Обеспечение защищенного взаимодействия между DHCP клиентами и серверами из удаленных подсетей

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 построение защищенного соединения между центральным офисом и двумя филиалами для обеспечения безопасного взаимодействия между DHCP клиентами из удаленных подсетей филиалов и DHCP серверами в центральном офисе с использованием DHCP Relay.

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, NAT, Firewall, routing, switching, DHCP.

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 https://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://testgost2012.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 криптошлюзами должна быть обеспечена IP связность.

Схема взаимодействия

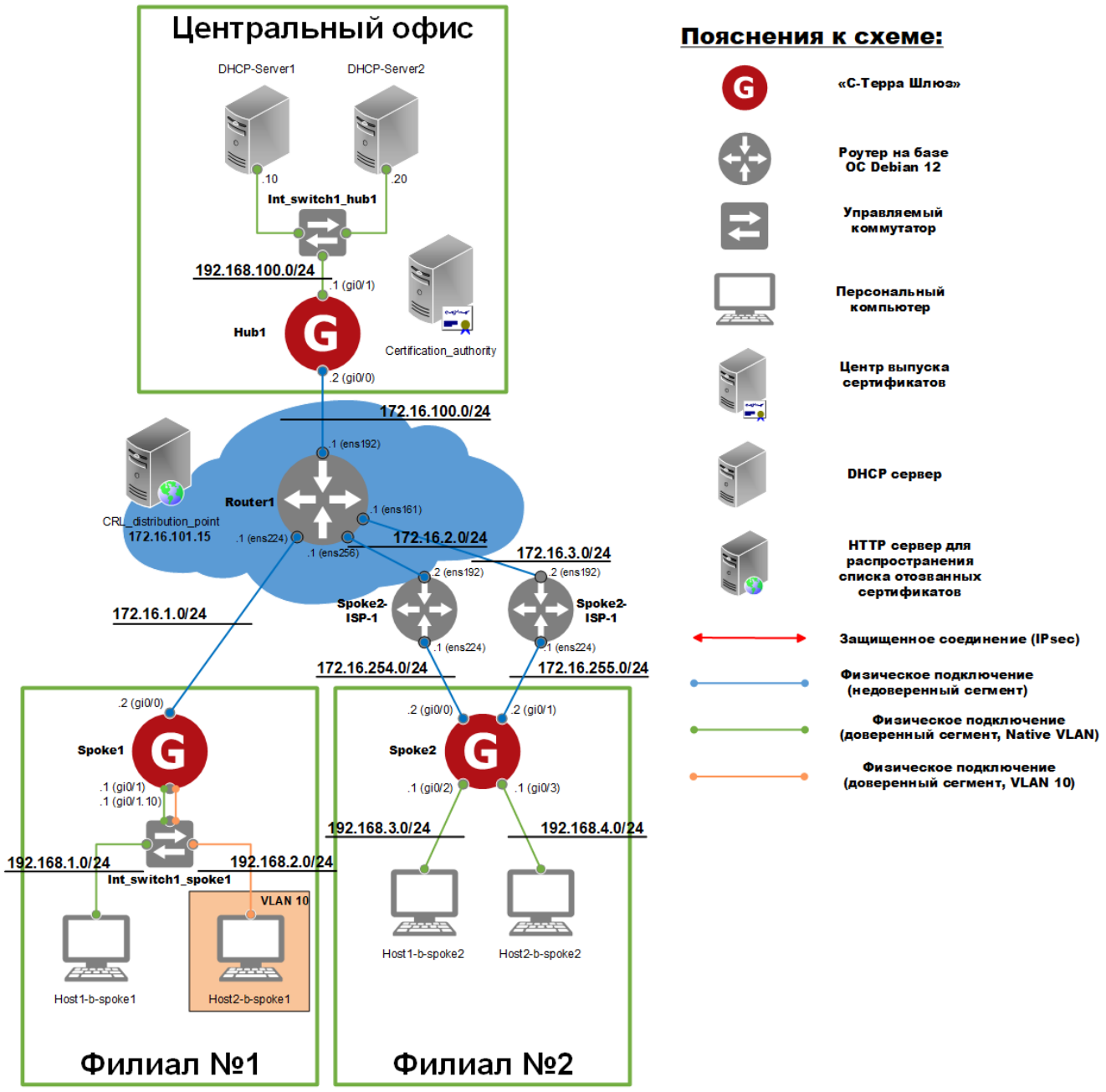


Рисунок 1. Схема взаимодействия

Общая логика работы

1. Размещение устройств.

   1.1. В центральном офисе размещаются: центр выпуска сертификатов (Certification_authority), криптошлюз «С-Терра Шлюз» (Hub1), управляемый коммутатор (Int_switch1_Hub1) и два DHCP сервера (DHCP-Server1 и DHCP-Server1).

   1.2. В филиале №1 размещаются: криптошлюз «С-Терра Шлюз» (Spoke1), управляемый коммутатор (Int_switch1_Spoke1) и персональные компьютеры (Host1-b-spoke1 и Host2-b-spoke1).

   1.3. В филиале №2 размещаются: криптошлюз «С-Терра Шлюз» (Spoke2), и персональные компьютеры (Host1-b-spoke2 и Host2-b-spoke2).

   1.4. В неконтролируемом сегменте (синее облако на схеме) размещаются: HTTP сервер для распространения списка отозванных сертификатов (CRL_distribution_point), маршрутизаторы (Router1, Spoke2-ISP-1 и Spoke2-ISP-2).

2. Подключение к сети Интернет.
   В данном сценарии для эмуляции сети Интернет используется маршрутизатор Router1.

   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).

   2.1. Криптошлюз Hub1 подключается к сети Интернет с помощью статической маршрутизации (маршрут по умолчанию через маршрутизатор Router1).

   2.2. Криптошлюз Spoke1 подключается к сети Интернет с помощью статической маршрутизации (маршрут по умолчанию через маршрутизатор Router1).

   2.3. Криптошлюз Spoke2 подключается к сети Интернет с помощью `multiwan` – метода резервирования каналов связи.

3. DHCP серверы.

   В центральном офисе находятся два DHCP сервера, работающие в режиме отказоустойчивости `DHCP failover`.

   Для того, чтобы DHCP клиенты из удаленных подсетей за криптошлюзами Spoke1 и Spoke2 могли получить сетевые настройки с серверов в центре, на Spoke1 и Spoke2 работает DHCP Relay Agent, который перехватывает широковещательные запросы клиентов, отправляет их на DHCP сервера и перенаправляет ответы обратно клиентам.

4. DHCP Relay.

   DHCP Relay используется в случаях, когда DHCP клиенты и DHCP сервера находятся в разных подсетях (разных широковещательных доменах).

   За работу DHCP Relay агента на шлюзе отвечает сервис `st-dhcp-relay`. По умолчанию данный сервис выключен и не добавлен в автозапуск. DHCP Relay настраивается с помощью конфигурационного файла в формате YAML `/opt/st-dhcp-relay/etc/config.yaml`.

   Принцип работы DHCP Relay.

   Рассмотрим принцип работы на примере следующего конфигурационного файла:

```
instances:
  ipv4:
    -
      instance_num: 1
      helper_addresses:
        - "192.168.100.10"
      downstream_interfaces:
        - name: "eth1"
```

```
    upstream_interfaces:
      - name: "eth0"
    log_level: "debug"
```

- `instances` – секция, содержащая в себе секции `ipv4` и/или `ipv6` (в данном примере секция `ipv6` отсутствует);
- `ipv4` – секция, содержащая список IPv4 инстансов (процессов) DHCP Relay. Инстанс представляет собой список параметров. В данном примере описывается один IPv4 инстанс DHCP Relay.
- `instance_num` – параметр, определяющий номер данного инстанса DHCP Relay. У каждого инстанса должен быть свой уникальный номер.
- `helper_addresses` – параметр, содержащий список адресов DHCP серверов, на которые перенаправляются запросы DHCP клиентов. Если указано несколько адресов, то запросы отправляются на все из них одновременно.
- `downstream_interfaces` – параметр, содержащий список интерфейсов (в нотации Linux), на которых слушаются и перехватываются широковещательные запросы DHCP клиентов.
- `upstream_interfaces` – параметр, содержащий список интерфейсов (в нотации Linux), с которых отправляются запросы и принимаются ответы от DHCP сервера.
- `log_level` – параметр, определяющий детальность сообщений, пишущихся в лог данным инстансом. Сообщения пишутся в файл `/var/log/cspvpngate.log`.

Подробный список допустимых параметров конфигурационного файла и их описание приведены в документации на сайте https://doc.s-terra.ru.

5. Параметры безопасного взаимодействия.

   Весь IP трафик между подсетями центрального офиса (192.168.100.0/24) и филиалов (192.168.1.0/24, 192.168.2.0/24, 192.168.3.0/24, 192.168.4.0/24) защищается с использованием алгоритмов ГОСТ и протокола IPsec в туннельном режиме.

   Инициировать защищенное соединение может как трафик из филиалов в центр, так и наоборот, из центра в филиалы.

   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 – 172.16.100.1 и маску – 255.255.255.0 на сетевом интерфейсе ens192.
2. Настройте IP адрес – 172.16.1.1 и маску – 255.255.255.0 на сетевом интерфейсе ens224.
3. Настройте IP адрес – 172.16.2.1 и маску – 255.255.255.0 на сетевом интерфейсе ens256.
4. Настройте IP адрес – 172.16.3.1 и маску – 255.255.255.0 на сетевом интерфейсе ens161.

5. Задайте маршрут до сети 172.16.254.0/24 через 172.16.2.2 (маршрутизатор Spoke2-ISP-1).
6. Задайте маршрут до сети 172.16.255.0/24 через 172.16.3.2 (маршрутизатор Spoke2-ISP-2).
7. Разрешите прохождение IP трафика.

Настройка устройства Spoke2-ISP-1

1. Настройте IP адрес – 172.16.2.2 и маску – 255.255.255.0 на сетевом интерфейсе ens192.
2. Настройте IP адрес – 172.16.254.1 и маску – 255.255.255.0 на сетевом интерфейсе ens224.
3. Задайте маршрут по умолчанию через 172.16.2.1 (маршрутизатор Router1).
4. Разрешите прохождение IP трафика.

Настройка устройства Spoke2-ISP-2

1. Настройте IP адрес – 172.16.3.2 и маску – 255.255.255.0 на сетевом интерфейсе ens192.
2. Настройте IP адрес – 172.16.255.1 и маску – 255.255.255.0 на сетевом интерфейсе ens224.
3. Задайте маршрут по умолчанию через 172.16.3.1 (маршрутизатор Router1).
4. Разрешите прохождение IP трафика.

Настройка устройства Int_switch1_Spoke1

1. Интерфейс `gi0/0` должен быть в trunk режиме. Должно быть разрешено прохождения фреймов с VLAN ID 10 и других.
2. Интерфейс `gi0/1` должен быть в access режиме (с VLAN тэгом или без него).
3. Интерфейс `gi0/2` должен быть в access режиме с VLAN ID 10.

Настройка устройства Host1-b-spoke1

1. Настройте получение IP адреса с помощью DHCP на сетевом интерфейсе.
2. Разрешите прием и отправку IP трафика.

Настройка устройства Host2-b-spoke1

1. Настройте получение IP адреса с помощью DHCP на сетевом интерфейсе.
2. Разрешите прием и отправку IP трафика.

Настройка устройства Host1-b-spoke2

1. Настройте получение IP адреса с помощью DHCP на сетевом интерфейсе.
2. Разрешите прием и отправку IP трафика.

Настройка устройства Host2-b-spoke2

1. Настройте получение IP адреса с помощью DHCP на сетевом интерфейсе.
2. Разрешите прием и отправку IP трафика.

Настройка устройства DHCP-Server1

1. Настройте IP адреса – 192.168.100.10 и маску – 255.255.255.0 на сетевом интерфейсе.
2. Сконфигурируйте работу DHCP сервера, задайте пулы адресов и настройте режим DHCP failover для отказоустойчивости. Пример конфигурации в Приложении.

Настройка устройства DHCP-Server2

1. Настройте IP адреса – 192.168.100.20 и маску – 255.255.255.0 на сетевом интерфейсе.
2. Сконфигурируйте работу DHCP сервера, задайте пулы адресов и настройте режим DHCP failover для отказоустойчивости. Пример конфигурации в Приложении.

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 (в нашем примере это подсети 192.168.100.0/24, 192.168.1.0/24, 192.168.2.0/24, 192.168.3.0/24, 192.168.4.0/24. 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 в cisco-like консоли применяются сразу после ввода соответствующих команд. Политика безопасности применяются после выхода из режима конфигурирования.

1. Задайте имя устройства:

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

2. Задайте IP адрес в соответствии со схемой стенда на внутреннем `GigabitEthernet0/1` интерфейсе:

```
Hub1(config)#interface GigabitEthernet 0/1
Hub1(config-if)#ip address 192.168.100.1 255.255.255.0
Hub1(config-if)#no shutdown
Hub1(config-if)#exit
```

3. Задайте IP адрес в соответствии со схемой стенда на внешнем `GigabitEthernet0/0` интерфейсе:

```
Hub1(config)#interface GigabitEthernet 0/0
Hub1(config-if)#ip address 172.16.100.2 255.255.255.0
Hub1(config-if)#no shutdown
Hub1(config-if)#exit
```

4. Задайте маршрут по умолчанию через устройство Router1:

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

5. Проверьте доступность устройства Router1:

```
Hub1(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. Задайте общие параметры IKE:

```
Hub1(config)#crypto isakmp identity dn
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
```

1.1. Создайте политику, описывающую параметры IKE туннеля:

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

1.2. Создайте профиль настроек ISAKMP (алгоритм IKE) под именем `NEW_ALGS`:

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

2. Параметры IPsec.

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

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

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

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

2.3. Создайте список доступа (ACL) для трафика, который нужно защищать между центральным офисом 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)#permit ip 192.168.100.0 0.0.0.255 192.168.2.0 0.0.0.255
Hub1(config-ext-nacl)#exit
Hub1(config)#
```

2.4. Создайте список доступа (ACL) для трафика, который нужно защищать между центральным офисом и филиалом №2:

```
Hub1(config)#ip access-list extended IPSEC_ACl_HUB1_AND_SPOKE1
Hub1(config-ext-nacl)#permit ip 192.168.100.0 0.0.0.255 192.168.3.0 0.0.0.255
Hub1(config-ext-nacl)#permit ip 192.168.100.0 0.0.0.255 192.168.4.0 0.0.0.255
Hub1(config-ext-nacl)#exit
Hub1(config)#
```

2.5. Создайте крипто-карту (имя VPN, раздел 1) для построения IPsec туннеля между центральным офисом и филиалом №1:

```
Hub1(config)#crypto map VPN 1 ipsec-isakmp
Hub1(config-crypto-map)#match address IPSEC_ACl_HUB1_AND_SPOKE1
Hub1(config-crypto-map)#set transform-set MAGMA_ENCRYPT_AND_INTEGRITY
Hub1(config-crypto-map)#set peer 172.16.1.2
Hub1(config-crypto-map)#set isakmp-profile NEW_ALGS
Hub1(config-crypto-map)#set pre-fragmentation
Hub1(config-crypto-map)#set security-association mss-fix out
Hub1(config-crypto-map)#set dead-connection history off
Hub1(config-crypto-map)#exit
```

2.6. Создайте крипто-карту (имя VPN, раздел 2) для построения IPsec туннеля между центральным офисом и филиалом №2:

```
Hub1(config)#crypto map VPN 2 ipsec-isakmp
Hub1(config-crypto-map)#match address IPSEC_ACl_HUB1_AND_SPOKE2
Hub1(config-crypto-map)#set transform-set GOST_ENCRYPT_AND_INTEGRITY
Hub1(config-crypto-map)#set peer 172.16.254.2
Hub1(config-crypto-map)#set peer 172.16.255.2
Hub1(config-crypto-map)#set isakmp-profile NEW_ALGS
Hub1(config-crypto-map)#set pre-fragmentation
Hub1(config-crypto-map)#set security-association mss-fix out
Hub1(config-crypto-map)#set dead-connection history off
Hub1(config-crypto-map)#exit
```

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

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

2.8. Примените настройки:

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

Настройка политики обработки СОС (CRL)

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

Включите проверку СОС 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 с целью дальнейшего его использования в ACL:

```
Hub1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Hub1(config)#object-group network SELF
Hub1(config-network-group)#remark WAN IP
Hub1(config-network-group)#host 172.16.100.2
Hub1(config-network-group)#exit
```

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

```
Hub1(config)#ip access-list extended FW_INPUT_WAN
Hub1(config-ext-nacl)#permit udp any object-group SELF eq 500
Hub1(config-ext-nacl)#permit udp any object-group SELF eq 4500
Hub1(config-ext-nacl)#permit icmp any object-group SELF
Hub1(config-ext-nacl)#permit esp any object-group SELF
Hub1(config-ext-nacl)#exit
```

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

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

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

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

5. Примените настройки:

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

Настройка криптошлюза Hub1 завершена.

В Приложении представлены тексты конфигураций для криптошлюза Hub1:

- текст консоли cisco-like;
- текст LSP.

Настройка криптошлюза Spoke1

Начальные настройки

Настройка происходит аналогично настройке Hub1.

Настройки PKI (запросы и сертификаты)

Настройка происходит аналогично настройке Hub1.

Настройка паролей доступа к консоли cisco-like

Настройка происходит аналогично настройке Hub1.

Настройка сетевых параметров

1. Задайте имя устройства:

```
sterragate(config)#hostname Spoke1
```

2. Задайте IP адрес в соответствии со схемой стенда на внешнем `GigabitEthernet0/0` интерфейсе:

```
Spoke1(config)#interface GigabitEthernet 0/0
Spoke1(config-if)#ip address 172.16.1.2 255.255.255.0
Spoke1(config-if)#no shutdown
Spoke1(config-if)#exit
```

3. Задайте IP адрес в соответствии со схемой стенда на внутреннем `GigabitEthernet0/1` интерфейсе:

```
Spoke1(config)#interface GigabitEthernet 0/1
Spoke1(config-if)#ip address 192.168.1.1 255.255.255.0
Spoke1(config-if)#no shutdown
Spoke1(config-if)#exit
```

4. Создайте 802.1Q интерфейс `GigabitEthernet0/1.10` (VLAN ID 10) (в нотации Linux имя этого интерфейса будет eth1_10) и задайте IP адрес в соответствии со схемой стенда:

```
Spoke1(config)#interface gigabitEthernet 0/1.10
Spoke1(config-if)#ip address 192.168.2.1 255.255.255.0
Spoke1(config-if)#no shutdown
Spoke1(config-if)#exit
```

5. Задайте маршрут по умолчанию через устройство Router1:

```
Spoke1(config)#ip route 0.0.0.0 0.0.0.0 172.16.1.1
```

6. Проверьте доступность устройства Router1:

```
Spoke1(config)#do 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=64 time=0.316 ms
108 bytes from 172.16.1.1: icmp_seq=2 ttl=64 time=0.306 ms
108 bytes from 172.16.1.1: icmp_seq=3 ttl=64 time=0.421 ms
108 bytes from 172.16.1.1: icmp_seq=4 ttl=64 time=0.500 ms
108 bytes from 172.16.1.1: icmp_seq=5 ttl=64 time=0.421 ms

--- 172.16.1.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4024ms
rtt min/avg/max/mdev = 0.306/0.392/0.500/0.072 ms
```

Настройка шифрования

Настройка происходит аналогично настройке Hub1.

Настройка политики обработки COC (CRL)

Настройка происходит аналогично настройке Hub1.

Настройка межсетевого экрана

Настройка происходит аналогично настройке Hub1.

Настройка DHCP Relay

Настройка выполняется из linux bash консоли криптошлюза.

1. Откройте на редактирование файл `/opt/st-dhcp-relay/etc/config.yaml` и внесите в него следующие настройки:

```
instances:
  ipv4:
    -
      instance_num: 1
      helper_addresses:
        - "192.168.100.10"
        - "192.168.100.20"
      upstream_interfaces:
        - name: "eth0"
      downstream_interfaces:
        - name: "eth1"
        - name: "eth1_10"
```

2. Запустите сервис `st-dhcp-relay` и добавьте его в автозапуск:

```
root@Spoke1:~# systemctl start st-dhcp-relay.service
root@Spoke1:~# systemctl enable st-dhcp-relay.service

root@Spoke1:~# systemctl status st-dhcp-relay.service
```

```
● st-dhcp-relay.service - S-Terra DHCP Relay Service
     Loaded: loaded (/lib/systemd/system/st-dhcp-relay.service; enabled; preset:
disabled)
     Active: active (exited) since Wed 2023-05-17 14:34:46 MSK; 18min ago
   Main PID: 55518 (code=exited, status=0/SUCCESS)
        CPU: 87ms

May 17 14:34:46 Spoke1 systemd[1]: Starting st-dhcp-relay.service - S-Terra DHCP Relay
Service...
May 17 14:34:46 Spoke1 dhcprelay_launcher[55518]: INFO: Starting S-Terra DHCP Relay.
May 17 14:34:46 Spoke1 dhcprelay_launcher[55518]: INFO: Starting
dhcrelay_ipv4_instance@1.service.
May 17 14:34:46 Spoke1 systemd[1]: Finished st-dhcp-relay.service - S-Terra DHCP Relay
Service.
```

Примечание:

В «С-Терра Шлюз» версии 5.0.25337 сервис DHCP Relay зависит от сервиса `vpngate.service` таким образом, что при остановке `vpngate.service` остановится и DHCP Relay, но при этом не будет автоматически запущен при повторном старте `vpngate.service` (потребуется запускать DHCP Relay вручную). Если такая логика работы не подходит по тем или иным причинам, то можно изменить параметры сервиса DHCP Relay так, чтобы он не останавливался при остановке `vpngate.service`. Для этого выполните следующие действия:

1. Файл `/usr/lib/systemd/system/st-dhcp-relay.service` приведите к следующему виду:

```
[Unit]
Description=S-Terra DHCP Relay Service
After=vpngate.service
Requires=networking.service
After=network-online.target

[Service]
Type=oneshot
```

```
TimeoutSec=60
StandardOutput=journal
RemainAfterExit=yes

ExecStart=/opt/st-dhcp-relay/bin/dhcprelay_launcher.py start
ExecStop=/opt/st-dhcp-relay/bin/dhcprelay_launcher.py stop

[Install]
WantedBy=vpngate.service
```

2. Файл `/usr/lib/systemd/system/dhcrelay_ipv4_instance@.service` приведите к следующему виду:

```
[Unit]
Description=S-Terra DHCP IPv4 Relay instance %i
After=vpngate.service

[Service]
Type=forking
TimeoutSec=60
StandardOutput=journal
KillSignal=SIGINT
EnvironmentFile=/run/st-dhcp-relay/configs/ipv4_instance_%i.conf
PIDFile=/run/st-dhcp-relay/ipv4_instance_%i.pid

ExecStart=/usr/sbin/dhcrelay -pf /run/st-dhcp-relay/ipv4_instance_%i.pid $ARGS
ExecStopPost=-/bin/rm -f /var/run/st-dhcp-relay/ipv4_instance_%i.pid
```

3. Файл `/usr/lib/systemd/system/dhcrelay_ipv6_instance@.service` приведите к следующему виду:

```
[Unit]
Description=S-Terra DHCP IPv6 Relay instance %i
After=vpngate.service

[Service]
Type=forking
TimeoutSec=60
StandardOutput=journal
KillSignal=SIGINT
EnvironmentFile=/run/st-dhcp-relay/configs/ipv6_instance_%i.conf
PIDFile=/run/st-dhcp-relay/ipv6_instance_%i.pid

ExecStart=/usr/sbin/dhcrelay -pf /run/st-dhcp-relay/ipv6_instance_%i.pid $ARGS
ExecStopPost=-/bin/rm -f /var/run/st-dhcp-relay/ipv6_instance_%i.pid
```

4. После редактирования файлов обязательно выполните следующие команды для применения обновленной конфигурации и пересчета контрольных сумм:

```
root@Spoke1:~# systemctl daemon-reload
root@Spoke1:~# /opt/VPNagent/bin/links_verify update
```

```
INFO [links_verify]: Creating new links hashes list...
INFO [links_verify]: Completed
```

В Приложении представлены тексты конфигураций для криптошлюза Spoke1:

- текст консоли cisco-like;
- текст LSP;
- текст конфигурации DHCP Relay.

Настройка криптошлюза Spoke2

Настройка происходит аналогично настройке Spoke1. Настройка multiwan выполняется в соответствии со сценарием «Резервирование каналов связи при помощи пакета «multiwan».

В Приложении представлены тексты конфигураций для криптошлюза Spoke2:

- текст консоли cisco-like;
- текст LSP;
- текст конфигурации multiwan;
- текст конфигурации DHCP Relay.

Проверка работоспособности стенда

Проверку работоспособности нужно выполнять после завершения настройки всех устройств стенда.

Проверка IP связности

1. Проверьте, что с криптошлюза Spoke1 доступен по ICMP криптошлюз Hub1 (по внешним интерфейсам). Для этого выполните команду `ping` из cisco-like консоли криптошлюзов.

   ```
   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=63 time=0.840 ms
   108 bytes from 172.16.100.2: icmp_seq=2 ttl=63 time=0.896 ms
   108 bytes from 172.16.100.2: icmp_seq=3 ttl=63 time=0.785 ms
   108 bytes from 172.16.100.2: icmp_seq=4 ttl=63 time=0.952 ms
   108 bytes from 172.16.100.2: icmp_seq=5 ttl=63 time=1.03 ms

   --- 172.16.100.2 ping statistics ---
   5 packets transmitted, 5 received, 0% packet loss, time 4003ms
   rtt min/avg/max/mdev = 0.785/0.899/1.025/0.083 ms
   ```

   Видно, что с криптошлюза Spoke1 доступен по ICMP криптошлюз Hub1.

2. Проверьте, что с криптошлюза Spoke2 доступен по ICMP криптошлюз Hub1 (по внешним интерфейсам). Для этого выполните команду `ping` из cisco-like консоли криптошлюзов.

   ```
   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=62 time=0.824 ms
   108 bytes from 172.16.100.2: icmp_seq=2 ttl=62 time=1.04 ms
   108 bytes from 172.16.100.2: icmp_seq=3 ttl=62 time=0.910 ms
   108 bytes from 172.16.100.2: icmp_seq=4 ttl=62 time=1.28 ms
   108 bytes from 172.16.100.2: icmp_seq=5 ttl=62 time=0.788 ms

   --- 172.16.100.2 ping statistics ---
   5 packets transmitted, 5 received, 0% packet loss, time 3998ms
   rtt min/avg/max/mdev = 0.788/0.968/1.278/0.177 ms
   ```

   Видно, что с криптошлюза Spoke2 доступен по ICMP криптошлюз Hub1.

Проверка PKI

1. Проверьте, что на криптошлюзах Hub1, Spoke1 и Spoke2 СОС импортирован в базу Продукта. Для этого выполните команду `run cert_mgr show` из cisco-like консоли криптошлюзов. Например, на Hub1:

   ```
   Hub1#run cert_mgr show
   ```

   ```
   Found 2 certificates. Found 1 CRL.
   1 Status: local   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Hub1
   2 Status: trusted C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root CA
   3 CRL: C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root CA
   ```

   Видно, что СОС импортирован в базу Продукта. Если этого не произошло, то проверьте файл журнала (команда `run grep getcrls_daemon /var/log/cspvpngate.log`) и, при необходимости, перезапустите сервис автоматической загрузки СОС (команда `run systemctl restart getcrls.service`).

2. Проверьте, что на криптошлюзах Hub1, Spoke1 и Spoke2 статус всех сертификатов Active. Для этого выполните команду `run cert_mgr check` из cisco-like консоли криптошлюзов. Например, на Hub1:

   ```
   Hub1#run cert_mgr check
   ```

   ```
   1 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Hub1
   2 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root CA
   ```

Видно, что статус всех сертификатов Active. Если статус Inactive, то проверьте загружен ли СОС в базу Продукта и правильность установки даты и времени.

Проверка построения защищенного соединения

1. Инициируйте построение защищенного соединения между криптошлюзами Hub1 и Spoke1. Для этого выполните запрос на получение IP адреса по DHCP с устройств host1-b-spoke1 и host2-b-spoke1. Пример для host1-b-spoke1:

```
root@host1-b-spoke1:~# dhclient -i ens192
```

2. Инициируйте построение защищенного соединения между криптошлюзами Hub1 и Spoke2. Для этого выполните запрос на получение IP адреса по DHCP с устройств host1-b-spoke2 и host2-b-spoke2. Пример для host1-b-spoke2:

```
root@host1-b-spoke2:~# dhclient -i ens192
```

3. Проверьте, что на криптошлюзах Hub1 и Spoke1 установлены защищенные IPsec соединения. Для этого выполните команду `run sa_mgr show` из cisco-like консоли криптошлюзов.

```
Hub1#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 (172.16.100.2,500)-(172.16.1.2,500) active 2288 2368
2 2 (172.16.100.2,500)-(172.16.254.2,500) active 2288 2368

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (192.168.100.0-192.168.100.255,*)-(192.168.1.0-192.168.1.255,*) * ESP tunn 1104
1032
2 2 (192.168.100.0-192.168.100.255,*)-(192.168.2.0-192.168.2.255,*) * ESP tunn 1472
1432
3 3 (192.168.100.0-192.168.100.255,*)-(192.168.3.0-192.168.3.255,*) * ESP tunn 1472
1432
4 4 (192.168.100.0-192.168.100.255,*)-(192.168.4.0-192.168.4.255,*) * ESP tunn 1104
1088
```

```
Spoke1#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 (172.16.1.2,500)-(172.16.100.2,500) active 2368 2288

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (192.168.1.0-192.168.1.255,*)-(192.168.100.0-192.168.100.255,*) * ESP tunn 1376
1104
2 2 (192.168.2.0-192.168.2.255,*)-(192.168.100.0-192.168.100.255,*) * ESP tunn 1432
1472
```

```
Spoke2#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 (172.16.254.2,500)-(172.16.100.2,500) active 2368 2288

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (192.168.3.0-192.168.3.255,*)-(192.168.100.0-192.168.100.255,*) * ESP tunn 1432
1472
```

```
2 2 (192.168.4.0-192.168.4.255,*)-(192.168.100.0-192.168.100.255,*) * ESP tunn 1432
1104
```

Видно, что на криптошлюзах Hub1, Spoke1 и Spoke2 установлены защищенные IPsec соединения. Если этого не произошло, то проверьте файл журнала (команда `run less /var/log/cspvpngate.log`). При необходимости увеличьте уровень логирования (команда `logging trap debugging` в консоли cisco-like) и заново инициируйте защищенное соединение.

Приложение

Конфигурации криптошлюза Hub1

4. Консоль 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 ipsec security-association lifetime kilobytes unlimited
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
aaa new-model
!
!
hostname Hub1
enable secret 5 HGJcyG+CRmCjINGFkW48VQ==
!
!
!
!
!
!
!
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_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
object-group network SELF
 remark WAN IP
 host 172.16.100.2
!
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
 permit ip 192.168.100.0 0.0.0.255 192.168.2.0 0.0.0.255
!
ip access-list extended IPSEC_ACl_HUB1_AND_SPOKE2
 permit ip 192.168.100.0 0.0.0.255 192.168.3.0 0.0.0.255
 permit ip 192.168.100.0 0.0.0.255 192.168.4.0 0.0.0.255
!
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_ACl_HUB1_AND_SPOKE1
 set transform-set MAGMA_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set security-association mss-fix out
 set peer 172.16.1.2
 set pre-fragmentation
 set dead-connection history off
crypto map VPN 2 ipsec-isakmp
 match address IPSEC_ACl_HUB1_AND_SPOKE2
 set transform-set MAGMA_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set security-association mss-fix out
 set peer 172.16.254.2
 set peer 172.16.255.2
 set pre-fragmentation
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 172.16.100.2 255.255.255.0
 ip access-group FW_INPUT_WAN in
 crypto map VPN
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 ip address 192.168.100.1 255.255.255.0
!
!
ip route 0.0.0.0 0.0.0.0 172.16.100.1
!
!
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
```

5. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Fri Jan 19 10:07:38 2024

GlobalParameters(
    Title                      = "This LSP was automatically generated by CSP Converter
at Fri Jan 19 10:07:38 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
        )

)

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

FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.100.2
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.2
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.2
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.2
        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 = 172.16.1.2
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

IKERule IKERule:VPN:2
(
    IKEPeerIPFilter = 172.16.254.2, 172.16.255.2
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 20
)

IPsecAction IPsecAction:VPN:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.1.2
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule:VPN:1
)

IPsecAction IPsecAction:VPN:2
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.254.2
        Prefragmentation = ENABLE
    ),
    TunnelEntry(
        PeerAddress = 172.16.255.2
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule:VPN:2
)

FilterChain IPsecPolicy:VPN (
    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:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_ACl_HUB1_AND_SPOKE1"
    ),
    Filter (
        SourceIP = 192.168.100.0/24
        DestinationIP = 192.168.2.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_ACl_HUB1_AND_SPOKE1"
    ),
    Filter (
        Label = "VPN:2:IPSEC_ACl_HUB1_AND_SPOKE2"
        SourceIP = 192.168.100.0/24
        DestinationIP = 192.168.3.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:2 >
        LogEventID = "IPsec:Protect:VPN:2:IPSEC_ACl_HUB1_AND_SPOKE2"
    ),
    Filter (
        SourceIP = 192.168.100.0/24
        DestinationIP = 192.168.4.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:2 >
        LogEventID = "IPsec:Protect:VPN:2:IPSEC_ACl_HUB1_AND_SPOKE2"
    )
)

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

NetworkInterface (
    LogicalName = "GigabitEthernet0/1"
    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 dn
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
aaa new-model
!
!
hostname Spoke1
enable secret 5 HGJcyG+CRmCjINGFkW48VQ==
!
!
!
!
```

```
!
!
!
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_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
object-group network SELF
 remark WAN IP
 host 172.16.1.2
!
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
 permit ip 192.168.2.0 0.0.0.255 192.168.100.0 0.0.0.255
!
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_ACl_HUB1_AND_SPOKE1
 set transform-set MAGMA_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 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 172.16.1.2 255.255.255.0
 ip access-group FW_INPUT_WAN in
 crypto map VPN
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 ip address 192.168.1.1 255.255.255.0
!
interface GigabitEthernet0/1.10
 ip address 192.168.2.1 255.255.255.0
!
!
ip route 0.0.0.0 0.0.0.0 172.16.1.1
!
!
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:   Fri Jan 19 10:09:13 2024

GlobalParameters(
    Title                       = "This LSP was automatically generated by CSP Converter
at Fri Jan 19 10:09:13 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
        )

)

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

```
FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.1.2
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.2
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.2
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.2
        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 = 172.16.100.2
    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 = 172.16.100.2
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = 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 = 192.168.1.0/24
        DestinationIP = 192.168.100.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_ACl_HUB1_AND_SPOKE1"
```

```
    ),
    Filter (
        SourceIP = 192.168.2.0/24
        DestinationIP = 192.168.100.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_ACl_HUB1_AND_SPOKE1"
    )
)

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 = "GigabitEthernet0/1.10"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)
```

3. Конфигурация DHCP Relay:

```
instances:
  ipv4:
    -
      instance_num: 1
      helper_addresses:
        - "192.168.100.10"
        - "192.168.100.20"
      upstream_interfaces:
        - name: "eth0"
      downstream_interfaces:
        - name: "eth1"
        - name: "eth1_10"
```

Конфигурации криптошлюза Spoke2

1. Консоль 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 ipsec security-association lifetime kilobytes unlimited
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
aaa new-model
!
!
hostname Spoke2
enable secret 5 HGJcyG+CRmCjINGFkW48VQ==
```

```
!
!
!
!
!
!
!
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_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
object-group network SELF
 remark WAN IP
 host 172.16.254.2
 host 172.16.255.2
!
ip access-list extended IPSEC_ACl_HUB1_AND_SPOKE2
 permit ip 192.168.3.0 0.0.0.255 192.168.100.0 0.0.0.255
 permit ip 192.168.4.0 0.0.0.255 192.168.100.0 0.0.0.255
!
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_MAIN 1 ipsec-isakmp
 match address IPSEC_ACl_HUB1_AND_SPOKE2
 set transform-set MAGMA_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set security-association mss-fix out
 set local-address 172.16.254.2
 set peer 172.16.100.2
 set pre-fragmentation
 set dead-connection history off
!
crypto map VPN_BACKUP 1 ipsec-isakmp
 match address IPSEC_ACl_HUB1_AND_SPOKE2
 set transform-set MAGMA_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set security-association mss-fix out
 set local-address 172.16.255.2
 set peer 172.16.100.2
 set pre-fragmentation
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 172.16.254.2 255.255.255.0
 ip access-group FW_INPUT_WAN in
```

```
 crypto map VPN_MAIN
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 ip address 172.16.255.2 255.255.255.0
 ip access-group FW_INPUT_WAN in
 crypto map VPN_BACKUP
 ip inspect INSPECT out
!
interface GigabitEthernet0/2
 ip address 192.168.3.1 255.255.255.0
!
interface GigabitEthernet0/3
 ip address 192.168.4.1 255.255.255.0
!
!
!
!
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:   Fri Jan 19 10:10:48 2024

GlobalParameters(
    Title                       = "This LSP was automatically generated by CSP Converter
at Fri Jan 19 10:10:48 2024 (user: cscons)"
    Version                     = LSP_5_0
    CRLHandlingMode             = ENABLE
    PreserveIPsecSA             = FALSE
)

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

FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.254.2
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.255.2
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.254.2
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.255.2
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.254.2
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.255.2
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.254.2
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.255.2
        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_MAIN:1
(
    IKEPeerIPFilter = 172.16.100.2
    IKELocalIPFilter = 172.16.254.2
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

IKERule IKERule:VPN_BACKUP:1
(
    IKEPeerIPFilter = 172.16.100.2
    IKELocalIPFilter = 172.16.255.2
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 20
)

IPsecAction IPsecAction:VPN_MAIN:1
(
    TunnelingParameters = TunnelEntry(
        LocalIPAddress = 172.16.254.2
        PeerAddress = 172.16.100.2
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule:VPN_MAIN:1
)

IPsecAction IPsecAction:VPN_BACKUP:1
(
    TunnelingParameters = TunnelEntry(
        LocalIPAddress = 172.16.255.2
        PeerAddress = 172.16.100.2
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule:VPN_BACKUP:1
)

FilterChain IPsecPolicy:VPN_MAIN (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 192.168.3.0/24
        DestinationIP = 192.168.100.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN_MAIN:1 >
```

```
        LogEventID = "IPsec:Protect:VPN_MAIN:1:IPSEC_ACl_HUB1_AND_SPOKE2"
    ),
    Filter (
        SourceIP = 192.168.4.0/24
        DestinationIP = 192.168.100.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN_MAIN:1 >
        LogEventID = "IPsec:Protect:VPN_MAIN:1:IPSEC_ACl_HUB1_AND_SPOKE2"
    )
)

FilterChain IPsecPolicy:VPN_BACKUP (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 192.168.3.0/24
        DestinationIP = 192.168.100.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN_BACKUP:1 >
        LogEventID = "IPsec:Protect:VPN_BACKUP:1:IPSEC_ACl_HUB1_AND_SPOKE2"
    ),
    Filter (
        SourceIP = 192.168.4.0/24
        DestinationIP = 192.168.100.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN_BACKUP:1 >
        LogEventID = "IPsec:Protect:VPN_BACKUP:1:IPSEC_ACl_HUB1_AND_SPOKE2"
    )
)

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

NetworkInterface (
    LogicalName = "GigabitEthernet0/1"
    InputFilter = FilterChain:FW_INPUT_WAN
    OutputFilter = InspectChain:INSPECT
    IPsecPolicy = IPsecPolicy:VPN_BACKUP
)

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

NetworkInterface (
    LogicalName = "GigabitEthernet0/3"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)
```

3. Конфигурация `multiwan`:

```
config_format_version: 1
links_params:
  check_links_reachability_on_start: true
  preempt: no
```

```
  links:
    -
      iface_name: eth0
      name: ISP 1
      prio: 200
      routes:
        ipv4:
          networks:
            - 0.0.0.0/0
          nexthop: 172.16.254.1
      check:
        hosts_reachability: any
        ipv4:
          hosts:
            - 172.16.101.15
        method:
          icmp:
            interval: 1
            restore_interval: 1
            restore_count: 3
            failure_count: 3

    -
      iface_name: eth1
      name: ISP 2
      prio: 100
      routes:
        ipv4:
          networks:
            - 0.0.0.0/0
          nexthop: 172.16.255.1
      check:
        hosts_reachability: any
        ipv4:
          hosts:
            - 172.16.101.15
        method:
          icmp:
            interval: 1
            restore_interval: 1
            restore_count: 3
            failure_count: 3
```

4. Конфигурация DHCP Relay:

```
instances:
  ipv4:
    -
      instance_num: 1
      helper_addresses:
        - "192.168.100.10"
        - "192.168.100.20"
      upstream_interfaces:
        - name: "eth0"
        - name: "eth1"
      downstream_interfaces:
        - name: "eth2"
        - name: "eth3"
```

Конфигурация устройства DHCP-Server1

1. Конфигурационный файл `/etc/dhcp/dhcpd.conf`:

```
default-lease-time 3600;
max-lease-time 3600;

failover peer "failover-partner" {
    primary;
    address 192.168.100.10;
    peer address 192.168.100.20;
    max-response-delay 60;
    max-unacked-updates 10;
    mclt 3600;
    split 128;
    load balance max seconds 3;
}

subnet 192.168.1.0 netmask 255.255.255.0 {
    option routers              192.168.1.1;
    option subnet-mask          255.255.255.0;
    option broadcast-address    192.168.1.255;

    pool {
        failover peer "failover-partner";
        range 192.168.1.101 192.168.1.150;
    }
}

subnet 192.168.2.0 netmask 255.255.255.0 {
    option routers              192.168.2.1;
    option subnet-mask          255.255.255.0;
    option broadcast-address    192.168.2.255;

    pool {
        failover peer "failover-partner";
        range 192.168.2.101 192.168.2.150;
    }
}

subnet 192.168.3.0 netmask 255.255.255.0 {
    option routers              192.168.3.1;
    option subnet-mask          255.255.255.0;
    option broadcast-address    192.168.3.255;

    pool {
        failover peer "failover-partner";
        range 192.168.3.101 192.168.3.150;
    }
}

subnet 192.168.4.0 netmask 255.255.255.0 {
    option routers              192.168.4.1;
    option subnet-mask          255.255.255.0;
    option broadcast-address    192.168.4.255;

    pool {
        failover peer "failover-partner";
        range 192.168.4.101 192.168.4.150;
    }
}

subnet 192.168.100.0 netmask 255.255.255.0 {
}
```

Конфигурация устройства DHCP-Server2

1. Конфигурационный файл `/etc/dhcp/dhcpd.conf`:

```
default-lease-time 3600;
```

```
max-lease-time 3600;

failover peer "failover-partner" {
    secondary;
    address 192.168.100.20;
    peer address 192.168.100.10;
    max-response-delay 60;
    max-unacked-updates 10;
    load balance max seconds 3;
}

subnet 192.168.1.0 netmask 255.255.255.0 {
    option routers              192.168.1.1;
    option subnet-mask          255.255.255.0;
    option broadcast-address    192.168.1.255;

    pool {
        failover peer "failover-partner";
        range 192.168.1.101 192.168.1.150;
    }
}

subnet 192.168.2.0 netmask 255.255.255.0 {
    option routers              192.168.2.1;
    option subnet-mask          255.255.255.0;
    option broadcast-address    192.168.2.255;

    pool {
        failover peer "failover-partner";
        range 192.168.2.101 192.168.2.150;
    }
}

subnet 192.168.3.0 netmask 255.255.255.0 {
    option routers              192.168.3.1;
    option subnet-mask          255.255.255.0;
    option broadcast-address    192.168.3.255;

    pool {
        failover peer "failover-partner";
        range 192.168.3.101 192.168.3.150;
    }
}

subnet 192.168.4.0 netmask 255.255.255.0 {
    option routers              192.168.4.1;
    option subnet-mask          255.255.255.0;
    option broadcast-address    192.168.4.255;

    pool {
        failover peer "failover-partner";
        range 192.168.4.101 192.168.4.150;
    }
}

subnet 192.168.100.0 netmask 255.255.255.0 {
}
```