Построение отказоустойчивого решения с балансировкой трафика (GRE+OSPF)

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 содержит пример настройки отказоустойчивого решения, где трафик между защищаемыми подсетями центрального офиса и филиала балансируется по нескольким параллельным GRE туннелям, каждый из которых защищается шлюзами безопасности «С-Терра Шлюз 5.0». Инкапсуляция трафика в GRE и балансировка выполняются на маршрутизаторах, находящихся в доверенной зоне за криптошлюзами.

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 сертификаты в формате x509. В качестве криптопровайдера будет использована криптографическая библиотека, разработанная компанией «С-Терра СиЭсПи». Шлюзы безопасности (или криптошлюзы) – «С-Терра Шлюз» версии 5.0.

На криптошлюзах будет настроено межсетевое экранирование с применением статических списков доступа (ACL) и пакетной фильтрации с контролем состояния сессий (stateful, аналог CBAC в Cisco IOS) для протоколов TCP и UDP.

Предварительные требования

Требования к материально-техническому обеспечению

Для переноса запросов и сертификатов между криптошлюзами и центром выпуска сертификатов 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: PKI, IPsec, GRE, OSPF, Firewall, routing, switching.

Администратор должен быть ознакомлен со сценарием «Организация защищенного взаимодействия между подсетями центрального офиса и филиалов посредством IPsec VPN туннеля (протокол IPv4)» (далее по тексту он будет называться базовым сценарием).

Требования к инфраструктуре

1. Требования к устройствам.

   1.1. «С-Терра Шлюз».

      1.1.1. Устройства С-Терра Шлюз должны быть инициализированы (подробнее на http://doc.s-terra.ru раздел С-Терра Шлюз -> С-Терра Шлюз <версия> -> «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.

   1.2. Центр выпуска сертификатов.

      1.2.1. Должен быть настроен центр выпуска сертификатов (удостоверяющий центр, далее УЦ) для IPsec. Устройство с именем Certification-authority на схеме.

      1.2.2. Для выпуска цифровых сертификатов допускается использование встроенного в ОС Windows Server 2016 (или новее) удостоверяющего центра совместно с сертифицированным СКЗИ «КриптоПро» CSP 5.0 (или новее).

      1.2.3. Для тестовых целей можно использовать тестовый УЦ от «КриптоПро» (веб-интерфейс: https://www.cryptopro.ru/certsrv/certrqxt.asp).

Категорически запрещено использование тестового УЦ от «КриптоПро» в производственной (боевой) эксплуатации, так как в данном случае отсутствует возможность контролировать процесс выпуска сертификатов и, соответственно, процедуру аутентификации.

   1.3. HTTP сервер для распространения списка отозванных сертификатов.

      1.3.1. Функционирующий HTTP сервер для распространения списка отозванных сертификатов (COC). Устройство с именем CRL-distribution-point на схеме. Если по объективным причинам использование COC не представляется возможным или не требуется, то проверку COC можно отключить в CLI «С-Терра Шлюз» (при соблюдении правил пользования).

Доставка нового списка отозванных сертификатов с удостоверяющего центра на HTTP сервер должна происходить заблаговременно, до истечения срока действия предыдущего списка.

2. Требования к сетевому взаимодействию.

   2.1. Между устройствами стенда должна быть обеспечена IP связность.

Схема взаимодействия

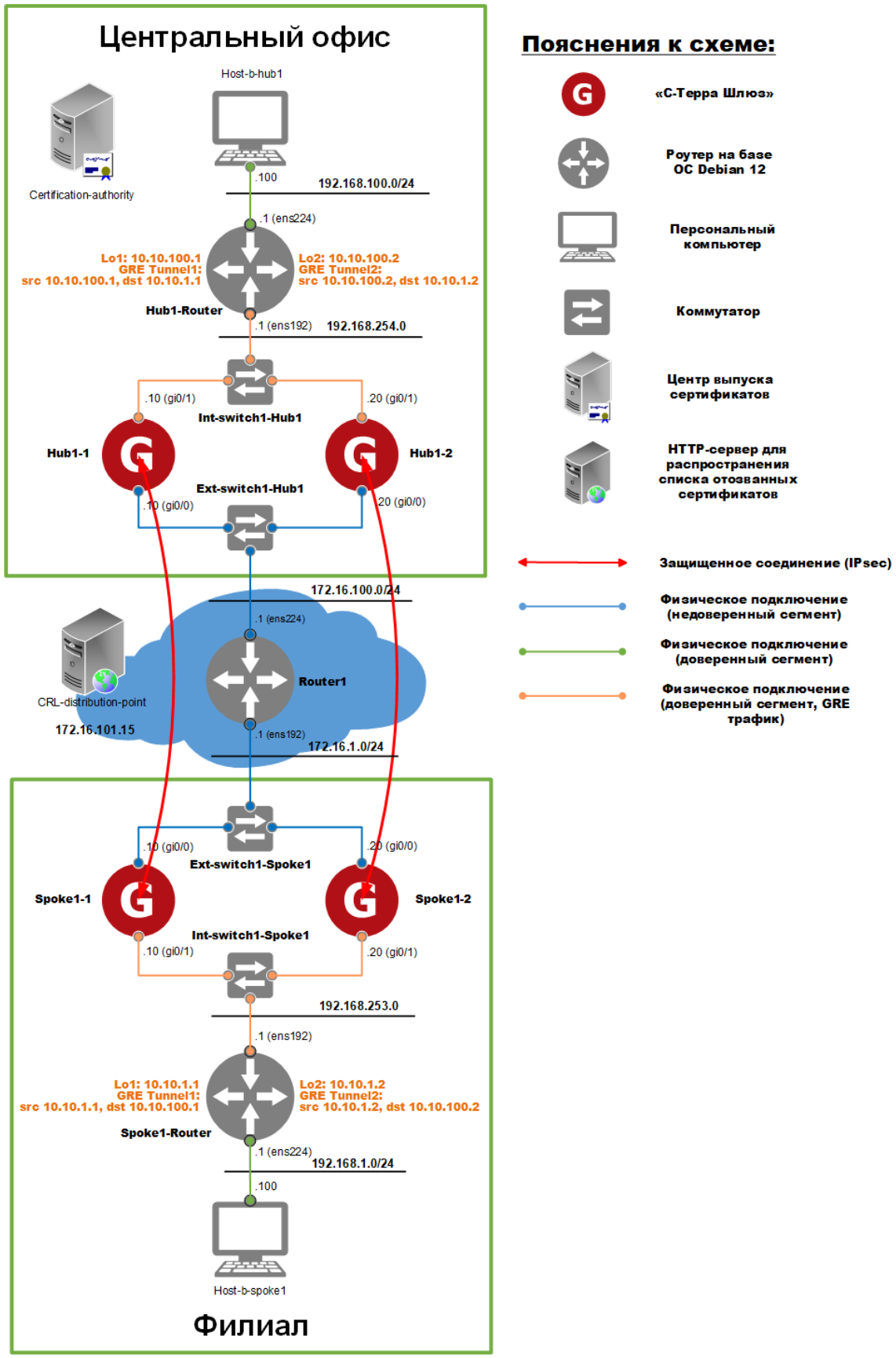


Рисунок 1. Схема взаимодействия

Общая логика работы

1. Размещение устройств.

   1.1. В центральном офисе размещаются: центр выпуска сертификатов (Certification-authority), коммутаторы (Int-switch1-Hub1 и Ext-switch1-Hub1), криптошлюзы С-Терра Шлюз (Hub1-1 и Hub1-2), маршрутизатор Hub1-Router и персональный компьютер (Host-b-hub1).

   1.2. В филиале размещаются: криптошлюзы С-Терра Шлюз (Spoke1-1 и Spoke1-2), коммутаторы (Int-switch1-Spoke1, Ext-switch1-Spoke1), маршрутизатор Spoke1-Router и персональный компьютер (Host-b-spoke1).

   1.3. В неконтролируемом сегменте (синее облако на схеме) размещаются: HTTP сервер для распространения списка отозванных сертификатов (CRL-distribution-point), маршрутизатор (Router1).

2. Балансировка трафика и отказоустойчивость.

   В данном сценарии трафик, идущий из сети центрального офиса в филиал, инкапсулируется и балансируется по двум GRE туннелям (Tunnel1 и Tunnel2) на маршрутизаторе Hub1-Router.

   В качестве tunnel source для первого GRE-туннеля будет использоваться интерфейс Lo1 (10.10.100.1), в качестве tunnel destination – IP-адрес интерфейса Lo1 маршрутизатора Spoke1-Router (10.10.1.1).

   В качестве tunnel source для второго GRE-туннеля будет использоваться интерфейс Lo2 (10.10.100.2), в качестве tunnel destination – IP-адрес интерфейса Lo2 маршрутизатора Spoke1-Router (10.10.1.2).

   GRE трафик первого туннеля будет направляться через криптошлюз Hub1-1, трафик второго туннеля – через криптошлюз Hub1-2. GRE трафик маршрутизируется при помощи статических заданных маршрутов.

   Аналогичным образом выполняется инкапсуляция и балансировка трафика из филиала в центральный офис на маршрутизаторе Spoke1-Router.

   Маршрутизаторы Hub1-Router и Spoke1-Router будут обмениваться информацией о маршрутах между защищаемыми подсетями центрального офиса и филиала с использованием протокола динамической маршрутизации OSPF. Обмен информацией будет происходить по GRE туннелям.

   В ситуации, когда все устройства стенда работают нормально, у маршрутизатора Hub1-Router будут доступны два маршрута до подсети филиала (через GRE туннели Tunnel1 и Tunnel2) с одинаковой метрикой. В случае отказа одного из криптошлюзов, маршрут через соответствующий GRE туннель удалится из таблицы маршрутизации, и трафик будет направлен по оставшемуся активным GRE туннелю.

3. Подключение к сети Интернет.В данном сценарии для эмуляции сети Интернет используются маршрутизатор Router1. Для обеспечения доступа в Интернет с устройств Host-b-hub1 и Host-b-spoke1 необходимо произвести дополнительные настройки. Подробнее в главе Дополнительные настройки.

   Подключение к сети Интернет на устройствах С-Терра Шлюз будет считаться успешным, если по протоколу ICMP (или «ping») будет доступен HTTP сервер для распространения списка отозванных сертификатов (устройство CRL-distribution-point на схеме).

4. Параметры межсетевого экранирования.Для пакетов, которые не защищаются IPsec применяются следующие правила фильтрации:

   4.1. На LAN интерфейсах IPv4 трафик не контролируется.

   4.2. На LAN интерфейсах IPv6 трафик заблокирован (за исключением capture интерфейсов L2, на которых захватывается весь трафик, включая IPv6).

   4.3. На WAN интерфейсах разрешен входящий IPv4 трафик:

      4.3.1. Протокол UDP порты 500 и 4500 – для IKE и NAT-Traversal (ESP in UDP).

      4.3.2. Протокол ICMP – для проверки доступности криптошлюза и работы механизма PathMTUd.

4.3.3. Протокол ESP (номер 50) – для работы IPsec.

4.3.4. Весь TCP и UDP трафик, для которого есть исходящее соединение (это динамические правила, которые создаются при помощи stateful фильтрации).

4.4. На WAN интерфейсах IPv6 трафик заблокирован.

4.5. На WAN интерфейсах инспектируются все исходящие локальные и транзитные IPv4 соединения для протоколов TCP и UDP. На основе результатов инспектирования осуществляется фильтрация входящего трафика с учетом состояния сессии (stateful).

5. Параметры шифрования IPsec.

Трафик GRE туннелей Tunnel1 и Tunnel2 между маршрутизаторами Hub-Router и Spoke-Router защищается с использованием алгоритмов ГОСТ и протокола IPsec в туннельном режиме.

Инициировать защищенное соединение может как GRE трафик от Hub-Router к Spoke-Router, так и наоборот.

5.1. Параметры протокола IKE:

- Аутентификация при помощи цифровых сертификатов, алгоритм подписи – ГОСТ Р 34.10-2012 (ключ 256 бит);
- 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_GOSTR3410_2012_256 (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 – 172.16.1.1 и маску – 255.255.255.0 на сетевом интерфейсе ens192.
2. Настройте IP адрес – 172.16.100.1 и маску – 255.255.255.0 на сетевом интерфейсе ens224.
3. 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 криптошлюза Hub1-1

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 и УЦ должны быть одинаковы, так как для аутентификации используются цифровые сертификаты, в которых зафиксированы дата и время начала их действия и окончания. Также одинаковые дата и время на всей инфраструктуре облегчают поиск неисправностей по лог-файлам.

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.

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройки PKI (запросы и сертификаты)

Продукт не поддерживает разностные (delta) списки отзыва сертификатов, только базовые. Учитывайте это при выборе или развертывании УЦ.

Для аутентификации партнеров по IPsec можно использовать цифровые сертификаты, выпущенные при помощи сертифицированного СКЗИ, или предварительно распределенные ключи, созданные и введённые в соответствии с правилами пользования СКЗИ (появились в версии 5.0).

Сгенерируйте запрос на сертификат криптошлюза и сохраните этот запрос в файл на USB Flash накопителе. Доставьте файл с запросом на УЦ и выпустите по нему сертификат. Сертификат криптошлюза вместе с сертификатом УЦ доставьте на шлюз и импортируйте их в базу Продукта.

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка паролей доступа к консоли cisco-like

Войдите в cisco-like консоль и смените пароль для пользователя `cscons` (пароль по умолчанию: csp) и пароль на enable (пароль по умолчанию: csp).

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка сетевых параметров

Изменение состояния интерфейсов и назначение IP адресов применяются сразу после ввода соответствующих команд. Политика безопасности применяется после выхода из режима конфигурирования.

1. Задайте имя устройства:

```
sterragate(config)#hostname Hub1-1
```

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

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

3. Убедитесь, что интерфейсы административно включены и line protocol (способность интерфейса передавать пакеты в данный момент) находится в состоянии up:

```
Hub1-1(config)#do show interfaces
```

```
GigabitEthernet0/0 is up, line protocol is up
  Hardware address is 0050.569e.b06d
  MTU 1500 bytes
GigabitEthernet0/1 is up, line protocol is up
  Hardware address is 0050.569e.d3c3
  MTU 1500 bytes
...
```

Если line protocol находится в состоянии down, то проверьте подключение сетевого интерфейса криптошлюза к коммутационному или прочему оборудованию.

4. Задайте IP адреса в соответствии со схемой стенда на внешнем `GigabitEthernet0/0` и внутреннем `GigabitEthernet0/1` интерфейсах:

```
Hub1-1(config)#interface GigabitEthernet 0/0
Hub1-1(config-if)#ip address 172.16.100.10 255.255.255.0
Hub1-1(config-if)#exit
Hub1-1(config)#interface GigabitEthernet 0/1
Hub1-1(config-if)#ip address 192.168.254.10 255.255.255.0
Hub1-1(config-if)#exit
```

5. Задайте маршрут до интерфейса Loopback1 маршрутизатора Hub1-Router через IP-адрес его интерфейса `ens192`:

```
Hub1-1(config)#ip route 10.10.100.1 255.255.255.255 192.168.254.1
```

6. Задайте маршрут по умолчанию через устройство Router1:

```
Hub1-1(config)#ip route 0.0.0.0 0.0.0.0 172.16.100.1
```

7. Проверьте доступность устройства Router1:

```
Hub1-1(config)#do 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=64 time=1.13 ms
108 bytes from 172.16.100.1: icmp_seq=2 ttl=64 time=0.237 ms
108 bytes from 172.16.100.1: icmp_seq=3 ttl=64 time=0.192 ms
108 bytes from 172.16.100.1: icmp_seq=4 ttl=64 time=0.200 ms
108 bytes from 172.16.100.1: icmp_seq=5 ttl=64 time=0.299 ms

--- 172.16.100.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4078ms
rtt min/avg/max/mdev = 0.192/0.411/1.131/0.362 ms
```

Настройка шифрования

1. Параметры IKE.

1.1. Задайте общие параметры IKE:

```
Hub1-1(config)#crypto isakmp identity dn
Hub1-1(config)#crypto isakmp keepalive 3 2
Hub1-1(config)#crypto isakmp keepalive retry-count 5
Hub1-1(config)#crypto isakmp fragmentation
Hub1-1(config)#crypto isakmp security-association lifetime delta 50
```

```
Hub1-1(config)#crypto isakmp initiator-sessions-max 100
Hub1-1(config)#crypto isakmp responder-sessions-max 100
```

1.2. Создайте политику, описывающую параметры IKE туннеля:

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

1.3. Создайте профиль настроек ISAKMP (алгоритм IKE) под именем `NEW_ALGS`, чтобы далее его прикрепить к криптокарте:

```
Hub1-1(config)#crypto isakmp profile NEW_ALGS
Hub1-1(conf-isa-prof)#match identity dn
Hub1-1(conf-isa-prof)#set identity mapping
Hub1-1(conf-isa-prof)#set policy 1
```

2. Параметры IPsec.

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

```
Hub1-1(config)#crypto ipsec security-association lifetime kilobytes unlimited
```

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

```
Hub1-1(config)# crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-
mgm
Hub1-1(cfg-crypto-trans)#exit
```

2.3. Создайте список доступа (ACL) для трафика, который нужно защищать между центральным офисом и филиалом:

```
Hub1-1(config)#ip access-list extended IPSEC_GRE_TUNNEL_1
Hub1-1(config-ext-nacl)#permit gre host 10.10.100.1 host 10.10.1.1
Hub1-1(config-ext-nacl)#exit
Hub1-1(config)#
```

2.4. Создайте крипто-карту (имя `VPN`, раздел 1):

```
Hub1-1(config)#crypto map VPN 1 ipsec-isakmp
Hub1-1(config-crypto-map)# match match address IPSEC_GRE_TUNNEL_1
Hub1-1(config-crypto-map)# set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
Hub1-1(config-crypto-map)# set peer 172.16.1.10
Hub1-1(config-crypto-map)# set isakmp-profile NEW_ALGS
Hub1-1(config-crypto-map)# set dead-connection history off
Hub1-1(config-crypto-map)#exit
```

2.5. Прикрепите созданную крипто-карту `VPN` к внешнему интерфейсу `GigabitEthernet0/0`:

```
Hub1-1(config)#interface GigabitEthernet0/0
Hub1-1(config-if)#crypto map VPN
```

2.6. Примените настройки:

```
Hub1-1(config-if)#end
```

В случае, когда по какой-либо причине полностью отсутствует IPsec соединение, на маршрутизаторах Hub1-Router и Spoke1-Router будет динамически удалена информация о маршрутах через GRE туннели, и таким образом трафик, идущий из подсети центрального офиса в филиал (и наоборот), не будет инкапсулирован в GRE и будет направлен через маршруты по умолчанию, заданные на Hub1-Router и Spoke1-Router (см. п. «Настройка маршрутов по умолчанию»).

Настройка политики обработки COC (CRL)

В целях безопасности настоятельно рекомендуется включать проверку списков отзыва сертификатов. Разностные списки отозванных сертификатов (delta CRL) не поддерживаются.

Включите проверку COC и настройте автоматическую загрузку и импортирование в базу Продукта списка отозванных сертификатов с HTTP сервера CRL-distribution-point.

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка межсетевого экрана

1. Создайте сетевой `object-group` с именем `SELF` для хранения в нем WAN адресов криптошлюза:

```
Hub1-1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Hub1-1(config)#object-group network SELF
Hub1-1(config-network-group)#remark WAN IP
Hub1-1(config-network-group)#host 172.16.100.10
Hub1-1(config-network-group)#exit
```

2. Создайте ACL с именем `FW_INPUT_WAN` для входящего на WAN интерфейс трафика:

```
Hub1-1(config)#ip access-list extended FW_INPUT_WAN
Hub1-1(config-ext-nacl)#permit udp any object-group SELF eq isakmp
Hub1-1(config-ext-nacl)#permit udp any object-group SELF eq non500-isakmp
Hub1-1(config-ext-nacl)#permit icmp any object-group SELF
Hub1-1(config-ext-nacl)#permit esp any object-group SELF
Hub1-1(config-ext-nacl)#exit
```

3. Создайте правила инспектирования всего TCP и UDP трафика и задайте необходимое максимальное количество сессий:

```
Hub1-1(config)#ip inspect name INSPECT tcp
Hub1-1(config)#ip inspect name INSPECT udp
Hub1-1(config)#ip inspect tcp sessions-max 256000
Hub1-1(config)#ip inspect udp sessions-max 65536
```

4. Прикрепите правила межсетевого экранирования на WAN интерфейс `GigabitEthernet0/0`:

```
Hub1-1(config)#interface GigabitEthernet0/0
Hub1-1(config-if)#ip access-group FW_INPUT_WAN in
Hub1-1(config-if)#ip inspect INSPECT out
Hub1-1(config-if)#exit
```

5. Примените настройки:

```
Hub1-1(config)#end
```

Подробная настройка с описанием команд приведена в базовом сценарии.

Если были настроены маршруты по умолчанию через Hub-Router и Spoke-Router, то при отсутствии IPsec соединения, трафик, идущий из подсети центрального офиса в филиал (и наоборот) будет направлен через них и не будет инкапсулирован в GRE. Во избежание утечки трафика необходимо настроить дополнительные списки доступа (см. п. «Настройка списков доступа для предотвращения утечки трафика»). Подробнее в главе «Дополнительные настройки».

В Приложении представлены тексты конфигураций для криптошлюза Hub1-1:

- текст консоли cisco-like;
- текст LSP.

Настройка криптошлюза Spoke1-1

Настройка криптошлюза Spoke1-1 происходит аналогично настройке Hub1-1.

В Приложении представлены тексты конфигураций для криптошлюза Spoke1-1:

- текст консоли cisco-like;
- текст LSP.

Настройка криптошлюза Hub1-2

Настройка криптошлюза Hub1-2 происходит аналогично настройке Hub1-1.

В Приложении представлены тексты конфигураций для криптошлюза Hub1-2:

- текст консоли cisco-like;
- текст LSP.

Настройка криптошлюза Spoke1-2

Настройка криптошлюза Spoke1-2 происходит аналогично настройке Hub1-1.

В Приложении представлены тексты конфигураций для криптошлюза Spoke1-2:

- текст консоли cisco-like;
- текст LSP.

Настройка маршрутизатора Hub1-Router

В данном сценарии используются маршрутизаторы на базе ОС Debian 12. В качестве протокола динамической маршрутизации будет использоваться OSPF, входящий в состав пакета FRRouting (https://frrouting.org).

Настройка сетевых параметров

1. Настройте IP адрес – 192.168.100.1 и маску – 255.255.255.0 на сетевом интерфейсе ens224.
2. Настройте IP адрес – 192.168.254.1 и маску – 255.255.255.0 на сетевом интерфейсе ens192.
3. Разрешите прохождение IP трафика (если заблокировано).

Настройка Loopback интерфейсов

Для создания интерфейсов, которые будут использоваться в качестве source и destination для GRE туннелей, нужно добавить в систему два интерфейса типа dummy.

1. Создайте файл `/etc/network/interfaces.d/lo1` со следующим содержимым:

```
auto lo1
iface lo1 inet static
address 10.10.100.1
netmask 255.255.255.255
pre-up ip link add lo1 type dummy
post-down ip link delete lo1
```

2. Создайте файл `/etc/network/interfaces.d/lo2` со следующим содержимым:

```
auto lo2
iface lo2 inet static
address 10.10.100.2
netmask 255.255.255.255
pre-up ip link add lo2 type dummy
post-down ip link delete lo2
```

3. Включите интерфейсы `lo1` и `lo2`:

```
root@Hub1-Router:~# ifup lo1
root@Hub1-Router:~# ifup lo2
```

4. Убедитесь, что данные интерфейсы доступны в системе и им назначены IP-адреса, например, в выводе команды `ip a`:

```
root@Hub1-Router:~# ip a
```

```
...
5: lo1: <BROADCAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/ether 1a:cb:76:48:e3:0f brd ff:ff:ff:ff:ff:ff
    inet 10.10.100.1/32 brd 10.10.100.1 scope global lo1
       valid_lft forever preferred_lft forever
6: lo2: <BROADCAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/ether 2e:a9:bf:cb:93:de brd ff:ff:ff:ff:ff:ff
    inet 10.10.100.2/32 brd 10.10.100.2 scope global lo2
       valid_lft forever preferred_lft forever
...
```

Настройка GRE интерфейсов

Для добавления GRE туннелей в систему, нужно создать два интерфейса в режиме `gre`.

В данном сценарии адреса для туннельных интерфейсов будут назначаться из подсети 100.100.101.0/30 для первого GRE туннеля и из подсети 100.100.102.0/30 для второго GRE туннеля.

1. Создайте файл `/etc/network/interfaces.d/tunnel1` со следующим содержимым:

```
auto tunnel1
iface tunnel1 inet static
address 100.100.101.1
netmask 255.255.255.252

pre-up ip tunnel add tunnel1 mode gre local 10.10.100.1 remote 10.10.1.1 ttl 64
pre-up ip link set tunnel1 multicast on
pre-up ip link set tunnel1 mtu 1400
post-down ip link del tunnel1
```

2. Создайте файл `/etc/network/interfaces.d/tunnel2` со следующим содержимым:

```
auto tunnel2
iface tunnel2 inet static
address 100.100.102.1
netmask 255.255.255.252

pre-up ip tunnel add tunnel2 mode gre local 10.10.100.2 remote 10.10.1.2 ttl 64
pre-up ip link set tunnel2 multicast on
pre-up ip link set tunnel2 mtu 1400
post-down ip link del tunnel2
```

3. Включите интерфейсы `tunnel1` и `tunnel2`:

```
root@Hub1-Router:~# ifup tunnel1
root@Hub1-Router:~# ifup tunnel2
```

4. Убедитесь, что данные интерфейсы доступны в системе и им назначены IP-адреса, например, в выводе команды `ip a`:

```
root@Hub1-Router:~# ip a
```

```
...
10: tunnel1@NONE: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1400 qdisc noqueue
state UNKNOWN group default qlen 1000
    link/gre 10.10.100.1 peer 10.10.1.1
    inet 100.100.101.1/30 brd 100.100.101.3 scope global tunnel1
       valid_lft forever preferred_lft forever
11: tunnel2@NONE: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1400 qdisc noqueue
state UNKNOWN group default qlen 1000
    link/gre 10.10.100.2 peer 10.10.1.2
    inet 100.100.102.1/30 brd 100.100.102.3 scope global tunnel2
       valid_lft forever preferred_lft forever
...
```

Настройка статических маршрутов

1. Задайте статический маршрут до destination IP-адреса GRE туннеля Tunnel1 (10.10.1.1) через криптошлюз Hub1-1 (192.168.254.10).
2. Задайте статический маршрут до destination IP-адреса GRE туннеля Tunnel2 (10.10.1.2) через криптошлюз Hub1-2 (192.168.254.20).

Настройка динамической маршрутизации

Настройка FRR

1. В конфигурационном файле `/etc/frr/daemons` установите значение `yes` для демона `ospfd`:

```
ospfd=yes
```

2. Перезапустите сервис FRR:

```
root@Hub1-Router:~# service frr restart
```

3. Убедитесь, что демон `ospfd` запущен:

```
root@Hub1-Router:~# ps -ef | grep "ospfd"
root        484      1  0 12:34 ?        00:00:00 /usr/lib/frr/watchfrr -d -F
traditional zebra ospfd staticd
frr         543      1  0 12:34 ?        00:00:00 /usr/lib/frr/ospfd -d -F
traditional -A 127.0.0.1
```

Настройка OSPF

1. Войдите в консоль FRR:

```
root@Hub1-Router:~# vtysh
Hello, this is FRRouting (version 9.0.4).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

Hub1-Router#
```

2. Перейдите в режим конфигурации:

```
Hub1-Router# configure terminal
Hub1-Router(config)#
```

3. Включите процесс OSPF:

```
Hub1-Router(config)# router ospf
Hub1-Router(config-router)#
```

4. Задайте `router-id`:

```
Hub1-Router(config-router)# ospf router-id 192.168.100.1
```

5. Отключите отправку Hello пакетов на всех интерфейсах по умолчанию:

```
Hub1-Router(config-router)# passive-interface default
```

6. Включите OSPF для подсети центрального офиса (192.168.100.0/24) и для подсетей GRE туннелей (100.100.101.0/30 и 100.100.101.0/30):

```
Hub1-Router(config-router)# network 192.168.100.0/24 area 0
Hub1-Router(config-router)# network 100.100.101.0/30 area 0
Hub1-Router(config-router)# network 100.100.102.0/30 area 0
```

7. Выйдите из режима конфигурации OSPF:

```
Hub1-Router(config-router)# exit
Hub1-Router(config)#
```

8. Включите отправку Hello пакетов на интерфейсах `tunnel1` и `tunnel2`:

```
Hub1-Router(config)# interface tunnel1
Hub1-Router(config-if)# no ip ospf passive
Hub1-Router(config-if)# exit
Hub1-Router(config)# interface tunnel2
Hub1-Router(config-if)# no ip ospf passive
```

9. Настройте протокол OSPF для работы в широковещательном режиме на интерфейсах `tunnel1` и `tunnel2`:

```
Hub1-Router(config)# interface tunnel1
Hub1-Router(config-if)# ip ospf network broadcast
Hub1-Router(config-if)# exit
Hub1-Router(config)# interface tunnel2
Hub1-Router(config-if)# ip ospf network broadcast
```

10. Завершите процедуру конфигурации:

```
Hub1-Router(config)# end
```

11. Сохраните текущую конфигурацию (она будет записана в файл `/etc/frr/frr.conf`):

```
Hub1-Router# write memory
```

```
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Integrated configuration saved to /etc/frr/frr.conf
[OK]
```

12. Выйдите из консоли FRR обратно в linux bash консоль:

```
Hub1-Router# exit
```

```
root@Hub1-Router:~#
```

Настройка маршрутизатора Spoke1-Router

Настройка маршрутизатора Spoke1-Router осуществляется аналогично Hub1-Router. В Приложении представлены текст конфигурации FRR и настройки для Loopback и GRE интерфейсов.

Дополнительные настройки

Настройка маршрутов по умолчанию

Если из подсетей центрального офиса и филиала нужно обеспечить доступ в Интернет, то необходимо настроить маршруты по умолчанию на маршрутизаторах Hub1-Router и Spoke1-Router. Для обеспечения отказоустойчивости необходимо настроить получение маршрута по умолчанию с помощью динамической маршрутизации. Предположим, нужно обеспечить доступ в интернет для устройства Host-b-hub1. Сделать это можно следующим образом:

1. Настройте динамическую маршрутизацию на устройствах Hub1-1 и Hub1-2.
   1.1. Настройте FRR по аналогии с главой «Настройка FRR» устройства Hub1-Router.
   1.2. Настройте OSPF:
      1.2.1. Зайдите в FRR консоль и перейдите в режим конфигурирования.
      1.2.2. Настройте протокол OSPF для работы в широковещательном режиме на интерфейсе eth1:

```
Hub1-1(config)# interface eth1
Hub1-1(config-if)# ip ospf network broadcast
Hub1-1(config-if)# exit
```

      1.2.3. Включите процесс OSPF, задайте `router-id`, включите OSPF для подсети маршрутизатора Hub1_Router (192.168.254.0/24) и включите анонсирование маршрутов:

```
Hub1-1(config)# router ospf
Hub1-1(config-router)# ospf router-id 192.168.254.10
Hub1-1(config-router)# network 192.168.254.0/24 area 0
Hub1-1(config-router)# default-information originate
```

      Таким образом криптошлюз отправит свой маршрут по умолчанию маршрутизатору Hub1-Router.
   1.3. Включите SNAT для трафика, идущего подсети устройства Host-b-hub1.

```
root@Hub1-1:~# iptables -t nat -A POSTROUTING -o eth0 -s 192.168.100.0/24 -j SNAT --
to-source 172.16.100.10
```

   1.4. Произведите аналогичные настройки на устройстве Hub1-2.
2. Настройте маршрутизатор Hub1-Router.
   2.1. Зайдите в FRR консоль и перейдите в режим конфигурирования.
   2.2. Включите отправку Hello пакетов, а также настройте протокол OSPF для работы в широковещательном режиме на интерфейсе ens192.

```
Hub1-Router(config)# interface ens192
Hub1-Router(config-if)# ip ospf network broadcast
Hub1-Router(config-if)# no ip ospf passive
Hub1-Router(config-if)# exit
```

   2.3. Включите OSPF для подсети криптошлюзов:

```
Hub1-Router(config-router)# network 192.168.254.0/24 area 0
```

3. После завершения настроек убедитесь в том, что маршруты по умолчанию получены на Hub1-Router.

```
root@Hub1-Router:~# ip r
default nhid 382 proto ospf metric 20
        nexthop via 192.168.254.10 dev ens192 weight 1
        nexthop via 192.168.254.20 dev ens192 weight 1
```

4. Удостоверьтесь в том, что у устройства Host-b-hub1 есть доступ в интернет.

Помимо приведенных выше настроек необходимо описать дополнительные списки доступа, чтобы не допустить утечку трафика (см. ниже)

Настройка списков доступа для предотвращения утечки трафика

В случае отсутствия IPsec соединения маршрутизаторы Hub1-Router и Spoke1-Router автоматически уберут данные о маршрутах через GRE туннели. Это приведет к тому, что трафик между подсетью центрального офиса и филиалом не будет инкапсулирован в GRE и будет отправляться по маршрутам по умолчанию, настроенным на Hub1-Router и Spoke1-Router. В целях безопасности, на криптошлюзах следует заблокировать прохождение этого трафика. Сделать это можно с помощью описания расширенных списков доступа. При этом, при наличии NAT необходимо использовать следующую логику: блокировать весь трафик, у которого исходный адрес соответствует адресу, на который выполняется NAT, а целевой адрес принадлежит приватным диапазонам (192.168.0.0/16, 172.16.0.0/12). Важно отметить, что перед применением данного правила необходимо разрешить трафик к определённым адресам, который не должен шифроваться, таким как адреса сервера CRL и прокси-серверов, поскольку трафик к ним не должен подвергаться блокировке.

Например, можно создать две объектные группы: одну для разрешенных адресов и другую для запрещенных. Для устройства Hub1-1 настройка будет следующей:

1. Создайте `object-group network` для разрешенных адресов:

```
Hub1-1(config)#object-group network ALLOWED_ADDRESSES
Hub1-1(config-network-group)# host 172.16.100.1
Hub1-1(config-network-group)# host 172.16.101.15
Hub1-1(config-network-group)# host 172.16.1.10
```

2. Создайте `object-group network` для запрещенных подсетей:

```
Hub1-1(config)#object-group network DENIED_ADDRESSES
Hub1-1(config-network-group)# 192.168.0.0 255.255.0.0
Hub1-1(config-network-group)# 172.16.0.0 255.240.0.0
```

3. Создайте список доступа для защищаемого трафика:

```
Hub1-1(config)# ip access-list extended ACL_TRAFFIC_LEAK
Hub1-1(config-ext-nacl)# deny   gre any any
Hub1-1(config-ext-nacl)# permit ip host 172.16.100.10 object-group ALLOWED_ADDRESSES
Hub1-1(config-ext-nacl)# deny   ip host 172.16.100.10 object-group DENIED_ADDRESSES
Hub1-1(config-ext-nacl)# permit ip any any
```

4. Прикрепите созданный список доступа на WAN интерфейс `GigabitEthernet0/0` (обратите особое внимание на направление правил фильтрации, список доступа `ACL_TRAFFIC_LEAK` прикрепляется для выходящих с интерфейса пакетов – **out**):

```
Hub1-1(config)# interface GigabitEthernet0/0
Hub1-1(config-if)# ip access-group ACL_TRAFFIC_LEAK out
Hub1-1(config-if)# exit
```

5. С помощью утилиты `tcpdump`, удостоверьтесь в том, что на устройство Router1 не попадает трафик с устройства Host-b-hub1.

Проверка работоспособности стенда

Проверка IPsec соединений

Проверьте наличие IPsec туннелей на криптошлюзах с помощью команды `run sa_mgr show` из cisco-like консоли криптошлюзов. Обратите внимание на номер протокола в разделе IPsec connections, номер 47 соответствует протоколу GRE.

Эти IPsec туннели должны появиться сразу после настройки маршрутизаторов Hub1-Router и Spoke1-Router, так как построение защищенного соединения будет инициировано GRE трафиком в связи с началом обмена информацией по протоколу OSPF.

1. На криптошлюзе Hub1-1:

```
Hub1-1#run 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 (172.16.100.10,500)-(172.16.1.10,500) active 39952 39880

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (10.10.100.1,*)-(10.10.1.1,*) 47 ESP tunn 29608 28808
```

2. На криптошлюзе Spoke1-1:

```
Spoke1-1#run 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 (172.16.1.10,500)-(172.16.100.10,500) active 39880 39952

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (10.10.1.1,*)-(10.10.100.1,*) 47 ESP tunn 28808 29608
```

3. На криптошлюзе Hub1-2:

```
Hub1-2#run 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 (172.16.100.20,500)-(172.16.1.20,500) active 40252 40180

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (10.10.100.2,*)-(10.10.1.2,*) 47 ESP tunn 29248 28256
```

4. На криптошлюзе Spoke1-2:

```
Spoke1-2#run 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 (172.16.1.20,500)-(172.16.100.20,500) active 40180 40252

IPsec connections:
```

```
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (10.10.1.2,*)-(10.10.100.2,*) 47 ESP tunn 28256 29248
```

Проверка наличия маршрутов

Проверьте, что на маршрутизаторах Hub1-Router и Spoke1-Router появилась информация о маршрутах до удаленных подсетей:

1. На устройстве Hub1-Router должен появиться маршрут до подсети филиала:

```
root@Hub1-Router:~# ip r
```

```
10.10.1.1 via 192.168.254.10 dev ens192
10.10.1.2 via 192.168.254.20 dev ens192
100.100.101.0/30 dev tunnel1 proto kernel scope link src 100.100.101.1
100.100.102.0/30 dev tunnel2 proto kernel scope link src 100.100.102.1
192.168.100.0/24 dev ens224 proto kernel scope link src 192.168.100.1
192.168.254.0/24 dev ens192 proto kernel scope link src 192.168.254.1
192.168.1.0/24 proto ospf metric 20
      nexthop via 100.100.101.2 dev tunnel1 weight 1
      nexthop via 100.100.102.2 dev tunnel2 weight 1
```

2. На устройстве Spoke1-Router должен появиться маршрут до подсети центрального офиса:

```
root@Spoke1-Router:~# ip r
```

```
10.10.100.1 via 192.168.253.10 dev ens192
10.10.100.2 via 192.168.253.20 dev ens192
100.100.101.0/30 dev tunnel1 proto kernel scope link src 100.100.101.2
100.100.102.0/30 dev tunnel2 proto kernel scope link src 100.100.102.2
192.168.1.0/24 dev ens224 proto kernel scope link src 192.168.1.1
192.168.253.0/24 dev ens192 proto kernel scope link src 192.168.253.1
192.168.100.0/24 proto ospf metric 20
      nexthop via 100.100.101.1 dev tunnel1 weight 1
      nexthop via 100.100.102.1 dev tunnel2 weight 1
```

Проверка сетевого соединения между подсетями центрального офиса и филиала.

С устройства Host-b-hub1 выполните проверку доступности устройства Host-b-spoke1 (можно и наоборот) с помощью отправки ICMP трафика. Для этого выполните команду `ping` из linux bash консоли устройства Host-b-hub1:

```
root@Host-b-hub1:~# ping 192.168.1.100 -c 5
```

```
PING 192.168.1.100 (192.168.1.100) 56(84) bytes of data.
64 bytes from 192.168.1.100: icmp_seq=1 ttl=62 time=3.30 ms
64 bytes from 192.168.1.100: icmp_seq=2 ttl=62 time=2.19 ms
64 bytes from 192.168.1.100: icmp_seq=3 ttl=62 time=2.20 ms
64 bytes from 192.168.1.100: icmp_seq=4 ttl=62 time=2.13 ms
64 bytes from 192.168.1.100: icmp_seq=5 ttl=62 time=2.16 ms

--- 192.168.1.100 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 12ms
rtt min/avg/max/mdev = 2.130/2.397/3.303/0.454 ms
```

Проверка отказоустойчивости

1. Посылайте с устройства Host-b-spoke1 на Host-b-hub1 защищаемый ICMP-трафик:

```
root@Host-b-spoke1:~# ping 192.168.100.100
PING 192.168.100.100 (192.168.100.100) 56(84) bytes of data.
64 bytes from 192.168.100.100: icmp_seq=1 ttl=64 time=3.54 ms
64 bytes from 192.168.100.100: icmp_seq=2 ttl=64 time=2.33 ms
```

```
64 bytes from 192.168.100.100: icmp_seq=3 ttl=64 time=2.49 ms
64 bytes from 192.168.100.100: icmp_seq=4 ttl=64 time=2.26 ms
64 bytes from 192.168.100.100: icmp_seq=5 ttl=64 time=2.32 ms
...
```

2. Смоделируйте отказ канала связи, например, отключив сетевой кабель внешнего интерфейса, на одном из криптошлюзов (Spoke1-1 или Spoke1-2).

   При помощи диагностических команд, выполняемых на маршрутизаторах Hub1-Router и Spoke1-Router, убедитесь в том, что в случае отказа одного из криптошлюзов, маршрут через соответствующий GRE туннель удалится из таблицы маршрутизации, и трафик будет направлен по оставшемуся активным GRE туннелю. Например, в случае отключения внешнего интерфейса криптошлюза Spoke1-2, таблица маршрутизации выглядит следующим образом:

   ```
   root@Hub1-Router:~# ip r
   ```

   ```
   10.10.1.1 via 192.168.254.10 dev ens192
   10.10.1.2 via 192.168.254.20 dev ens192
   100.100.101.0/30 dev tunnel1 proto kernel scope link src 100.100.101.1
   100.100.102.0/30 dev tunnel2 proto kernel scope link src 100.100.102.1
   192.168.1.0/24 nhid 46 via 100.100.101.2 dev tunnel1 proto ospf metric 20
   192.168.100.0/24 dev ens224 proto kernel scope link src 192.168.100.1
   192.168.254.0/24 dev ens192 proto kernel scope link src 192.168.254.1
   ```

   Видно, что в таблице маршрутизации Hub1-Router оставлен один маршрут через GRE туннель рабочего устройства Spoke1-1.

3. Подождите некоторое время, пока трафик будет перебалансирован на рабочую пару криптошлюзов.

4. Как только произошла перебалансировка трафика, убедитесь в том, что в IPsec туннелях рабочей пары криптошлюзов симметрично увеличиваются счетчики `Sent` и `Rcvd`. Для просмотра счетчиков выполните команду `run sa_mgr show` из cisco-like консоли криптошлюзов.

   Для рабочей пары криптошлюзов вывод `run sa_mgr show` команды будет следующим:

   ```
   Hub1-1#run sa_mgr show
   ```

   ```
   ISAKMP sessions: 0 initiated, 0 responded

   ISAKMP connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
   1 4 (172.16.100.10,500)-(172.16.1.10,500) active 153036 153488

   IPsec connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
   1 76 (10.10.100.1,*)-(10.10.1.1,*) 47 ESP tunn 64400 62068
   ```

   ```
   Spoke1-1#run sa_mgr show
   ```

   ```
   ISAKMP sessions: 0 initiated, 0 responded

   ISAKMP connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
   1 4 (172.16.1.10,500)-(172.16.100.10,500) active 153488 153036

   IPsec connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
   1 76 (10.10.1.1,*)-(10.10.100.1,*) 47 ESP tunn 62068 64400
   ```

   Видно, что счетчики `Sent` и `Rcvd` увеличиваются симметрично. При наличии отклонений проанализируйте журналы на криптошлюзах.

Приложение

Конфигурация криптошлюза Hub1-1

1. Конфигурация cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity dn
crypto isakmp fragmentation
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
hostname Hub1-1
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
!
!
!
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp profile NEW_ALGS
 match identity dn
 set identity mapping
 set policy 1
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
object-group network SELF
 remark WAN IP
 host 172.16.100.10
!
ip access-list extended IPSEC_GRE_TUNNEL_1
 permit gre host 10.10.100.1 host 10.10.1.1
!
ip access-list extended FW_INPUT_WAN
 permit udp any object-group SELF eq isakmp
 permit udp any object-group SELF eq non500-isakmp
 permit icmp any object-group SELF
 permit esp any object-group SELF
!
ipv6 default filtering deny
!
```

```
!
ip inspect tcp sessions-max 256000
ip inspect udp sessions-max 65536
ip inspect name INSPECT tcp
ip inspect name INSPECT udp
!
crypto map VPN 1 ipsec-isakmp
 match address IPSEC_GRE_TUNNEL_1
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set peer 172.16.1.10
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 172.16.100.10 255.255.255.0
 ip access-group FW_INPUT_WAN in
 crypto map VPN
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 ip address 192.168.254.10 255.255.255.0
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
!
ip route 0.0.0.0 0.0.0.0 172.16.100.1
ip route 10.10.100.1 255.255.255.255 192.168.254.1
!
!
crypto pki trustpoint s-terra_technological_trustpoint
 revocation-check crl
 crl download group ROOTCA http://172.16.101.15/certcrl.crl
 crl download time 60
crypto pki certificate chain s-terra_technological_trustpoint
certificate 58E026BFD6D625BE4582C16C6189C183
...
AD4F8901771632E0A0AF83

quit
!
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Thu Nov 14 16:43:31 2024

GlobalParameters(
    Title                      = "This LSP was automatically generated by CSP Converter
at Thu Nov 14 16:43:31 2024 (user: cscons)"
    Version                      = LSP_5_0
    CRLHandlingMode              = ENABLE
    PreserveIPsecSA              = FALSE
)

RoutingTable(
    Routes =
        Route(
```

```
            Destination = 0.0.0.0/0
            Gateway = 172.16.100.1
        ),
        Route(
            Destination = 10.10.100.1/32
            Gateway = 192.168.254.1
        )

)

FirewallParameters(
    TCPSynSentTimeout = 30
    TCPFinTimeout = 5
    TCPClosedTimeout = 30
    TCPSynRcvdTimeout = 30
    TCPEstablishedTimeout = 3600
    TCPHalfOpenLow = 400
    TCPHalfOpenMax = 500
    TCPSessionRateLow = 400
    TCPSessionRateMax = 500
    TCPSessionsMax = 256000
    TCPStrictnessLevel = 3
    UDPNotEstablishedTimeout = 30
    UDPEstablishedTimeout = 120
    UDPSessionsMax = 65536
)

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

ESPProposal MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP
(
    Transform* = ESPTransform
    (
        CipherAlg*          = "GR341215M-K288-MGM-33"
        LifetimeSeconds     = 3600
        LifetimeKilobytes   = 4608000
    )
)

FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.100.10
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.10
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
```

```
    ),
    Filter (
        DestinationIP = 172.16.100.10
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.10
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = DROP
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

FilterChain InspectChain:INSPECT (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 6
        Action = PASS
        ExtendedAction = inspect_tcp<>
        LogEventID = "Inspect:INSPECT:tcp"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 17
        Action = PASS
        ExtendedAction = inspect_udp<>
        LogEventID = "Inspect:INSPECT:udp"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = PASS
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

IdentityEntry Remote:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

IdentityEntry Local:Cert:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

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

```
        DestinationIP = ::/0
        Action = DROP
    ))

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

AuthMethodGOSTSign GOST:Sign:NEW_ALGS
(
    RemoteID             = Remote:NEW_ALGS
    LocalID              = Local:Cert:NEW_ALGS
    SendRequestMode      = ALWAYS
    SendCertMode         = ALWAYS
)

IKERule IKERule:VPN:1
(
    IKEPeerIPFilter = 172.16.1.10
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

IPsecAction IPsecAction:VPN:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.1.10
    )
    ContainedProposals = ( MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    DFHandling = COPY
    IKERule = IKERule:VPN:1
)

FilterChain IPsecPolicy:VPN (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 10.10.100.1
        DestinationIP = 10.10.1.1
        ProtocolID = 47
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_GRE_TUNNEL_1"
    )
)

NetworkInterface (
```

```
    LogicalName = "GigabitEthernet0/0"
    InputFilter = FilterChain:FW_INPUT_WAN
    OutputFilter = InspectChain:INSPECT
    IPsecPolicy = IPsecPolicy:VPN
)

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

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

Конфигурация криптошлюза Hub1-2

1. Конфигурация cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity dn
crypto isakmp fragmentation
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
hostname Hub1-2
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
!
!
!
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp profile NEW_ALGS
 match identity dn
 set identity mapping
 set policy 1
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
```

```
!
object-group network SELF
 remark WAN IP
 host 172.16.100.20
!
ip access-list extended IPSEC_GRE_TUNNEL_2
 permit gre host 10.10.100.2 host 10.10.1.2
!
ip access-list extended FW_INPUT_WAN
 permit udp any object-group SELF eq isakmp
 permit udp any object-group SELF eq non500-isakmp
 permit icmp any object-group SELF
 permit esp any object-group SELF
!
ipv6 default filtering deny
!
!
ip inspect tcp sessions-max 256000
ip inspect udp sessions-max 65536
ip inspect name INSPECT tcp
ip inspect name INSPECT udp
!
crypto map VPN 1 ipsec-isakmp
 match address IPSEC_GRE_TUNNEL_2
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set peer 172.16.1.20
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 172.16.100.20 255.255.255.0
 ip access-group FW_INPUT_WAN in
 crypto map VPN
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 ip address 192.168.254.20 255.255.255.0
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
!
ip route 0.0.0.0 0.0.0.0 172.16.100.1
ip route 10.10.100.2 255.255.255.255 192.168.254.1
!
!
crypto pki trustpoint s-terra_technological_trustpoint
 revocation-check crl
 crl download group ROOTCA http://172.16.101.15/certcrl.crl
 crl download time 60
crypto pki certificate chain s-terra_technological_trustpoint
certificate 58E026BFD6D625BE4582C16C6189C183
...
AD4F8901771632E0A0AF83

quit
!
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Thu Nov 14 16:45:00 2024

GlobalParameters(
    Title                     = "This LSP was automatically generated by CSP Converter
at Thu Nov 14 16:45:00 2024 (user: cscons)"
    Version                     = LSP_5_0
    CRLHandlingMode             = ENABLE
    PreserveIPsecSA             = FALSE
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 172.16.100.1
        ),
        Route(
            Destination = 10.10.100.2/32
            Gateway = 192.168.254.1
        )

)

FirewallParameters(
    TCPSynSentTimeout = 30
    TCPFinTimeout = 5
    TCPClosedTimeout = 30
    TCPSynRcvdTimeout = 30
    TCPEstablishedTimeout = 3600
    TCPHalfOpenLow = 400
    TCPHalfOpenMax = 500
    TCPSessionRateLow = 400
    TCPSessionRateMax = 500
    TCPSessionsMax = 256000
    TCPStrictnessLevel = 3
    UDPNotEstablishedTimeout = 30
    UDPEstablishedTimeout = 120
    UDPSessionsMax = 65536
)

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

ESPProposal MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP
(
    Transform* = ESPTransform
    (
        CipherAlg*          = "GR341215M-K288-MGM-33"
        LifetimeSeconds     = 3600
        LifetimeKilobytes   = 4608000
    )
)
```

```
FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.100.20
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.20
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.20
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.20
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = DROP
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

FilterChain InspectChain:INSPECT (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 6
        Action = PASS
        ExtendedAction = inspect_tcp<>
        LogEventID = "Inspect:INSPECT:tcp"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 17
        Action = PASS
        ExtendedAction = inspect_udp<>
        LogEventID = "Inspect:INSPECT:udp"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = PASS
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))
```

```
IdentityEntry Remote:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

IdentityEntry Local:Cert:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

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

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

AuthMethodGOSTSign GOST:Sign:NEW_ALGS
(
    RemoteID             = Remote:NEW_ALGS
    LocalID              = Local:Cert:NEW_ALGS
    SendRequestMode      = ALWAYS
    SendCertMode         = ALWAYS
)

IKERule IKERule:VPN:1
(
    IKEPeerIPFilter = 172.16.1.20
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

IPsecAction IPsecAction:VPN:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.1.20
    )
    ContainedProposals = ( MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    DFHandling = COPY
    IKERule = IKERule:VPN:1
)

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

```
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 10.10.100.2
        DestinationIP = 10.10.1.2
        ProtocolID = 47
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_GRE_TUNNEL_2"
    )
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/0"
    InputFilter = FilterChain:FW_INPUT_WAN
    OutputFilter = InspectChain:INSPECT
    IPsecPolicy = IPsecPolicy:VPN
)

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

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

Конфигурация криптошлюза Spoke1-1

1. Конфигурация cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity dn
crypto isakmp fragmentation
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
hostname Spoke1-1
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
!
!
!
```

```
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp profile NEW_ALGS
 match identity dn
 set identity mapping
 set policy 1
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
object-group network SELF
 remark WAN IP
 host 172.16.1.10
!
ip access-list extended IPSEC_GRE_TUNNEL_1
 permit gre host 10.10.1.1 host 10.10.100.1
!
ip access-list extended FW_INPUT_WAN
 permit udp any object-group SELF eq isakmp
 permit udp any object-group SELF eq non500-isakmp
 permit icmp any object-group SELF
 permit esp any object-group SELF
!
ipv6 default filtering deny
!
!
ip inspect tcp sessions-max 256000
ip inspect udp sessions-max 65536
ip inspect name INSPECT tcp
ip inspect name INSPECT udp
!
crypto map VPN 1 ipsec-isakmp
 match address IPSEC_GRE_TUNNEL_1
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set peer 172.16.100.10
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 172.16.1.10 255.255.255.0
 ip access-group FW_INPUT_WAN in
 crypto map VPN
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 ip address 192.168.253.10 255.255.255.0
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
!
ip route 0.0.0.0 0.0.0.0 172.16.1.1
ip route 10.10.1.1 255.255.255.255 192.168.253.1
!
```

```
!
crypto pki trustpoint s-terra_technological_trustpoint
 revocation-check crl
 crl download group ROOTCA http://172.16.101.15/certcrl.crl
 crl download time 60
crypto pki certificate chain s-terra_technological_trustpoint
certificate 58E026BFD6D625BE4582C16C6189C183
...
AD4F8901771632E0A0AF83

quit
!
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Thu Nov 14 16:46:22 2024

GlobalParameters(
    Title                     = "This LSP was automatically generated by CSP Converter
at Thu Nov 14 16:46:22 2024 (user: cscons)"
    Version                     = LSP_5_0
    CRLHandlingMode             = ENABLE
    PreserveIPsecSA             = FALSE
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 172.16.1.1
        ),
        Route(
            Destination = 10.10.1.1/32
            Gateway = 192.168.253.1
        )

)

FirewallParameters(
    TCPSynSentTimeout = 30
    TCPFinTimeout = 5
    TCPClosedTimeout = 30
    TCPSynRcvdTimeout = 30
    TCPEstablishedTimeout = 3600
    TCPHalfOpenLow = 400
    TCPHalfOpenMax = 500
    TCPSessionRateLow = 400
    TCPSessionRateMax = 500
    TCPSessionsMax = 256000
    TCPStrictnessLevel = 3
    UDPNotEstablishedTimeout = 30
    UDPEstablishedTimeout = 120
    UDPSessionsMax = 65536
)

IKETransform crypto:isakmp:policy:1
(
    CipherAlg   = "GR341215M-K256-CFB-65526"
```

```
    HashAlg     = "GR341112_256TC26-65128"
    GroupID     = VKO2_1B
    RestrictAuthenticationTo = GOST_SIGN
    LifetimeSeconds = 86400
)

ESPProposal MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP
(
    Transform* = ESPTransform
    (
        CipherAlg*          = "GR341215M-K288-MGM-33"
        LifetimeSeconds     = 3600
        LifetimeKilobytes   = 4608000
    )
)

FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.1.10
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.10
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.10
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.10
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = DROP
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

FilterChain InspectChain:INSPECT (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 6
        Action = PASS
        ExtendedAction = inspect_tcp<>
        LogEventID = "Inspect:INSPECT:tcp"
    ),
```

```
    Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 17
        Action = PASS
        ExtendedAction = inspect_udp<>
        LogEventID = "Inspect:INSPECT:udp"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = PASS
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

IdentityEntry Remote:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

IdentityEntry Local:Cert:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

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

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

AuthMethodGOSTSign GOST:Sign:NEW_ALGS
(
    RemoteID             = Remote:NEW_ALGS
    LocalID              = Local:Cert:NEW_ALGS
    SendRequestMode      = ALWAYS
    SendCertMode         = ALWAYS
)

IKERule IKERule:VPN:1
(
    IKEPeerIPFilter = 172.16.100.10
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)
```

```
IPsecAction IPsecAction:VPN:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.100.10
    )
    ContainedProposals = ( MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    DFHandling = COPY
    IKERule = IKERule:VPN:1
)

FilterChain IPsecPolicy:VPN (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 10.10.1.1
        DestinationIP = 10.10.100.1
        ProtocolID = 47
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_GRE_TUNNEL_1"
    )
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/0"
    InputFilter = FilterChain:FW_INPUT_WAN
    OutputFilter = InspectChain:INSPECT
    IPsecPolicy = IPsecPolicy:VPN
)

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

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

Конфигурация криптошлюза Spoke1-2

1. Конфигурация cisco-like:

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

```
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
hostname Spoke1-2
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
!
!
!
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp profile NEW_ALGS
 match identity dn
 set identity mapping
 set policy 1
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
object-group network SELF
 remark WAN IP
 host 172.16.1.20
!
ip access-list extended IPSEC_GRE_TUNNEL_2
 permit gre host 10.10.1.2 host 10.10.100.2
!
ip access-list extended FW_INPUT_WAN
 permit udp any object-group SELF eq isakmp
 permit udp any object-group SELF eq non500-isakmp
 permit icmp any object-group SELF
 permit esp any object-group SELF
!
ipv6 default filtering deny
!
!
ip inspect tcp sessions-max 256000
ip inspect udp sessions-max 65536
ip inspect name INSPECT tcp
ip inspect name INSPECT udp
!
crypto map VPN 1 ipsec-isakmp
 match address IPSEC_GRE_TUNNEL_2
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set peer 172.16.100.20
 set dead-connection history off
!
```

```
interface GigabitEthernet0/0
 ip address 172.16.1.20 255.255.255.0
 ip access-group FW_INPUT_WAN in
 crypto map VPN
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 ip address 192.168.253.20 255.255.255.0
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
!
ip route 0.0.0.0 0.0.0.0 172.16.1.1
ip route 10.10.1.2 255.255.255.255 192.168.253.1
!
!
crypto pki trustpoint s-terra_technological_trustpoint
 revocation-check crl
 crl download group ROOTCA http://172.16.101.15/certcrl.crl
 crl download time 60
crypto pki certificate chain s-terra_technological_trustpoint
certificate 58E026BFD6D625BE4582C16C6189C183
...
AD4F8901771632E0A0AF83

quit
!
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Thu Nov 14 16:47:36 2024

GlobalParameters(
    Title                      = "This LSP was automatically generated by CSP Converter
at Thu Nov 14 16:47:36 2024 (user: cscons)"
    Version                      = LSP_5_0
    CRLHandlingMode              = ENABLE
    PreserveIPsecSA              = FALSE
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 172.16.1.1
        ),
        Route(
            Destination = 10.10.1.2/32
            Gateway = 192.168.253.1
        )

)

FirewallParameters(
    TCPSynSentTimeout = 30
    TCPFinTimeout = 5
```

```
    TCPClosedTimeout = 30
    TCPSynRcvdTimeout = 30
    TCPEstablishedTimeout = 3600
    TCPHalfOpenLow = 400
    TCPHalfOpenMax = 500
    TCPSessionRateLow = 400
    TCPSessionRateMax = 500
    TCPSessionsMax = 256000
    TCPStrictnessLevel = 3
    UDPNotEstablishedTimeout = 30
    UDPEstablishedTimeout = 120
    UDPSessionsMax = 65536
)

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

ESPProposal MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP
(
    Transform* = ESPTransform
    (
        CipherAlg*          = "GR341215M-K288-MGM-33"
        LifetimeSeconds     = 3600
        LifetimeKilobytes   = 4608000
    )
)

FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.1.20
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.20
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.20
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.20
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
```

```
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = DROP
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

FilterChain InspectChain:INSPECT (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 6
        Action = PASS
        ExtendedAction = inspect_tcp<>
        LogEventID = "Inspect:INSPECT:tcp"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 17
        Action = PASS
        ExtendedAction = inspect_udp<>
        LogEventID = "Inspect:INSPECT:udp"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = PASS
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

IdentityEntry Remote:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

IdentityEntry Local:Cert:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

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

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

AuthMethodGOSTSign GOST:Sign:NEW_ALGS
(
```

```
    RemoteID             = Remote:NEW_ALGS
    LocalID              = Local:Cert:NEW_ALGS
    SendRequestMode      = ALWAYS
    SendCertMode         = ALWAYS
)

IKERule IKERule:VPN:1
(
    IKEPeerIPFilter = 172.16.100.20
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

IPsecAction IPsecAction:VPN:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.100.20
    )
    ContainedProposals = ( MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    DFHandling = COPY
    IKERule = IKERule:VPN:1
)

FilterChain IPsecPolicy:VPN (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 10.10.1.2
        DestinationIP = 10.10.100.2
        ProtocolID = 47
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_GRE_TUNNEL_2"
    )
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/0"
    InputFilter = FilterChain:FW_INPUT_WAN
    OutputFilter = InspectChain:INSPECT
    IPsecPolicy = IPsecPolicy:VPN
)

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

NetworkInterface (
```

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

Конфигурация маршрутизатора Hub1-Router

1. Конфигурация FRR:

```
frr version 9.0.4
frr defaults traditional
hostname Hub1-Router
log syslog informational
no ipv6 forwarding
service integrated-vtysh-config
!
interface tunnel1
 ip ospf network broadcast
 no ip ospf passive
exit
!
interface tunnel2
 ip ospf network broadcast
 no ip ospf passive
exit
!
router ospf
 ospf router-id 192.168.100.1
 passive-interface default
 network 100.100.101.0/30 area 0
 network 100.100.102.0/30 area 0
 network 192.168.100.0/24 area 0
!
line vty
!
```

2. Конфигурация интерфейсов `lo1, lo2, tunnel1, tunnel2`:

```
auto lo1
iface lo1 inet static
    address 10.10.100.1
    netmask 255.255.255.255
pre-up ip link add lo1 type dummy
post-down ip link delete lo1

auto lo2
iface lo2 inet static
    address 10.10.100.2
    netmask 255.255.255.255
pre-up ip link add lo2 type dummy
post-down ip link delete lo2

auto tunnel1
iface tunnel1 inet static
    address 100.100.101.1
    netmask 255.255.255.252
pre-up ip tunnel add tunnel1 mode gre local 10.10.100.1 remote 10.10.1.1 ttl 64
pre-up ip link set tunnel1 multicast on
pre-up ip link set tunnel1 mtu 1400
post-down ip link del tunnel1

auto tunnel2
```

```
iface tunnel2 inet static
    address 100.100.102.1
    netmask 255.255.255.252
pre-up ip tunnel add tunnel2 mode gre local 10.10.100.2 remote 10.10.1.2 ttl 64
pre-up ip link set tunnel2 multicast on
pre-up ip link set tunnel2 mtu 1400
post-down ip link del tunnel2
```

Конфигурация маршрутизатора Spoke1-Router

1. Конфигурация FRR:

```
frr version 9.0.4
frr defaults traditional
hostname Spoke1-Router
log syslog informational
no ipv6 forwarding
service integrated-vtysh-config
!
interface tunnel1
 ip ospf network broadcast
 no ip ospf passive
exit
!
interface tunnel2
 ip ospf network broadcast
 no ip ospf passive
exit
!
router ospf
 ospf router-id 192.168.1.1
 passive-interface default
 network 100.100.101.0/30 area 0
 network 100.100.102.0/30 area 0
 network 192.168.1.0/24 area 0
!
line vty
!
```

2. Конфигурация интерфейсов `lo1, lo2, tunnel1, tunnel2`:

```
auto lo1
iface lo1 inet static
    address 10.10.1.1
    netmask 255.255.255.255
pre-up ip link add lo1 type dummy
post-down ip link delete lo1

auto lo2
iface lo2 inet static
    address 10.10.1.2
    netmask 255.255.255.255
pre-up ip link add lo2 type dummy
post-down ip link delete lo2

auto tunnel1
iface tunnel1 inet static
    address 100.100.101.2
    netmask 255.255.255.252
pre-up ip tunnel add tunnel1 mode gre local 10.10.1.1 remote 10.10.100.1 ttl 64
pre-up ip link set tunnel1 multicast on
pre-up ip link set tunnel1 mtu 1400
```

```
post-down ip link del tunnel1

auto tunnel2
iface tunnel2 inet static
    address 100.100.102.2
    netmask 255.255.255.252
pre-up ip tunnel add tunnel2 mode gre local 10.10.1.2 remote 10.10.100.2 ttl 64
pre-up ip link set tunnel2 multicast on
pre-up ip link set tunnel2 mtu 1400
post-down ip link del tunnel2
```