Резервирование каналов связи при помощи сервиса MultiWAN на «С-Терра Шлюз КВ»

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.

Версия ПО MultiWAN

Данная инструкция описывает работу с ПО MultiWAN версии `2.4.1+sterra2`.

Если установленная версия ниже указанной, то нужно **обязательно** обновиться (см. «Обновление пакета»).

Введение

Данный сценарий описывает настройку устройств «С-Терра Шлюз КВ» версии 5.0 для обеспечения безопасного взаимодействия между защищаемыми подсетями центрального офиса и филиала. В центральном офисе обеспечивается отказоустойчивость WAN каналов связи с использованием сервиса `multiwan`.

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».

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.

Краткое описание сервиса multiwan

Сервис `multiwan` предназначен для резервирования каналов связи (Интернет-провайдеров) в режиме активный/резервный.

Сервис `multiwan` конфигурируется через файл `/etc/multiwan/config.yaml`.

Доступность канала определяется по доступности хостов посредством протокола ICMP. В зависимости от режима, доступность канала определяется по доступности одного или всех хостов.

Сервис `multiwan` считает канал неработоспособным или работоспособным, если канал был недоступен или доступен несколько проверок подряд соответственно. Сервис `multiwan` считает состояние канала неизвестным, если со старта сервиса канал не определил свою доступность.

Если установлено определение доступности канала по всем хостам, то сервис `multiwan` считает канал неработоспособным, если при каждой проверке хотя бы один хост был недоступен. Если установлено определение доступности канала по любому хосту, то сервис `multiwan` считает канал неработоспособным, если при каждой проверке были недоступны все хосты.

Сервис произведет переключение на другой работоспособный канал в следующих случаях:

1. Текущий активный канал неработоспособен.
2. Если включен режим preempt и есть работоспособный канал с большим приоритетом.

Ограничения и особенности сервиса multiwan

1. Сервис `multiwan` может резервировать произвольные маршруты;
2. При работе `multiwan`, если в системе присутствуют маршруты, адреса назначения которых совпадают с адресом назначения `multiwan` маршрута, то такие маршруты могут быть удалены, если находятся в той же таблице маршрутизации;
3. При переключении канала автоматически происходит очистка текущих ISAKMP/IPsec туннелей, перезагрузка LSP и очистка таблицы трансляций NAT;
4. Каждый канал связи проверяется отдельно. Для проверки доступности каналов используются ICMP запросы. Для ICMP запросов сервиса `multiwan` используются отдельные таблицы маршрутизации. Данные таблицы содержат маршруты до проверяемого хоста через адрес следующего перехода и очищаются от посторонних маршрутов;
5. Reverse path filtering должен быть отключен. Для проверки доступности хостов используются собственные таблицы отличные от основной таблицы маршрутизации. Поэтому проверки доступности будут неудачными при включенном reverse path filtering.

   По умолчанию данный параметр отключен на криптошлюзах «С-Терра Шлюз» ОС Debian 12 классы защиты КС1, КС2 и КС3.

   Пример проверки состояния reverse path filtering на интерфейсе `eth0`:

   ```
   root@Hub:~# cat /proc/sys/net/ipv4/conf/eth0/rp_filter
   0
   ```

   Ноль в выводе команды означает, что reverse path filtering отключен.

Предварительные требования

Требования к материально-техническому обеспечению

В данном сценарии потребуются несколько USB Flash накопителей. Два накопителя будут использоваться в качестве рабочих ключевых носителей, один накопитель будет использоваться в качестве административного носителя, еще один – для переноса запросов и сертификатов между криптошлюзами и центром выпуска сертификатов.

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.

Администратор должен иметь опыт развертывания сценария «Организация защищенного взаимодействия между подсетями центрального офиса и филиала посредством IPsec VPN туннеля (протокол IPv4) с использованием «С-Терра Шлюз КВ» (далее по тексту он будет называться базовым сценарием).

Требования к инфраструктуре

1. Требования к устройствам.

   1.1. «С-Терра Шлюз КВ».

1.1.1. Устройства «С-Терра Шлюз КВ» должны быть в заводском состоянии.

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 R2 (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 КВ» (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

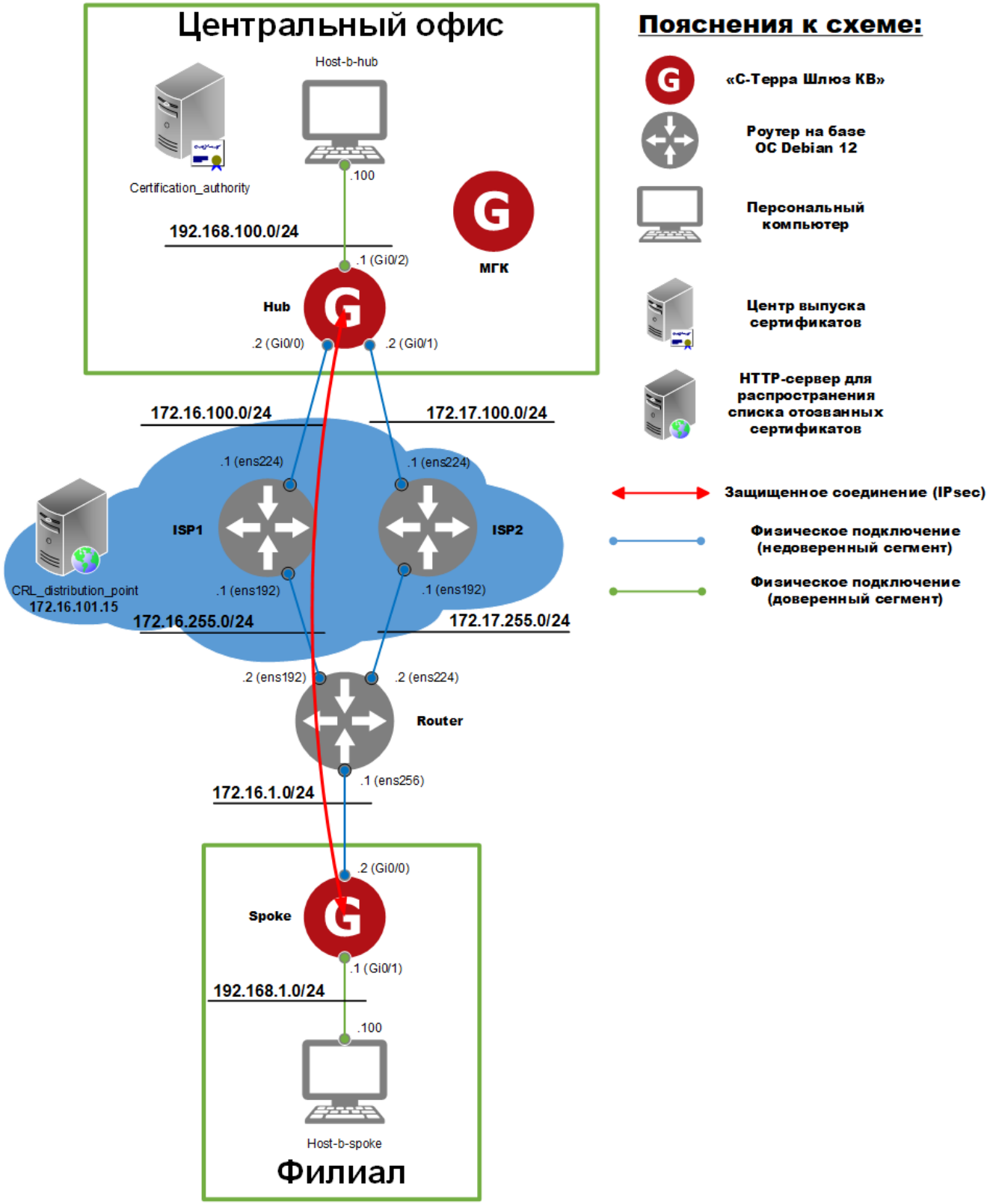


Рисунок 1. Схема взаимодействия

Общая логика работы

1. Устройства «С-Терра Шлюз КВ».

   1.1. Криптошлюз – в данном сценарии так будет называться «С-Терра Шлюз КВ» («С-Терра Шлюз NF КВ», исполнение «ЗМВ»), инициализированный в роли network и использующийся для защиты сетевого трафика (устройства Hub и Spoke на схеме).

   1.2. МГК (модуль генерации ключей) – в данном сценарии так будет называться «С-Терра Шлюз КВ» («С-Терра МГК КВ», исполнение «МГК»), инициализированный в роли management и использующийся для подготовки ключевых носителей для криптошлюзов (устройство МГК на схеме).

2. Ключевые носители.

   В данном сценарии используются следующие типы ключевых носителей:

   2.1. Шифроблокнот – CD-диск однократной записи, содержащий случайную последовательность. Шифроблокнот используется устройством МГК при подготовке рабочих ключевых носителей.

   2.2. Административный носитель – USB Flash накопитель, использующийся на МГК и содержащий ключевую и конфигурационную информацию.

   2.3. Рабочий ключевой носитель (РКН) – USB Flash накопитель, содержащий доверенный сертификат, контейнер с закрытым ключом и состояние ПДСЧ. Рабочий носитель используется при введении криптошлюза в эксплуатацию и при запуске СКЗИ.

3. Администраторские роли:

   3.1. Администратор безопасности – это администратор устройства МГК.

   3.2. Администратор сети – это администратор криптошлюза.

4. Размещение устройств.

   4.1. В центральном офисе размещаются: центр выпуска сертификатов (Certification_authority), устройства «С-Терра Шлюз КВ» (МГК и Hub) и персональный компьютер (Host-b-hub).

   4.2. В филиале размещаются: криптошлюз «С-Терра Шлюз КВ» (Spoke) и персональный компьютер (Host-b-spoke).

   4.3. В неконтролируемом сегменте (синее облако на схеме) размещаются: HTTP сервер для распространения списка отозванных сертификатов (CRL_distribution_point) и маршрутизаторы (Router1, ISP1, ISP2).

5. Подключение к сети Интернет.

   В данном сценарии для эмуляции сети Интернет используются маршрутизаторы Router1, ISP1 и ISP2.

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

   5.1. Криптошлюз Hub подключается к сети Интернет с помощью сервиса `multiwan`. Основной маршрут по умолчанию через маршрутизатор ISP1. Резервный маршрут по умолчанию через маршрутизатор ISP2.

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

   5.3. Устройство МГК к сети не подключается.

6. Отказоустойчивость.

   За резервирование каналов связи отвечает сервис `multiwan`. `multiwan` добавляет маршрут по умолчанию через ISP1 или ISP2. Выбор зависит от доступности каналов. Если канал через ISP1 недоступен, то `multiwan` переключается на работу через ISP2, если он доступен. Если `multiwan` работает на канале через ISP2 и канал через ISP1 становится доступен, то `multiwan` переключится на работу через ISP1, если включен preempt режим.

Для проверки доступности каналов используются ICMP запросы. Для каждого проверяемого хоста и для каждого интерфейса создается своё правило для ICMP запроса. Для каждого канала создаётся своя таблица с маршрутами до проверяемых хостов через ISP1 или через ISP2 с указанием src адреса. Данная таблица будет содержать только маршруты до проверяемых хостов, все остальные маршруты из таблиц для каналов будут удалены.

7. Параметры шифрования IPsec.

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

   Инициировать защищенное соединение может как трафик из филиала в центр, так и наоборот, из центра в филиал.

   7.1. Параметры протокола IKEv2:

   - Аутентификация при помощи цифровых сертификатов, алгоритм подписи – ГОСТ Р 34.10-2012 (ключ 256 бит);
   - Алгоритм шифрования – ГОСТ Р 34.12-2015 «Магма» в режиме MGM (ключ 256 бит);
   - Алгоритм псевдослучайной функции – ГОСТ Р 34.11-2012 «Стрибог» (ключ 512 бит);
   - Алгоритм выработки общего ключа (аналог алгоритма Диффи-Хеллмана) – группа 33 (ГОСТ Р 34.10-2012-256-paramSetA).

   7.2. Параметры протокола ESP:

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

8. Параметры межсетевого экранирования.

   Для пакетов, которые не защищаются IPsec применяются следующие правила фильтрации:

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

   8.2. На LAN интерфейсах IPv4 трафик не контролируется.

   8.3. На WAN интерфейсах разрешен входящий IPv4 трафик:

   8.3.1. Протокол UDP порты 500 и 4500 – для IKEv2 и NAT-Traversal (ESP in UDP).

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

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

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

   8.3.5. 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).

Настройка стенда

Настройка устройства Host-b-hub

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 (если заблокировано).

Настройка устройства Host-b-spoke

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 (если заблокировано).

Настройка устройства ISP1

1. Настройте IP адрес – 172.16.255.1 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. Задайте маршрут по умолчанию через 172.16.255.2.
4. Разрешите прохождение IP трафика (если заблокировано).

Настройка устройства ISP2

1. Настройте IP адрес – 172.17.255.1 и маску – 255.255.255.0 на сетевом интерфейсе ens192.
2. Настройте IP адрес – 172.17.100.1 и маску – 255.255.255.0 на сетевом интерфейсе ens224.
3. Задайте маршрут по умолчанию через 172.17.255.2.
4. Разрешите прохождение IP трафика (если заблокировано).

Настройка устройства Router

1. Настройте IP адрес – 172.16.255.2 и маску – 255.255.255.0 на сетевом интерфейсе ens192.
2. Настройте IP адрес – 172.17.255.2 и маску – 255.255.255.0 на сетевом интерфейсе ens224.
3. Настройте IP адрес – 172.16.1.1 и маску – 255.255.255.0 на сетевом интерфейсе ens256.
4. Задайте маршрут до сети 172.16.100.0/24 через 172.16.255.1.
5. Задайте маршрут до сети 172.17.100.0/24 через 172.17.255.1.
6. Разрешите прохождение IP трафика (если заблокировано).

Настройка устройства МГК

Настройте устройство МГК в соответствии с базовым сценарием. Подготовьте ключевые носители для криптошлюзов Hub и Spoke.

Настройка криптошлюза Hub

Настройка будет происходить локально при помощи консольного подключения. Все действия на криптошлюзе Hub выполняются Администратором сети.

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 консоль управления под учетной записью администратора (пользователь и пароль по умолчанию: `admin`, `s-terra`, уровень целостности – 63), установите правильный тип терминала и требуемую ширину, смените пароль для пользователя `admin`.

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

Подробная настройка с описанием команд приведена в разделе «Начальные настройки» базового сценария.

Инициализация

Подключите рабочий ключевой носитель с сессионными данными для криптошлюза Hub и инициализируйте криптошлюз Hub в роли network.

Подробная настройка с описанием команд приведена в разделе «Инициализация» базового сценария.

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 в соответствии с правилами пользования СКЗИ.

Сгенерируйте запрос на сертификат криптошлюза с использованием ключевого контейнера с рабочего ключевого носителя, сохраните запрос в файл на USB Flash накопителе. Доставьте файл с запросом на УЦ и выпустите по нему сертификат. Сертификат криптошлюза доставьте на шлюз и импортируйте его в базу Продукта.

Подробная настройка с описанием команд приведена в разделе «Настройки PKI (запросы и сертификаты)» базового сценария.

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 применяется после выхода из режима конфигурирования**.

1. Задайте имя устройства:

```
sterragate(config)#hostname Hub
```

2. Включите WAN и LAN интерфейсы и настройте IP адреса в соответствии со схемой стенда:

```
Hub(config)#interface GigabitEthernet0/0
Hub(config-if)# ip address 172.16.100.2 255.255.255.0
Hub(config-if)# no shutdown
Hub(config-if)#interface GigabitEthernet0/1
Hub(config-if)# ip address 172.17.100.2 255.255.255.0
Hub(config-if)# no shutdown
Hub(config-if)#interface GigabitEthernet0/2
Hub(config-if)# ip address 192.168.100.2 255.255.255.0
Hub(config-if)# no shutdown
Hub(config-if)#exit
```

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка шифрования

1. Параметры IKEv2.

   1.1. Создайте набор параметров аутентификации с использованием сертификатов в рамках протокола IKEv2 под именем `IKEv2_AUTH`:

```
Hub(config)#crypto ikev2 authentication sig IKEv2_AUTH
```

   1.2. Укажите в качестве типа идентификатора (identity) отличительное имя (Distinguished Name, DN) x509 сертификата:

```
Hub(conf-ikev2-auth)#identity dn
Hub(conf-ikev2-auth)#exit
```

   1.3. Создайте набор алгоритмов под номером 1 для защиты IKEv2 туннеля:

```
Hub(config)#crypto ikev2 proposal 1
Hub(config-ikev2-proposal)#encryption magma
Hub(config-ikev2-proposal)#group gost3410-12-256
Hub(config-ikev2-proposal)#prf streebog512
```

   1.4. Создайте политику IKEv2 под номером 1 и добавьте в нее созданный выше набор алгоритмов:

```
Hub(config)#crypto ikev2 policy 1
Hub(config-ikev2-policy)#proposal 1
Hub(config-ikev2-policy)#exit
```

   1.5. Создайте профиль настроек IKEv2 под именем IKEv2_ALGS:

```
Hub(config)#crypto ikev2 profile IKEv2_ALGS
```

      1.5.1. Укажите использование политики 1:

```
Hub(config-ikev2-profile)#set policy 1
```

      1.5.2. Укажите созданный ранее набор параметров аутентификации `IKEv2_AUTH` для локальной аутентификации и аутентификации партнера:

```
Hub(config-ikev2-profile)#authentication local IKEv2_AUTH
Hub(config-ikev2-profile)#authentication remote IKEv2_AUTH
```

      1.5.3. Настройте параметры DPD (dead peer detection), срабатывающего по отсутствию входящего трафика в IPsec или в IKEv2 туннеле:

```
Hub(config-ikev2-profile)#dpd 3 on-demand
Hub(config-ikev2-profile)#exit
```

      Примечание: если в туннеле в течение 3 секунд отсутствует входящий трафик, но присутствует исходящий, то будет инициирована DPD сессия. Если партнер не отвечает на DPD сессию, то выполняется попытка перестроения IKEv2 и соответствующего IPsec туннеля.

2. Параметры IPsec.

   2.1. Отключите обновление ключа IPsec SA по достижению определенного объема переданного трафика (это позволит увеличить производительность и снизит потери при интенсивном трафике; разрешено для комбинированных алгоритмов шифрования, используемых в

настоящем документе, так как в них есть встроенные механизмы обновления ключей по достижению определенного объема переданного трафика):

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

2.2. Создайте набор алгоритмов с именем IKEv2_IPSEC_ALGS для защиты IPsec туннелей (в данном сценарии используется комбинированный алгоритм шифрования и имитозащиты ГОСТ 34.12-2015 Магма в режиме MGM):

```
Hub(config)#crypto ikev2 ipsec-proposal IKEv2_IPSEC_ALGS
Hub(ipsec-proposal)#esp encryption magma-mgm
Hub(ipsec-proposal)#exit
```

2.3. Создайте список доступа (ACL) для IPv4 трафика, который нужно защищать между центральным офисом и филиалом:

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

2.4. Создайте крипто-карту (имя `VPN_ISP1`, раздел 1) для построения IPsec туннеля между центральным офисом и филиалом через провайдера ISP1:

```
Hub(config)#crypto map VPN_ISP1 1 ipsec-isakmp
```

2.4.1. Укажите список доступа для защищаемого трафика:

```
Hub(config-crypto-map)#match address IPSEC_ACL_HUB_AND_SPOKE
```

2.4.2. Укажите набор алгоритмов для защиты IPsec туннелей:

```
Hub(config-crypto-map)#set ikev2-ipsec-proposal IKEv2_IPSEC_ALGS
```

2.4.3. Прикрепите IKEv2 профиль к криптокарте:

```
Hub(config-crypto-map)#set ikev2-profile IKEv2_ALGS
```

2.4.4. Укажите IPv4 адрес партнера по IPsec, в данном сценарии это внешний IPv4 адрес устройства Spoke (если криптошлюз находится за устройством, выполняющим NAT, то указывается его внешний адрес):

```
Hub(config-crypto-map)#set peer 172.16.1.2
```

2.4.5. Укажите локальный адрес (это адрес интерфейса `GigabitEthernet0/0`):

```
Hub(config-crypto-map)#set local-address 172.16.100.2
```

2.4.6. Включите туннельный режим IPsec:

```
Hub(config-crypto-map)#set mode tunnel
```

2.4.7. Включите автоматическую и интеллектуальную корректировку значения MSS (величина подбирается в зависимости от алгоритма шифрования и MTU трассы) для исходящего TCP трафика, чтобы его пакеты имели оптимальный размер и не фрагментировались криптошлюзом (данная настройка должна быть задача с двух сторон):

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

2.4.8. Включите префрагментацию (сначала трафик фрагментируется, потом зашифровывается), чтобы в недоверенной сети не было фрагментов (некоторые интернет-провайдеры блокируют фрагменты) и чтобы снизить вероятность ложного удаления фрагментированных пакетов при интенсивном трафике:

```
Hub(config-crypto-map)#set pre-fragmentation
Hub(config-crypto-map)#exit
```

2.5. Аналогичным образом создайте крипто-карту (имя `VPN_ISP2`, раздел 1) для построения IPsec туннеля между центральным офисом и филиалом через провайдера ISP2:

```
Hub(config)#crypto map VPN_ISP2 1 ipsec-isakmp
Hub(config-crypto-map)#match address IPSEC_ACL_HUB_AND_SPOKE
Hub(config-crypto-map)#set ikev2-ipsec-proposal IKEv2_IPSEC_ALGS
Hub(config-crypto-map)#set ikev2-profile IKEv2_ALGS
```

```
Hub(config-crypto-map)#set peer 172.16.1.2
Hub(config-crypto-map)#set local-address 172.17.100.2
Hub(config-crypto-map)#set mode tunnel
Hub(config-crypto-map)#set security-association mss-fix out
Hub(config-crypto-map)#set pre-fragmentation
Hub(config-crypto-map)#exit
```

2.6. Прикрепите созданные крипто-карты к соответствующим WAN интерфейсам:

```
Hub(config)#interface GigabitEthernet0/0
Hub(config-if)#crypto map VPN_ISP1
Hub(config-if)#interface GigabitEthernet0/1
Hub(config-if)#crypto map VPN_ISP2
Hub(config-if)#exit
```

3. Примените настройки:

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

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.

Из-за отсутствия маршрута по умолчанию COC не будет загружен в базу Продукта. Проверка осуществится после настройки сервиса `multiwan`.

Настройка межсетевого экрана

По умолчанию на всех сетевых интерфейсах IPv6 трафик заблокирован – команда `ipv6 default filtering deny`, если требуется разрешить IPv6 трафик, то нужно выполнить команду `ipv6 default filtering permit` и далее уже фильтровать IPv6 трафик по аналогии с IPv4. Подробное описание работы данной команды приведено в сценарии «Организация защищенного взаимодействия между подсетями центрального офиса и филиалов посредством IPsec VPN туннеля (протокол IPv6)».

1. Создайте сетевой `object-group` с именем `SELF` для хранения в нем WAN адресов криптошлюза с целью дальнейшего его использования в ACL:

```
Hub#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Hub(config)#object-group network SELF
Hub(config-network-group)#remark WAN IP
Hub(config-network-group)#host 172.16.100.2
Hub(config-network-group)#host 172.17.100.2
Hub(config-network-group)#exit
```

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

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

3. Создайте правила инспектирования (stateful) всего TCP и UDP трафика и задайте необходимое максимальное количество сессий (в версии 5.0 данная команда устанавливает максимальное количество сессий для IPv4 и IPv6):

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

4. Прикрепите правила межсетевого экранирования на WAN интерфейсы (обратите особое внимание на направление правил фильтрации, список доступа `FW_INPUT_WAN` прикрепляется для входящих в интерфейс пакетов – **in**, а правило `INSPECT` stateful фильтрации прикрепляется для исходящих с интерфейса пакетов – **out**):

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

5. Примените настройки:

```
Hub(config)#end
```

Настройка сервиса multiwan

При конфигурировании сервиса `multiwan` нужно **обязательно** соблюдать следующие правила:
1) Конфигурационный файл должен быть формата YAML.
2) Версия формата конфигурационного файла должна быть 1 (параметр `config_format_version`).
3) Приоритеты каналов связи, принадлежащих одной группе, должны отличаться.
4) Каждый канал связи должен содержать: название интерфейса, приоритет канала связи, маршруты, адрес следующего перехода и хосты для проверки.
5) Адрес следующего перехода может отсутствовать в секции `ipv4` только если он уже задан в секции `ipv6`. Аналогично адрес следующего перехода может отсутствовать в секции `ipv6` только если он уже задан для секции `ipv4`.

Настройка осуществляется из linux bash.

Настройка `multiwan` осуществляется через файл `/etc/multiwan/config.yaml`.

Использование очистки таблицы conntrack может привести к зависанию MultiWAN версии ниже 2.3.0.
На версиях ниже 2.3.0 рекомендуется отключить очистку таблицы conntrack:
`clear_all_conntrack_translations: False`

1. Приведите содержимое файла `/etc/multiwan/config.yaml` к следующему виду:

При расположении параметра не в своей секции, значение параметра будет проигнорировано.

```
config_format_version: 1
links_params:
  links:
  - name: ISP1
    iface_name: eth0
    prio: 300
    routes:
      ipv4:
        networks:
        - 0.0.0.0/0
        nexthop: 172.16.100.1
    check:
      hosts_reachability: any
      ipv4:
```

```
        hosts:
        - 172.16.101.15
        - 172.16.1.2
      method:
        icmp:
          max_rtt: 100
          interval: 3
          restore_interval: 1
          restore_count: 10
          failure_count: 10
          tos: 184
          data_size: 56
  - name: ISP2
    iface_name: eth1
    prio: 200
    routes:
      ipv4:
        networks:
        - 0.0.0.0/0
        nexthop: 172.17.100.1
    check:
      hosts_reachability: any
      ipv4:
        hosts:
        - 172.16.101.15
        - 172.16.1.2
      method:
        icmp:
          max_rtt: 100
          interval: 3
          restore_interval: 1
          restore_count: 10
          failure_count: 10
          tos: 184
          data_size: 56
```

Подробное описание настроек приведено в документации.

2. Проверьте правильность составленного конфигурационного файла с помощью команды `/opt/multiwan/multiwan_daemon --check_config`. Для отображения параметров, получаемых из конфигурационного файла, добавьте ключ `--result`:

```
root@Hub:~# /opt/multiwan/multiwan_daemon --check_config --result
```

```
INFO: Parsed parameters:
config_format_version: 1
daemon_params:
  watch_intervals:
    watch_foreign_routes: 10
    watch_rules: 10
    watch_self_routes: 10
links_params:
  check_links_reachability_on_start: false
  clear_all_conntrack_translations: true
  clear_routes_if_all_links_unreachable: false
  clear_sa_conn: true
  links:
  - check:
      hosts_reachability: any
      ipv4:
        hosts:
        - 172.16.101.15
        - 172.16.1.2
```

```
      source_address: null
    method:
      icmp:
        data_size: 56
        failure_count: 10
        fwmark: 1024
        interval: 3
        max_rtt: 100
        restore_count: 10
        restore_interval: 1
        rule_prio: 100
        rule_table_num: 1000
        tos: 184
        watch_rules: true
  group: null
  iface_name: eth0
  name: ISP1
  prio: 300
  routes:
    ipv4:
      networks:
      - 0.0.0.0/0
      nexthop: 172.16.100.1
      source_address: null
    routes_table_num: 254
    watch_foreign_routes: false
    watch_self_routes: true
- check:
    hosts_reachability: any
    ipv4:
      hosts:
      - 172.16.101.15
      - 172.16.1.2
      source_address: null
    method:
      icmp:
        data_size: 56
        failure_count: 10
        fwmark: 1025
        interval: 3
        max_rtt: 100
        restore_count: 10
        restore_interval: 1
        rule_prio: 100
        rule_table_num: 1001
        tos: 184
        watch_rules: true
  group: null
  iface_name: eth1
  name: ISP2
  prio: 200
  routes:
    ipv4:
      networks:
      - 0.0.0.0/0
      nexthop: 172.17.100.1
      source_address: null
    routes_table_num: 254
    watch_foreign_routes: false
    watch_self_routes: true
preempt: true
reload_lsp: true
```

```
notify:
  scripts:
    postchangelink: null
    poststart: null
    prechangelink: null
    prestop: null

INFO: Configuration file /etc/multiwan/config.yaml can be parsed
```

3. Включите сервис `multiwan` и добавьте его в автозапуск:

```
root@Hub:~# systemctl start multiwan.service
root@Hub:~# systemctl enable multiwan.service
```

4. Убедитесь, что в систему добавился маршрут `multiwan`:

```
root@Hub:~# ip route show proto multiwan
```

```
default via 172.16.100.1 dev eth0 metric 700
```

5. Перезапустите сервис `getcrls` (если была настроена загрузка СОС):

```
root@Hub:~# systemctl restart getcrls.service
```

Опционально: сохранение контрольных сумм конфигурации

После завершения процесса конфигурации есть возможность сохранить контрольную сумму настроек криптошлюза. Для этого перейдите в консоль управления и выполните команду `record checksum Hub`:

```
Hub] record checksum Hub
```

При этом на рабочем ключевом носителе создается файл с контрольными суммами текущих настроек, включая следующие:

- Cisco-like и LSP конфигурация;
- Локальные сертификаты;
- Настройки протоколирования;
- Настройки VPN драйвера.

После этого при выполнении команды `deploy` в консоли управления перед запуском VPN сервиса будет выполняться проверка контрольных сумм, в случае обнаружения несоответствия будет выведено предупреждение и VPN сервис запущен не будет. Для обновления контрольных сумм нужно будет повторно выполнить команду `record checksum Hub` в консоли управления криптошлюза.

Настройка криптошлюза Hub завершена.

В Приложении представлены тексты конфигураций для криптошлюза Hub:

- текст консоли cisco-like;
- текст LSP;

Настройка криптошлюза Spoke

Настройка криптошлюза Spoke происходит аналогично настройке Hub. Настройка `multiwan` на криптошлюзе Spoke не выполняется. Все действия на криптошлюзе Spoke выполняются Администратором сети.

В Приложении представлены тексты конфигураций для криптошлюза Spoke:

- текст консоли cisco-like;
- текст LSP;

Проверка работоспособности стенда

Проверку работоспособности нужно выполнять после завершения настройки всех устройств стенда.

Проверка IP связности

1. Проверьте, что с криптошлюзов по ICMP доступен шлюз по умолчанию (ISP1 и ISP2 для Hub, Router1 для Spoke). Для этого выполните команду `ping` из cisco-like консоли соответствующего криптошлюза.

   ```
   Hub#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=0.208 ms
   108 bytes from 172.16.100.1: icmp_seq=2 ttl=64 time=0.260 ms
   108 bytes from 172.16.100.1: icmp_seq=3 ttl=64 time=0.245 ms
   108 bytes from 172.16.100.1: icmp_seq=4 ttl=64 time=0.276 ms
   108 bytes from 172.16.100.1: icmp_seq=5 ttl=64 time=0.303 ms

   --- 172.16.100.1 ping statistics ---
   5 packets transmitted, 5 received, 0% packet loss, time 46ms
   rtt min/avg/max/mdev = 0.208/0.258/0.303/0.034 ms
   ```

   ```
   Hub#ping 172.17.100.1
   ```

   ```
   PING 172.17.100.1 (172.17.100.1) 100(128) bytes of data.
   108 bytes from 172.17.100.1: icmp_seq=1 ttl=64 time=0.257 ms
   108 bytes from 172.17.100.1: icmp_seq=2 ttl=64 time=0.184 ms
   108 bytes from 172.17.100.1: icmp_seq=3 ttl=64 time=0.266 ms
   108 bytes from 172.17.100.1: icmp_seq=4 ttl=64 time=0.305 ms
   108 bytes from 172.17.100.1: icmp_seq=5 ttl=64 time=0.280 ms

   --- 172.17.100.1 ping statistics ---
   5 packets transmitted, 5 received, 0% packet loss, time 46ms
   rtt min/avg/max/mdev = 0.184/0.258/0.305/0.043 ms
   ```

   ```
   Spoke#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=0.771 ms
   108 bytes from 172.16.1.1: icmp_seq=2 ttl=64 time=0.282 ms
   108 bytes from 172.16.1.1: icmp_seq=3 ttl=64 time=0.276 ms
   108 bytes from 172.16.1.1: icmp_seq=4 ttl=64 time=0.268 ms
   108 bytes from 172.16.1.1: icmp_seq=5 ttl=64 time=0.244 ms

   --- 172.16.1.1 ping statistics ---
   5 packets transmitted, 5 received, 0% packet loss, time 4096ms
   rtt min/avg/max/mdev = 0.244/0.368/0.771/0.202 ms
   ```

   Видно, что со всех криптошлюзов доступны по ICMP их соответствующие шлюзы по умолчанию.

2. Проверьте, что с криптошлюза Hub доступен по ICMP криптошлюз Spoke (по внешнему интерфейсу). Для этого выполните команду `ping` из cisco-like консоли криптошлюза Hub.

   ```
   Hub#ping 172.16.1.2
   ```

   ```
   PING 172.16.1.2 (172.16.1.2) 100(128) bytes of data.
   108 bytes from 172.16.1.2: icmp_seq=1 ttl=63 time=0.802 ms
   108 bytes from 172.16.1.2: icmp_seq=2 ttl=63 time=0.416 ms
   108 bytes from 172.16.1.2: icmp_seq=3 ttl=63 time=0.449 ms
   108 bytes from 172.16.1.2: icmp_seq=4 ttl=63 time=0.393 ms
   108 bytes from 172.16.1.2: icmp_seq=5 ttl=63 time=0.485 ms

   --- 172.16.1.2 ping statistics ---
   5 packets transmitted, 5 received, 0% packet loss, time 4083ms
   rtt min/avg/max/mdev = 0.393/0.509/0.802/0.149 ms
   ```

   Видно, что с криптошлюза Hub доступен по ICMP криптошлюз Spoke.

3. Проверьте, что с защищаемых устройств Host-b-hub и Host-b-spoke доступен по ICMP соответствующий криптошлюз, который является шлюзом по умолчанию. Для этого выполните команду `ping` из linux bash консоли защищаемых устройств.

```
root@Host-b-hub:~# ping 192.168.100.1 -c 5
```

```
PING 192.168.100.1 (192.168.100.1) 56(84) bytes of data.
64 bytes from 192.168.100.1: icmp_seq=1 ttl=64 time=0.211 ms
64 bytes from 192.168.100.1: icmp_seq=2 ttl=64 time=0.235 ms
64 bytes from 192.168.100.1: icmp_seq=3 ttl=64 time=0.223 ms
64 bytes from 192.168.100.1: icmp_seq=4 ttl=64 time=0.254 ms
64 bytes from 192.168.100.1: icmp_seq=5 ttl=64 time=0.233 ms

--- 192.168.100.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4088ms
rtt min/avg/max/mdev = 0.211/0.231/0.254/0.017 ms
```

```
root@Host-b-spoke:~# ping 192.168.1.1 -c 5
```

```
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.214 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.223 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.189 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=0.210 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=0.212 ms

--- 192.168.1.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4083ms
rtt min/avg/max/mdev = 0.189/0.209/0.223/0.019 ms
```

Видно, что с защищаемых устройств Host-b-hub и Host-b-spoke доступны по ICMP их соответствующие шлюзы по умолчанию.

Проверка PKI

1. Проверьте, что на криптошлюзах СОС импортирован в базу Продукта. Для этого выполните команду `run cert_mgr show` из cisco-like консоли соответствующего криптошлюза.

```
Hub#run cert_mgr show
```

```
Found 2 certificates. Found 1 CRL.
1 Status: local   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Hub
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
```

```
Spoke#run cert_mgr show
```

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

Видно, что на криптошлюзах 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 статус всех сертификатов Active. Для этого выполните команду `run cert_mgr check` из cisco-like консоли соответствующего криптошлюза.

```
Hub#run cert_mgr check
```

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

```
Spoke#run cert_mgr check
```

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

Видно, что на криптошлюзах статус всех сертификатов Active. Если статус Inactive, то проверьте:

- загружен ли СОС в базу Продукта;
- правильны и одинаковы ли дата и время на всех криптошлюзах.

Проверка построения защищенного соединения

1. Инициируйте построение защищенного соединения между криптошлюзами Hub и Spoke при помощи ICMP трафика, посылаемого с устройства Host-b-Spoke на Host-b-Hub (можно и наоборот). Для этого выполните команду `ping` из linux bash консоли защищаемого устройства Host-b-Spoke.

```
root@host-b-Spoke:~# ping 192.168.100.100 -c 5
```

```
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=6.70 ms
64 bytes from 192.168.100.100: icmp_seq=2 ttl=64 time=3.97 ms
64 bytes from 192.168.100.100: icmp_seq=3 ttl=64 time=3.69 ms
64 bytes from 192.168.100.100: icmp_seq=4 ttl=64 time=2.44 ms
64 bytes from 192.168.100.100: icmp_seq=5 ttl=64 time=2.61 ms

--- 192.168.100.100 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 10ms
rtt min/avg/max/mdev = 2.442/3.882/6.700/1.529 ms
```

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

```
Hub#run sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

IKEv2 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 1423 1395

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 440 440
```

```
Spoke#run sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

IKEv2 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 1395 1423

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 440 440
```

Видно, что на криптошлюзах Hub и Spoke 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.

Для дополнительной диагностики можно воспользоваться утилитой `tcpdump` на нужных интерфейсах.

Проверка состояния каналов связи

На криптошлюзе Hub проверьте состояние каналов связи с помощью утилиты `mwan_mgr`:

```
Hub#run mwan_mgr show --links_status
```

```
Link: 1
    Name: ISP1
    Interface: eth0
```

```
    Priority: 300
    State: reachable
    Active: True
    Hosts reachability: any
    IPv4 hosts: "172.16.101.15", "172.16.1.2"
    IPv4 nexthop: "172.16.100.1"
    IPv6 hosts: None
    IPv6 nexthop: None
Link: 2
    Name: ISP2
    Interface: eth1
    Priority: 200
    State: reachable
    Active: False
    Hosts reachability: any
    IPv4 hosts: "172.16.101.15", "172.16.1.2"
    IPv4 nexthop: "172.17.100.1"
    IPv6 hosts: None
    IPv6 nexthop: None
```

Видно, что канал с именем ISP1 является доступным (`State: reachable`) и активным (`Active: True`). Второй канал (ISP2) является доступным, но в данный момент не является активным (`Active: False`).

Проверка переключения на резервный канал

1. На криптошлюзе Hub для эмуляции отказа отключите интерфейс основного канала:

   ```
   Hub#run ip link set eth0 down
   ```

2. Подождите пока `multiwan` переключится на резервный канал, затем проверьте маршрут в системе:

   ```
   Hub#run ip route show proto multiwan
   ```

   ```
   default via 172.17.100.1 dev eth1 metric 800
   ```

3. Проверьте состояние каналов связи с помощью утилиты `mwan_mgr`:

   ```
   Hub#run mwan_mgr show --links_status
   Link: 1
       Name: ISP1
       Interface: eth0
       Priority: 300
       State: unreachable
       Active: False
       Hosts reachability: any
       IPv4 hosts: "172.16.101.15", "172.16.1.2"
       IPv4 nexthop: "172.16.100.1"
       IPv6 hosts: None
       IPv6 nexthop: None
   Link: 2
       Name: ISP2
       Interface: eth1
       Priority: 200
       State: reachable
       Active: True
       Hosts reachability: any
       IPv4 hosts: "172.16.101.15", "172.16.1.2"
       IPv4 nexthop: "172.17.100.1"
       IPv6 hosts: None
       IPv6 nexthop: None
   ```

Видно, что теперь резервный канал (ISP2) является активным, а основной канал (ISP1) стал недоступным (`State: unreachable`) и неактивным.

4. Запустите трафик между устройствами Host-b-Spoke и Host-b-Hub и убедитесь, что защищенные соединения успешно построились через резервный канал связи:

```
Hub#run sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

IKEv2 connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 2 (172.17.100.2,500)-(172.16.1.2,500) active 1423 1395

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 2 (192.168.100.0-192.168.100.255,*)-(192.168.1.0-192.168.1.255,*) * ESP tunn 440 440
```

5. Включите интерфейс основного канала:

```
Hub#run ip link set eth0 up
```

6. Убедитесь, что `multiwan` переключился обратно на основной канал:

```
Hub#run ip route show proto multiwan
```

```
default via 172.16.100.1 dev eth0 metric 700
```

7. Проверьте состояние каналов связи:

```
Hub#run mwan_mgr show --links_status
```

```
Link: 1
    Name: ISP1
    Interface: eth0
    Priority: 300
    State: reachable
    Active: True
    Hosts reachability: any
    IPv4 hosts: "172.16.101.15", "172.16.1.2"
    IPv4 nexthop: "172.16.100.1"
    IPv6 hosts: None
    IPv6 nexthop: None
Link: 2
    Name: ISP2
    Interface: eth1
    Priority: 200
    State: reachable
    Active: False
    Hosts reachability: any
    IPv4 hosts: "172.16.101.15", "172.16.1.2"
    IPv4 nexthop: "172.17.100.1"
    IPv6 hosts: None
    IPv6 nexthop: None
```

Проверка работы межсетевого экрана

Для проверки работы межсетевого экрана выполните последовательно следующие команды из cisco-like консоли криптошлюзов и проанализируйте их вывод.

```
Hub#run fwconn_mgr show
Hub#run klogview -f FW,drop
```

Приложение

Обновление пакета

Текущую версию пакета можно узнать с помощью команды из linux bash:

```
root@sterragate:~# dpkg -s multiwan | grep Version
Version: 2.2.6+sterra1-astra1.7.6
```

Актуальную версию пакета можно скачать на портале документации https://doc.s-terra.ru/.

Чтобы обновить пакет на Шлюзе, надо выполнить команду установки:

```
root@sterragate:~# dpkg -i <имя пакета>
```

Например, для пакета версии 2.4.1 на Шлюзе КС3 команда будет такая:

```
root@sterragate:~# dpkg -i multiwan_2.4.1+sterra2-astra1.7.6_all.deb
```

Конфигурации криптошлюза Hub

1. Консоль cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity address
crypto ipsec security-association lifetime kilobytes unlimited
username          cscons          privilege          15          secret          5
$6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOWZlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi
20ElR4wtMXTl0
aaa new-model
!
!
hostname Hub
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
!
!
!
!
logging console critical
!
!
crypto ikev2 proposal 1
 encryption magma
 group gost3410-12-256
 prf streebog512
!
crypto ikev2 policy 1
 proposal 1
!
crypto ikev2 authentication sig IKEv2_AUTH
 identity dn
 set identity-mapping auto
!
crypto ikev2 profile IKEv2_ALGS
 authentication remote IKEv2_AUTH
 authentication local IKEv2_AUTH
 set policy 1
 dpd 3 on-demand
```

```
!
crypto ikev2 ipsec-proposal IKEv2_IPSEC_ALGS
 esp encryption magma-mgm
!
object-group network SELF
 description IP on WAN interfaces
 host 172.16.100.2
 host 172.17.100.2
!
ip access-list extended IPSEC_ACL_HUB_AND_SPOKE
 permit ip 192.168.100.0 0.0.0.255 192.168.1.0 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_ISP1 1 ipsec-isakmp
 match address IPSEC_ACL_HUB_AND_SPOKE
 set ikev2-ipsec-proposal IKEv2_IPSEC_ALGS
 set ikev2-profile IKEv2_ALGS
 set security-association mss-fix out
 set local-address 172.16.100.2
 set mode tunnel
 set peer 172.16.1.2
 set pre-fragmentation
!
crypto map VPN_ISP2 1 ipsec-isakmp
 match address IPSEC_ACL_HUB_AND_SPOKE
 set ikev2-ipsec-proposal IKEv2_IPSEC_ALGS
 set ikev2-profile IKEv2_ALGS
 set security-association mss-fix out
 set local-address 172.17.100.2
 set mode tunnel
 set peer 172.16.1.2
 set pre-fragmentation
!
interface GigabitEthernet0/0
 ip address 172.16.100.2 255.255.255.0
 ip access-group FW_INPUT_WAN in
 crypto map VPN_ISP1
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 ip address 172.17.100.2 255.255.255.0
 ip access-group FW_INPUT_WAN in
 crypto map VPN_ISP2
 ip inspect INSPECT out
!
interface GigabitEthernet0/2
 ip address 192.168.100.1 255.255.255.0
!
!
```

```
!
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:   Wed Apr 29 11:13:39 2026

GlobalParameters(
    Title                       = "This LSP was automatically generated by CSP Converter
at Wed Apr 29 11:13:39 2026 (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
)

IKEParameters2 (
    SessionTimeout = 20
    ResponseTimeMax = 10
    SendRetries = 5
    RetryTimeBase = 1
    RetryTimeMax = 10
    CookieChallenge = 10
    NATKeepaliveInterval = 20
    NATKeepaliveRetain = 300
    FragmentSize = 576
    FragmentSize6 = 1280
)

IKEProposal2 n1 (
    EncryptionAlg = ENCR_MAGMA_MGM_KTREE
    PrfAlg        = PRF_HMAC_STREEBOG_512
    KeyAlg        = GOST3410_2012_256
)
```

```
IPsecProposal2 IKEv2_IPSEC_ALGS (
    ESPEncryptionAlg = ENCR_MAGMA_MGM_KTREE
)

IdentityEntry id:local:IKEv2_AUTH (
    DistinguishedName = USER_SPECIFIC_DATA
)

IKEAuthSignature2 local:IKEv2_AUTH (
    Identity  = id:local:IKEv2_AUTH
    Credential  = CertDescription()
    SendCertMode  = ALWAYS
)

IdentityEntry id:remote:IKEv2_AUTH (
    DistinguishedName = USER_SPECIFIC_DATA
)

IKEAuthSignature2 remote:IKEv2_AUTH (
    Identity  = id:remote:IKEv2_AUTH
    Credential  = CertDescription()
    SendRequestMode  = ALWAYS
)

FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.16.100.2
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.17.100.2
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.16.100.2
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.17.100.2
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.16.100.2
        ProtocolID = 1
```

```
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.17.100.2
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.16.100.2
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.17.100.2
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 0.0.0.0/0
        Action = DROP
    )
    IPv6Filters = Filter (
        SourceIP = ::/0
        DestinationIP = ::/0
        Action = DROP
    )
)

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

```
    )
)

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

IKEParameters(
    FragmentSize = 0
)

IKERule2 IKERule2:VPN_ISP1:1
(
    IKEPeerIPFilter = 172.16.1.2
    IKELocalIPFilter = 172.16.100.2
    Proposals = n1
    LocalAuth  = local:IKEv2_AUTH
    RemoteAuth  = remote:IKEv2_AUTH
    DPDMode             = ONDEMAND
    DPDIdleDuration     = 3
    Priority            = 10
)

IKERule2 IKERule2:VPN_ISP2:1
(
    IKEPeerIPFilter = 172.16.1.2
    IKELocalIPFilter = 172.17.100.2
    Proposals = n1
    LocalAuth  = local:IKEv2_AUTH
    RemoteAuth  = remote:IKEv2_AUTH
    DPDMode             = ONDEMAND
    DPDIdleDuration     = 3
    Priority            = 20
)

IPsecAction2 IPsecAction2:VPN_ISP1:1
(
    TunnelingParameters = TunnelEntry(
        LocalIPAddress = 172.16.100.2
        PeerAddress = 172.16.1.2
        Prefragmentation = ENABLE
    )
    ContainedProposals = IKEv2_IPSEC_ALGS
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule2:VPN_ISP1:1
)

IPsecAction2 IPsecAction2:VPN_ISP2:1
(
    TunnelingParameters = TunnelEntry(
        LocalIPAddress = 172.17.100.2
        PeerAddress = 172.16.1.2
```

```
        Prefragmentation = ENABLE
    )
    ContainedProposals = IKEv2_IPSEC_ALGS
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule2:VPN_ISP2:1
)

FilterChain IPsecPolicy:VPN_ISP1 (
    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 = IPsecAction2:VPN_ISP1:1 >
        LogEventID = "IPsec:Protect:VPN_ISP1:1:IPSEC_ACL_HUB_AND_SPOKE"
    )
)

FilterChain IPsecPolicy:VPN_ISP2 (
    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 = IPsecAction2:VPN_ISP2:1 >
        LogEventID = "IPsec:Protect:VPN_ISP2:1:IPSEC_ACL_HUB_AND_SPOKE"
    )
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/0"
    InputFilter = FilterChain:FW_INPUT_WAN
    OutputFilter = InspectChain:INSPECT
    IPsecPolicy = IPsecPolicy:VPN_ISP1
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/1"
    InputFilter = FilterChain:FW_INPUT_WAN
    OutputFilter = InspectChain:INSPECT
    IPsecPolicy = IPsecPolicy:VPN_ISP2
)

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

3. Конфигурационный файл `/etc/multiwan/config.yaml`:

```
config_format_version: 1
```

```
links_params:
  links:
  - name: ISP1
    iface_name: eth0
    prio: 300
    routes:
      ipv4:
        networks:
        - 0.0.0.0/0
        nexthop: 172.16.100.1
    check:
      hosts_reachability: any
      ipv4:
        hosts:
        - 172.16.101.15
        - 172.16.1.2
      method:
        icmp:
          max_rtt: 100
          interval: 3
          restore_interval: 1
          restore_count: 10
          failure_count: 10
          tos: 184
          data_size: 56
  - name: ISP2
    iface_name: eth1
    prio: 200
    routes:
      ipv4:
        networks:
        - 0.0.0.0/0
        nexthop: 172.17.100.1
    check:
      hosts_reachability: any
      ipv4:
        hosts:
        - 172.16.101.15
        - 172.16.1.2
      method:
        icmp:
          max_rtt: 100
          interval: 3
          restore_interval: 1
          restore_count: 10
          failure_count: 10
          tos: 184
          data_size: 56
```

Конфигурации криптошлюза Spoke

1. Консоль cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity address
crypto ipsec security-association lifetime kilobytes unlimited
```

```
username          cscons          privilege          15          secret          5
$6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOWZlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi
20ElR4wtMXTl0
aaa new-model
!
!
hostname Spoke
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
!
!
!
!
logging console critical
!
!
crypto ikev2 proposal 1
 encryption magma
 group gost3410-12-256
 prf streebog512
!
crypto ikev2 policy 1
 proposal 1
!
crypto ikev2 authentication sig IKEv2_AUTH
 identity dn
 set identity-mapping auto
!
crypto ikev2 profile IKEv2_ALGS
 authentication remote IKEv2_AUTH
 authentication local IKEv2_AUTH
 set policy 1
 dpd 3 on-demand
!
crypto ikev2 ipsec-proposal IKEv2_IPSEC_ALGS
 esp encryption magma-mgm
!
object-group network SELF
 description IP on WAN interfaces
 host 172.16.1.2
!
ip access-list extended IPSEC_ACL_HUB_AND_SPOKE
 permit ip 192.168.1.0 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_HUB_AND_SPOKE
 set ikev2-ipsec-proposal IKEv2_IPSEC_ALGS
```

```
 set ikev2-profile IKEv2_ALGS
 set security-association mss-fix out
 set mode tunnel
 set peer 172.16.100.2
 set peer 172.17.100.2
 set pre-fragmentation
!
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
!
!
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:   Wed Apr 29 11:14:25 2026

GlobalParameters(
    Title                       = "This LSP was automatically generated by CSP Converter
at Wed Apr 29 11:14:25 2026 (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
)

IKEParameters2 (
    SessionTimeout = 20
    ResponseTimeMax = 10
    SendRetries = 5
    RetryTimeBase = 1
    RetryTimeMax = 10
    CookieChallenge = 10
    NATKeepaliveInterval = 20
    NATKeepaliveRetain = 300
    FragmentSize = 576
    FragmentSize6 = 1280
)

IKEProposal2 n1 (
    EncryptionAlg = ENCR_MAGMA_MGM_KTREE
    PrfAlg        = PRF_HMAC_STREEBOG_512
    KeyAlg        = GOST3410_2012_256
)

IPsecProposal2 IKEv2_IPSEC_ALGS (
    ESPEncryptionAlg = ENCR_MAGMA_MGM_KTREE
)

IdentityEntry id:local:IKEv2_AUTH (
    DistinguishedName = USER_SPECIFIC_DATA
)

IKEAuthSignature2 local:IKEv2_AUTH (
    Identity  = id:local:IKEv2_AUTH
    Credential  = CertDescription()
    SendCertMode  = ALWAYS
)

IdentityEntry id:remote:IKEv2_AUTH (
    DistinguishedName = USER_SPECIFIC_DATA
)

IKEAuthSignature2 remote:IKEv2_AUTH (
    Identity  = id:remote:IKEv2_AUTH
    Credential  = CertDescription()
    SendRequestMode  = ALWAYS
)

FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.16.1.2
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
```

```
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.16.1.2
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.16.1.2
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 172.16.1.2
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        SourceIP = 0.0.0.0/0
        DestinationIP = 0.0.0.0/0
        Action = DROP
    )
    IPv6Filters = Filter (
        SourceIP = ::/0
        DestinationIP = ::/0
        Action = DROP
    )
)

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

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

IKEParameters(
    FragmentSize = 0
)

IKERule2 IKERule2:VPN:1
(
    IKEPeerIPFilter = 172.16.100.2, 172.17.100.2
    Proposals = n1
    LocalAuth  = local:IKEv2_AUTH
    RemoteAuth  = remote:IKEv2_AUTH
    DPDMode             = ONDEMAND
    DPDIdleDuration     = 3
    Priority            = 10
)

IPsecAction2 IPsecAction2:VPN:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.100.2
        Prefragmentation = ENABLE
    ),
    TunnelEntry(
        PeerAddress = 172.17.100.2
        Prefragmentation = ENABLE
    )
    ContainedProposals = IKEv2_IPSEC_ALGS
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule2: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 = IPsecAction2:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_ACL_HUB_AND_SPOKE"
    )
)

NetworkInterface (
```

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

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