1
PING 172.16.100.1 (172.16.100.1) 100(128) bytes of data.
108 bytes from 172.16.100.1: icmp_seq=1 ttl=63 time=0.729 ms
108 bytes from 172.16.100.1: icmp_seq=2 ttl=63 time=0.805 ms
108 bytes from 172.16.100.1: icmp_seq=3 ttl=63 time=0.491 ms
108 bytes from 172.16.100.1: icmp_seq=4 ttl=63 time=0.328 ms
108 bytes from 172.16.100.1: icmp_seq=5 ttl=63 time=0.648 ms

--- 172.16.100.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4089ms
rtt min/avg/max/mdev = 0.328/0.600/0.805/0.171 ms
```

```
Spoke1#ping 172.16.1.1
PING 172.16.1.1 (172.16.1.1) 100(128) bytes of data.
108 bytes from 172.16.1.1: icmp_seq=1 ttl=63 time=0.624 ms
```

```
108 bytes from 172.16.1.1: icmp_seq=2 ttl=63 time=0.716 ms
108 bytes from 172.16.1.1: icmp_seq=3 ttl=63 time=0.709 ms
108 bytes from 172.16.1.1: icmp_seq=4 ttl=63 time=0.606 ms
108 bytes from 172.16.1.1: icmp_seq=5 ttl=63 time=0.905 ms

--- 172.16.1.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4079ms
rtt min/avg/max/mdev = 0.606/0.712/0.905/0.106 ms
```

```
Spoke2#ping 172.16.1.1
```

```
PING 172.16.1.1 (172.16.1.1) 100(128) bytes of data.
108 bytes from 172.16.1.1: icmp_seq=1 ttl=63 time=0.594 ms
108 bytes from 172.16.1.1: icmp_seq=2 ttl=63 time=0.441 ms
108 bytes from 172.16.1.1: icmp_seq=3 ttl=63 time=0.474 ms
108 bytes from 172.16.1.1: icmp_seq=4 ttl=63 time=0.599 ms
108 bytes from 172.16.1.1: icmp_seq=5 ttl=63 time=0.708 ms

--- 172.16.1.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4086ms
rtt min/avg/max/mdev = 0.441/0.563/0.708/0.096 ms
```

Видно, что устройство Router1 доступно по ICMP со всех криптошлюзов.

2. Проверьте, что с криптошлюзов Spoke1 и Spoke2 доступен по протоколу ICMP адрес партнера по IPsec. Для этого из cisco-like консоли каждого из криптошлюзов выполните команду `ping` до внешнего интерфейса маршрутизатора Hub1-ISP:

```
Spoke1#ping 172.16.100.2
```

```
PING 172.16.100.2 (172.16.100.2) 100(128) bytes of data.
108 bytes from 172.16.100.2: icmp_seq=1 ttl=61 time=1.05 ms
108 bytes from 172.16.100.2: icmp_seq=2 ttl=61 time=1.25 ms
108 bytes from 172.16.100.2: icmp_seq=3 ttl=61 time=1.10 ms
108 bytes from 172.16.100.2: icmp_seq=4 ttl=61 time=1.14 ms
108 bytes from 172.16.100.2: icmp_seq=5 ttl=61 time=1.38 ms

--- 172.16.100.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 1.051/1.182/1.375/0.116 ms
```

```
Spoke2#ping 172.16.100.2
```

```
PING 172.16.100.2 (172.16.100.2) 100(128) bytes of data.
108 bytes from 172.16.100.2: icmp_seq=1 ttl=61 time=1.02 ms
108 bytes from 172.16.100.2: icmp_seq=2 ttl=61 time=1.00 ms
108 bytes from 172.16.100.2: icmp_seq=3 ttl=61 time=1.00 ms
108 bytes from 172.16.100.2: icmp_seq=4 ttl=61 time=1.25 ms
108 bytes from 172.16.100.2: icmp_seq=5 ttl=61 time=1.05 ms

--- 172.16.100.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 1.000/1.064/1.252/0.095 ms
```

Видно, что криптошлюзам Spoke1 и Spoke2 доступны по протоколу icmp их партнеры по защищенному соединению.

3. Проверьте, что с криптошлюза Hub1 доступны по протоколу ICMP адреса партнеров по IPsec. Для этого из cisco-like консоли криптошлюза Hub1 выполните команды `ping` до обоих адресов внешнего интерфейса маршрутизатора Spokes-ISP:

```
Hub1#ping 172.16.1.10
```

```
PING 172.16.1.10 (172.16.1.10) 100(128) bytes of data.
108 bytes from 172.16.1.10: icmp_seq=1 ttl=62 time=0.900 ms
108 bytes from 172.16.1.10: icmp_seq=2 ttl=62 time=1.62 ms
108 bytes from 172.16.1.10: icmp_seq=3 ttl=62 time=1.22 ms
```

```
108 bytes from 172.16.1.10: icmp_seq=4 ttl=62 time=1.24 ms
108 bytes from 172.16.1.10: icmp_seq=5 ttl=62 time=1.27 ms

--- 172.16.1.10 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 0.900/1.251/1.621/0.228 ms
```

```
Hub1#ping 172.16.1.20
```

```
PING 172.16.1.20 (172.16.1.20) 100(128) bytes of data.
108 bytes from 172.16.1.20: icmp_seq=1 ttl=62 time=0.851 ms
108 bytes from 172.16.1.20: icmp_seq=2 ttl=62 time=0.863 ms
108 bytes from 172.16.1.20: icmp_seq=3 ttl=62 time=0.943 ms
108 bytes from 172.16.1.20: icmp_seq=4 ttl=62 time=0.830 ms
108 bytes from 172.16.1.20: icmp_seq=5 ttl=62 time=0.987 ms

--- 172.16.1.20 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4035ms
rtt min/avg/max/mdev = 0.830/0.894/0.987/0.059 ms
```

Видно, что криптошлюзу Hub1 доступен по протоколу icmp его партнер по защищенному соединению.

4. Проверьте, что с защищаемых устройств Host-b-hub1, Host-b-spoke1, Host-b-spoke2 доступен по протоколу ICMP соответствующий криптошлюз, который является шлюзом по умолчанию . Для этого выполните команду `ping` из консоли защищаемых устройств:

```
root@Host-b-hub1:~# ping 192.168.100.1 -c 4
```

```
PING 192.168.100.1 (192.168.100.1) 56(84) bytes of data.
64 bytes from 192.168.100.1: icmp_seq=1 ttl=64 time=0.551 ms
64 bytes from 192.168.100.1: icmp_seq=2 ttl=64 time=0.723 ms
64 bytes from 192.168.100.1: icmp_seq=3 ttl=64 time=0.569 ms
64 bytes from 192.168.100.1: icmp_seq=4 ttl=64 time=0.384 ms

--- 192.168.100.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3072ms
rtt min/avg/max/mdev = 0.384/0.556/0.723/0.120 ms
```

```
root@Host-b-spoke1:~# ping 192.168.1.1 -c 4
```

```
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.368 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.620 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.560 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=0.431 ms

--- 192.168.1.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3037ms
rtt min/avg/max/mdev = 0.368/0.494/0.620/0.100 ms
```

```
root@Host-b-spoke2:~# ping 192.168.2.1 -c 4
```

```
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data.
64 bytes from 192.168.2.1: icmp_seq=1 ttl=64 time=0.497 ms
64 bytes from 192.168.2.1: icmp_seq=2 ttl=64 time=0.593 ms
64 bytes from 192.168.2.1: icmp_seq=3 ttl=64 time=0.446 ms
64 bytes from 192.168.2.1: icmp_seq=4 ttl=64 time=0.618 ms

--- 192.168.2.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3077ms
rtt min/avg/max/mdev = 0.446/0.538/0.618/0.069 ms
```

Видно, что со всех защищаемых устройств соответствующие криптошлюзы доступны по ICMP.

Проверка аутентификации на предварительно распределенных ключах

1. Проверьте, что предварительно распределенные ключи партнеров Hub1 и Spoke1 совпадают. Для этого сравните содержание файл-контейнеров Hub1 и Spoke1 - из linux bash криптошлюзов выполните следующую команду:

   ```
   Hub1#show crypto isakmp key
   ```

   ```
   Name         : key_Hub1_Spoke1
   State Details: Active
   Valid from   : 01.07.2024
   Valid to     : 01.10.2025
   Key body size: 32
   Content hash : D82248231D85C01759BE185448B5F70B1AE97441580BCAFA0E6017F651495582

   Name         : key_Hub1_Spoke2
   State Details: Active
   Valid from   : 01.07.2024
   Valid to     : 01.10.2025
   Key body size: 32
   Content hash : 293AB9BDEE8010DB05D658414FE4C664A1EF8392862AD79A2CC34911ADFBE4DA
   ```

   ```
   Spoke1#show crypto isakmp key
   ```

   ```
   Name         : key_Hub1_Spoke1
   State Details: Active
   Valid from   : 01.07.2024
   Valid to     : 01.10.2025
   Key body size: 32
   Content hash : D82248231D85C01759BE185448B5F70B1AE97441580BCAFA0E6017F651495582
   ```

   Видно, что ключи совпадают. Если ключи не совпали, следует повторно осуществить перенос соответствующего ключа с Hub1 на Spoke1 и его импорт.

2. Проверьте, что предварительно распределенные ключи партнеров Hub1 и Spoke2 совпадают. Для этого сравните содержание файл-контейнеров Hub1 и Spoke2 - из linux bash криптошлюзов выполните следующую команду:

   ```
   Hub1#show crypto isakmp key
   ```

   ```
   Name         : key_Hub1_Spoke1
   State Details: Active
   Valid from   : 01.07.2024
   Valid to     : 01.10.2025
   Key body size: 32
   Content hash : D82248231D85C01759BE185448B5F70B1AE97441580BCAFA0E6017F651495582

   Name         : key_Hub1_Spoke2
   State Details: Active
   Valid from   : 01.07.2024
   Valid to     : 01.10.2025
   Key body size: 32
   Content hash : 293AB9BDEE8010DB05D658414FE4C664A1EF8392862AD79A2CC34911ADFBE4DA
   ```

   ```
   Spoke2#show crypto isakmp key
   ```

   ```
   Name         : key_Hub1_Spoke2
   State Details: Active
   Valid from   : 01.07.2024
   Valid to     : 01.10.2025
   Key body size: 32
   Content hash : 293AB9BDEE8010DB05D658414FE4C664A1EF8392862AD79A2CC34911ADFBE4DA
   ```

Видно, что ключи совпадают. Если ключи не совпали, следует повторно осуществить перенос соответствующего ключа с Hub1 на Spoke2 и его импорт.

Проверка построения защищенного соединения

1. Инициируйте построение защищенного соединения между криптошлюзом Hub1 и криптошлюзом Spoke1. Для этого отправьте ICMP запросы с устройства Host-b-hub1 на устройство Host-b-spoke1. Аналогичным образом инициируйте построение защищенного соединения между криптошлюзом Hub1 и криптошлюзом Spoke2 за счет отправки ICMP запросов с устройства Host-b-hub1 на устройство Host-b-spoke2.

   Для просмотра установленных соединений выполните следующую команду из linux bash криптошлюзов:

   ```
   root@Hub1:~# sa_mgr show
   ```

   ```
   ISAKMP sessions: 0 initiated, 0 responded

   ISAKMP connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
   1 1 (100.100.100.2,4500)-(172.16.1.10,4500) active 1510 1626
   2 2 (100.100.100.2,4500)-(172.16.1.20,4500) active 1142 1174

   IPsec connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
   1 2 (192.168.100.0-192.168.100.255,*)-(192.168.1.0-192.168.1.255,*) * ESP nat-t-tunn
   2464 2464
   2 3 (192.168.100.0-192.168.100.255,*)-(192.168.2.0-192.168.2.255,*) * ESP nat-t-tunn
   1672 1672
   ```

   ```
   root@Spoke1:~# sa_mgr show
   ```

   ```
   ISAKMP sessions: 0 initiated, 0 responded

   ISAKMP connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
   1 1 (100.100.1.10,4500)-(172.16.100.2,4500) active 1626 1510

   IPsec connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
   1 2 (192.168.1.0-192.168.1.255,*)-(192.168.100.0-192.168.100.255,*) * ESP nat-t-tunn
   9064 9064
   ```

   ```
   root@Spoke2:~# sa_mgr show
   ```

   ```
   ISAKMP sessions: 0 initiated, 0 responded

   ISAKMP connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
   1 1 (100.100.1.20,4500)-(172.16.100.2,4500) active 1174 1142

   IPsec connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
   1 1 (192.168.2.0-192.168.2.255,*)-(192.168.100.0-192.168.100.255,*) * ESP nat-t-tunn
   11792 11792
   ```

   Видно, что между криптошлюзами Hub1, Spoke1 и Hub1, Spoke2 установлено защищенное IPsec соединение и количество полученных и отправленных пакетов изменилось на счётчиках обоих IPsec соединений.

   Если защищенные соединения не построились, проверьте файл журнала `/var/log/cspvpngate.log` (при необходимости увеличьте уровень логирования командой `logging trap debugging` в cisco-like консоли).

Приложение

Конфигурации криптошлюза Hub1

1. Консоль cisco-like:

```
!
version 12.4
!
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity address
crypto isakmp fragmentation
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto ipsec security-association lifetime kilobytes unlimited
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $y$j9T$NOQlZVBX/PvYoVmlqZqAR/$HaY00aXA/Mm
jer4nG8nPZwF5.k/opuO.qHe6lO.MlC2
aaa new-model
!
!
hostname Hub1
enable secret 5 JLI9qp52IVCYdKnFL5Xk/A==
!
!
crypto keyring Hub1_Spoke1
 pre-shared-key key_Hub1_Spoke1
crypto keyring Hub1_Spoke2
 pre-shared-key key_Hub1_Spoke2
!
!
!
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication pre-share
 group vko2
!
crypto isakmp profile PSK_Spoke1
 match identity address ipv4 100.100.1.10
 keyring Hub1_Spoke1
 self-identity address ipv4 100.100.100.2
 set identity mapping
 set policy 1
!
crypto isakmp profile PSK_Spoke2
 match identity address ipv4 100.100.1.20
 keyring Hub1_Spoke2
 self-identity address ipv4 100.100.100.2
 set identity mapping
 set policy 1
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
ip access-list extended IPSEC_ACL_HUB1_AND_SPOKE1
```

```
 permit ip 192.168.100.0 0.0.0.255 192.168.1.0 0.0.0.255
!
ip access-list extended IPSEC_ACL_HUB1_AND_SPOKE2
 permit ip 192.168.100.0 0.0.0.255 192.168.2.0 0.0.0.255
!
ipv6 default filtering deny
!
!
crypto map MAP_PSK 1 ipsec-isakmp
 match address IPSEC_ACL_HUB1_AND_SPOKE1
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile PSK_Spoke1
 set security-association mss-fix out
 set peer 172.16.1.10
 set pre-fragmentation
 set dead-connection history off
crypto map MAP_PSK 2 ipsec-isakmp
 match address IPSEC_ACL_HUB1_AND_SPOKE2
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile PSK_Spoke2
 set security-association mss-fix out
 set peer 172.16.1.20
 set pre-fragmentation
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 100.100.100.2 255.255.255.0
 crypto map MAP_PSK
!
interface GigabitEthernet0/1
 ip address 192.168.100.1 255.255.255.0
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
!
ip route 0.0.0.0 0.0.0.0 100.100.100.1
!
!
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Tue Aug 20 16:59:32 2024

GlobalParameters(
    Title                     = "This LSP was automatically generated by CSP Converter
at Tue Aug 20 16:59:32 2024 (user: cscons)"
    Version                     = LSP_5_0
    PreserveIPsecSA             = FALSE
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 100.100.100.1
        )

)
```

```
FirewallParameters(
    TCPSynSentTimeout = 30
    TCPFinTimeout = 5
    TCPClosedTimeout = 30
    TCPSynRcvdTimeout = 30
    TCPEstablishedTimeout = 3600
    TCPHalfOpenLow = 400
    TCPHalfOpenMax = 500
    TCPSessionRateLow = 400
    TCPSessionRateMax = 500
    TCPSessionsMax = 65536
    TCPStrictnessLevel = 3
    UDPNotEstablishedTimeout = 30
    UDPEstablishedTimeout = 120
    UDPSessionsMax = 0
)

IKETransform crypto:isakmp:policy:1
(
    CipherAlg   = "GR341215M-K256-CFB-65526"
    HashAlg     = "GR341112_256TC26-65128"
    GroupID     = VKO2_1B
    RestrictAuthenticationTo = PRESHARED
    LifetimeSeconds = 86400
)

ESPProposal MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP
(
    Transform* = ESPTransform
    (
        CipherAlg*          = "GR341215M-K288-MGM-33"
        LifetimeSeconds     = 3600
    )
)

FilterChain IPv6Default (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        Action = PASS
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

IdentityEntry Local:Preshared:PSK_Spoke1(
    IPv4Address = 100.100.100.2
)

IdentityEntry Remote:PSK_Spoke1(
    IPv4Address = 100.100.1.10
)

IdentityEntry Local:Preshared:PSK_Spoke2(
    IPv4Address = 100.100.100.2
)

IdentityEntry Remote:PSK_Spoke2(
    IPv4Address = 100.100.1.20
)
```

```
IKEParameters(
    FragmentSize = 576
    SALifetimeDelta = 50
    InitiatorSessionsMax = 100
    ResponderSessionsMax = 100
)

AuthMethodPreshared Preshared:PSK_Spoke1
(
    LocalID  = Local:Preshared:PSK_Spoke1
    RemoteID  = Remote:PSK_Spoke1
    SharedIKESecret = "cs_preshared_key_Hub1_Spoke1"
)

AuthMethodPreshared Preshared:PSK_Spoke2
(
    LocalID  = Local:Preshared:PSK_Spoke2
    RemoteID  = Remote:PSK_Spoke2
    SharedIKESecret = "cs_preshared_key_Hub1_Spoke2"
)

IKERule IKERule:MAP_PSK:1
(
    IKEPeerIPFilter = 172.16.1.10
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = Preshared:PSK_Spoke1
    MainModeAuthMethod  = Preshared:PSK_Spoke1
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

IKERule IKERule:MAP_PSK:2
(
    IKEPeerIPFilter = 172.16.1.20
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = Preshared:PSK_Spoke2
    MainModeAuthMethod  = Preshared:PSK_Spoke2
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 20
)

IPsecAction IPsecAction:MAP_PSK:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.1.10
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule:MAP_PSK:1
)

IPsecAction IPsecAction:MAP_PSK:2
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.1.20
```

```
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule:MAP_PSK:2
)

FilterChain IPsecPolicy:MAP_PSK (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 192.168.100.0/24
        DestinationIP = 192.168.1.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:MAP_PSK:1 >
        LogEventID = "IPsec:Protect:MAP_PSK:1:IPSEC_ACL_HUB1_AND_SPOKE1"
    ),
    Filter (
        Label = "MAP_PSK:2:IPSEC_ACL_HUB1_AND_SPOKE2"
        SourceIP = 192.168.100.0/24
        DestinationIP = 192.168.2.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:MAP_PSK:2 >
        LogEventID = "IPsec:Protect:MAP_PSK:2:IPSEC_ACL_HUB1_AND_SPOKE2"
    )
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/0"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
    IPsecPolicy = IPsecPolicy:MAP_PSK
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/1"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/2"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)
```

Конфигурации криптошлюза Spoke1

1. Консоль cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity address
```

```
crypto isakmp fragmentation
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto ipsec security-association lifetime kilobytes unlimited
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $y$j9T$bAS4lGk7Z.u6ywISPF5PT.$mG4W8fo.vRH
CztUaO/wwU/MLacW7MCKYTV.BSy7xeO4
aaa new-model
!
!
hostname Spoke1
enable secret 5 JLI9qp52IVCYdKnFL5Xk/A==
!
crypto keyring Hub1_Spoke1
 pre-shared-key key_Hub1_Spoke1
!
!
!
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication pre-share
 group vko2
!
crypto isakmp profile PSK_Spoke1
 match identity address ipv4 100.100.100.2
 keyring Hub1_Spoke1
 self-identity address ipv4 100.100.1.10
 set identity mapping
 set policy 1
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
ip access-list extended IPSEC_ACL_HUB1_AND_SPOKE1
 permit ip 192.168.1.0 0.0.0.255 192.168.100.0 0.0.0.255
!
ipv6 default filtering deny
!
!
crypto map MAP_PSK 1 ipsec-isakmp
 match address IPSEC_ACL_HUB1_AND_SPOKE1
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile PSK_Spoke1
 set security-association mss-fix out
 set peer 172.16.100.2
 set pre-fragmentation
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 100.100.1.10 255.255.255.0
 crypto map MAP_PSK
!
interface GigabitEthernet0/1
 ip address 192.168.1.1 255.255.255.0
!
interface GigabitEthernet0/2
 no ip address
 shutdown
```

```
!
!
ip route 0.0.0.0 0.0.0.0 100.100.1.1
!
!
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Tue Aug 20 17:05:44 2024

GlobalParameters(
    Title                     = "This LSP was automatically generated by CSP Converter
at Tue Aug 20 17:05:44 2024 (user: cscons)"
    Version                     = LSP_5_0
    PreserveIPsecSA             = FALSE
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 100.100.1.1
        )

)

FirewallParameters(
    TCPSynSentTimeout = 30
    TCPFinTimeout = 5
    TCPClosedTimeout = 30
    TCPSynRcvdTimeout = 30
    TCPEstablishedTimeout = 3600
    TCPHalfOpenLow = 400
    TCPHalfOpenMax = 500
    TCPSessionRateLow = 400
    TCPSessionRateMax = 500
    TCPSessionsMax = 65536
    TCPStrictnessLevel = 3
    UDPNotEstablishedTimeout = 30
    UDPEstablishedTimeout = 120
    UDPSessionsMax = 0
)

IKETransform crypto:isakmp:policy:1
(
    CipherAlg   = "GR341215M-K256-CFB-65526"
    HashAlg     = "GR341112_256TC26-65128"
    GroupID     = VKO2_1B
    RestrictAuthenticationTo = PRESHARED
    LifetimeSeconds = 86400
)

ESPProposal MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP
(
    Transform* = ESPTransform
    (
        CipherAlg*          = "GR341215M-K288-MGM-33"
        LifetimeSeconds     = 3600
    )
```

```
)

FilterChain IPv6Default (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        Action = PASS
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

IdentityEntry Local:Preshared:PSK_Spoke1(
    IPv4Address = 100.100.1.10
)

IdentityEntry Remote:PSK_Spoke1(
    IPv4Address = 100.100.100.2
)

IKEParameters(
    FragmentSize = 576
    SALifetimeDelta = 50
    InitiatorSessionsMax = 100
    ResponderSessionsMax = 100
)

AuthMethodPreshared Preshared:PSK_Spoke1
(
    LocalID  = Local:Preshared:PSK_Spoke1
    RemoteID  = Remote:PSK_Spoke1
    SharedIKESecret = "cs_preshared_key_Hub1_Spoke1"
)

IKERule IKERule:MAP_PSK:1
(
    IKEPeerIPFilter = 172.16.100.2
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = Preshared:PSK_Spoke1
    MainModeAuthMethod  = Preshared:PSK_Spoke1
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

IPsecAction IPsecAction:MAP_PSK:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.100.2
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule:MAP_PSK:1
)

FilterChain IPsecPolicy:MAP_PSK (
    Filters = Filter (
        ProtocolID = 17
```

```
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 192.168.1.0/24
        DestinationIP = 192.168.100.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:MAP_PSK:1 >
        LogEventID = "IPsec:Protect:MAP_PSK:1:IPSEC_ACL_HUB1_AND_SPOKE1"
    )
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/0"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
    IPsecPolicy = IPsecPolicy:MAP_PSK
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/1"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/2"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)
```

Конфигурации криптошлюза Spoke2

1. Консоль cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity address
crypto isakmp fragmentation
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto ipsec security-association lifetime kilobytes unlimited
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $y$j9T$AI/n/DzJ8HDawzrfljr1N.$9WiLZf27Twl
XxCQoH9O.ieYl0.tUVIQG9G3a/KyFTy/
aaa new-model
!
!
hostname Spoke2
enable secret 5 JLI9qp52IVCYdKnFL5Xk/A==
!
crypto keyring Hub1_Spoke2
 pre-shared-key key_Hub1_Spoke2
!
!
!
```

```
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication pre-share
 group vko2
!
crypto isakmp profile PSK_Spoke2
 match identity address ipv4 100.100.100.2
 keyring Hub1_Spoke2
 self-identity address ipv4 100.100.1.20
 set identity mapping
 set policy 1
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
ip access-list extended IPSEC_ACL_HUB1_AND_SPOKE2
 permit ip 192.168.2.0 0.0.0.255 192.168.100.0 0.0.0.255
!
ipv6 default filtering deny
!
!
crypto map MAP_PSK 1 ipsec-isakmp
 match address IPSEC_ACL_HUB1_AND_SPOKE2
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile PSK_Spoke2
 set security-association mss-fix out
 set peer 172.16.100.2
 set pre-fragmentation
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 100.100.1.20 255.255.255.0
 crypto map MAP_PSK
!
interface GigabitEthernet0/1
 ip address 192.168.2.1 255.255.255.0
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
!
ip route 0.0.0.0 0.0.0.0 100.100.1.1
!
!
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Tue Aug 20 17:23:40 2024

GlobalParameters(
    Title                      = "This LSP was automatically generated by CSP Converter
at Tue Aug 20 17:23:40 2024 (user: cscons)"
    Version                      = LSP_5_0
    PreserveIPsecSA              = FALSE
)

RoutingTable(
```

```
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 100.100.1.1
        )

)

FirewallParameters(
    TCPSynSentTimeout = 30
    TCPFinTimeout = 5
    TCPClosedTimeout = 30
    TCPSynRcvdTimeout = 30
    TCPEstablishedTimeout = 3600
    TCPHalfOpenLow = 400
    TCPHalfOpenMax = 500
    TCPSessionRateLow = 400
    TCPSessionRateMax = 500
    TCPSessionsMax = 65536
    TCPStrictnessLevel = 3
    UDPNotEstablishedTimeout = 30
    UDPEstablishedTimeout = 120
    UDPSessionsMax = 0
)

IKETransform crypto:isakmp:policy:1
(
    CipherAlg   = "GR341215M-K256-CFB-65526"
    HashAlg     = "GR341112_256TC26-65128"
    GroupID     = VKO2_1B
    RestrictAuthenticationTo = PRESHARED
    LifetimeSeconds = 86400
)

ESPProposal MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP
(
    Transform* = ESPTransform
    (
        CipherAlg*          = "GR341215M-K288-MGM-33"
        LifetimeSeconds     = 3600
    )
)

FilterChain IPv6Default (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        Action = PASS
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

IdentityEntry Local:Preshared:PSK_Spoke2(
    IPv4Address = 100.100.1.20
)

IdentityEntry Remote:PSK_Spoke2(
    IPv4Address = 100.100.100.2
)

IKEParameters(
```

```
    FragmentSize = 576
    SALifetimeDelta = 50
    InitiatorSessionsMax = 100
    ResponderSessionsMax = 100
)

AuthMethodPreshared Preshared:PSK_Spoke2
(
    LocalID  = Local:Preshared:PSK_Spoke2
    RemoteID  = Remote:PSK_Spoke2
    SharedIKESecret = "cs_preshared_key_Hub1_Spoke2"
)

IKERule IKERule:MAP_PSK:1
(
    IKEPeerIPFilter = 172.16.100.2
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = Preshared:PSK_Spoke2
    MainModeAuthMethod  = Preshared:PSK_Spoke2
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

IPsecAction IPsecAction:MAP_PSK:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.100.2
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule:MAP_PSK:1
)

FilterChain IPsecPolicy:MAP_PSK (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 192.168.2.0/24
        DestinationIP = 192.168.100.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:MAP_PSK:1 >
        LogEventID = "IPsec:Protect:MAP_PSK:1:IPSEC_ACL_HUB1_AND_SPOKE2"
    )
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/0"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
    IPsecPolicy = IPsecPolicy:MAP_PSK
)

NetworkInterface (
```

```
    LogicalName = "GigabitEthernet0/1"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/2"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)
```