Настройка DMVPN (phase 2) dual cloud с протоколом динамической маршрутизации iBGP

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

Данная инструкция описывает настройку DMVPN для сетевых инженеров, знакомых с данной технологией по оборудованию Cisco, а также разбирающихся в базовых настройках «С-Терра Шлюз».

Краткое описание технологии DMVPN

Технология DMVPN (англ. Dynamic Multipoint Virtual Private Network – динамическая многоточечная виртуальная частная сеть) позволяет динамически строить защищенные соединения без необходимости описывать каждое соединение в политике безопасности, что существенно упрощает масштабирование крупных сетей с полносвязной топологией. Автоматическое построение полносвязной защищенной топологии достигается за счет применения динамических протоколов маршрутизации и таких протоколов как NHRP (англ. Next Hop Resolution Protocol – протокол разрешения следующего узла) и multipoint GRE (англ. Generic Routing Encapsulation).

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, NHRP, GRE, OSPF, BGP, VRRP.

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.

© 2003-2025 S-Terra CSP

Требования к инфраструктуре

1. Требования к устройствам.

   1.1. «С-Терра Шлюз».

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

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

   1.2. Центр выпуска сертификатов.

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

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

© 2003-2025 S-Terra CSP

Общая схема стенда

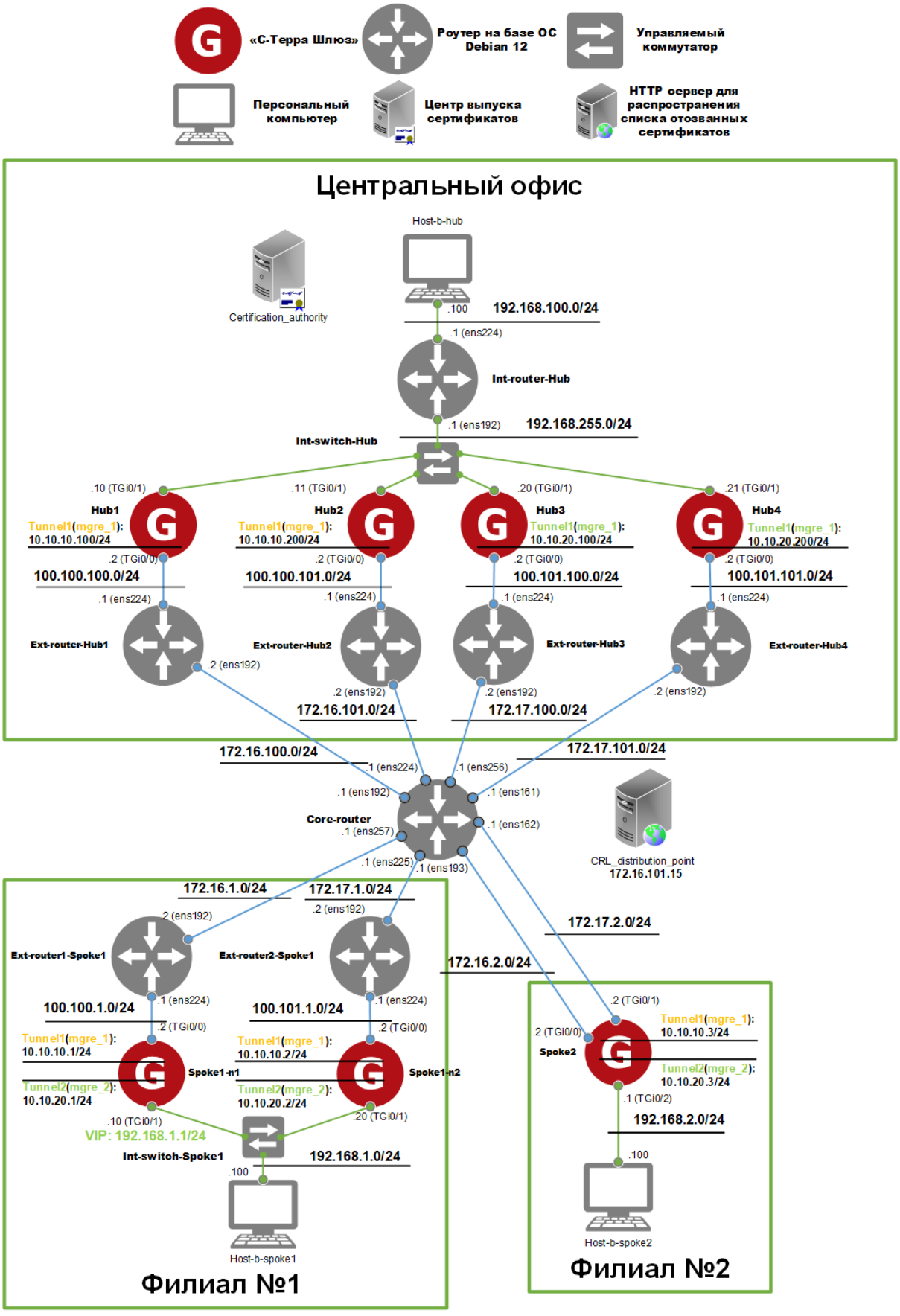


Рисунок 1. Общая схема стенда

Общая логика работы

Данная инструкция описывает настройку технологии DMVPN на криптошлюзах безопасности «С-Терра Шлюз» 5.0. Будет настроено две mGRE подсети (dual cloud) и использован протокол динамической маршрутизации iBGP. В каждой mGRE подсети будет по два Hub. По окончании настройки реализуется возможность построения динамических защищенных соединений между всеми подсетями филиалов, а также подсетями филиалов и центрального офиса (см. Рисунок 1).

Стенд состоит из 20 устройств:

- «С-Терра Шлюз 5.0»: Hub1, Hub2, Hub3, Hub4, Spoke1-n1, Spoke1-n2, Spoke2.
- Роутеры на базе ОС Debian 12: Int-router-Hub, Ext-router-Hub1, Ext-router-Hub2, Ext-router-Hub3, Ext-router-Hub4, Core-router, Ext-router1-Spoke1, Ext-router2-Spoke1.
- Управляемые коммутаторы: Int-switch-Hub, Int-switch-Spoke1.
- Персональные компьютеры: Host-b-hub, Host-b-spoke1, Host-b-spoke2.

Роутеры на базе ОС Debian 12 выбраны в качестве примера. В общем случае могут быть использованы любые маршрутизаторы, поддерживающие функции NAT и динамической маршрутизации (NAT в ОС Debian 12 будет реализован при помощи iptables, а динамическая маршрутизация при помощи пакета FRR).

Из-за особенностей реализации криптошлюзов безопасности «С-Терра Шлюз» 5.0, для совместимости работы «С-Терра Шлюз» 4.3 потребуется дополнительная настройка. Подробнее про совместимость «С-Терра Шлюз» 5.0 с «С-Терра Шлюз» 4.3 описано в главе «Совместимость «С-Терра Шлюз» 5.0 с «С-Терра Шлюз» 4.3».

1. Размещение устройств.

   1.1. В центральном офисе размещаются: центр выпуска сертификатов (Certification_authority), 4 криптошлюза «С-Терра Шлюз» (Hub1, Hub2, Hub3 и Hub4) версии 5.0, 4 маршрутизатора, преобразующие IPv4 адреса (Ext-router-Hub1, Ext-router-Hub2, Ext-router-Hub3, Ext-router-Hub4), маршрутизатор, анонсирующий в BGP маршрут до защищаемой подсети центрального офиса (Int-router-Hub), и персональный компьютер (Host-b-hub).

   1.2. В филиале № 1 размещаются: 2 криптошлюза «С-Терра Шлюз» (Spoke1-n1 и Spoke1-n2) версии 5.0, 2 маршрутизатора, преобразующие IPv4 адрес, и персональный компьютер (Host-b-spoke1).

   1.3. В филиале № 2 размещаются: криптошлюз «С-Терра Шлюз» (Spoke2) версии 5.0 и персональный компьютер (Host-b-spoke2).

   1.4. В неконтролируемом сегменте размещаются: маршрутизатор (Core-router) и HTTP сервер для распространения списка отозванных сертификатов (CRL_distribution_point).

2. Подключение к сети Интернет.

   В данном сценарии для эмуляции сети Интернет используется маршрутизатор Core-router.

   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. Криптошлюзы Hub1, Hub2, Hub3 и Hub4 подключаются к сети Интернет с помощью статической маршрутизации (маршрут по умолчанию через маршрутизаторы Ext-router-Hub1, Ext-router-Hub2, Ext-router-Hub3 и Ext-router-Hub4 соответственно).

   2.2. Криптошлюзы Spoke1-n1 и Spoke1-n2 подключается к сети Интернет с помощью статической маршрутизации (маршрут по умолчанию через маршрутизатор Ext-router1-Spoke и Ext-router2-Spoke соответственно).

© 2003-2025 S-Terra CSP

2.3. Криптошлюз Spoke2 подключается к сети Интернет с помощью статической маршрутизации (маршрут по умолчанию через маршрутизатор Core-router).

3. Параметры шифрования IPsec.

Весь трафик между центральным офисом (192.168.100.0/24) и филиалами (192.168.1.0/24, 192.168.2.0/24) инкапсулируется в GRE, а GRE трафик защищается с использованием алгоритмов ГОСТ и протокола IPsec в транспортном IPv4 режиме.

Между криптошлюзами в филиалах и центральным офисом устанавливаются статические IPsec туннели. Инициатором подключения является служебный трафик протоколов NHRP/BGP.

Между филиальными криптошлюзами устанавливаются динамические IPsec туннели. Инициатором построения туннелей является реальный трафик защищаемых подсетей филиалов.

Режим работы IPsec туннеля должен быть транспортным. Работа IPsec туннеля в туннельном режиме не поддерживается для сети DMVPN на криптошлюзах безопасности «С-Терра Шлюз 5.0».

3.1. Параметры протокола IKE

3.1.1 Параметры протокола IKE для версии 5.0:

- 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).

3.1.2 Параметры протокола IKE для совместимости с версией 4.3:

- Аутентификация при помощи цифровых сертификатов, алгоритм подписи – ГОСТ Р 34.10-2012 (ключ 256 бит);
- Алгоритм шифрования – ГОСТ 28147-89 (ключ 256 бит);
- Алгоритм вычисления хеш-функции – ГОСТ Р 34.11-2012 ТК26 (ключ 256 бит);
- Алгоритм выработки общего ключа (аналог алгоритма Диффи-Хеллмана) – VKO_GOSTR3410_2012_256 (ключ 256 бит).

3.2. Параметры протокола ESP

3.2.1 Параметры протокола ESP для версии 5.0:

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

3.2.2 Параметры протокола ESP для совместимости с версией 4.3:

- Комбинированный алгоритм шифрования и имитозащиты (контроль целостности) – ESP_GOST-4M-IMIT (ключ 256 бит).

4. Параметры межсетевого экранирования.

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

4.1. Для «С-Терра Шлюз» версии 5.0:

4.1.1 На LAN интерфейсах IPv4 трафик не контролируется.

4.1.2 На LAN интерфейсах IPv6 трафик заблокирован.

4.1.3 На WAN интерфейсах разрешен входящий IPv4 трафик:

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

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

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

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

4.1.4 На WAN интерфейсах заблокирован исходящий незашифрованный GRE трафик и транзитный трафик до приватных IP адресов.

4.1.5 На WAN интерфейсах IPv6 трафик заблокирован.

4.1.6 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).

4.2. Для «С-Терра Шлюз» версии 4.3(в сценарии не используется, указано для совместимости«С-Терра Шлюз» версии 5.0 с «С-Терра Шлюз» версии 4.3):

Использование stateful фильтрации для версии 4.3 нежелательно, поэтому рекомендуется использовать статические правила (ACL).

4.2.1 На LAN интерфейсах IPv4 трафик не контролируется.

4.2.2 На WAN интерфейсах заблокирован исходящий незашифрованный GRE трафик и трафик до приватных IP адресов.

4.2.3 На LAN и WAN интерфейсах IPv6 трафик заблокирован (протокол IPv6 отключен в ядре операционной системы).

5. Отказоустойчивость.

Отказоустойчивость обеспечивается протоколом NHRP и протоколом динамической маршрутизации BGP. Филиальные криптошлюзы одновременно регистрируются по протоколу NHRP на Hub1, Hub2, Hub3 и Hub4. Hub1 и Hub2 находятся в одной mGRE-подсети (10.10.10.0/24), а Hub3 и Hub4 находятся во второй mGRE-подсети (10.10.20.0/24).

Поддержка нескольких Hub в одной mGRE-подсети осуществляется, начиная с «С-Терра Шлюз» версии 5.0.

5.1. Динамическая маршрутизация.

Динамическая маршрутизация между криптошлюзами филиалов и центрального офиса обеспечивается протоколом Internal BGP. Все криптошлюзы безопасности находятся в одной автономной системе №65000.

Криптошлюзы безопасности, установленные в филиалах, отсылают маршрутную информацию о защищаемых подсетях только криптошлюзам в центральный офис, соответственно, получение маршрутной информации происходит тоже только из центрального офиса. Маршруты до защищаемых подсетей устанавливаются через mGRE-интерфейсы. Центральные криптошлюзы являются роут-рефлекторами (route-reflector), а филиальные – роут-рефлектор клиентами (route-reflector client).

Между криптошлюзами в центральном офисе и роутером Int-router-Hub (AS №65001) передача маршрутной информации обеспечивается при помощи External BGP. Между криптошлюзами в центральном офисе передача маршрутной информации не обеспечивается.

На каждом криптошлюзе филиала и на роутере Int-router-Hub будет набор резервирующих друг друга маршрутов для каждой из защищаемых подсетей, из которых будет использоваться лучший из доступных в настоящий момент. Выбор лучшего из доступных маршрутов будет настроен по правилу статического приоритета, которые будут определены при конфигурировании.

© 2003-2025 S-Terra CSP

5.2. Отказоустойчивость филиала № 1.

Повышение отказоустойчивости Филиала № 1 обеспечивается использованием протокола VRRP на LAN интерфейсе криптошлюзов. Криптошлюзы Spoke1-n1 и Spoke1-n2 объединяются в отказоустойчивый кластер, работающий на базе протокола VRRP в режиме Master/Backup. Протокол VRRP реализуется сервисом keepalived.

Конфигурация между нодами кластера не синхронизируется. При изменении настроек на одной ноде нужно выполнить аналогичные настройки на другой.

Нода, находящаяся в состоянии Master, владеет VIP адресом в доверенном сегменте и, соответственно, обрабатывает трафик. Нода, находящаяся в состоянии Backup или Fault, не владеет VIP адресом.

При помощи скрипта `/etc/keepalived/scripts/check_ping` сервис `keepalived` дополнительно контролирует состояние канала Интернет-провайдера посредством проверки доступности заданных узлов по протоколу ICMP. Если все заданные узлы недоступны определенное время, то нода кластера переходит в состояние FAULT, что приводит к переключению кластера на резервную ноду (в случае, если она была в состоянии BACKUP), которая подключена к другому Интернет-провайдеру. Резервирование подключения к сети Интернет обеспечивается следующим принципом – каждая нода кластера филиала подключается к соответствующему Интернет-провайдеру.

При помощи конфигурационного файла `/etc/keepalived/notify_common.conf` и скрипта `/etc/keepalived/scripts/notify_common` обеспечивается следующая логика. Если нода переходит в состояние BACKUP или FAULT, то BGP пиринг с центральными криптошлюзами выключается. Если нода переходит в состояние MASTER, то BGP пиринг с центральными криптошлюзами включается.

Также в настройках FRR добавляется маршрут до защищаемой сети центрального офиса через VIP адрес с дистанцией 240. Таким образом, если был получен маршрут по BGP, он будет более приоритетным чем маршрут через VIP адрес.

На портах коммутатора Int-switch-Spoke1, подключенных к криптошлюзам, обязательно должен быть включен режим `portfast`.

5.3. Отказоустойчивость филиала № 2.

Отказоустойчивость филиала реализована резервированием каналов связи и управлением резервированными каналами связи при помощи сервиса «multiwan». Режим работы каналов основной/резервный. При отказе основного канала связи происходит переключение на работу через резервный канал связи (если он был доступен). При восстановлении работы основного канала связи происходит переключение на работу через основной канал связи.

6. Масштабирование.

Реальные сети DMVPN могут содержать существенно больше Spoke’ов, чем в настоящем сценарии, где демонстрируются основные возможности реализации DMVPN сетей на криптошлюзах версии 5.0.

Для DMVPN сетей, построенных на основе криптошлюзов версии 5.0, максимальное рекомендуемое количество Spoke’ов составляет не более 1000 Spoke’ов (при условии достаточной производительности Hub’а).

Рекомендации по настройке больших DMVPN сетей приведены в Приложении Рекомендации по настройке больших DMVPN сетей.

Настройка стенда

Настройка устройства Host-b-hub

На данном устройстве должны быть настроены IP-адреса и маршрутизация в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).

Настройка устройства Host-b-spoke1

На данном устройстве должны быть настроены IP-адреса и маршрутизация в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).

Настройка устройства Host-b-spoke2

На данном устройстве должны быть настроены IP-адреса и маршрутизация в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).

Настройка устройства Int-switch-Hub

На данном устройстве сетевые интерфейсы, коммутирующие криптошлюзы Hub1, Hub2, Hub3, Hub4 и роутер Int-router-Hub, должны быть в одном VLAN и в режиме portfast.

Настройка устройства Int-switch-Spoke1

На данном устройстве сетевые интерфейсы, коммутирующие криптошлюзы Spoke1-n1, Spoke1-n2 и персональный компьютер Host-b-spoke1, должны быть в одном VLAN и в режиме portfast.

Настройка устройства Core-router

На данном устройстве должны быть настроены IP-адреса в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).

Настройка устройства Ext-router-Hub1

На данном устройстве должно быть настроено:

1. IP-адреса и маршрутизация в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).
2. Source NAT/Destination NAT. В приложении представлены настройки iptables (содержимое файла `/etc/iptables.rules`).

Настройка устройства Ext-router-Hub2

На данном устройстве должно быть настроено:

1. IP-адреса и маршрутизация в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).
2. Source NAT/Destination NAT. В приложении представлены настройки iptables (содержимое файла `/etc/iptables.rules`).

Настройка устройства Ext-router-Hub3

На данном устройстве должно быть настроено:

1. IP-адреса и маршрутизация в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).
2. Source NAT/Destination NAT. В приложении представлены настройки iptables (содержимое файла `/etc/iptables.rules`).

Настройка устройства Ext-router-Hub4

На данном устройстве должно быть настроено:

1. IP-адреса и маршрутизация в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).
2. Source NAT/Destination NAT. В приложении представлены настройки iptables (содержимое файла `/etc/iptables.rules`).

Настройка устройства Ext-router1-Spoke1

На данном устройстве должно быть настроено:

1. IP-адреса и маршрутизация в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).
2. Source NAT/Destination NAT. В приложении представлены настройки iptables (содержимое файла `/etc/iptables.rules`).

Настройка устройства Ext-router2-Spoke1

На данном устройстве должно быть настроено:

1. IP-адреса и маршрутизация в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).
2. Source NAT/Destination NAT. В приложении представлены настройки iptables (содержимое файла `/etc/iptables.rules`).

Настройка устройства Int-router-Hub

На данном устройстве должно быть настроено:

1. IP-адреса в соответствии с общей схемой стенда (см. Рисунок 1). В приложении представлены сетевые настройки (содержимое файла `/etc/network/interfaces`).
2. Динамическая маршрутизация. В приложении представлены настройки FRR.

Настройка устройства Hub1

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). 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 применяется после выхода из режима конфигурирования.

1. Задайте имя устройства:

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

2. Включите WAN и LAN интерфейсы и настройте IP адрес на WAN интерфейсе в соответствии со схемой стенда:

```
Hub1(config)#interface TenGigabitEthernet0/0
Hub1(config-if)# ip address 100.100.100.2 255.255.255.0
Hub1(config-if)# no shutdown
Hub1(config-if)#interface TenGigabitEthernet0/1
Hub1(config-if)# ip address 192.168.255.10 255.255.255.0
Hub1(config-if)# no shutdown
Hub1(config-if)# exit
```

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

3. Проверьте доступность маршрута по умолчанию:

```
Hub1#ping 100.100.100.1
```

```
PING 100.100.100.1 (100.100.100.1) 100(128) bytes of data.
108 bytes from 100.100.100.1: icmp_req=1 ttl=64 time=0.404 ms
108 bytes from 100.100.100.1: icmp_req=2 ttl=64 time=0.216 ms
108 bytes from 100.100.100.1: icmp_req=3 ttl=64 time=0.232 ms
108 bytes from 100.100.100.1: icmp_req=4 ttl=64 time=0.212 ms
108 bytes from 100.100.100.1: icmp_req=5 ttl=64 time=0.230 ms

--- 100.100.100.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4002ms
rtt min/avg/max/mdev = 0.212/0.258/0.404/0.075 ms
```

Настройка шифрования

Настройки шифрования осуществляются из cisco-like консоли криптошлюза.

1. Параметры IKE:

   1.1. Задайте общие параметры IKE:

```
Hub1(config)#crypto isakmp identity dn
Hub1(config)#crypto isakmp keepalive 3 3
Hub1(config)#crypto isakmp keepalive retry-count 3
Hub1(config)#crypto isakmp fragmentation
Hub1(config)#crypto isakmp security-association lifetime delta 50
Hub1(config)#crypto isakmp initiator-sessions-max 100
Hub1(config)#crypto isakmp responder-sessions-max 100
```

   1.2. Создайте политику **1**, описывающую алгоритмы IKE туннеля для совместимости с версией 5.0:

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

   1.3. Опционально: создайте политику **2**, описывающую алгоритмы IKE туннеля для совместимости с версией 4.3:

```
Hub1(config)#crypto isakmp policy 2
Hub1(config-isakmp)# encr gost
Hub1(config-isakmp)# hash gost341112-256-tc26
Hub1(config-isakmp)# authentication gost-sig
Hub1(config-isakmp)# group vko2
Hub1(config-isakmp)#exit
```

   1.4. Создайте профиль настроек ISAKMP (алгоритм IKE) под именем `ALGS_PROFILE`, чтобы далее его прикрепить к соответствующему разделу крипто-карты:

```
Hub1(config)#crypto isakmp profile ALGS_PROFILE
```

```
% A profile is deemed incomplete until it has match identity statements
```

```
Hub1(conf-isa-prof)# match identity dn
Hub1(conf-isa-prof)# set identity mapping
Hub1(conf-isa-prof)# set policy 1
Hub1(conf-isa-prof)# set policy 2
Hub1(conf-isa-prof)#exit
```

Примечание - В базовом сценарии для указания различающихся политик, используемых для подключения криптошлюзов версий 4.3 и 5.0, использовалось два раздела криптокарты. Эти разделы были обособлены друг от друга двумя списками доступа с непересекающимися адресными пространствами. Однако, для перевода существующей DMVPN сети с криптошлюзов версии 4.3 на версию 5.0 представляется актуальной задача сохранения существующего единого адресного пространства. В связи с этим в настоящем сценарии в отличии от базового сценария используется один раздел криптокарты как для подключения криптошлюзов версии 4.3, так и для подключения криптошлюзов версии 5.0. Так как к разделу криптокарты можно привязать только один isakmp профиль используется isakmp профиль с двумя привязанными isakmp политиками – одна для подключения криптошлюзов версии 4.3, вторая для подключения криптошлюзов версии 5.0.

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. Создайте IPsec трансформ, задающий комбинированный алгоритм шифрования и имитозащиты ГОСТ 34.12-2015 Магма в режиме MGM для трафика (для версий 5.0):

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

Режим работы IPsec туннеля должен быть транспортным. В отличие от криптошлюзов версии 4.3 туннельный режим IPsec в сценарии DMVPN для криптошлюзов версии 5.0 не поддерживается и не может быть использован даже при отсутствии NAT.

2.3. Опционально: для совместимости с 4.3 создайте IPsec трансформ, задающий комбинированный алгоритм шифрования и имитозащиты на базе ГОСТ 28147-89 для трафика:

```
Hub1(config)#crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-
4m-imit
Hub1(cfg-crypto-trans)# mode transport
Hub1(cfg-crypto-trans)#exit
Hub1(config)#
```

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

```
Hub1(config)#object-group network SELF
Hub1(config-network-group)#description IP on WAN interface
Hub1(config-network-group)#host 100.100.100.2
Hub1(config-network-group)#exit
```

2.5. Создайте сетевой `object-group` с именем `PRIVATE` для хранения в нем приватных подсетей с целью дальнейшего его использования в ACL:

```
Hub1(config)#object-group network PRIVATE
Hub1(config-network-group)#description private LANs
Hub1(config-network-group)#100.64.0.0 255.192.0.0
Hub1(config-network-group)#172.16.0.0 255.240.0.0
Hub1(config-network-group)#192.168.0.0 255.255.0.0
Hub1(config-network-group)#10.0.0.0 255.0.0.0
Hub1(config-network-group)#exit
```

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

```
Hub1(config)#ip access-list extended IPSEC_DMVPN
Hub1(config-ext-nacl)# remark special DMVPN filter
Hub1(config-ext-nacl)# permit gre object-group SELF each
Hub1(config-ext-nacl)#exit
Hub1(config)#
```

Примечание - Ключевое слово each позволяет строить узкие IPsec туннели для любого адреса назначения. Адрес назначения берется из IP заголовка GRE пакета. в котором находится NBMA адрес криптошлюза.

2.7. Опционально: создайте список доступа (ACL) для IPv4 трафика, который нужно защищать между центральным офисом и филиалом, работающим на версии криптошлюза 4.3:

```
Hub1(config)#object-group network MULTICAST
Hub1(config-network-group)#host 224.0.0.5
Hub1(config-network-group)#host 224.0.0.6
Hub1(config-network-group)#host 224.0.0.18
Hub1(config-network-group)#exit
Hub1(config)#ip access-list extended IPSEC_DMVPN_FOR_4.3
Hub1(config-ext-nacl)# remark special DMVPN filter for 4.3
Hub1(config-ext-nacl)# deny   icmp object-group SELF any
Hub1(config-ext-nacl)# deny   udp object-group SELF any
Hub1(config-ext-nacl)# deny   tcp object-group SELF any
Hub1(config-ext-nacl)# deny   ip object-group SELF object-group MULTICAST
Hub1(config-ext-nacl)# permit ip object-group SELF each
```

2.8. Создайте динамическую крипто-карту (имя DMAP, раздел 1) и привяжите её к статической (имя VPN, раздел 1):

```
Hub1(config)#crypto dynamic-map DMAP 1
Hub1(config-crypto-map)# match address IPSEC_DMVPN
Hub1(config-crypto-map)# set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
```

Для совместимости с 4.3 требуется использовать список доступа для 4.3. А также указать комбинированный алгоритм шифрования и имитозащиты на базе ГОСТ 28147-89:

```
Hub1(config-crypto-map)#match address IPSEC_DMVPN_FOR_4.3
Hub1(config-crypto-map)#set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
GOST_ENCRYPT_AND_INTEGRITY
```

Обратите внимание, что в команде set transform-set важен порядок указания алгоритмов шифрования. В случае постепенного перевода криптошлюзов на объекте внедрения с версии 4.3 на 5.0 необходимо указываль алгоритмы шифрования в последовательности `MAGMA_GOST_ENCRYPT_AND_INTEGRITY` (обязательно первым) `GOST_ENCRYPT_AND_INTEGRITY` (вторым). Если указать алгоритмы шифрования в обратном порядке криптошлюзы версии 5.0 будут взаимодействовать между собой на старом алгоритме `GOST_ENCRYPT_AND_INTEGRITY`.

```
Hub1(config-crypto-map)# set isakmp-profile ALGS_PROFILE
Hub1(config-crypto-map)# set security-association mss-fix out
Hub1(config-crypto-map)# set pre-fragmentation
Hub1(config-crypto-map)# set dead-connection history off
Hub1(config-crypto-map)#exit
Hub1(config)#crypto map VPN 1 ipsec-isakmp dynamic DMAP
Hub1(config)#
```

2.9. Прикрепите созданную статическую крипто-карту к внешнему интерфейсу `TenGigabitEthernet0/0`:

```
Hub1(config)#interface TenGigabitEthernet0/0
```

```
Hub1(config-if)#crypto map VPN
Hub1(config-if)#exit
```

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка политики обработки COC (CRL)

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

Включите проверку COC и настройте автоматическую загрузку и импортирование в базу Продукта списка отозванных сертификатов с HTTP сервера CRL_distribution_point.

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка межсетевого экрана

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.

Настройка межсетевого экрана осуществляются из cisco-like консоли.

1. Создайте ACL с именем `FIREWALL_OUT` для исходящего с WAN интерфейса трафика:

   Список доступа для исходящего трафика должен блокировать открытый трафик из защищаемой подсети криптошлюза в подсети RFC1918/RFC6598 (считаем, что трафик непубличных подсетей должен обязательно шифроваться), а также незашифрованный GRE.

   ```
   Hub1(config)#ip access-list extended FIREWALL_OUT
   Hub1(config-ext-nacl)# permit ip object-group SELF object-group PRIVATE
   Hub1(config-ext-nacl)# deny ip any object-group PRIVATE
   Hub1(config-ext-nacl)# deny gre any any
   Hub1(config-ext-nacl)# permit ip any any
   Hub1(config-ext-nacl)#exit
   ```

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

   ```
   Hub1(config)#ip access-list extended FIREWALL_IN
   Hub1(config-ext-nacl)# permit udp any object-group SELF eq isakmp
   Hub1(config-ext-nacl)# permit udp any object-group SELF eq non500-isakmp
   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
   ```

   Примечание - Список доступа для входящего трафика разрешает входящие пакеты протоколов IKE, ESP, ICMP, а остальные блокирует только для случая, когда криптошлюз является ответчиком **впервые.** Если криптошлюз является инициатором или ответчиком в рамках ранее созданной сессии, то ответные пакеты будут разрешены, так как согласно алгоритму работы stateful фильтрации динамические разрешающие правила более приоритетны чем статические. Stateful фильтрация будет настроена далее в данном разделе. Динамические правила stateful фильтрации в cisco-like консоли НЕ отображаются, их можно посмотреть при помощи утилиты `fwconn_mgr` с параметром `show`.

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 списки доступа и правила инспектирования на внешний интерфейс:

```
Hub1(config)#interface TenGigabitEthernet0/0
Hub1(config-if)# ip access-group FIREWALL_IN in
Hub1(config-if)# ip access-group FIREWALL_OUT out
Hub1(config-if)# ip inspect INSPECT out
Hub1(config-if)#exit
```

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка mGRE-интерфейса

На центральных криптошлюзах должно быть по одному mGRE-интерфейсу

Настройки применяются при выходе из настройки интерфейса.

Настройки mGRE-интерфейса осуществляются из cisco-like консоли криптошлюза:

```
Hub1(config)#interface Tunnel1
Hub1(config-if)#ip address 10.10.10.100 255.255.255.0
Hub1(config-if)#mtu 1400
Hub1(config-if)#tunnel key 1
Hub1(config-if)#tunnel mode gre multipoint
Hub1(config-if)#tunnel ttl 64
Hub1(config-if)#tunnel tos inherit
Hub1(config-if)#end
```

Настройка NHRP

Настройки NHRP интерфейсов осуществляются из linux bash, чтобы попасть в linux bash выполните следующую команду в CLI разграничения доступа:

```
Hub1#exit
administrator@Hub1] system
```

```
Entering system shell...
root@Hub1:~#
```

1. Приведите файл `/etc/opennhrp/opennhrp.conf` к виду:

```
interface mgre_1
 multicast dynamic
 holding-time 90
 cisco-authentication secret
 non-caching
```

Примечание – Допустимый диапазон значения `holding-time` – от 30 до 150 секунд. Подробное описание параметра `holding-time` и ограничения по его изменению приведено в разделе Описание параметра holding-time.

2. Добавьте сервис OpenNHRP в автозагрузку и запустите его:

```
root@Hub1:~# systemctl enable opennhrp
```

```
Created symlink /etc/systemd/system/multi-user.target.wants/opennhrp.service -> /lib/systemd/system/opennhrp.service.
```

```
root@Hub1:~# systemctl start opennhrp
```

Настройка BGP

Динамическая маршрутизация настраивается через консоль сервиса FRR.

1. Добавьте сервис FRR в автозагрузку и запустите его:

```
root@Hub1:~# systemctl enable frr
```

```
Synchronizing state of frr.service with SysV service script with
/lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable frr
Created symlink /etc/systemd/system/multi-user.target.wants/frr.service →
/lib/systemd/system/frr.service.
```

```
root@Hub1:~# systemctl start frr
```

2. Войдите в консоль сервиса FRR, для этого нужно из linux bash набрать следующую команду:

```
root@Hub1:~# vtysh
```

```

Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

hub1#
```

3. Настройте BGP (см. описание http://docs.frrouting.org):

```
Hub1# configure terminal
```

3.1. Создайте route-map `FROM-HUB1-OUT`, которая будет менять `local-preference` для маршрутов, которые Hub1 получает из автономной системы 65001 (192.168.100.0/24) и ретранслирует в криптошлюзы филиалов:

```
Hub1(config)# route-map FROM-HUB1-OUT permit 1
Hub1(config-route-map)# set local-preference 100
Hub1(config-route-map)# exit
```

Примечания:

1. route-map будет изменять статический приоритет (`local-preference`) для объявляемых маршрутов или маршрутов, полученных по eBGP и не будет влиять на `local-preference` маршрутов, полученных от Spoke-ов;
2. Обратите внимание, что рекомендуемая логика использования route-map BGP для задания статических приоритетов в версии 5.0 отличается от рекомендовавшейся логики использования route-map BGP в версии 4.3. Это необходимо учитывать при миграции DMVPN сети с криптошлюзов версии 4.3 на 5.0. В версии 4.3 `route-map bgp` настраивались на Spoke'ах в направлении in. В версии 5.0 рекомендуется настраивать route-map BGP и на Hub'ах, и на Spoke'ах в направлении out. Перед началом миграции DMVPN сети с криптошлюзов версии 4.3 на версию 5.0 следует проверить настройки route-map на Spoke'ах и при необходимости предпринять меры по согласованию route-map на Spoke'ах и Hub'ах во избежание ассиметричности трафика между Hub и Spoke'ами.

3.2. Включите процесс протокола BGP для автономной системы 65000 и перейдите в режим его конфигурирования:

```
Hub1(config)# router bgp 65000
```

3.3. Установите идентификатор маршрутизатора:

```
Hub1(config-router)# bgp router-id 10.10.10.100
```

3.4. Включите логирование изменения статуса (например, up, down и другие) BGP соседа в журнале работы:

```
Hub1(config-router)# bgp log-neighbor-changes
```

3.5. Отключите обязательное наличие BGP фильтров для обмена маршрутами по eBGP сессий:

```
Hub1(config-router)# no bgp ebgp-requires-policy
```

Примечание - Команда `bgp ebgp-requires-policy` включена по умолчанию и требует наличие фильтров при работе с eBGP соседями. При отсутствии входящих фильтров, маршруты не будут приниматься от eBGP соседа. При отсутствии исходящих фильтров, маршруты не будут транслироваться eBGP соседу. В настоящем сценарии отключение ebgp-requires-policy производится с целью упрощения конфигурирования.

3.6. Установите таймеры keepalive и hold time для BGP соседей:

```
Hub1(config-router)# timers bgp 3 9
```

Примечание - Данные настройки производятся с целью ускорения сходимости iBGP и по согласованию с Производителем могут изменяться.

3.7. Создайте группу spokes-ibgp с номером автономной системы 65000 для BGP соседей из mgre подсети (криптошлюзов филиалов):

```
Hub1(config-router)# neighbor spokes-ibgp peer-group
Hub1(config-router)# neighbor spokes-ibgp remote-as 65000
Hub1(config-router)# bgp listen range 10.10.10.0/24 peer-group spokes-ibgp
```

3.8. Измените максимальное количество BGP соседей, которые могут подключиться к криптошлюзу по BGP одновременно:

```
Hub1(config-router)# bgp listen limit 5000
```

3.9. Добавьте bgp соседа 192.168.255.1 с номером автономной системы 65001:

```
Hub1(config-router)# neighbor 192.168.255.1 remote-as 65001
```

3.10. Включите пересылку маршрутов для группы `spokes-ibgp`:

```
Hub1(config-router)# address-family ipv4 unicast
Hub1(config-router-af)# neighbor spokes-ibgp route-reflector-client
```

3.11. Включите изменение nexthop адреса для маршрутов, полученных по eBGP, на адрес Hub1:

```
Hub1(config-router-af)# neighbor spokes-ibgp next-hop-self
```

Примечание - Чтобы изменять next-hop адреса маршрутов, полученных по iBGP, можно использовать команду «`neighbor spokes-ibgp next-hop-self force`». В этом случае трафик между филиалами будет идти через центр (Аналог первой фазы DMVPN).

3.12. Примените route-map `FROM-HUB1-OUT` на отправленные маршруты к соседям из группы `spokes-ibgp`:

```
Hub1(config-router-af)# neighbor spokes-ibgp route-map FROM-HUB1-OUT out
```

Примечание - Для данного сценария `local-preference` будет изменён для ретранслируемого маршрута до 192.168.100.0/24.

3.13. Сохраните изменения:

```
Hub1(config-router-af)# end
Hub1# write
```

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

В Приложении представлены тексты конфигураций для криптошлюза Hub1:

- текст консоли cisco-like;
- Содержимое `/etc/opennhrp/opennhrp.conf`;
- Содержимое `/etc/frr/frr.conf`.

Настройка устройства Hub2

Настройка устройства Hub2 осуществляется аналогично устройству Hub1 с учетом соответствующих IP-адресов (см. Рисунок 1). В настройках FRR значение `local-preference` должно быть равно 75.

Значение mGRE key на Hub1 и на Hub2 должны совпадать. Но должен отличаться от значения mGRE key на Hub3 и Hub4.

В Приложении представлены тексты конфигураций для криптошлюза Hub2:

- текст консоли cisco-like;
- Содержимое `/etc/opennhrp/opennhrp.conf`;
- Содержимое `/etc/frr/frr.conf`.

Настройка устройства Hub3

Настройка устройства Hub3 осуществляется аналогично устройству Hub1 с учетом соответствующих IP-адресов (см. Рисунок 1). В настройках FRR значение `local-preference` должно быть равно 50.

Значение mGRE key на Hub3 и на Hub4 должны совпадать. Но должен отличаться от значения mGRE key на Hub1 и Hub2.

В Приложении представлены тексты конфигураций для криптошлюза Hub3:

- текст консоли cisco-like;
- Содержимое `/etc/opennhrp/opennhrp.conf`;
- Содержимое `/etc/frr/frr.conf`.

Настройка устройства Hub4

Настройка устройства Hub4 осуществляется аналогично устройству Hub1 с учетом соответствующих IP-адресов (см. Рисунок 1). В настройках FRR значение `local-preference` должно быть равно 25.

Значение mGRE key на Hub3 и на Hub4 должны совпадать. Но должен отличаться от значения mGRE key на Hub1 и Hub2.

В Приложении представлены тексты конфигураций для криптошлюза Hub4:

- текст консоли cisco-like;
- Содержимое `/etc/opennhrp/opennhrp.conf`;
- Содержимое `/etc/frr/frr.conf`.

Настройка устройства Spoke1-n1

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). Доверенным каналом связи также считается канал, защищенный при помощи протокола IPsec.

Начальные настройки, настройки PKI, настройка паролей доступа к консоли cisco-like, настройка сетевых параметров, настройка шифрования, настройка политики обработки COC (CRL) и настройка межсетевого экрана осуществляется аналогично устройству Hub1 с учетом соответствующих IP-адресов (см. Рисунок 1).

© 2003-2025 S-Terra CSP

Настройка mGRE-интерфейсов

На филиальных криптошлюзах должно быть по два mGRE-интерфейса (по одному на каждую mGRE-подсеть).

Настройки применяются при выходе из настройки интерфейса.

1. Настройки mGRE-интерфейса осуществляются из cisco-like консоли криптошлюза:

```
Spoke1-n1#conf t
Spoke1-n1(config)#interface Tunnel1
Spoke1-n1(config-if)#ip address 10.10.10.1 255.255.255.0
Spoke1-n1(config-if)#mtu 1400
Spoke1-n1(config-if)#tunnel key 1
Spoke1-n1(config-if)#tunnel mode gre multipoint
Spoke1-n1(config-if)#tunnel ttl 64
Spoke1-n1(config-if)#tunnel tos inherit
Spoke1-n1(config-if)#exit
Spoke1-n1(config)#interface Tunnel2
Spoke1-n1(config-if)#ip address 10.10.20.1 255.255.255.0
Spoke1-n1(config-if)#mtu 1400
Spoke1-n1(config-if)#tunnel key 2
Spoke1-n1(config-if)#tunnel mode gre multipoint
Spoke1-n1(config-if)#tunnel ttl 64
Spoke1-n1(config-if)#tunnel tos inherit
Spoke1-n1(config-if)#end
```

Настройка NHRP

Настройка NHRP осуществляются из linux bash.

1. Приведите файл `/etc/opennhrp/opennhrp.conf` к виду:

Маска в настройках opennhrp должна совпадать с маской на mgre интерфейсе.

```
interface mgre_1
 map 10.10.10.100/24 172.16.100.2 register
 map 10.10.10.200/24 172.16.101.2 register
 multicast nhs
 holding-time 90
 cisco-authentication secret
 non-caching
interface mgre_2
 map 10.10.20.200/24 172.17.200.2 register
 map 10.10.20.200/24 172.17.101.2 register
 multicast nhs
 holding-time 90
 cisco-authentication secret
 non-caching
```

Примечание – В больших DMVPN сетях, содержащих более сотни устройств, может наблюдаться повышение нагрузки на криптошлюзы Центрального офиса после потери связи со всеми или большинством филиалов, например, из-за отказа провайдера. В этом случае криптошлюзы филиалов могут направлять Registration Request на Hub практически одновременно создавая пик нагрузки на него. Дополнительно, количество одновременно создаваемых туннелей на Hub ограничено. Это может привести к увеличению длительности подключения филиальных криптошлюзов и полного восстановления DMVPN сети после отказа. Для повышения равномерности распределения по времени запросов Registration Request от филиальных

криптошлюзов введен параметр `reregister-time` – время до начала перерегистрации Spoke на Hub после потери связи между ними. Пример настроек с использованием `reregister-time`:

```
interface mgre_1
 map 10.10.10.100/24 172.16.100.2 register reregister-time 60
 map 10.10.10.200/24 172.16.101.2 register reregister-time 45-75
 multicast nhs
 holding-time 90
 cisco-authentication secret
 non-caching
interface mgre_2
 map 10.10.20.200/24 172.17.200.2 register
 map 10.10.20.200/24 172.17.101.2 register
 multicast nhs
 reregister-time 45-75
 holding-time 90
 cisco-authentication secret
 non-caching
```

Команда `reregister-time A` изменяет время перерегистрации на постоянное указанное значение.

Команда `reregister-time A-B` изменяет время перерегистрации на случайное число из указанного диапазона.

Указание `reregister-time` в начале строки изменяет время перерегистрации на всех Hub'ах, объявленных под этим интерфейсом.

Команда `reregister-time` для интерфейса должна отделяться от команды `map A.B.C.D/X A.B.C.D register` значащей командой из набора допустимых в opennhrp.conf команд. Если команда `reregister-time` для интерфейса после команды `map A.B.C.D/X A.B.C.D register` будет отделяться пробелом, нескольким пробелами или даже символом перевода строки, она будет восприниматься частью команды `map`.

Пример идентичных команд:

```
map 10.10.10.100/24 172.16.100.2 register reregister-time 60
```

и

```
map 10.10.10.100/24 172.16.100.2 register

reregister-time
60
```

Пример, когда `reregister-time` задаётся для интерфейса:

```
interface mgre_2
 map 10.10.20.200/24 172.17.200.2 register
 map 10.10.20.200/24 172.17.101.2 register
 multicast nhs
 reregister-time 45-75
```

Указание `reregister-time` в строке `map A.B.C.D/X A.B.C.D register` изменяет время перерегистрации только для указанного криптошлюза. Указание `reregister-time` для интерфейса изменяет время перерегистрации для всех криптошлюзов, описанных в интерфейсе и без `reregister-time`.

По умолчанию время перерегистрации берётся из диапазона 60-180 секунд.

2. Добавьте сервис OpenNHRP в автозагрузку и запустите его:

```
root@Spoke1-n1:~# systemctl enable opennhrp
```

```
Created symlink /etc/systemd/system/multi-user.target.wants/opennhrp.service -> /lib/systemd/system/opennhrp.service.
```

```
root@Spoke1-n1:~# systemctl start opennhrp
```

3. Через некоторое время проверьте, что между Spoke1-n1 и центральными криптошлюзами Hub1, Hub2, Hub3, Hub4 поднялись IPsec туннели (если туннели не поднялись – проверьте настройки сети/IPsec/Firewall/NHRP, лог-файлы `/var/log/cspvpngate.log` и `/var/log/syslog`):

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 5 (100.100.1.2,4500)-(172.16.100.2,4500) active 2208 2232
2 6 (100.100.1.2,4500)-(172.17.100.2,4500) active 1784 1744
3 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 2240 2200
4 8 (100.100.1.2,4500)-(172.16.101.2,4500) active 1848 1744
5 9 (100.100.1.2,4500)-(172.17.100.2,4500) active 1752 1776

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 5 (100.100.1.2,*)-(172.16.100.2,*) 47 ESP nat-t-trans 308 168
2 6 (100.100.1.2,*)-(172.16.100.2,*) 47 ESP nat-t-trans 16160 15732
3 7 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 348 380
4 8 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 15780 16220
5 9 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 0 16208
6 10 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 13692 0
7 11 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 15316 15744
```

Настройка BGP

Динамическая маршрутизация настраивается через консоль сервиса FRR.

1. Добавьте сервис FRR в автозагрузку и запустите его:

```
root@Spoke1-n1:~# systemctl enable frr
```

```
Synchronizing state of frr.service with SysV service script with /lib/system/system-sysv-install.
Executing: /lib/system/system-sysv-install enable frr
```

```
root@Spoke1-n1:~# systemctl start frr
```

2. Войдите в консоль сервиса FRR, для этого нужно из linux bash набрать следующую команду:

```
root@Spoke1-n1:~# vtysh
```

```

Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

```

3. Добавьте маршрут 192.168.100.0/24 через VIP адрес (192.168.1.1) с дистанцией 240:

```
Spoke1-n1#configure terminal
Spoke1-n1(config)# ip route 192.168.100.0/24 192.168.1.1 240
```

Примечание – Данный маршрут будет устанавливаться в систему на резервной ноде кластера VRRP для обеспечения возможности опроса ее LAN интерфейса из подсети Центрального офиса, например, с целью централизованного сбора диагностической информации из Центрального офиса. На основной ноде кластера VRRP будут находится 2 маршрута: добавленный нами с дистанцией 240 и полученный по BGP с дистанцией 200. FRR добавит в систему маршрут с меньшей дистанцией. Резервная нода кластера VRRP будет находится в состоянии BACKUP/FAULT и BGP пиринг с криптошлюзами Центрального офиса на ней будет остановлен

(смотри настройки в следующем разделе - Настройки VRRP). При остановке BGP пиринга на резервную ноду кластера VRRP перестанут поступать маршруты по BGP. В сценарии для версии 4.3 аналогичный функционал настраивался через конфигурационный файл `/etc/keepalived/notify_common.conf`, управляющий работой скрипта `/etc/keepalived/scripts/notify_common`.

4. Настройте BGP (см. описание http://docs.frrouting.org):

   4.1. Создайте route-map `TO-HUB1-OUT`, `TO-HUB2-OUT`, `TO-HUB3-OUT`, `TO-HUB4-OUT`. Данные route-map будет менять `local-preference` для маршрутов, анонсируемых филиалом в сторону криптошлюзов Центрального офиса (192.168.1.0/24):

```
Spoke1-n1(config)# route-map TO-HUB1-OUT permit 1
Spoke1-n1(config-route-map)#set local-preference 100
Spoke1-n1(config-route-map)#exit
Spoke1-n1(config)#route-map TO-HUB2-OUT permit 1
Spoke1-n1(config-route-map)#set local-preference 75
Spoke1-n1(config-route-map)#exit
Spoke1-n1(config)#route-map TO-HUB3-OUT permit 1
Spoke1-n1(config-route-map)#set local-preference 50
Spoke1-n1(config-route-map)#exit
Spoke1-n1(config)#route-map TO-HUB4-OUT permit 1
Spoke1-n1(config-route-map)#set local-preference 25
Spoke1-n1(config-route-map)#exit
```

   Примечание - На криптошлюзах всех остальных филиалах маршрут до защищаемой подсети Spoke1-n1, полученный через Hub1, будет иметь наибольший приоритет, а маршрут до защищаемой подсети Spoke1-n1, полученный через Hub4, будет иметь наименьший приоритет.

   4.2. Включите процесс протокола BGP для автономной системы 65000 и перейдите в режим его конфигурирования:

```
Spoke1-n1(config)#router bgp 65000
```

   4.3. Установите идентификатор маршрутизатора:

```
Spoke1-n1(config-router)#bgp router-id 192.168.1.10
```

   4.4. Включите логирование изменения статуса (например, up, down и другие) BGP соседа в журнале работы:

```
Spoke1-n1(config-router)#bgp log-neighbor-changes
```

   4.5. Установите таймеры keepalive и holdtime для BGP соседа:

```
Spoke1-n1(config-router)#timers bgp 3 9
```

   4.6. Добавьте mGRE-адреса Hub-ов в качестве BGP соседей с номером автономной системы 65000 и установите время повторной попытки построения BGP соседства в 30 секунд:

```
Spoke1-n1(config-router)#neighbor 10.10.10.100 remote-as 65000
Spoke1-n1(config-router)#neighbor 10.10.10.100 timers connect 30
Spoke1-n1(config-router)#neighbor 10.10.20.100 remote-as 65000
Spoke1-n1(config-router)#neighbor 10.10.20.100 timers connect 30
Spoke1-n1(config-router)#neighbor 10.10.10.200 remote-as 65000
Spoke1-n1(config-router)#neighbor 10.10.10.200 timers connect 30
Spoke1-n1(config-router)#neighbor 10.10.20.200 remote-as 65000
Spoke1-n1(config-router)#neighbor 10.10.20.200 timers connect 30
```

   4.7. Включите анонсирование маршрута до защищаемой подсети Филиала №1:

```
Spoke1-n1(config-router)#address-family ipv4 unicast
Spoke1-n1Spoke1-n1(config-router-af)#network 192.168.1.0/24
```

   4.8. Примените route-map к соседям:

```
Spoke1-n1(config-router-af)#neighbor 10.10.10.100 route-map TO-HUB1-OUT out
Spoke1-n1(config-router-af)#neighbor 10.10.10.200 route-map TO-HUB2-OUT out
Spoke1-n1(config-router-af)#neighbor 10.10.20.100 route-map TO-HUB3-OUT out
Spoke1-n1(config-router-af)#neighbor 10.10.20.200 route-map TO-HUB4-OUT out
```

4.9. Сохраните изменения:

```
Spoke1-n1(config-router-af)#end
Spoke1-n1(config-router)# exit
Spoke1-n1# write
```

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

Настройка VRRP

В отличие от версии 4.3 в версии 5.0 значительная часть настроек VRRP перенесена в cisco-like консоль. Но настройка VRRP конфигурационного файла `/etc/keepalived/notify_common.conf` осуществляется из linux bash.

1. Сохраните копию файла `/etc/keepalived/notify_common.conf` в `/etc/keepalived/notify_common.conf.backup` и приведите файл `/etc/keepalived/notify_common.conf` к виду:

```
FLAG_DMVPN_SCENARIO_ON_CLUSTERED_SPOKE="true"
DMVPN_INTERNAL_ROUTING_PROTOCOL="IBGP"
DMVPN_LOCAl_ASN="65000"
DMVPN_BGP_NEIGHBORS="10.10.10.100 10.10.10.200 10.10.20.100 10.10.20.200"
```

Конфигурационный файл `/etc/keepalived/notify_common.conf` управляет скриптом `/etc/keepalived/scripts/notify_common`. Логика следующая: если нода переходит в состояние BACKUP или FAULT, то BGP пиринг с центральными криптошлюзами (10.10.10.100, 10.10.10.200, 10.10.20.100, 10.10.20.200) выключается. Если нода переходит в состояние MASTER, то BGP пиринг с центральными криптошлюзами включается.

Изменение параметров DMVPN в файле `/etc/keepalived/notify_common.conf` **после настройки keepalived (в процессе работы)**:
При необходимости изменения параметров следует:
1) Перезапустить FRR с помощью команды "`systemctl restart frr`" (это действие загрузит конфигурацию, которая соответствует состоянию MASTER). ВАЖНО: после перезапуска FRR может возникнуть ассиметричная маршрутизация. Это связано с тем, что устройство в состоянии BACKUP или FAULT будет участвовать в обмене BGP маршрутами;
2) Изменить требуемые настройки FRR, которые соответствует состоянию MASTER;
3) Сохранить настройки FRR (команда write).
4) После настройки FRR требуется изменить соответствующие параметры DMVPN в файле `/etc/keepalived/notify_common.conf`.
5) Для устройств, находящихся в состоянии BACKUP или FAULT, следует перезапустить сервис keepalived, чтобы применить настройки из файла `/etc/keepalived/notify_common.conf`. Для устройства в состоянии MASTER настройки применяются после перехода в состояние BACKUP или FAULT.
Если была допущена ошибка в файле `/etc/keepalived/notify_common.conf` в части DMVPN (например, номер автономной системы или IP-адрес BGP пира не совпадает с настройками FRR), то при переходе ноды в состояние BACKUP или FAULT в логах `/var/log/cspvpngate.log` появится сообщение "`VRRP notify common script: ERROR: can not set shutdown on neighbor`". После исправления ошибок в файле достаточно перезапустить сервис keepalived.

2. Остальные настройки VRRP осуществляются из cisco-like консоли криптошлюза. Включите использование vrrp v3:

```
Spoke1-n1# configure terminal
Spoke1-n1(config)# fhrp version vrrp v3
```

3. Включите запуск общего скрипт-файла (`/etc/keepalived/scripts/notify_common`) при переходе VRRP инстанса между состояниями:

```
Spoke1-n1(config)# vrrp notify common
```

4. Создайте правило SLA (по умолчанию для IPv4):

```
Spoke1-n1(config)#vrrp track sla check_isp_link
```

5. Укажите целевые хосты для проверки связи:

```
Spoke1-n1(vrrp-track-sla)# host 172.16.1.1
Spoke1-n1(vrrp-track-sla)# host 172.16.101.15
```

6. Укажите режим SLA на проверку с помощью ICMP:

```
Spoke1-n1(vrrp-track-sla)# mode icmp-echo
```

7. Задайте условие успешности проверки, как доступность любого из указанного хостов:

```
Spoke1-n1(vrrp-track-sla)# reachability any
```

8. Задайте интерфейс, с которого будут отправляться ICMP:

```
Spoke1-n1(vrrp-track-sla)# source TenGigabitEthernet0/0
```

9. Задайте количество допустимых неуспешных попыток проверки доступности хостов:

```
Spoke1-n1(vrrp-track-sla)# fall 15
```

10. Задайте количество необходимых успешных попыток проверки доступности хостов:

```
Spoke1-n1(vrrp-track-sla)# rise 10
```

11. Задайте время ожидания проверки доступности хоста:

```
Spoke1-n1(vrrp-track-sla)# timeout 2
```

12. Интервал между проверками доступности хоста:

```
Spoke1-n1(vrrp-track-sla)# interval 3
Spoke1-n1(vrrp-track-sla)# exit
```

13. Включите VRRP на внутреннем интерфейсе (создает VRRP инстанс):

```
Spoke1-n1(config)# interface TenGigabitEthernet0/1
Spoke1-n1(config-if)# vrrp 1 address-family ipv4
Spoke1-n1(config-if-vrrp)#
```

14. Задайте кластерный IPv4 адрес:

```
Spoke1-n1(config-if-vrrp)# address 192.168.1.1 255.255.255.0
```

15. Установите интервал отправки advertisement-сообщений:

```
Spoke1-n1(config-if-vrrp)# timers advertise 3
```

16. Установите интервал отправки gratuitous ARP-сообщения:

```
Spoke1-n1(config-if-vrrp)# timers garp 5
```

17. Прикрепите к интерфейсу SLA правило:

```
Spoke1-n1(config-if-vrrp)# track sla check_isp_link
```

18. Включите отслеживание внешнего интерфейса:

```
Spoke1-n1(config-if-vrrp)# track interface TenGigabitEthernet0/0
```

19. Отключите preempt mode:

```
Spoke1-n1(config-if-vrrp)# no preempt
```

20. Установите приоритет для VRRP инстанса:

```
Spoke1-n1(config-if-vrrp)# priority 100
Spoke1-n1(config-if-vrrp)#end
```

21. Убедитесь, что нода перешла в состояние MASTER. Для этого войдите в cisco-like консоль криптошлюза и посмотрите настройки VRRP:

```
Spoke1-n1#show vrrp
```

```
Interface                      VRID  State    AddrFamily
-------------------------------------------------------
TenGigabitEthernet0/1          1     Master   IPv4
```

22. Убедитесь, что доступен один из хостов указанных в SLA:

```
Spoke1-n1#sh vrrp sla check_isp_link
```

```


Rule name: check_isp_link
Result:    Success

Hostname                                State       Last checked at
172.16.1.1                              reachable   2022-07-07|13:21:58.593
172.16.101.15                           unknown     None
```

Примечание - Из-за того, что проверяется доступность любого хоста, после получения ответа от 172.16.1.1 проверка второго хоста не происходит.

23. Убедитесь, что на внутреннем интерфейсе eth1 появился кластерный VIP-адрес:

```
root@Spoke1-n1:~# ip address show dev eth1
```

```
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether 00:50:56:9e:c3:fe brd ff:ff:ff:ff:ff:ff
    altname enp11s0
    altname ens192
    altname TenGigabitEthernet0_1
    inet 192.168.1.10/24 brd 192.168.1.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet 192.168.1.1/24 scope global secondary eth1:900
       valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:fe9e:c3fe/64 scope link
       valid_lft forever preferred_lft forever
```

В Приложении представлены тексты конфигураций для криптошлюза Spoke1-n1:

- текст консоли cisco-like;
- Содержимое `/etc/opennhrp/opennhrp.conf`;
- Содержимое `/etc/frr/frr.conf`;
- Содержимое `/etc/opennhrp/opennhrp.conf`;
- Содержимое `/etc/keepalived/keepalived.conf`.

Настройка устройства Spoke1-n2

Настройка устройства Spoke1-n2 осуществляется аналогично устройству Spoke1-n1 с учетом соответствующих IP-адресов (см. Рисунок 1).

Значение bgp router-id у резервной ноды должно отличаться от значения основной ноды (Spoke1-n1).

Значение priority в VRRP у резервной ноды должно быть меньше, чем у основной ноды.

В Приложении представлены тексты конфигураций для криптошлюза Spoke1-n1:

- текст консоли cisco-like;
- Содержимое `/etc/opennhrp/opennhrp.conf`;
- Содержимое `/etc/frr/frr.conf`;
- Содержимое `/etc/opennhrp/opennhrp.conf`;
- Содержимое `/etc/keepalived/keepalived.conf`.

Настройка устройства Spoke2

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). Доверенным каналом связи также считается канал, защищенный при помощи протокола IPsec.

Начальные настройки, настройки PKI, настройка паролей доступа к консоли cisco-like, настройка политики обработки COC (CRL), настройка межсетевого экрана, настройка mGRE-интерфейсов, настройка NHRP и настройка BGP осуществляется аналогично устройству Spoke1-n1 с учетом соответствующих IP-адресов (см. Рисунок 1). В настройках BGP не нужно добавлять маршрут через VIP адрес. Настройки VRRP на Spoke2 не выполняются.

Из-за отсутствия маршрута по умолчанию до настройки multiwan, COC не будет загружен в базу Продукта во время настройки политики обработки COC (CRL).

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 Spoke2
```

2. Включите WAN и LAN интерфейсы и настройте IP адрес на WAN интерфейсе в соответствии со схемой стенда:

```
Spoke2(config)#interface TenGigabitEthernet0/0
Spoke2(config-if)# ip address 172.16.2.2 255.255.255.0
Spoke2(config-if)# no shutdown
Spoke2(config-if)#interface TenGigabitEthernet0/1
Spoke2(config-if)# ip address 172.17.2.2 255.255.255.0
Spoke2(config-if)# no shutdown
Spoke2(config-if)#interface TenGigabitEthernet0/2
Spoke2(config-if)# ip address 192.168.2.1 255.255.255.0
Spoke2(config-if)# no shutdown
Spoke2(config)#end
```

Примечание – маршрут по умолчанию будет добавлен с помощью сервиса multiwan.

Настройка шифрования

Настройка шифрования осуществляется аналогично устройству Spoke1-n1 с учетом соответствующих IP-адресов (см. Рисунок 1) и указанием двух WAN адресов в `object-group` с именем `SELF`:

```
Spoke2(config)#object-group network SELF
Spoke2(config-network-group)# description IP on WAN interface
Spoke2(config-network-group)# host 172.16.2.2
Spoke2(config-network-group)# host 172.17.2.2
Spoke2(config-network-group)#exit
```

Настройка multiwan

Настройка осуществляются из linux bash. Подробное описание параметров приведено в сценарии «Резервирование каналов связи при помощи пакета «multiwan»».

1. Приведите файл `/etc/multiwan/config.yaml` к виду:

```
config_format_version: 1
links_params:
  links:
  - name: ISP1
    iface_name: eth0
    prio: 300
    routes:
      ipv4:
        networks:
        - 0.0.0.0/0
        nexthop: 172.16.2.1
      watch_foreign_routes: true
    check:
      hosts_reachability: any
      ipv4:
        hosts:
        - 172.16.101.15
        - 172.16.1.1
      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: 172.17.2.1
      watch_foreign_routes: True
    check:
      hosts_reachability: any
      ipv4:
        hosts:
        - 172.16.101.15
        - 172.16.1.1
      method:
        icmp:
          max_rtt: 100
          interval: 3
          restore_interval: 1
          restore_count: 10
```

```
          failure_count: 10
          tos: 184
          data_size: 56
```

Завершение настройки Spoke2

1. После настройки запустите сервис `multiwan`, а также добавьте его в автозагрузку:

```
root@Spoke2:~# systemctl start multiwan.service
root@Spoke2:~# systemctl enable multiwan.service
```

2. Посмотреть лог работы сервиса можно в файле `/var/log/cspvpngate.log`:

```
root@Spoke2:~# tail -n 1000 /var/log/cspvpngate.log | grep multiwan
```

```
2023-12-04T15:10:14.853421+03:00 Spoke2 multiwan: INFO: Starting multiwan...
2023-12-04T15:10:15.908671+03:00 Spoke2 multiwan: INFO: NAT translations table has
been cleared.
2023-12-04T15:10:16.923643+03:00 Spoke2 multiwan: INFO: LSP was reloaded.
2023-12-04T15:10:16.930806+03:00 Spoke2 multiwan: INFO: Used interface "eth0".
Nexthop "172.16.100.1".
```

3. Проверьте, что добавлены маршруты объявлены в настройках `multiwan`:

```
root@Spoke2:~# ip route show proto multiwan
```

```
default via 172.16.2.1 dev eth0 src 172.16.2.2 metric 700
```

4. Проверьте что маршрут до IP адреса HTTP сервера CRL_distribution_point в системе:

```
root@Spoke2:~# ip route get 172.16.101.15
```

```
172.16.101.15 via 172.16.2.1 dev eth0 src 172.16.2.2 uid 0
    cache
```

5. Если всё настроено верно, то перезагрузите сервис `getcrls` для загрузки СОС в базу Продукта:

```
root@Spoke2:~# systemctl restart getcrls.service
```

6. Выполните проверку статуса сертификатов в базе Продукта:

```
root@Spoke2:~# cert_mgr check
```

```
1 State: Active    C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root
CA
2 State: Active    C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Spoke2
```

В Приложении представлены тексты конфигураций для криптошлюза Spoke2:

- текст консоли cisco-like;
- Содержимое `/etc/opennhrp/opennhrp.conf`;
- Содержимое `/etc/frr/frr.conf`;
- Содержимое `/etc/multiwan/config.yaml`

Проверка работоспособности стенда

Проверка работоспособности стенда проводится в трех состояниях:

1. Все узлы в центральном офисе и филиалах функционируют и исправны.
2. Проблема на основном криптошлюзе в центральном офисе: отказ центрального криптошлюза, отказ внешнего сетевого интерфейса.
3. Проблема на основной ноде кластера в филиале: отказ основной ноды кластера, недоступность Интернет-провайдера, недоступность сервиса opennhrp.

Все узлы в центральном офисе и филиалах функционируют и исправны

1. Убедитесь, что криптошлюз второго филиала работает через основной интерфейс:

```
root@Spoke2:~# 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", "172.16.1.1"
    IPv6 hosts: None
Link: 2
    Name: ISP2
    Interface: eth1
    Priority: 200
    State: reachable
    Active: False
    Hosts reachability: any
    IPv4 hosts: "172.16.101.15", "172.16.1.1"
    IPv6 hosts: None
```

   Видно, что активным интерфейсом является eth0.

2. Убедитесь, что на центральных криптошлюзах установлены IPsec-туннели со всеми филиальными криптошлюзами, для этого в linux bash выполните следующую команду:

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

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 2 (100.100.100.2,4500)-(172.16.1.2,4500) active 2232 2208
2 3 (100.100.100.2,4500)-(172.17.1.2,4500) active 1808 1848
3 4 (100.100.100.2,4500)-(172.16.2.2,4500) active 1908 1948

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 2 (100.100.100.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 168 140
2 3 (100.100.100.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 7340 7092
3 4 (100.100.100.2,*)-(172.17.1.2,*) 47 ESP nat-t-trans 876 656
4 5 (100.100.100.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 6108 5928
```

```
root@Hub2:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded
```

```
ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 1 (100.100.101.2,4500)-(172.16.1.2,4500) active 1808 1848
2 2 (100.100.101.2,4500)-(172.16.1.2,4500) active 1744 1848
3 3 (100.100.101.2,4500)-(172.17.1.2,4500) active 1808 1848
4 4 (100.100.101.2,4500)-(172.16.2.2,4500) active 1808 1848

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (100.100.101.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 1312 996
2 2 (100.100.101.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 7500 7240
3 3 (100.100.101.2,*)-(172.17.1.2,*) 47 ESP nat-t-trans 876 656
4 4 (100.100.101.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 6256 6076
```

```
root@Hub3:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 2 (100.101.100.2,4500)-(172.16.1.2,4500) active 1776 1752
2 3 (100.101.100.2,4500)-(172.16.1.2,4500) active 1744 1784
3 4 (100.101.100.2,4500)-(172.17.1.2,4500) active 1808 1848
4 5 (100.101.100.2,4500)-(172.16.2.2,4500) active 1908 1948

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 2 (100.101.100.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 0 7316
2 3 (100.101.100.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 7584 0
3 4 (100.101.100.2,*)-(172.17.1.2,*) 47 ESP nat-t-trans 876 656
4 5 (100.101.100.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 6136 6192
```

```
root@Hub4:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 3 (100.101.101.2,4500)-(172.16.1.2,4500) active 2200 2240
2 4 (100.101.101.2,4500)-(172.17.1.2,4500) active 1808 1848
3 5 (100.101.101.2,4500)-(172.16.2.2,4500) active 1808 1848

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 2 (100.101.101.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 548 348
2 3 (100.101.101.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 7268 7040
3 4 (100.101.101.2,*)-(172.17.1.2,*) 47 ESP nat-t-trans 876 656
4 5 (100.101.101.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 6476 6340
```

Видно, что на каждом центральном криптошлюзе установлено по три IPsec-туннеля – два с нодами кластера в первом филиале (IP-адреса 172.16.1.2, 172.17.1.2) и третий с криптошлюзом во втором филиале (IP-адрес: 172.16.2.2). Строка «`ESP nat-t-trans`» свидетельствует о том, что IPsec-туннель работает в транспортном режиме через NAT. Состояние, когда один IPsec-туннель отображается как два IPsec-туннеля (один для входящего трафика, другой для исходящего) является нормальным. Так же состояние, когда происходит дублирование записи IPsec-туннеля и счётчики меняются только у последней записи, является нормальным.

3. Убедитесь, что на центральных криптошлюзах протокол NHRP верно отображает информацию о филиальных криптошлюзах, для этого в linux bash выполните следующую команду:

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

```
Status: ok

Interface: mgre_1
Type: local
Protocol-Address: 10.10.10.255/32
Alias-Address: 10.10.10.100
Flags: up

Interface: mgre_1
Type: local
Protocol-Address: 10.10.10.100/32
Flags: up

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.10.2/32
NBMA-Address: 172.17.1.2
NBMA-NAT-OA-Address: 100.101.1.2
Flags: used up
Expires-In: 1:28

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.10.1/32
NBMA-Address: 172.16.1.2
NBMA-NAT-OA-Address: 100.100.1.2
Flags: up
Expires-In: 1:20

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.10.3/32
NBMA-Address: 172.16.2.2
Flags: up
Expires-In: 1:02
```

```
root@Hub2:~# opennhrpctl show
```

```
Status: ok

Interface: mgre_1
Type: local
Protocol-Address: 10.10.10.255/32
Alias-Address: 10.10.10.200
Flags: up

Interface: mgre_1
Type: local
Protocol-Address: 10.10.10.200/32
Flags: up

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.10.2/32
NBMA-Address: 172.17.1.2
NBMA-NAT-OA-Address: 100.101.1.2
Flags: used up
Expires-In: 1:27

Interface: mgre_1
Type: dynamic
```

```
Protocol-Address: 10.10.10.1/32
NBMA-Address: 172.16.1.2
NBMA-NAT-OA-Address: 100.100.1.2
Flags: up
Expires-In: 1:18

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.10.3/32
NBMA-Address: 172.16.2.2
Flags: up
Expires-In: 1:01
```

```
root@Hub3:~# opennhrpctl show
```

```
Status: ok

Interface: mgre_1
Type: local
Protocol-Address: 10.10.20.255/32
Alias-Address: 10.10.20.100
Flags: up

Interface: mgre_1
Type: local
Protocol-Address: 10.10.20.100/32
Flags: up

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.2/32
NBMA-Address: 172.17.1.2
NBMA-NAT-OA-Address: 100.101.1.2
Flags: up
Expires-In: 1:26

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.1/32
NBMA-Address: 172.16.1.2
NBMA-NAT-OA-Address: 100.100.1.2
Flags: up
Expires-In: 1:17

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.3/32
NBMA-Address: 172.16.2.2
Flags: up
Expires-In: 1:00
```

```
root@Hub4:~# opennhrpctl show
```

```
Status: ok

Interface: mgre_1
Type: local
Protocol-Address: 10.10.20.255/32
Alias-Address: 10.10.20.200
Flags: up
```

```
Interface: mgre_1
Type: local
Protocol-Address: 10.10.20.200/32
Flags: up

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.3/32
NBMA-Address: 172.16.2.2
Flags: up
Expires-In: 1:29

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.2/32
NBMA-Address: 172.17.1.2
NBMA-NAT-OA-Address: 100.101.1.2
Flags: used up
Expires-In: 1:24

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.1/32
NBMA-Address: 172.16.1.2
NBMA-NAT-OA-Address: 100.100.1.2
Flags: up
Expires-In: 1:15
```

Видно, что на каждом центральном криптошлюзе протокол NHRP корректно отображает информацию о филиальных криптошлюзах (записи с пометкой «`Type: dynamic`»), а именно mGRE-адрес (`Protocol-Address`) соответствует NBMA IP-адресу (`NBMA-Address`). Также правильно отображается информация об исходном IP-адресе до выполнения процедуры трансляции (`NBMA-NAT-OA-Address`) для тех криптошлюзов, которые находятся за устройствами, выполняющими функции NAT.

4. Убедитесь, что на центральных криптошлюзах подсистема ip neighbor ядра ОС Linux корректно отображает информацию о маппинге mGRE-адреса в NBMA-адрес филиальных криптошлюзов, для этого в linux bash выполните следующую команду:

```
root@Hub1:~# ip neighbor show
```

```
192.168.255.1 dev eth1 lladdr 00:50:56:9e:19:47 REACHABLE
10.10.10.2 dev mgre_1 lladdr 172.17.1.2 REACHABLE
10.10.10.1 dev mgre_1 lladdr 172.16.1.2 REACHABLE
100.100.100.1 dev eth0 lladdr 00:50:56:9e:47:c6 REACHABLE
10.10.10.3 dev mgre_1 lladdr 172.16.2.2 REACHABLE
```

```
root@Hub2:~# ip neighbor show
```

```
10.10.10.3 dev mgre_1 lladdr 172.16.2.2 REACHABLE
192.168.255.1 dev eth1 lladdr 00:50:56:9e:19:47 REACHABLE
100.100.101.1 dev eth0 lladdr 00:50:56:9e:11:d8 REACHABLE
10.10.10.1 dev mgre_1 lladdr 172.16.1.2 REACHABLE
10.10.10.2 dev mgre_1 lladdr 172.17.1.2 REACHABLE
```

```
root@Hub3:~# ip neighbor show
```

```
100.101.100.1 dev eth0 lladdr 00:50:56:9e:e0:5d REACHABLE
10.10.20.1 dev mgre_1 lladdr 172.16.1.2 REACHABLE
192.168.255.1 dev eth1 lladdr 00:50:56:9e:19:47 REACHABLE
10.10.20.3 dev mgre_1 lladdr 172.16.2.2 REACHABLE
10.10.20.2 dev mgre_1 lladdr 172.17.1.2 REACHABLE
```

```
root@Hub4:~# ip neighbor show
```

```
10.10.20.3 dev mgre_1 lladdr 172.16.2.2 REACHABLE
10.10.20.2 dev mgre_1 lladdr 172.17.1.2 REACHABLE
100.101.101.1 dev eth0 lladdr 00:50:56:9e:43:b9 REACHABLE
10.10.20.1 dev mgre_1 lladdr 172.16.1.2 REACHABLE
192.168.255.1 dev eth1 lladdr 00:50:56:9e:19:47 REACHABLE
```

Видно, что на каждом центральном криптошлюзе маппинг mGRE-адреса в NBMA-адрес филиальных криптошлюзов выполнен верно.

5. Убедитесь, что на центральных криптошлюзах корректно работает протокол BGP, для этого зайдите в linux bash выполните следующие команды:

```
root@Hub1:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Hub1# show ip bgp summary
```

```
IPv4 Unicast Summary (VRF default):
BGP router identifier 10.10.10.100, local AS number 65000 vrf-id 0
BGP table version 3
RIB entries 5, using 960 bytes of memory
Peers 3, using 61 KiB of memory
Peer groups 1, using 64 bytes of memory

Neighbor          V           AS    MsgRcvd    MsgSent    TblVer   InQ OutQ   Up/Down
State/PfxRcd   PfxSnt Desc
*10.10.10.1      4       65000        116        118         3     0     0 00:05:37
1        3 N/A
*10.10.10.3      4       65000        112        114         3     0     0 00:05:24
1        3 N/A
192.168.255.1    4       65001        131        131         3     0     0 00:06:15
1        3 N/A

Total number of neighbors 3
* - dynamic neighbor
2 dynamic neighbor(s), limit 5000
```

```
Hub1# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.10.1, mgre_1, weight 1, 00:05:48
B>* 192.168.2.0/24 [200/0] via 10.10.10.3, mgre_1, weight 1, 00:05:36
B>* 192.168.100.0/24 [20/0] via 192.168.255.1, eth1, weight 1, 00:06:27
```

```
root@Hub2:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Hub2# show ip bgp summary
```

```
IPv4 Unicast Summary (VRF default):
BGP router identifier 10.10.10.200, local AS number 65000 vrf-id 0
BGP table version 3
RIB entries 5, using 960 bytes of memory
Peers 3, using 61 KiB of memory
Peer groups 1, using 64 bytes of memory

Neighbor          V           AS    MsgRcvd    MsgSent    TblVer   InQ OutQ   Up/Down
State/PfxRcd   PfxSnt Desc
*10.10.10.1      4       65000        117        119         3     0     0 00:05:40
1        3 N/A
*10.10.10.3      4       65000        113        115         3     0     0 00:05:28
1        3 N/A
192.168.255.1    4       65001        130        130         3     0     0 00:06:15
1        3 N/A

Total number of neighbors 3
* - dynamic neighbor
2 dynamic neighbor(s), limit 5000
```

```
Hub2# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.10.1, mgre_1, weight 1, 00:05:53
B>* 192.168.2.0/24 [200/0] via 10.10.10.3, mgre_1, weight 1, 00:05:41
B>* 192.168.100.0/24 [20/0] via 192.168.255.1, eth1, weight 1, 00:06:28
```

```
root@Hub3:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Hub3# show ip bgp summary
```

```
IPv4 Unicast Summary (VRF default):
BGP router identifier 10.10.20.100, local AS number 65000 vrf-id 0
BGP table version 3
RIB entries 5, using 960 bytes of memory
Peers 3, using 61 KiB of memory
Peer groups 1, using 64 bytes of memory

Neighbor          V           AS    MsgRcvd    MsgSent    TblVer   InQ OutQ   Up/Down
State/PfxRcd   PfxSnt Desc
*10.10.20.1      4       65000        117        119         3     0     0 00:05:41
1        3 N/A
*10.10.20.3      4       65000        113        115         3     0     0 00:05:29
1        3 N/A
192.168.255.1    4       65001        130        130         3     0     0 00:06:14
1        3 N/A

Total number of neighbors 3
* - dynamic neighbor
2 dynamic neighbor(s), limit 5000
```

```
Hub3# show ip route bgp

Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.20.1, mgre_1, weight 1, 00:05:54
B>* 192.168.2.0/24 [200/0] via 10.10.20.3, mgre_1, weight 1, 00:05:42
B>* 192.168.100.0/24 [20/0] via 192.168.255.1, eth1, weight 1, 00:06:27
```

```
root@Hub4:~# vtysh

Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Hub4# show ip bgp summary

IPv4 Unicast Summary (VRF default):
BGP router identifier 10.10.20.200, local AS number 65000 vrf-id 0
BGP table version 3
RIB entries 5, using 960 bytes of memory
Peers 3, using 61 KiB of memory
Peer groups 1, using 64 bytes of memory

Neighbor          V           AS    MsgRcvd    MsgSent    TblVer   InQ OutQ   Up/Down
State/PfxRcd   PfxSnt Desc
*10.10.20.1      4       65000        118        120         3     0     0 00:05:43
1        3 N/A
*10.10.20.3      4       65000        114        116         3     0     0 00:05:30
1        3 N/A
192.168.255.1    4       65001        130        130         3     0     0 00:06:15
1        3 N/A

Total number of neighbors 3
* - dynamic neighbor
2 dynamic neighbor(s), limit 5000
```

```
Hub4# show ip route bgp

Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.20.1, mgre_1, weight 1, 00:05:55
B>* 192.168.2.0/24 [200/0] via 10.10.20.3, mgre_1, weight 1, 00:05:43
B>* 192.168.100.0/24 [20/0] via 192.168.255.1, eth1, weight 1, 00:06:28
```

Видно, что на каждом центральном криптошлюзе по протоколу BGP установлено соседство с основной нодой кластера в филиале №1, с криптошлюзом в филиале №2 и роутером Int-router-Hub. Соседство с резервной нодой кластера в филиале №1 отсутствует, как и требуется (нода, которая находится в состоянии BACKUP/FAULT не участвует в BGP).

Маршруты до филиальных подсетей (192.168.1.0/24 и 192.168.2.0/24) в таблице маршрутизации присутствуют, а также до 192.168.100.0/24.

6. Убедитесь, что основная нода кластера в филиале №1 находится в состоянии MASTER, для этого в linux bash выполните следующие команды:

```
root@Spoke1-n1:~# ip address show dev eth1
```

```
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether 00:50:56:9e:82:c4 brd ff:ff:ff:ff:ff:ff
    altname enp11s0
    altname ens192
    altname TenGigabitEthernet0_1
    inet 192.168.1.10/24 brd 192.168.1.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet 192.168.1.1/24 scope global secondary eth1:900
       valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:fe9e:82c4/64 scope link
       valid_lft forever preferred_lft forever
```

```
root@Spoke1-n1:~# tcpdump -i eth1 -n -c 4 vrrp
```

```
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), snapshot length 262144 bytes
09:44:27.771740 IP 192.168.1.10 > 224.0.0.18: VRRPv3, Advertisement, vrid 1, prio
100, intvl 300cs, length 12
09:44:30.771915 IP 192.168.1.10 > 224.0.0.18: VRRPv3, Advertisement, vrid 1, prio
100, intvl 300cs, length 12
09:44:33.772098 IP 192.168.1.10 > 224.0.0.18: VRRPv3, Advertisement, vrid 1, prio
100, intvl 300cs, length 12
09:44:36.772248 IP 192.168.1.10 > 224.0.0.18: VRRPv3, Advertisement, vrid 1, prio
100, intvl 300cs, length 12
4 packets captured
4 packets received by filter
0 packets dropped by kernel
```

```
root@Spoke1-n1:~# ip route show
```

```
default via 100.100.1.1 dev eth0 proto st_policy
10.10.10.0/24 dev mgre_1 proto kernel scope link src 10.10.10.1
10.10.20.0/24 dev mgre_2 proto kernel scope link src 10.10.20.1
100.100.1.0/24 dev eth0 proto kernel scope link src 100.100.1.2
192.168.1.0/24 dev eth1 proto kernel scope link src 192.168.1.10
192.168.2.0/24 nhid 65 via 10.10.10.3 dev mgre_1 proto bgp metric 20
192.168.100.0/24 nhid 125 via 10.10.10.100 dev mgre_1 proto bgp metric 20
```

Видно, что на внутреннем интерфейсе eth1 имеется VIP-адрес 192.168.1.1, а также с данного интерфейса (с IP-адреса 192.168.1.10) отсылаются VRRP пакеты. В таблице маршрутизации присутствуют маршруты до защищаемых подсетей центрального офиса и филиала №2.

Также в cisco-like консоли криптошлюза можно увидеть состояние VRRP:

```
Spoke1-n1#show vrrp
```

```
Interface                     VRID  State   AddrFamily
-----------------------------------------------------
TenGigabitEthernet0/1         1     Master  IPv4
```

7. Убедитесь, что на резервной ноде есть маршрут до 192.168.100.0/24 через VIP-адрес.

```
root@Spoke1-n2:~# ip route show
```

```
default via 100.101.1.1 dev eth0 proto st_policy
10.10.10.0/24 dev mgre_1 proto kernel scope link src 10.10.10.2
10.10.20.0/24 dev mgre_2 proto kernel scope link src 10.10.20.2
100.101.1.0/24 dev eth0 proto kernel scope link src 100.101.1.2
192.168.1.0/24 dev eth1 proto kernel scope link src 192.168.1.20
192.168.100.0/24 nhid 14 via 192.168.1.1 dev eth1 proto static metric 20
```

8. Убедитесь, что на внутреннем маршрутизаторе центрального офиса есть марштрты до защищищаемых подсетей филиалов через Hub1.

```
root@Int-router-Hub:~# ip route show
```

```
192.168.1.0/24 nhid 171 via 192.168.255.10 dev ens192 proto bgp metric 20
192.168.2.0/24 nhid 171 via 192.168.255.10 dev ens192 proto bgp metric 20
192.168.100.0/24 dev ens224 proto kernel scope link src 192.168.100.1
192.168.255.0/24 dev ens192 proto kernel scope link src 192.168.255.1
```

9. Убедитесь, что присутствует трафик между защищаемым подсетями филиала №1 и центрального офиса. Для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-hub и удостоверьтесь в прохождении пакетов:

```
root@Host-b-spoke1:~# ping 192.168.100.100 -c 4
```

```
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=61 time=2.05 ms
64 bytes from 192.168.100.100: icmp_seq=2 ttl=61 time=2.12 ms
64 bytes from 192.168.100.100: icmp_seq=3 ttl=61 time=2.05 ms
64 bytes from 192.168.100.100: icmp_seq=4 ttl=61 time=2.42 ms

--- 192.168.100.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 2.048/2.157/2.415/0.151 ms
```

10. Убедитесь, что защищенное соединение между основной нодой кластера в филиале №1 и криптошлюзом в филиале №2 успешно строится, для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-spoke2 и проверьте наличие IPsec-туннеля на основной ноде кластера в филиале №1:

```
root@Host-b-spoke1:~# ping 192.168.2.100 -c 4
```

```
PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.
64 bytes from 192.168.2.100: icmp_seq=1 ttl=61 time=2.81 ms
64 bytes from 192.168.2.100: icmp_seq=2 ttl=62 time=339 ms
64 bytes from 192.168.2.100: icmp_seq=3 ttl=62 time=2.27 ms
64 bytes from 192.168.2.100: icmp_seq=4 ttl=62 time=1.53 ms

--- 192.168.2.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.531/86.441/339.151/145.902 ms
```

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 5 (100.100.1.2,4500)-(172.16.100.2,4500) active 2208 2232
2 6 (100.100.1.2,4500)-(172.17.100.2,4500) active 1784 1744
3 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 2240 2200
4 8 (100.100.1.2,4500)-(172.16.101.2,4500) active 1848 1744
5 9 (100.100.1.2,4500)-(172.17.100.2,4500) active 1752 1776
6 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 1848 1816

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 5 (100.100.1.2,*)-(172.16.100.2,*) 47 ESP nat-t-trans 308 168
2 6 (100.100.1.2,*)-(172.16.100.2,*) 47 ESP nat-t-trans 16160 15732
3 7 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 348 380
4 8 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 15780 16220
5 9 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 0 16208
6 10 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 13692 0
```

```
7 11 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 15316 15744
8 12 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 288 288
```

Видно, что пакеты успешно дошли от Host-b-spoke1 до Host-b-spoke2, а также построился IPsec-туннель между основной нодой кластера в филиале №1 и криптошлюзом в филиале №2.

Проблема на криптошлюзах в центральном офисе

Смоделируем поочерёдный отказ внешнего сетевого интерфейса eth0 на криптошлюзах центрального офиса Hub1, Hub2 и Hub3 и проверим работоспособность стенда.

1. Переведите внешний сетевой интерфейс eth0 на криптошлюзе Hub1 в состояние DOWN, для этого в linux bash выполните следующую команду:

```
root@Hub1:~# ip link set dev eth0 down
```

2. Подождите некоторое время и убедитесь в том, что на основной ноде кластера в филиале №1 и на криптошлюзе в филиале №2 IPsec-туннель с Hub1 отсутствует. Маршруты в защищаемые подсети доступны через mGRE-интерфейс mgre_1. Маршрут до защищаемую подсеть центрального офиса доступна через mGRE-адрес Hub2. Для этого в linux bash выполните следующие команды:

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 6 (100.100.1.2,4500)-(172.17.100.2,4500) active 2568 2696
2 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 3024 3152
3 8 (100.100.1.2,4500)-(172.16.101.2,4500) active 2716 2528
4 9 (100.100.1.2,4500)-(172.17.100.2,4500) active 1752 1776
5 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 2324 2208

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 16 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 115120 117976
2 19 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 34700 35540
3 20 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 23360 32120
4 21 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 1004 1044
```

```
root@Spoke1-n1:~# vtysh
```

```

Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

```

```
Spoke1-n1# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.2.0/24 [200/0] via 10.10.10.3, mgre_1, weight 1, 00:01:44
B>* 192.168.100.0/24 [200/0] via 10.10.10.200, mgre_1, weight 1, 00:01:44
```

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

```
ISAKMP sessions: 1 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
```

```
1 1 (172.16.2.2,4500)-(172.17.101.2,4500) active 2800 2592
2 2 (172.16.2.2,4500)-(172.16.101.2,4500) active 2324 2200
3 3 (172.16.2.2,4500)-(172.17.100.2,4500) active 2424 2300
4 5 (172.16.2.2,4500)-(172.16.1.2,4500) active 2208 2324

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 8 (172.16.2.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 105760 79724
2 11 (172.16.2.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 45872 32096
3 12 (172.16.2.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 16624 22920
```

```
root@Spoke2:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke2# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.10.1, mgre_1, weight 1, 00:02:42
B>* 192.168.100.0/24 [200/0] via 10.10.10.200, mgre_1, weight 1, 00:02:42
```

3. Убедитесь, что ходит трафик между защищаемым подсетями филиала №1 и центрального офиса, для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-hub и удостоверьтесь в прохождении пакетов:

```
root@Host-b-spoke1:~# ping 192.168.100.100 -c 4
```

```
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=61 time=1.86 ms
64 bytes from 192.168.100.100: icmp_seq=2 ttl=61 time=1.91 ms
64 bytes from 192.168.100.100: icmp_seq=3 ttl=61 time=1.73 ms
64 bytes from 192.168.100.100: icmp_seq=4 ttl=61 time=1.98 ms

--- 192.168.100.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 1.734/1.870/1.975/0.088 ms
```

4. Убедитесь, что защищенное соединение между основной нодой кластера в филиале №1 и криптошлюзом в филиале №2 успешно строится, для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-spoke2 и проверьте наличие IPsec-туннеля на основной ноде кластера в филиале №1:

```
root@Host-b-spoke1:~# ping 192.168.2.100 -c 4
```

```
PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.
64 bytes from 192.168.2.100: icmp_seq=1 ttl=61 time=3.00 ms
64 bytes from 192.168.2.100: icmp_seq=2 ttl=62 time=1.99 ms
64 bytes from 192.168.2.100: icmp_seq=3 ttl=62 time=1.36 ms
64 bytes from 192.168.2.100: icmp_seq=4 ttl=62 time=1.96 ms

--- 192.168.2.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 1.355/2.075/2.999/0.590 ms
```

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 6 (100.100.1.2,4500)-(172.17.100.2,4500) active 2568 2696
2 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 3024 3152
3 8 (100.100.1.2,4500)-(172.16.101.2,4500) active 2800 2528
4 9 (100.100.1.2,4500)-(172.17.100.2,4500) active 1752 1776
5 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 2324 2208

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 17 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 288 288
2 19 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 42532 43572
3 20 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 30808 40152
4 21 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 9884 10236
```

Видно, что защищенное соединение между основной нодой кластера в филиале №1 и криптошлюзом в филиале №2 успешно построилось.

5. Переведите внешний сетевой интерфейс eth0 на криптошлюзе Hub2 в состояние DOWN, для этого в linux bash выполните следующую команду:

```
root@Hub2:~# ip link set dev eth0 down
```

6. Подождите некоторое время и убедитесь в том, что на основной ноде кластера в филиале №1 и на криптошлюзе в филиале №2 IPsec-туннель с Hub2 отсутствует. Маршруты в защищаемые подсети доступны через mGRE-интерфейс mgre_2. Маршрут до защищаемой подсети центрального офиса доступен через mGRE-адрес Hub3. Для этого в linux bash выполните следующие команды:

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 6 (100.100.1.2,4500)-(172.17.100.2,4500) active 2568 2696
2 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 3024 3152
3 9 (100.100.1.2,4500)-(172.17.100.2,4500) active 1752 1776
4 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 2324 2208

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 17 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 288 288
2 19 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 49064 50264
3 20 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 36764 46844
```

```
root@Spoke1-n1:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke1-n1# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.2.0/24 [200/0] via 10.10.20.3, mgre_2, weight 1, 00:04:31
```

```
B>* 192.168.100.0/24 [200/0] via 10.10.20.100, mgre_2, weight 1, 00:04:31
```

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

```
ISAKMP sessions: 1 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 1 (172.16.2.2,4500)-(172.17.101.2,4500) active 2800 2592
2 3 (172.16.2.2,4500)-(172.17.100.2,4500) active 2424 2300
3 5 (172.16.2.2,4500)-(172.16.1.2,4500) active 2208 2324
4 14 (172.16.2.2,500)-(172.16.100.2,500) incompleted 356 0

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 9 (172.16.2.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 288 288
2 11 (172.16.2.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 69400 50464
3 12 (172.16.2.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 37016 46344
```

```
root@Spoke2:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke2# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.20.1, mgre_2, weight 1, 00:05:45
B>* 192.168.100.0/24 [200/0] via 10.10.20.100, mgre_2, weight 1, 00:05:45
```

7. Убедитесь, что ходит трафик между защищаемым подсетями филиала №1 и центрального офиса, для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-hub и удостоверьтесь в прохождении пакетов:

```
root@Host-b-spoke1:~# ping 192.168.100.100 -c 4
```

```
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=61 time=1.82 ms
64 bytes from 192.168.100.100: icmp_seq=2 ttl=61 time=2.44 ms
64 bytes from 192.168.100.100: icmp_seq=3 ttl=61 time=2.05 ms
64 bytes from 192.168.100.100: icmp_seq=4 ttl=61 time=2.15 ms

--- 192.168.100.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 1.819/2.114/2.442/0.223 ms
```

8. Убедитесь, что защищенное соединение между основной нодой кластера в филиале №1 и криптошлюзом в филиале №2 функционирует исправно, для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-spoke2 и проверьте наличие IPsec-туннеля на основной ноде кластера в филиале №1:

```
root@Host-b-spoke1:~# ping 192.168.2.100 -c 4
```

```
PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.
64 bytes from 192.168.2.100: icmp_seq=1 ttl=62 time=2.13 ms
64 bytes from 192.168.2.100: icmp_seq=2 ttl=62 time=1.96 ms
64 bytes from 192.168.2.100: icmp_seq=3 ttl=62 time=2.01 ms
```

```
64 bytes from 192.168.2.100: icmp_seq=4 ttl=62 time=2.46 ms

--- 192.168.2.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.956/2.139/2.463/0.196 ms
```

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 1 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 6 (100.100.1.2,4500)-(172.17.100.2,4500) active 2568 2696
2 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 3024 3152
3 9 (100.100.1.2,4500)-(172.17.100.2,4500) active 1752 1776
4 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 2324 2208

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 17 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 768 768
2 19 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 72860 74648
3 20 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 61744 71268
```

Так как туннель был до проведения теста, то после отправки пакетов с помощью PING увеличились счётчики отправки и получения в туннеле.

9. Переведите внешний сетевой интерфейс eth0 на криптошлюзе Hub3 в состояние DOWN, для этого в linux bash выполните следующую команду:

```
root@Hub3:~# ip link set dev eth0 down
```

10. Подождите некоторое время и убедитесь в том, что на основной ноде кластера в филиале №1 и на криптошлюзе в филиале №2 IPsec-туннель с Hub3 отсутствует. Маршруты в защищаемые подсети доступны через mGRE-интерфейс mgre_2. Маршрут до защищаемую подсеть центрального офиса доступна через mGRE-адрес Hub4. Для этого в linux bash выполните следующие команды:

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 3024 3152
2 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 2324 2208

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 17 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 768 768
2 19 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 82900 85056
```

```
root@Spoke1-n1:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke1-n1# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure
```

```
B>* 192.168.2.0/24 [200/0] via 10.10.20.3, mgre_2, weight 1, 00:02:42
B>* 192.168.100.0/24 [200/0] via 10.10.20.200, mgre_2, weight 1, 00:02:42
```

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

```
ISAKMP sessions: 2 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 1 (172.16.2.2,4500)-(172.17.101.2,4500) active 2800 2592
2 5 (172.16.2.2,4500)-(172.16.1.2,4500) active 2208 2324

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 9 (172.16.2.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 768 768
2 12 (172.16.2.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 61148 71388
```

```
root@Spoke2:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke2# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.20.1, mgre_2, weight 1, 00:03:28
B>* 192.168.100.0/24 [200/0] via 10.10.20.200, mgre_2, weight 1, 00:03:28
```

11. Убедитесь, что ходит трафик между защищаемым подсетями филиала №1 и центрального офиса, для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-hub и удостоверьтесь в прохождении пакетов:

```
root@Host-b-spoke1:~# ping 192.168.100.100 -c 4
```

```
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=61 time=2.08 ms
64 bytes from 192.168.100.100: icmp_seq=2 ttl=61 time=2.29 ms
64 bytes from 192.168.100.100: icmp_seq=3 ttl=61 time=2.29 ms
64 bytes from 192.168.100.100: icmp_seq=4 ttl=61 time=50.0 ms

--- 192.168.100.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 2.079/14.176/50.043/20.707 ms
```

12. Убедитесь, что защищенное соединение между основной нодой кластера в филиале №1 и криптошлюзом в филиале №2 функционирует исправно, для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-spoke2 и проверьте наличие IPsec-туннеля на основной ноде кластера в филиале №1:

```
root@Host-b-spoke1:~# ping 192.168.2.100 -c 4
```

```
PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.
64 bytes from 192.168.2.100: icmp_seq=1 ttl=61 time=2.78 ms
64 bytes from 192.168.2.100: icmp_seq=2 ttl=62 time=1.55 ms
64 bytes from 192.168.2.100: icmp_seq=3 ttl=62 time=1.87 ms
64 bytes from 192.168.2.100: icmp_seq=4 ttl=62 time=1.66 ms
```

```
--- 192.168.2.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 1.548/1.964/2.781/0.485 ms
```

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 2 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 3024 3152
2 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 2324 2208
3 29 (100.100.1.2,500)-(172.16.100.2,500) incompleted 356 0
4 30 (100.100.1.2,500)-(172.17.100.2,500) incompleted 356 0

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 17 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 1056 1056
2 19 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 93884 96532
```

Так как туннель был до проведения теста, то после отправки пакетов с помощью PING увеличились счётчики отправки и получения в туннеле.

13. Переведите внешний сетевой интерфейс eth0 на криптошлюзе Hub3 в состояние UP, для этого в linux bash выполните следующую команду:

```
root@Hub3:~# ip link set dev eth0 up
```

14. Подождите некоторое время (до пяти минут) и убедитесь в том, что на основной ноде кластера в филиале №1 и на криптошлюзе в филиале №2 IPsec-туннель с Hub3 создан. Маршруты в защищаемые подсети доступны через mGRE-интерфейс mgre_2. Маршрут до защищаемой подсети центрального офиса доступнен через mGRE-адрес Hub3. Для этого в linux bash выполните следующие команды:

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 3416 3628
2 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 2324 2208
3 33 (100.100.1.2,4500)-(172.17.100.2,4500) active 1848 1808

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 17 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 1056 1056
2 22 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 4696 4908
3 23 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 3900 4020
```

```
root@Spoke1-n1:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke1-n1# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
```

```
       t - trapped, o - offload failure

B>* 192.168.2.0/24 [200/0] via 10.10.20.3, mgre_2, weight 1, 00:00:50
B>* 192.168.100.0/24 [200/0] via 10.10.20.100, mgre_2, weight 1, 00:01:29
```

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

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 1 (172.16.2.2,4500)-(172.17.101.2,4500) active 2800 2592
2 5 (172.16.2.2,4500)-(172.16.1.2,4500) active 2600 2716
3 26 (172.16.2.2,4500)-(172.17.100.2,4500) active 1848 1808

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 9 (172.16.2.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 1056 1056
2 12 (172.16.2.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 81356 92588
3 13 (172.16.2.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 5484 5664
4 14 (172.16.2.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 0 0
```

```
root@Spoke2:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke2# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.20.1, mgre_2, weight 1, 00:01:30
B>* 192.168.100.0/24 [200/0] via 10.10.20.100, mgre_2, weight 1, 00:01:30
```

15. Переведите внешний сетевой интерфейс eth0 на криптошлюзе Hub2 в состояние UP, для этого в linux bash выполните следующую команду:

```
root@Hub2:~# ip link set dev eth0 up
```

16. Подождите некоторое время (до пяти минут) и убедитесь в том, что на основной ноде кластера в филиале №1 и на криптошлюзе в филиале №2 IPsec-туннель с Hub2 создан. Маршруты в защищаемые подсети доступны через mGRE-интерфейс mgre_1. Маршрут до защищаемую подсеть центрального офиса доступна через mGRE-адрес Hub2. Для этого в linux bash выполните следующие команды:

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 1 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 3416 3628
2 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 2800 2600
3 33 (100.100.1.2,4500)-(172.17.100.2,4500) active 1848 1808
4 38 (100.100.1.2,4500)-(172.16.101.2,4500) active 1848 1808
5 39 (100.100.1.2,500)-(172.16.100.2,500) incompleted 356 0

IPsec connections:
```

```
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 22 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 16908 19252
2 23 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 17524 17944
3 24 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 288 288
4 25 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 2516 2720
```

```
root@Spoke1-n1:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke1-n1# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.2.0/24 [200/0] via 10.10.10.3, mgre_1, weight 1, 00:00:32
B>* 192.168.100.0/24 [200/0] via 10.10.10.200, mgre_1, weight 1, 00:00:32
```

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

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 1 (172.16.2.2,4500)-(172.17.101.2,4500) active 2800 2592
2 5 (172.16.2.2,4500)-(172.16.1.2,4500) active 2600 2800
3 26 (172.16.2.2,4500)-(172.17.100.2,4500) active 1848 1808
4 31 (172.16.2.2,4500)-(172.16.101.2,4500) active 1848 1808

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 12 (172.16.2.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 91492 104184
2 13 (172.16.2.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 17044 15100
3 14 (172.16.2.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 288 288
4 15 (172.16.2.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 5356 7308
```

```
root@Spoke2:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke2# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.10.1, mgre_1, weight 1, 00:00:50
B>* 192.168.100.0/24 [200/0] via 10.10.10.200, mgre_1, weight 1, 00:01:58
```

17. Переведите внешний сетевой интерфейс eth0 на криптошлюзе Hub1 в состояние UP, для этого в linux bash выполните следующую команду:

```
root@Hub1:~# ip link set dev eth0 up
```

18. Подождите некоторое время (до пяти минут) и убедитесь в том, что на основной ноде кластера в филиале №1 и на криптошлюзе в филиале №2 IPsec-туннель с Hub1 создан. Маршруты в защищаемые подсети доступны через mGRE-интерфейс mgre_1. Маршрут до защищаемую подсеть центрального офиса доступна через mGRE-адрес Hub1. Для этого в linux bash выполните следующие команды:

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 3416 3628
2 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 2800 2600
3 33 (100.100.1.2,4500)-(172.17.100.2,4500) active 1848 1808
4 38 (100.100.1.2,4500)-(172.16.101.2,4500) active 1848 1808
5 40 (100.100.1.2,4500)-(172.16.100.2,4500) active 1848 1808

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 22 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 25584 30704
2 23 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 28432 29092
3 24 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 288 288
4 25 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 13092 14416
5 26 (100.100.1.2,*)-(172.16.100.2,*) 47 ESP nat-t-trans 2444 2548
```

```
root@Spoke1-n1:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke1-n1# show ip route bgp
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.2.0/24 [200/0] via 10.10.10.3, mgre_1, weight 1, 00:04:03
B>* 192.168.100.0/24 [200/0] via 10.10.10.100, mgre_1, weight 1, 00:00:35
```

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

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 1 (172.16.2.2,4500)-(172.17.101.2,4500) active 3276 2984
2 5 (172.16.2.2,4500)-(172.16.1.2,4500) active 2600 2800
3 26 (172.16.2.2,4500)-(172.17.100.2,4500) active 1848 1808
4 31 (172.16.2.2,4500)-(172.16.101.2,4500) active 1848 1808
5 33 (172.16.2.2,4500)-(172.16.100.2,4500) active 1848 1808

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 13 (172.16.2.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 29696 23476
2 14 (172.16.2.2,*)-(172.16.1.2,*) 47 ESP nat-t-trans 288 288
3 15 (172.16.2.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 15124 21136
4 16 (172.16.2.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 8516 8956
5 17 (172.16.2.2,*)-(172.16.100.2,*) 47 ESP nat-t-trans 1472 1396
```

```
root@Spoke2:~# vtysh

Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

Spoke2# show ip route bgp
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.1.0/24 [200/0] via 10.10.10.1, mgre_1, weight 1, 00:00:10
B>* 192.168.100.0/24 [200/0] via 10.10.10.100, mgre_1, weight 1, 00:00:10
```

Видно, что в случае отказа внешнего сетевого интерфейса eth0 на криптошлюзах Hub1, Hub2 и Hub3 в центральном офисе динамическое построение защищенных соединений продолжает функционировать, защищаемая подсеть центрального офиса доступна с филиальных криптошлюзов.

При восстановлении внешнего сетевого интерфейса eth0 все также корректно отрабатывает. Проверку с полным отключением Hub1, Hub2, Hub3 повторите по аналогии.

Проблема на основной ноде кластера в филиале

Смоделируем полный отказ основной ноды кластера в филиале №1, а также отказ Интернет-провайдера и проверим работоспособность стенда.

1. Полный отказ основной ноды кластера в филиале №1.

   1.1. Выключите основную ноду, для этого в linux bash выполните следующую команду:

   ```
   root@Spoke1-n1:~# poweroff
   ```

   1.2. Убедитесь, что резервная нода перешла в состояние MASTER, для этого проверьте наличие VIP адреса на внутреннем интерфейсе и состояние VRRP в cisco-like консоли криптошлюза:

   ```
   root@Spoke1-n2:~# ip address show dev eth1
   3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
   qlen 1000
       link/ether 00:50:56:9e:96:28 brd ff:ff:ff:ff:ff:ff
       altname enp11s0
       altname ens192
       altname TenGigabitEthernet0_1
       inet 192.168.1.20/24 brd 192.168.1.255 scope global eth1
          valid_lft forever preferred_lft forever
       inet 192.168.1.1/24 scope global secondary eth1:900
          valid_lft forever preferred_lft forever
       inet6 fe80::250:56ff:fe9e:9628/64 scope link
          valid_lft forever preferred_lft forever
   ```

   ```
   Spoke1-n2#show vrrp
   Interface                    VRID  State   AddrFamily
   -----------------------------------------------------
   TenGigabitEthernet0/1        1     Master  IPv4
   ```

   Видно, что кластерный VIP-адрес (192.168.1.1) присутствует на внутреннем интерфейсе eth1 и VRRP находится в состоянии MASTER.

1.3. Убедитесь, что на резервной ноде по протоколу BGP установлено соседство с центральными криптошлюзами и получены маршруты в защищаемые подсети, для этого в linux bash выполните следующие команды:

```
root@Spoke1-n2:~# vtysh

Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
Spoke1-n2# show ip bgp summary

IPv4 Unicast Summary (VRF default):
BGP router identifier 192.168.1.20, local AS number 65000 vrf-id 0
BGP table version 3
RIB entries 5, using 960 bytes of memory
Peers 4, using 82 KiB of memory

Neighbor          V           AS    MsgRcvd    MsgSent    TblVer   InQ OutQ   Up/Down
State/PfxRcd   PfxSnt Desc
10.10.10.100     4       65000          34          30          3     0     0 00:01:19
2        1 N/A
10.10.10.200     4       65000          34          30          3     0     0 00:01:19
2        1 N/A
10.10.20.100     4       65000          34          30          3     0     0 00:01:19
2        1 N/A
10.10.20.200     4       65000          34          30          3     0     0 00:01:19
2        1 N/A

Total number of neighbors 4
```

```
Spoke1-n2# show ip route bgp

Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

B>* 192.168.2.0/24 [200/0] via 10.10.10.3, mgre_1, weight 1, 00:01:23
B>* 192.168.100.0/24 [200/0] via 10.10.10.100, mgre_1, weight 1, 00:01:23
```

1.4. Убедитесь, что защищенное соединение между резервной нодой кластера в филиале №1 и криптошлюзом в филиале №2 успешно строится, для этого запустите ping с устройства Host-b-spoke1 на устройство Host-b-spoke2 и проверьте наличие IPsec-туннеля на резервной ноде кластера в филиале №1:

```
root@Host-b-spoke1:~# ping 192.168.2.100 -c 4

PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.
64 bytes from 192.168.2.100: icmp_seq=1 ttl=61 time=3.37 ms
64 bytes from 192.168.2.100: icmp_seq=2 ttl=62 time=886 ms
64 bytes from 192.168.2.100: icmp_seq=3 ttl=62 time=2.04 ms
64 bytes from 192.168.2.100: icmp_seq=4 ttl=62 time=1.79 ms

--- 192.168.2.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.792/223.281/885.929/382.580 ms
```

```
root@Spoke1-n2:~# sa_mgr show

ISAKMP sessions: 0 initiated, 0 responded
```

```
ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 31 (100.101.1.2,4500)-(172.17.101.2,4500) active 1816 1840
2 32 (100.101.1.2,4500)-(172.16.100.2,4500) active 1816 1840
3 33 (100.101.1.2,4500)-(172.17.100.2,4500) active 1816 1840
4 34 (100.101.1.2,4500)-(172.16.101.2,4500) active 2208 2232
5 35 (100.101.1.2,4500)-(172.16.2.2,4500) active 1848 1816

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 16 (100.101.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 8488 8756
2 17 (100.101.1.2,*)-(172.16.100.2,*) 47 ESP nat-t-trans 8824 9168
3 18 (100.101.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 8564 7800
4 19 (100.101.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 232 148
5 20 (100.101.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 8392 8572
6 21 (100.101.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 288 288
```

1.5. Запустите основную ноду кластера, дождитесь ее полной загрузки и убедитесь, что она перешла в состояние BACKUP (основная нода не перехватит роль MASTER, так как в настройках VRRP стоит опция `nopreempt`), для этого проверьте отсутствие VIP адреса на внутреннем интерфейсе, и состояние VRRP в cisco-like консоли криптошлюза:

```
root@Spoke1-n1:~# ip address show dev eth1
```

```
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether 00:50:56:9e:82:c4 brd ff:ff:ff:ff:ff:ff
    altname enp11s0
    altname ens192
    altname TenGigabitEthernet0_1
    inet 192.168.1.10/24 brd 192.168.1.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:fe9e:82c4/64 scope link
       valid_lft forever preferred_lft forever
```

```
Spoke1-n1#show vrrp
```

```
Interface                      VRID  State   AddrFamily
-------------------------------------------------------
TenGigabitEthernet0/1          1     Backup  IPv4
```

Видно, что в случае полного отказа основной ноды кластера роль MASTER на себя берет резервная нода и успешно обрабатывает трафик.

2. Отказ Интернет-провайдера.

2.1. Чтобы смоделировать отказ Интернет-провайдера заблокируем ICMP пакеты от резервной ноды кластера (предполагается, что резервная нода находится в состоянии MASTER, а основная в BACKUP) на роутере Core_router, для этого в linux bash выполните следующую команду:

```
root@Core-router:~# iptables -A INPUT -s 172.17.1.2 -p icmp -j DROP
```

2.2. Убедитесь, что резервная нода кластера перешла в состояние FAULT, для этого состояние VRRP в cisco-like консоли криптошлюза:

```
Spoke1-n2#show vrrp
```

```
Interface                      VRID  State   AddrFamily
-------------------------------------------------------
TenGigabitEthernet0/1          1     Fault   IPv4
```

2.3. Убедитесь, что основная нода кластера перешла в состояние MASTER, для этого в linux bash выполните следующую команду:

```
Spoke1-n1#show vrrp
```

```
Interface                      VRID  State   AddrFamily
-------------------------------------------------------
TenGigabitEthernet0/1          1     Master  IPv4
```

2.4. Удалите на роутере Core_router правила, блокирующие ICMP добавленные ранее, для этого в linux bash выполните следующую команду:

```
root@Core-router:~# iptables -D INPUT -s 172.17.1.2 -p icmp -j DROP
```

2.5. Убедитесь, что резервная нода кластера перешла в состояние BACKUP, для этого в linux bash выполните следующую команду:

```
Spoke1-n2#show vrrp
```

```
Interface                      VRID  State   AddrFamily
-------------------------------------------------------
TenGigabitEthernet0/1          1     Backup  IPv4
```

При отказе Интернет-провайдера нода кластера переходит в состояние FAULT, а в случае восстановления в BACKUP.

Отказ интернет-провайдера второго филиала

1. Чтобы смоделировать отказ Интернет-провайдера заблокируем ICMP пакеты от основного интерфейса на роутере Core_router, для этого в linux bash выполните следующую команду:

```
root@Core-router:~# iptables -A INPUT -s 172.16.2.2 -p icmp -j DROP
```

2. Проверьте состояние линков:

```
root@Spoke2:~# 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", "172.16.1.1"
    IPv6 hosts: None
Link: 2
    Name: ISP2
    Interface: eth1
    Priority: 200
    State: reachable
    Active: True
    Hosts reachability: any
    IPv4 hosts: "172.16.101.15", "172.16.1.1"
    IPv6 hosts: None
```

Видно, что хосты стали недоступны через основной интерфейс(eth0) и активным стал второй интерфейс(eth1).

3. Убедитесь, что защищенное соединение между основной нодой кластера в филиале №1 и криптошлюзом в филиале №2 успешно строится, для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-spoke2 и проверьте наличие IPsec-туннеля на основной ноде кластера в филиале №1:

```
root@Host-b-spoke1:~# ping 192.168.2.100 -c 4
```

```
PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.
64 bytes from 192.168.2.100: icmp_seq=1 ttl=61 time=3.32 ms
64 bytes from 192.168.2.100: icmp_seq=2 ttl=62 time=121 ms
```

```
64 bytes from 192.168.2.100: icmp_seq=3 ttl=62 time=1.74 ms
64 bytes from 192.168.2.100: icmp_seq=4 ttl=62 time=2.10 ms

--- 192.168.2.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 1.739/32.100/121.247/51.471 ms
```

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 5 (100.100.1.2,4500)-(172.16.100.2,4500) active 1848 1808
2 6 (100.100.1.2,4500)-(172.16.101.2,4500) active 2240 2200
3 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 2240 2200
4 8 (100.100.1.2,4500)-(172.17.100.2,4500) active 2304 2264
5 9 (100.100.1.2,4500)-(172.17.2.2,4500) active 1848 1816

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 5 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 232 224
2 6 (100.100.1.2,*)-(172.16.100.2,*) 47 ESP nat-t-trans 50544 53788
3 7 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 232 224
4 8 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 0 44144
5 9 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 52280 0
6 10 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 52088 53404
7 11 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 51972 53308
8 12 (100.100.1.2,*)-(172.17.2.2,*) 47 ESP nat-t-trans 288 288
```

4. Проверьте правильность определения NBMA адреса криптошлюза филиала №2 в OpenNHRP на смежных ему устройствах:

```
root@Hub1:~# opennhrpctl show nbma 172.16.2.2
```

```
Status: ok
```

```
root@Hub1:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.10.3/32
NBMA-Address: 172.17.2.2
Flags: up
Expires-In: 1:10
```

```
root@Hub2:~# opennhrpctl show nbma 172.16.2.2
```

```
Status: ok
```

```
root@Hub2:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.10.3/32
NBMA-Address: 172.17.2.2
Flags: up
Expires-In: 1:25
```

```
root@Hub3:~# opennhrpctl show nbma 172.16.2.2
```

```
Status: ok
```

```
root@Hub3:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.3/32
NBMA-Address: 172.17.2.2
Flags: up
Expires-In: 1:17
```

```
root@Hub4:~# opennhrpctl show nbma 172.16.2.2
```

```
Status: ok
```

```
root@Hub4:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.3/32
NBMA-Address: 172.17.2.2
Flags: up
Expires-In: 1:28
```

```
root@Spoke1-n1:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok

Interface: mgre_1
Type: cached
Protocol-Address: 10.10.10.3/32
NBMA-Address: 172.17.2.2
Flags: used up
Expires-In: 1:14
```

Видно, что все устройства правильно определяют NBMA адрес криптошлюза филиала №2.

5. Удалите на роутере Core_router правила, блокирующие ICMP, добавленные ранее, для этого в linux bash выполните следующую команду:

```
root@Core-router:~# iptables -D INPUT -s 172.16.2.2 -p icmp -j DROP
```

6. Проверьте состояние линков:

```
root@Spoke2:~# 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", "172.16.1.1"
    IPv6 hosts: None
Link: 2
    Name: ISP2
```

```
    Interface: eth1
    Priority: 200
    State: reachable
    Active: False
    Hosts reachability: any
    IPv4 hosts: "172.16.101.15", "172.16.1.1"
    IPv6 hosts: None
```

Видно, что хосты стали доступны через основной интерфейс(eth0), и канал связи 1 (ISP1) стал активным.

7. Убедитесь, что защищенное соединение между основной нодой кластера в филиале №1 и криптошлюзом в филиале №2 успешно строится, для этого запустите `ping` с устройства Host-b-spoke1 на устройство Host-b-spoke2 и проверьте наличие IPsec-туннеля на основной ноде кластера в филиале №1:

```
root@Host-b-spoke1:~# ping 192.168.2.100 -c 4
```

```
PING 192.168.2.100 (192.168.2.100) 56(84) bytes of data.
64 bytes from 192.168.2.100: icmp_seq=1 ttl=61 time=3.07 ms
64 bytes from 192.168.2.100: icmp_seq=2 ttl=62 time=1027 ms
64 bytes from 192.168.2.100: icmp_seq=3 ttl=62 time=20.6 ms
64 bytes from 192.168.2.100: icmp_seq=4 ttl=62 time=2.00 ms

--- 192.168.2.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3009ms
rtt min/avg/max/mdev = 1.999/263.070/1026.629/440.902 ms, pipe 2
```

```
root@Spoke1-n1:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 5 (100.100.1.2,4500)-(172.16.100.2,4500) active 1848 1808
2 6 (100.100.1.2,4500)-(172.16.101.2,4500) active 2240 2200
3 7 (100.100.1.2,4500)-(172.17.101.2,4500) active 2240 2200
4 8 (100.100.1.2,4500)-(172.17.100.2,4500) active 2304 2264
5 9 (100.100.1.2,4500)-(172.17.2.2,4500) active 1848 1816
6 10 (100.100.1.2,4500)-(172.16.2.2,4500) active 1848 1816

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 5 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 232 224
2 6 (100.100.1.2,*)-(172.16.100.2,*) 47 ESP nat-t-trans 94704 95620
3 7 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 232 224
4 8 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 0 89952
5 9 (100.100.1.2,*)-(172.17.100.2,*) 47 ESP nat-t-trans 96968 0
6 10 (100.100.1.2,*)-(172.16.101.2,*) 47 ESP nat-t-trans 96968 99404
7 11 (100.100.1.2,*)-(172.17.101.2,*) 47 ESP nat-t-trans 96788 99340
8 12 (100.100.1.2,*)-(172.17.2.2,*) 47 ESP nat-t-trans 576 576
9 13 (100.100.1.2,*)-(172.16.2.2,*) 47 ESP nat-t-trans 288 288
```

8. Проверьте правильность определения NBMA адреса криптошлюза филиала №2 в OpenNHRP на смежных ему устройствах:

```
root@Hub1:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok
```

```
root@Hub1:~# opennhrpctl show nbma 172.16.2.2
```

```
Status: ok
```

```
Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.10.3/32
NBMA-Address: 172.16.2.2
Flags: up
Expires-In: 1:07
```

```
root@Hub2:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok
```

```
root@Hub2:~# opennhrpctl show nbma 172.16.2.2
```

```
Status: ok

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.10.3/32
NBMA-Address: 172.16.2.2
Flags: up
Expires-In: 1:26
```

```
root@Hub3:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok
```

```
root@Hub3:~# opennhrpctl show nbma 172.16.2.2
```

```
Status: ok

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.3/32
NBMA-Address: 172.16.2.2
Flags: up
Expires-In: 1:04
```

```
root@Hub4:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok
```

```
root@Hub4:~# opennhrpctl show nbma 172.16.2.2
```

```
Status: ok

Interface: mgre_1
Type: dynamic
Protocol-Address: 10.10.20.3/32
NBMA-Address: 172.16.2.2
Flags: up
Expires-In: 1:21
```

```
root@Spoke1-n1:~# opennhrpctl show nbma 172.17.2.2
```

```
Status: ok
```

```
root@Spoke1-n1:~# opennhrpctl show nbma 172.16.2.2
```

```
Status: ok
```

```
Interface: mgre_1
Type: cached
Protocol-Address: 10.10.10.3/32
NBMA-Address: 172.16.2.2
Flags: used up
Expires-In: 1:10
```

Видно, что все устройства правильно определяют NBMA адрес криптошлюза филиала №2. Дополнительно видно, что все криптошлюзы определяют, что криптошлюз филиала №2 не находится за NAT.

Дополнение

Балансировка нагрузки от филиальных криптошлюзов на центральные криптошлюзы

В данной главе будет описана настройка балансировки пакетов от филиальных криптошлюзов между криптошлюзами центрального офиса.

На устройстве Int-router-Hub должны отсутствовать route-map, регулирующие приоритет входящих маршрутов, полученных от Hub1, Hub2, Hub3 и Hub4. В таком случае в таблице маршрутизации Int-router-Hub будут присутствовать маршруты, полученные от всех Hub-ов.

На всех криптошлюзах центрального офиса требуется изменить настройки FRR. Далее показана настройка Hub1.

1. Войдите в консоль сервиса FRR, для этого нужно из linux bash набрать следующую команду:

```
root@Hub1:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

Hub1#
```

2. Требуется удалить route-map на Hub1, Hub2, Hub3 и Hub4. После этого криптошлюзы филиалов будут получать маршруты до защищаемых подсетей центрального офиса с одинаковым приоритетом:

```
Hub1# conf t
Hub1(config)# no route-map FROM-HUB1-OUT
Hub1(config)# router bgp 65000
Hub1(config-router)# address-family ipv4 unicast
Hub1(config-router-af)# no neighbor spokes-ibgp route-map FROM-HUB1-OUT out
Hub1(config-router-af)# end
```

3. Сохраните настройки FRR:

```
Hub1# write
```

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

После настройки на устройстве Int-router-Hub должно быть по четыре маршрута до защищаемых сетей филиалов, а на криптошлюзах филиалов будет четыре маршрута до защищаемой сети центрального офиса.

```
root@Int-router-Hub:~# ip route
```

```
192.168.1.0/24 nhid 14 proto bgp metric 20
        nexthop via 192.168.255.10 dev ens192 weight 1
        nexthop via 192.168.255.11 dev ens192 weight 1
        nexthop via 192.168.255.20 dev ens192 weight 1
        nexthop via 192.168.255.21 dev ens192 weight 1
192.168.2.0/24 nhid 14 proto bgp metric 20
        nexthop via 192.168.255.10 dev ens192 weight 1
        nexthop via 192.168.255.11 dev ens192 weight 1
        nexthop via 192.168.255.20 dev ens192 weight 1
        nexthop via 192.168.255.21 dev ens192 weight 1
192.168.100.0/24 dev ens224 proto kernel scope link src 192.168.100.1
192.168.255.0/24 dev ens192 proto kernel scope link src 192.168.255.1
```

```
root@Spoke1-n1:~# ip route
```

```
default via 100.100.1.1 dev eth0 proto 201
10.10.10.0/24 dev mgre_1 proto kernel scope link src 10.10.10.1
10.10.20.0/24 dev mgre_2 proto kernel scope link src 10.10.20.1
100.100.1.0/24 dev eth0 proto kernel scope link src 100.100.1.2
192.168.1.0/24 dev eth1 proto kernel scope link src 192.168.1.10
192.168.2.0/24 nhid 52 via 10.10.10.3 dev mgre_1 proto bgp metric 20
192.168.100.0/24 nhid 99 proto bgp metric 20
        nexthop via 10.10.10.100 dev mgre_1 weight 1
        nexthop via 10.10.20.100 dev mgre_2 weight 1
        nexthop via 10.10.20.200 dev mgre_2 weight 1
        nexthop via 10.10.10.200 dev mgre_1 weight 1
```

```
root@Spoke2:~# ip route
```

```
default via 172.16.2.1 dev eth0 proto 201
10.10.10.0/24 dev mgre_1 proto kernel scope link src 10.10.10.3
10.10.20.0/24 dev mgre_2 proto kernel scope link src 10.10.20.3
172.16.2.0/24 dev eth0 proto kernel scope link src 172.16.2.2
192.168.1.0/24 nhid 69 via 10.10.10.1 dev mgre_1 proto bgp metric 20
192.168.2.0/24 dev eth1 proto kernel scope link src 192.168.2.1
192.168.100.0/24 nhid 121 proto bgp metric 20
        nexthop via 10.10.10.100 dev mgre_1 weight 1
        nexthop via 10.10.20.100 dev mgre_2 weight 1
        nexthop via 10.10.20.200 dev mgre_2 weight 1
        nexthop via 10.10.10.200 dev mgre_1 weight 1
```

Совместимость «С-Терра Шлюз» 5.0 с «С-Терра Шлюз» 4.3

При использовании режима совместимости присутствуют следующие ограничения:
- В «С-Терра Шлюз» 4.3 не поддерживают наличие нескольких Hub в одной mGRE-подсети;
- В «С-Терра Шлюз» 4.3 должны работать только в транспортном режиме IPsec;
- В «С-Терра Шлюз» 4.3 нельзя указывать в списках доступа для шифрования только GRE трафик;
- Route-map для frr должны быть настроены одинаково.

Для совместной работы «С-Терра Шлюз» 5.0 с «С-Терра Шлюз» 4.3 требуется:

1. Установить на «С-Терра Шлюз» 5.0 значение параметра `NATTransport` равным 3 в файле `/opt/VPNagent/etc/agent.ini`.

   1.1. Пересчитать контрольную сумму файла, выполнив команду:

   ```
   root@Hub1:~# integr_mgr calc -f /opt/VPNagent/etc/agent.ini
   ```

   ```
   SUCCESS:  Operation was successful.
   ```

   1.2. Перезапустить сервис `vpngate`:

   ```
   root@Hub1:~# systemctl restart vpngate.service
   ```

2. Пример настройки IKE, списки доступа для трафика, комбинированный алгоритм шифрования и имитозащиты на базе ГОСТ 28147-89 и криптокарты приведён в настройках шифрования для криптошлюзов Hub1, Spoke1-n1 и Spoke2.

3. Проверьте режим работы IPsec туннеля в конфигурации cisco-like консоли криптошлюзов «С-Терра Шлюз» 4.3.

   Пример настройки IPsec туннеля в транспортном режиме:

   ```
   ...
   !
   crypto ipsec transform-set GOST esp-gost28147-4m-imit
   ```

```
 mode transport
!
...
```

Если отсутствует строка `mode transport`, то режим туннельный. Пример настройки IPsec туннеля в туннельном режиме:

```
...
!
crypto ipsec transform-set GOST esp-gost28147-4m-imit
!
...
```

Если IPsec туннель настроен в тунельном режиме, измените режим работы на транспортный:

```
Hub1(config)#crypto ipsec transform-set GOST esp-gost28147-4m-imit
Hub1(cfg-crypto-trans)# mode transport
Hub1(cfg-crypto-trans)#end
```

Ограничение полосы пропускания (shaping) на mGRE интерфейсе.

В данном разделе будет описано конфигурирование ограничения полосы пропускания до 100 Мбит/с суммарно по всем типам трафика, из них 10 на UDP, 85 на TCP и 5 на остальной трафик.

Подробнее об ограничениях при настройке полосы пропускания описано в инструкции «Обеспечение качества обслуживания для трафика (QoS)».

Для исключения срабатываний механизма защиты от атак воспроизведения на принимающем криптошлюзе при перегрузке полосы пропускания на передающем криптошлюзе рекомендуется отказаться от shaping на WAN-интерфейсе и выполнять его ДО зашифрования трафика, например, на внешнем оборудовании (LAN-сегмент) или mGRE-интерфейсе (подробнее в разделе «Ограничение полосы пропускания (shaping)» инструкции «Обеспечение качества обслуживания для трафика (QoS)»).

Примечание - Срабатывание механизма защиты от атак воспроизведения на приемной стороне можно диагностировать по увеличению счетчика потерь «ipsec replay drops» в утилите «kstat_show».

Для начала нужно маркировать UDP и TCP трафик. Настройка маркирования осуществляется в cisco-like консоли криптошлюза:

1. Создайте необходимые списки доступа, которые описывают маркируемый трафик:

```
Hub1#conf t
Hub1(config)#ip access-list extended SET_DSCP_AF11
Hub1(config-ext-nacl)#permit tcp 192.168.100.0 0.0.0.255 any
Hub1(config-ext-nacl)#ip access-list extended SET_DSCP_AF12
Hub1(config-ext-nacl)#permit udp 192.168.100.0 0.0.0.255 any
Hub1(config-ext-nacl)#ip access-list extended CLEAR_DSCP
Hub1(config-ext-nacl)#deny tcp 192.168.100.0 0.0.0.255 any
Hub1(config-ext-nacl)#deny udp 192.168.100.0 0.0.0.255 any
Hub1(config-ext-nacl)#permit ip any any
Hub1(config-ext-nacl)#exit
Hub1(config)#
```

2. Опишите классы трафика на основе списков доступа:

```
Hub1(config)#class-map match-all CLEAR_DSCP
Hub1(config-cmap)#match access-group name CLEAR_DSCP
Hub1(config-cmap)#class-map match-all SET_DSCP_AF11
Hub1(config-cmap)#match access-group name SET_DSCP_AF11
```

```
Hub1(config-cmap)#class-map match-all SET_DSCP_AF12
Hub1(config-cmap)#match access-group name SET_DSCP_AF12
Hub1(config-cmap)#exit
```

3. Опишите политику для трафика (выставление DSCP метки):

```
Hub1(config)#policy-map DSCP_POLICY
Hub1(config-pmap)#class SET_DSCP_AF11
Hub1(config-pmap-c)#set dscp af11
Hub1(config-pmap-c)#class SET_DSCP_AF12
Hub1(config-pmap-c)#set dscp af12
Hub1(config-pmap-c)#class CLEAR_DSCP
Hub1(config-pmap-c)#set dscp default
Hub1(config-pmap-c)#exit
Hub1(config-pmap)#exit
Hub1(config)#
```

4. Привяжите политику для трафика к mgre интерфейсу на исходящее направление:

```
Hub1(config)#interface Tunnel1
Hub1(config-if)#service-policy output DSCP_POLICY
Hub1(config-if)#end
```

Настройка ограничения полосы пропускания осуществляется из linux bash.

1. Добавьте следующее содержимое в файл `/etc/network/if-up.d/zshaper`:

Скрипты из `/etc/network/if-up.d/` выполняются в лексикографическом порядке. Скрипт должен выполняться позже `st-virtual-if`.

```
#!/bin/bash

function enable_shaper(){
# del root queue discipline
tc qdisc del dev $IFACE root || true

# add root queue discipline
tc qdisc add dev $IFACE root handle 1:0 htb default 4

# add root class
tc class add dev $IFACE parent 1:0 classid 1:1 htb rate 100mbit ceil 100mbit

# add sub classes
tc class add dev $IFACE parent 1:1 classid 1:2 htb rate 85mbit ceil 100mbit
tc class add dev $IFACE parent 1:1 classid 1:3 htb rate 10mbit ceil 100mbit
tc class add dev $IFACE parent 1:1 classid 1:4 htb rate 5mbit ceil 100mbit

# add filters for traffic
tc filter add dev $IFACE parent 1:0 protocol ip prio 0 u32 match ip tos 0x28 0xfc
flowid 1:2
tc filter add dev $IFACE parent 1:0 protocol ip prio 0 u32 match ip tos 0x30 0xfc
flowid 1:3
}

if [[ $IFACE == "mgre_1" ]]; then
    enable_shaper $IFACE
fi
```

Пояснения:

1.1. Удалить корневую дисциплину с интерфейса:

```
tc qdisc del dev $IFACE root || true
```

Примечание – При включении интерфейса с IPv6 адресом включается алгоритм Duplicate Address Detection (DAD), который обеспечивает уникальность IPv6 адреса в пределах сети. При попытке удаления корневой дисциплины (команда `tc qdisc del dev $IFACE root`) в период работы DAD возможно появление сообщений об ошибках следующего вида:

```
Error: Cannot delete qdisc with handle of zero.
```

Для безусловного выполнения других команд скрипта даже при выполнении команды `tc qdisc del dev $IFACE root` в период работы механизма DAD добавлены команды `|| true`.

1.2. Добавить подкласс 4 по умолчанию в корневую дисциплину интерфейса с классом HTB (трафик, который не попал под фильтры, направляется в этот класс):

```
tc qdisc add dev $IFACE root handle 1:0 htb default 4
```

1.3. Ограничить скорость на интерфейсе до 100 Мбит/с:

```
tc class add dev $IFACE parent 1:0 classid 1:1 htb rate 100mbit ceil 100mbit
```

1.4. Определить подклассы (1:2, 1:3, 1:4) и ограничить в них скорость:

```
tc class add dev $IFACE parent 1:1 classid 1:2 htb rate 85mbit ceil 100mbit
tc class add dev $IFACE parent 1:1 classid 1:3 htb rate 10mbit ceil 100mbit
tc class add dev $IFACE parent 1:1 classid 1:4 htb rate 5mbit ceil 100mbit
```

1.5. При помощи фильтров направить трафик в нужные классы. Пакеты с меткой AF11 (0x28) должны попадать в класс 1:2 (`flowid 1:2`). Пакеты с меткой AF12 (0x30) должны попадать в класс 1:3 (`flowid 1:3`):

Требуется указывать шестнадцатеричное значение ToS байта. Для двоичного значения DSCP добавляются два нуля справа и переводится полученное значение в шестнадцатеричное представление:

- двоичное представление для AF11 001010 -> 00101000 -> 0x28;
- двоичное представление для AF12 001100 -> 00110000 -> 0x30.

```
tc filter add dev $IFACE parent 1:0 protocol ip prio 0 u32 match ip tos 0x28 0xfc
flowid 1:2
tc filter add dev $IFACE parent 1:0 protocol ip prio 0 u32 match ip tos 0x30 0xfc
flowid 1:3
```

1.6. В данном сценарии shaping будет производиться на внутреннем интерфейсе `mgre_1` до зашифрования трафика.

2. Дайте файлу `/etc/network/if-up.d/zshaper` права на исполнение:

```
root@Hub1:~# chmod u+x /etc/network/if-up.d/zshaper
```

Примечание - Данный файл будет запускаться автоматически при старте системы во время конфигурирования интерфейса `mgre_1` (а также при выполнении команды `ifup mgre_1`).

3. Примените настройки ограничения полосы пропускания:

```
root@Hub1:~# ifdown mgre_1; ifup mgre_1
```

4. Посмотреть статистику можно при помощи команд:

```
root@Hub1:~# tc -s qdisc show dev mgre_1
root@Hub1:~# tc -s filter show dev mgre_1
root@Hub1:~# tc -s class show dev mgre_1
```

Примечание - Из-за ограничения полосы пропускания в `kstat_show` будут расти как счётчики `tx errors`, так и количество отброшенных пакетов (`dropped`) в `tc -s qdisc show dev mgre_1`. Допускается небольшое отклонение между количеством отброшенных пакетов и `tx error`.

Настройки для других криптошлюзов будут отличаться количеством интерфейсов и защищаемой подсетью.

Описание параметра holding-time

1. `holding-time` — время удержания в секундах, в течение которого информация о NHRP партнёре считается достоверной. Если отсутствует обновление информации от NHRP партнёра, то информация о NHRP партнёре удаляется из OpenNHRP. Значение `holding-time` по умолчанию – 7200 секунд. Рекомендуемое типовое значение `holding-time`, задаваемое в файле `/etc/opennhrp/opennhrp.conf` – 90 секунд. Допустимый диапазон значения `holding-time` – от 30 до 150 секунд.
2. `holding-time` определяет периоды выполнения служебных обменов NHRP с NHRP партнёрами DMVPN сети для подтверждения взаимной доступности. От `holding-time` на узлах DMVPN сети зависит скорость сходимости DMVPN сети и потребление CPU на Hub. В зависимости от фактических условий функционирования DMVPN сети может появиться потребность в выборе другого значения `holding-time`. При выборе значения `holding-time` следует учитывать следующее:

   2.1. Уменьшение `holding-time` позволяет сокращать время сходимости DMVPN сети, но приводит к увеличению нагрузки на Hub по следующим причинам:

   2.1.1 Увеличивается частота служебных обменов NHRP как при взаимодействии Spoke-to-Hub, так и при взаимодействии Spoke-to-Spoke, так как служебные обмены NHRP при Spoke-to-Spoke взаимодействии идут через Hub.

   2.1.2 Увеличивается частота кратковременной отправки через Hub пользовательского Spoke-to-Spoke трафика при некоторых характеристиках Spoke-to-Spoke трафика.

   При старте Spoke-to-Spoke взаимодействия до разрешения адреса партнёрского Spoke'а neighbor-запись указывает, что партнёрский Spoke доступен через Hub. Поэтому при новом обмене Spoke-to-Spoke трафиком он всегда кратковременно идет через Hub.

   После разрешения адреса партнёрского Spoke'а трафик идет по туннелю Spoke-to-Spoke до момента пока neighbour запись партнёрского Spoke'а действительна. При поступлении трафика от NHRP партнёра состояние neighbour записи обновляется системой. Таким трафиком может быть пользовательский Spoke-to-Spoke трафик или служебный NHRP трафик.

   Если трафик непрерывный и длительный, то neighbour запись поддерживается действительной пока трафик отправляется. Поэтому отправка пользовательского Spoke-to-Spoke трафика через Hub не происходит. После завершения пользовательского Spoke-to-Spoke трафика состояние neighbor-записи партнёрского Spoke'а будет переведено системой из состояния REACHABLE в состояние STALE и далее по списку состояний соседа (STALE, FAILED и так далее). Состояние neighbour записи обновляется системой циклически. Длительность каждого обновления состояния является случайной величиной от 15 до 45 секунд.

   Одновременно с системой состоянием neighbour записи NHRP партнёра управляет OpenNHRP. OpenNHRP проверяет состояние neighbour записи с периодом 2/3 `holding-time`. Если во время проверки состояние записи REACHABLE или состояние записи было не REACHABLE, но перешло в состояние REACHABLE из-за Spoke-to-Spoke трафика до окончания `holding-time`, то OpenNHRP с помощью служебных обменов производит проверку доступности NHRP партнёра. Если NHRP партнёр доступен, то обновляет состояние neighbour записи и начинает новый цикл отсчета `holding-time`.

   Если `holding-time` истек, то OpenNHRP переводит neighbour запись NHRP партнёра в состояние FAILED, то есть neighbour запись объявляется недействительной. Таким

образом, если произойдёт перерыв в отправке Spoke-to-Spoke трафике, то при уменьшении `holding-time` neighbour запись станет недействительной быстрее. Потребуется новый цикл разрешения адреса партнёрского Spoke'а с кратковременной отправкой Spoke-to-Spoke трафика через Hub. Если Spoke-to-Spoke трафик характеризуется длительностью менее 2/3 `holding-time` и периодом больше `holding-time`, то каждый такой цикл обмена пользовательским Spoke-to-Spoke трафиком будет приводить к кратковременной пересылке Spoke-to-Spoke трафика через Hub. Это приводит к импульсным, кратковременным нагрузкам на Hub'ы при Spoke-to-Spoke взаимодействии даже при существовании Spoke-to-Spoke IPsec туннелей.

2.1.3 Отправка служебных пакетов NHRP между Spoke'ами после прекращения пользовательского Spoke-to-Spoke трафика, если `holding-time` менее 70 секунд.

При уменьшении `holding-time` менее 70 секунд система может не успеть перевести neighbor-запись в состояние STALE. Тогда служебный обмен NHRP между Spoke'ами будет продолжаться еще некоторое время, пока на очередном цикле система не успеет перевести neighbour-запись в состояние STALE. Данное поведение частично сокращает частоту отправки пользовательского трафика Spoke-to-Spoke через Hub, описанную в предыдущем пункте.

При задании `holding-time` менее 23 секунд период отправки служебного NHRP трафика будет меньше, чем время изменения состояния neighbour записи. Таким образом служебный обмен NHRP будет продолжаться неограниченно долго до завершения работы OpenNHRP и neighbour запись будет оставаться действительной все это время.

2.2. Увеличение `holding-time` позволяет снизить нагрузку на CPU на Hub, но при этом увеличивается и время реконфигурации DMVPN сети.

Снижение нагрузки на CPU Hub'а обусловлено снижением количества служебных обменов NHRP и частоты отправки Spoke-to-Spoke трафика через Hub.

Увеличение времени реконфигурации DMVPN сети вызвано увеличением длительности выявления потери связи с NHRP партнёром, которое определяется значением `holding-time`. Период отправки служебных пакетов для регистрации NHRP-клиента на NHRP-сервере составляет 1/3 `holding-time`. При непоступлении ответа от NHRP-сервера NHRP-клиент дополнительно отправляет 6 пакетов с периодом 5 секунд. При значении `holding-time` 150 секунд время выявления неработоспособности NHRP сервера составляет до 80 секунд (1/3 `holding-time` плюс 6*5=30 секунд) по сравнению с 60 секундами при `holding-time` 90 секунд.

3. В криптошлюзах версии 4.3 нижняя граница ограничения `holding-time` (30 секунд) задавалась из угрозы переполнения логов при большом количестве (сотни) криптошлюзов, участвующих в DMVPN. Начиная с версии 5.0 предприняты меры для предотвращения переполнения логов - введена настройка уровня важности логируемых сообщений. По умолчанию логируются только предупреждения (`warning`) и ошибки (`error`). Возможные уровни логирования - `error`, `warning`, `info` и `debug`.

   На уровнях логирования `info` и `debug` объем логирования близок к объему логирования в версии 4.3 и включать их рекомендуется на ограниченное время.

   Для изменения уровня логирования введите следующую команду:

   ```
   root@Hub1:~# opennhrpctl log level debug
   ```

   ```
   Status: ok
   ```

   Для изменения уровня логирования со старта сервиса OpenNHRP добавьте дополнительные параметры в `/etc/default/opennhrp`:

```
ADDITIONAL_PARAMS="--log-level debug"
```

Рекомендации по настройке больших DMVPN сетей

Общие настройки

Настройка возможности подключения к Hub'ам большого количества Spoke'ов включает в себя три этапа:

1. Проверьте значение параметра ядра `net.ipv4.neigh.default.gc_thresh3`:

   ```
   root@Hub1:~# cat /etc/sysctl.d/01-sterra.conf | grep gc_thresh3
   ```

   ```
   net.ipv4.neigh.default.gc_thresh3 = 2048
   net.ipv6.neigh.default.gc_thresh3 = 2048
   ```

   Примечание - Параметр ядра `net.ipv4.neigh.default.gc_thresh3` отвечает за предельное количество соседей, при достижении которого система начинает очистку таблицы соседей (neighbours). Значение параметра `net.ipv4.neigh.default.gc_thresh3` должно быть равно удвоенному количеству максимально рекомендуемого количества Spoke'ов плюс некоторый запас. Например, при количестве Spoke'ов равному 1000 рекомендуется задать значение `net.ipv4.neigh.default.gc_thresh3` равным 2048 (задано по умолчанию). Если необходимо изменить эту настройку, после изменения параметра следует выполнить команду:

   ```
   root@Hub1:~# sysctl -p /etc/sysctl.d/01-sterra.conf
   ```

   ```
   ...
   net.ipv4.neigh.default.gc_thresh3 = 2048
   net.ipv6.neigh.default.gc_thresh3 = 2048
   ```

2. Настройте на криптошлюзах предельное количество соседей для используемого протокола маршрутизации. Для выполнения настройки следует раскомментировать и задать в файле `/etc/frr/daemons` параметр `MAX_FDS`:

   ```
   root@Hub1:~# cat /etc/frr/daemons
   ```

   ```
   ...
   # This is the maximum number of FD's that will be available.  Upon startup this
   # is read by the control files and ulimit is called.  Uncomment and use a
   # reasonable value for your setup if you are expecting a large number of peers
   # in say BGP.
   #
   MAX_FDS=2048
   ...
   ```

   Примечания:

   1. Значение `MAX_FDS` задано равным удвоенному количеству предельного числа соседей протокола BGP (2048). При таком количестве, остаётся значительный запас для добавления новых BGP соседей.
   2. Данная настройка выполняется для разрешения демону BGP использовать необходимое количество файловых дескрипторов, к которым относятся и сокеты, и обеспечения возможности взаимодействия Hub'ов с большим количеством Spoke'ов по протоколу BGP.
   3. По умолчанию демон BGP имеет ограничение - не более 1024 открытых дескрипторов (#MAX_FDS=1024), часть из которых используется демоном для внутренних задач (IPC). Рекомендуемая предельно допустимая для демона iBGP граница – удвоенное предельное количество соседей по протоколу BGP. Это значение с запасом. При необходимости запас может быть уменьшен. Не рекомендуется снижать запас меньше 256 файловых дескрипторов.

4. Признаком недостаточного количества разрешенных к использованию файловых дескрипторов для демона BGP является появление в логах (`/var/log/error.log`, `journalctl`) сообщений вида:

```
Oct 31 12:51:28 Hub1 bgpd[9287]: [RZYGD-06VQ1][EC 100663299] [Error] BGP socket
accept failed (Too many open files); retrying

Oct 31 12:52:28 Hub1 bgpd[9287]: [ZM9CM-ZXAVM][EC 100663299] can't accept vty socket
: Too many open files
```

Для применения настройки следует перезапустить демон `frr.service` командой:

```
root@Hub1:~# systemctl restart frr.service
```

3. Настройте размер буфера OpenNHRP для приема сообщений от системы об изменении состояния соседей (neighbours) по Netlink. Для выполнения настройки следует задать в файле `opennhrp.conf` параметр `netlink-neigh-buff-size`. По умолчанию размер буфера 256 Кбайт, что соответствует приблизительно 500 соседям. Для настройки выполните следующие действия:

3.1. Определите требуемую величину буфера `netlink-neigh-buff-size`. Зависимость размера буфера (параметр `netlink-neigh-buff-size`) от количества соседей приведена в `man opennhrp.conf`. Согласно `man opennhrp.conf` для 1000 Spoke'ов требуется размер `netlink-neigh-buff-size=512kb`.

3.2. Проверьте значение параметра ядра `net.core.rmem_max`:

```
root@Hub1:~# sysctl net.core.rmem_max
```

```
net.core.rmem_max = 4194304
```

Примечание - Параметр ядра `net.core.rmem_max` отвечает за предельный размер буфера Netlink. Параметр `net.core.rmem_max` должен быть равен или больше требуемому размеру буфера `netlink-neigh-buff-size` OpenNHRP. По умолчанию настроено 4 Мбайта, что достаточно для подключения 1000 Spoke'ов. Изменение значения параметра `net.core.rmem_max` может быть выполнено в соответствии с описанием в примечании к пункту 1 настоящего раздела.

3.3. Задайте значение `netlink-neigh-buff-size` в файле `/etc/opennhrp/opennhrp.conf`:

```
root@hub2:~# cat /etc/opennhrp/opennhrp.conf
```

```
interface mgre_1
 multicast dynamic
 holding-time 90
 cisco-authentication secret
 non-caching

netlink-neigh-buff-size 524288
```

3.4. Перезапустите OpenNHRP командой:

```
root@Hub1:~# systemctl restart opennhrp.service
```

Особые случаи поведения

В процессе работы возможно появление следующих сообщений об ошибках, не препятствующих функционированию DMVPN (phase 2):

1. Сообщение от `opennhrp.service` о недостаточном размере буфера для приема сообщений от подсистемы Netlink:

```
Aug 15 03:16:58 HUB2 opennhrp[53532]: Netlink overrun: No buffer space available
```

Настройка размера буфера `netlink-neigh-buff-size` описана в пункте 3 предыдущего раздела.

2. Сообщение о включении механизма SYN Cookie для сетевого порта протокола iBGP:

```
[    51.199558] TCP: request_sock_TCP: Possible SYN flooding on port 179. Sending
cookies.  Check SNMP counters.
```

Данное сообщение может появляться в системном журнале ядра (`dmesg` или `/var/log/kern.log`) после запуска сервиса `frr.service` при высокой интенсивности подключения клиентов iBGP к Hub'ам (при большом количестве Spoke'ов и малом значении `holding-time`). Появление данного сообщения не препятствует функционированию DMVPN (phase 2) и не замедляет его работу.

3. Сообщение о переполнении таблицы соседей (neighbour):

```
Jul 31 13:03:00 HUB2 kernel: [257888.071391] neighbour: arp_cache: neighbor table
overflow!
```

Данное сообщение может появляться в системном журнале ядра (`dmesg` или `/var/log/kern.log`) в случае, если фактическое количество соседей превышает предельные значения в настройках системы. В случае появления данного сообщения необходимо проверить не превышает ли фактическое количество соседей (neighbour) криптошлюза предельного значения 2048, соответствующего 1000 Spoke'ов.

4. Сообщение о недостатке разрешенных к использованию файловых дескрипторов в демоне BGP:

```
Oct 31 12:51:28 Hub1 bgpd[9287]: [RZYGD-06VQ1][EC 100663299] [Error] BGP socket
accept failed (Too many open files); retrying

Oct 31 12:52:28 Hub1 bgpd[9287]: [ZM9CM-ZXAVM][EC 100663299] can't accept vty socket
: Too many open files
```

Данное сообщение может появляться в системных журнала `/var/log/error.log` и `journalctl` в случае, если фактическое количество соседей по протоколу BGP превышает предельные значения в настройках FRR. Для предотвращения появления данных сообщений и устранения препятствий для взаимодействия Hub'ов и Spoke'ов по протоколу BGP следует выполнить настройку параметра `MAX_FDS` в соответствии с указаниями в общей части настоящего приложения.

Приложение

Устройство Core-router

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
# To Ext-router-Hub1
auto ens192
iface ens192 inet static
address 172.16.100.1
netmask 255.255.255.0
```

```
# To Ext-router-Hub2
auto ens224
iface ens224 inet static
address 172.16.101.1
netmask 255.255.255.0

# To Ext-router-Hub3
auto ens256
iface ens256 inet static
address 172.17.100.1
netmask 255.255.255.0

# To Ext-router-Hub4
auto ens161
iface ens161 inet static
address 172.17.101.1
netmask 255.255.255.0

# To Spoke2-eth0
auto ens193
iface ens193 inet static
address 172.16.2.1
netmask 255.255.255.0

# To Spoke1-n2
auto ens225
iface ens225 inet static
address 172.17.1.1
netmask 255.255.255.0

# To Spoke1-n1
auto ens257
iface ens257 inet static
address 172.16.1.1
netmask 255.255.255.0

# To Spoke2-eth1
auto ens162
iface ens162 inet static
address 172.17.2.1
netmask 255.255.255.0

auto ens194
iface ens194 inet dhcp

# For CDP.
auto ens256:0
iface ens256:0 inet static
address 172.16.101.15
netmask 255.255.255.0
```

Устройство Ext-router-Hub1

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
```

```
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
# To Core-router
auto ens192
iface ens192 inet static
address 172.16.100.2
netmask 255.255.255.0
gateway 172.16.100.1
pre-up /etc/iptables.rules

# To Hub1
auto ens224
iface ens224 inet static
address 100.100.100.1
netmask 255.255.255.0
```

Настройки iptables

Содержимое файла `/etc/iptables.rules`:

```
EXTIF="ens192"
INTIF="ens224"
LAN_RANGE="100.100.100.0/24"
INET_IP1="172.16.100.2"
IPSECGW="100.100.100.2"

# Flush previous rules, delete chains and reset counters
iptables -F -t filter
iptables -F -t nat
iptables -F -t mangle
iptables -F

# Default policies
iptables -P INPUT   DROP
iptables -P OUTPUT  DROP
iptables -P FORWARD DROP

# Enable loopback traffic
iptables -A INPUT  -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Enable statefull rules (after that, only need to allow NEW conections)
iptables -A INPUT   -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT  -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -i $EXTIF -o $INTIF -m conntrack --ctstate DNAT -j ACCEPT

# Drop invalid state packets
iptables -A INPUT   -m conntrack --ctstate INVALID -j DROP
iptables -A OUTPUT  -m conntrack --ctstate INVALID -j DROP
iptables -A FORWARD -m conntrack --ctstate INVALID -j DROP


### INPUT ###
# SSH
iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
```

```
# ICMP
iptables -A INPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### OUTPUT ###
# SSH
iptables -A OUTPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A OUTPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### FORWARD ###
iptables -A FORWARD -i $INTIF -o $EXTIF -m conntrack --ctstate NEW -j ACCEPT

### SNAT ###
iptables -t nat -A POSTROUTING -o $EXTIF -j SNAT --to-source $INET_IP1

### DNAT ###
# IKE/500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 500 -j DNAT --to-
destination $IPSECGW:500
# NAT-T/4500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 4500 -j DNAT --to-
destination $IPSECGW:4500

### LOGGING ###
iptables -A INPUT    -j LOG --log-level debug --log-prefix '[FW INPUT]:    '
iptables -A OUTPUT   -j LOG --log-level debug --log-prefix '[FW OUTPUT]:   '
iptables -A FORWARD  -j LOG --log-level debug --log-prefix '[FW FORWARD]:  '
```

Устройство Ext-router-Hub2

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
# To Core-router
auto ens192
iface ens192 inet static
address 172.16.101.2
netmask 255.255.255.0
gateway 172.16.101.1
pre-up /etc/iptables.rules

# To Hub2
auto ens224
iface ens224 inet static
address 100.100.101.1
netmask 255.255.255.0
```

Настройки iptables

Содержимое файла `/etc/iptables.rules`:

```
EXTIF="ens192"
INTIF="ens224"
LAN_RANGE="100.100.100.0/24"
INET_IP1="172.16.101.2"
IPSECGW="100.100.101.2"

# Flush previous rules, delete chains and reset counters
iptables -F -t filter
iptables -F -t nat
iptables -F -t mangle
iptables -F

# Default policies
iptables -P INPUT   DROP
iptables -P OUTPUT  DROP
iptables -P FORWARD DROP

# Enable loopback traffic
iptables -A INPUT  -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Enable statefull rules (after that, only need to allow NEW conections)
iptables -A INPUT   -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT  -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -i $EXTIF -o $INTIF -m conntrack --ctstate DNAT -j ACCEPT

# Drop invalid state packets
iptables -A INPUT   -m conntrack --ctstate INVALID -j DROP
iptables -A OUTPUT  -m conntrack --ctstate INVALID -j DROP
iptables -A FORWARD -m conntrack --ctstate INVALID -j DROP


### INPUT ###
# SSH
iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A INPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### OUTPUT ###
# SSH
iptables -A OUTPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A OUTPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### FORWARD ###
iptables -A FORWARD -i $INTIF -o $EXTIF -m conntrack --ctstate NEW -j ACCEPT

### SNAT ###
iptables -t nat -A POSTROUTING -o $EXTIF -j SNAT --to-source $INET_IP1

### DNAT ###
# IKE/500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 500 -j DNAT --to-
destination $IPSECGW:500
# NAT-T/4500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 4500 -j DNAT --to-
destination $IPSECGW:4500

### LOGGING ###
iptables -A INPUT    -j LOG --log-level debug --log-prefix '[FW INPUT]:    '
```

```
iptables -A OUTPUT   -j LOG --log-level debug --log-prefix '[FW OUTPUT]:   '
iptables -A FORWARD  -j LOG --log-level debug --log-prefix '[FW FORWARD]:  '
```

Устройство Ext-router-Hub3

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
# To Core-router
auto ens192
iface ens192 inet static
address 172.17.100.2
netmask 255.255.255.0
gateway 172.17.100.1
pre-up /etc/iptables.rules

# To Hub3
auto ens224
iface ens224 inet static
address 100.101.100.1
netmask 255.255.255.0
```

Настройки iptables

Содержимое файла `/etc/iptables.rules`:

```
EXTIF="ens192"
INTIF="ens224"
LAN_RANGE="100.100.200.0/24"
INET_IP1="172.17.100.2"
IPSECGW="100.101.100.2"

# Flush previous rules, delete chains and reset counters
iptables -F -t filter
iptables -F -t nat
iptables -F -t mangle
iptables -F

# Default policies
iptables -P INPUT   DROP
iptables -P OUTPUT  DROP
iptables -P FORWARD DROP

# Enable loopback traffic
iptables -A INPUT  -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Enable statefull rules (after that, only need to allow NEW conections)
iptables -A INPUT   -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT  -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
```

```
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -i $EXTIF -o $INTIF -m conntrack --ctstate DNAT -j ACCEPT

# Drop invalid state packets
iptables -A INPUT   -m conntrack --ctstate INVALID -j DROP
iptables -A OUTPUT  -m conntrack --ctstate INVALID -j DROP
iptables -A FORWARD -m conntrack --ctstate INVALID -j DROP


### INPUT ###
# SSH
iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A INPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### OUTPUT ###
# SSH
iptables -A OUTPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A OUTPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### FORWARD ###
iptables -A FORWARD -i $INTIF -o $EXTIF -m conntrack --ctstate NEW -j ACCEPT

### SNAT ###
iptables -t nat -A POSTROUTING -o $EXTIF -j SNAT --to-source $INET_IP1

### DNAT ###
# IKE/500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 500 -j DNAT --to-
destination $IPSECGW:500
# NAT-T/4500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 4500 -j DNAT --to-
destination $IPSECGW:4500

### LOGGING ###
iptables -A INPUT    -j LOG --log-level debug --log-prefix '[FW INPUT]:    '
iptables -A OUTPUT   -j LOG --log-level debug --log-prefix '[FW OUTPUT]:   '
iptables -A FORWARD  -j LOG --log-level debug --log-prefix '[FW FORWARD]:  '
```

Устройство Ext-router-Hub4

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
# To Core-router
auto ens192
iface ens192 inet static
address 172.17.101.2
netmask 255.255.255.0
```

```
gateway 172.17.101.1
pre-up /etc/iptables.rules

# To Hub4
auto ens224
iface ens224 inet static
address 100.101.101.1
netmask 255.255.255.0
```

Настройки iptables

Содержимое файла `/etc/iptables.rules`:

```
EXTIF="ens192"
INTIF="ens224"
LAN_RANGE="100.100.200.0/24"
INET_IP1="172.17.101.2"
IPSECGW="100.101.101.2"

# Flush previous rules, delete chains and reset counters
iptables -F -t filter
iptables -F -t nat
iptables -F -t mangle
iptables -F

# Default policies
iptables -P INPUT   DROP
iptables -P OUTPUT  DROP
iptables -P FORWARD DROP

# Enable loopback traffic
iptables -A INPUT  -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Enable statefull rules (after that, only need to allow NEW conections)
iptables -A INPUT   -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT  -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -i $EXTIF -o $INTIF -m conntrack --ctstate DNAT -j ACCEPT

# Drop invalid state packets
iptables -A INPUT   -m conntrack --ctstate INVALID -j DROP
iptables -A OUTPUT  -m conntrack --ctstate INVALID -j DROP
iptables -A FORWARD -m conntrack --ctstate INVALID -j DROP


### INPUT ###
# SSH
iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A INPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### OUTPUT ###
# SSH
iptables -A OUTPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A OUTPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### FORWARD ###
iptables -A FORWARD -i $INTIF -o $EXTIF -m conntrack --ctstate NEW -j ACCEPT
```

```
### SNAT ###
iptables -t nat -A POSTROUTING -o $EXTIF -j SNAT --to-source $INET_IP1

### DNAT ###
# IKE/500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 500 -j DNAT --to-
destination $IPSECGW:500
# NAT-T/4500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 4500 -j DNAT --to-
destination $IPSECGW:4500

### LOGGING ###
iptables -A INPUT    -j LOG --log-level debug --log-prefix '[FW INPUT]:    '
iptables -A OUTPUT   -j LOG --log-level debug --log-prefix '[FW OUTPUT]:   '
iptables -A FORWARD  -j LOG --log-level debug --log-prefix '[FW FORWARD]:  '
```

Устройство Ext-router1-Spoke1

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
# To Core-router
auto ens192
iface ens192 inet static
address 172.16.1.2
netmask 255.255.255.0
gateway 172.16.1.1
pre-up /etc/iptables.rules

# To Spoke1-n1
auto ens224
iface ens224 inet static
address 100.100.1.1
netmask 255.255.255.0
```

Настройки iptables

Содержимое файла `/etc/iptables.rules`:

```
EXTIF="ens192"
INTIF="ens224"
LAN_RANGE="100.100.1.0/24"
INET_IP1="172.16.1.2"
IPSECGW="100.100.1.2"

# Flush previous rules, delete chains and reset counters
iptables -F -t filter
iptables -F -t nat
iptables -F -t mangle
iptables -F
```

```
# Default policies
iptables -P INPUT   DROP
iptables -P OUTPUT  DROP
iptables -P FORWARD DROP

# Enable loopback traffic
iptables -A INPUT  -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Enable statefull rules (after that, only need to allow NEW conections)
iptables -A INPUT   -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT  -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -i $EXTIF -o $INTIF -m conntrack --ctstate DNAT -j ACCEPT

# Drop invalid state packets
iptables -A INPUT   -m conntrack --ctstate INVALID -j DROP
iptables -A OUTPUT  -m conntrack --ctstate INVALID -j DROP
iptables -A FORWARD -m conntrack --ctstate INVALID -j DROP


### INPUT ###
# SSH
iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A INPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### OUTPUT ###
# SSH
iptables -A OUTPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A OUTPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### FORWARD ###
iptables -A FORWARD -i $INTIF -o $EXTIF -m conntrack --ctstate NEW -j ACCEPT

### SNAT ###
iptables -t nat -A POSTROUTING -o $EXTIF -j SNAT --to-source $INET_IP1

### DNAT ###
# IKE/500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 500 -j DNAT --to-
destination $IPSECGW:500
# NAT-T/4500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 4500 -j DNAT --to-
destination $IPSECGW:4500


### LOGGING ###
iptables -A INPUT    -j LOG --log-level debug --log-prefix '[FW INPUT]:    '
iptables -A OUTPUT   -j LOG --log-level debug --log-prefix '[FW OUTPUT]:   '
iptables -A FORWARD  -j LOG --log-level debug --log-prefix '[FW FORWARD]:  '
```

Устройство Ext-router2-Spoke1

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
```

```
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
# To Core-router
auto ens192
iface ens192 inet static
address 172.17.1.2
netmask 255.255.255.0
gateway 172.17.1.1
pre-up /etc/iptables.rules

# To Spoke1-n2
auto ens224
iface ens224 inet static
address 100.101.1.1
netmask 255.255.255.0
```

Настройки iptables

Содержимое файла `/etc/iptables.rules`:

```
EXTIF="ens192"
INTIF="ens224"
LAN_RANGE="100.101.1.0/24"
INET_IP1="172.17.1.2"
IPSECGW="100.101.1.2"

# Flush previous rules, delete chains and reset counters
iptables -F -t filter
iptables -F -t nat
iptables -F -t mangle
iptables -F

# Default policies
iptables -P INPUT   DROP
iptables -P OUTPUT  DROP
iptables -P FORWARD DROP

# Enable loopback traffic
iptables -A INPUT  -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Enable statefull rules (after that, only need to allow NEW conections)
iptables -A INPUT   -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT  -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -i $EXTIF -o $INTIF -m conntrack --ctstate DNAT -j ACCEPT

# Drop invalid state packets
iptables -A INPUT   -m conntrack --ctstate INVALID -j DROP
iptables -A OUTPUT  -m conntrack --ctstate INVALID -j DROP
iptables -A FORWARD -m conntrack --ctstate INVALID -j DROP


### INPUT ###
```

```
# SSH
iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A INPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### OUTPUT ###
# SSH
iptables -A OUTPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT
# ICMP
iptables -A OUTPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT

### FORWARD ###
iptables -A FORWARD -i $INTIF -o $EXTIF -m conntrack --ctstate NEW -j ACCEPT

### SNAT ###
iptables -t nat -A POSTROUTING -o $EXTIF -j SNAT --to-source $INET_IP1

### DNAT ###
# IKE/500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 500 -j DNAT --to-
destination $IPSECGW:500
# NAT-T/4500
iptables -t nat -A PREROUTING --dst $INET_IP1 -p udp --dport 4500 -j DNAT --to-
destination $IPSECGW:4500


### LOGGING ###
iptables -A INPUT    -j LOG --log-level debug --log-prefix '[FW INPUT]:    '
iptables -A OUTPUT   -j LOG --log-level debug --log-prefix '[FW OUTPUT]:   '
iptables -A FORWARD  -j LOG --log-level debug --log-prefix '[FW FORWARD]:  '
```

Устройство Host-b-hub

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens192
iface ens192 inet static
address 192.168.100.100
netmask 255.255.255.0
gateway 192.168.100.1
```

Устройство Host-b-spoke1

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
```

```
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens192
iface ens192 inet static
address 192.168.1.100
netmask 255.255.255.0
gateway 192.168.1.1
```

Устройство Host-b-spoke2

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens192
iface ens192 inet static
address 192.168.2.100
netmask 255.255.255.0
gateway 192.168.2.1
```

Устройство Int-router-Hub

Сетевые настройки

Содержимое файла `/etc/network/interfaces`:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
# To Hubs
auto ens192
iface ens192 inet static
address 192.168.255.1
netmask 255.255.255.0

# To LAN
```

```
auto ens224
iface ens224 inet static
address 192.168.100.1
netmask 255.255.255.0
```

Настройки динамической маршрутизации

Настройки сервиса FRR (`vtysh`):

```
frr version 8.4.4
frr defaults traditional
hostname Int-router-Hub
log syslog informational
service integrated-vtysh-config
!
router bgp 65001
 bgp router-id 192.168.255.1
 no bgp ebgp-requires-policy
 coalesce-time 1000
 no bgp network import-check
 timers bgp 3 9
 neighbor 192.168.255.10 remote-as 65000
 neighbor 192.168.255.11 remote-as 65000
 neighbor 192.168.255.20 remote-as 65000
 neighbor 192.168.255.21 remote-as 65000
 !
 address-family ipv4 unicast
  network 192.168.100.0/24
  neighbor 192.168.255.10 route-map FROM-HUB1-IN in
  neighbor 192.168.255.11 route-map FROM-HUB2-IN in
  neighbor 192.168.255.20 route-map FROM-HUB3-IN in
  neighbor 192.168.255.21 route-map FROM-HUB4-IN in
 exit-address-family
exit
!
route-map FROM-HUB1-IN permit 1
 set local-preference 100
exit
!
route-map FROM-HUB2-IN permit 1
 set local-preference 75
exit
!
route-map FROM-HUB3-IN permit 1
 set local-preference 50
exit
!
route-map FROM-HUB4-IN permit 1
 set local-preference 25
exit
!
end
```

Устройство Hub1

cisco-like консоль

```
!
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 3
crypto isakmp keepalive retry-count 3
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
hostname Hub1
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp policy 2
 encr gost
 hash gost341112-256-tc26
 authentication gost-sig
 group vko2
!
crypto isakmp profile ALGS_PROFILE
 match identity dn
 set identity mapping
 set policy 1
 set policy 2
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
 mode transport
!
crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
 mode transport
!
object-group network SELF
 description IP on WAN interface
 host 100.100.100.2
!
object-group network PRIVATE
 description private LANs
 100.64.0.0 255.192.0.0
 172.16.0.0 255.240.0.0
 192.168.0.0 255.255.0.0
 10.0.0.0 255.0.0.0
!
object-group network MULTICAST
 host 224.0.0.5
 host 224.0.0.6
 host 224.0.0.18
!
ip access-list extended FIREWALL_OUT
 permit ip object-group SELF object-group PRIVATE
 deny   ip any object-group PRIVATE
```

```
 deny   gre any any
 permit ip any any
!
ip access-list extended FIREWALL_IN
 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
!
ip access-list extended IPSEC_DMVPN
 remark special DMVPN filter
 permit gre object-group SELF each
!
ip access-list extended IPSEC_DMVPN_FOR_4.3
 remark special DMVPN filter for 4.3
 deny   icmp object-group SELF any
 deny   udp object-group SELF any
 deny   tcp object-group SELF any
 deny   ip object-group SELF object-group MULTICAST
 permit ip object-group SELF each
!
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 dynamic-map DMAP 1
 match address IPSEC_DMVPN
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile ALGS_PROFILE
 set security-association mss-fix out
 set pre-fragmentation
 set dead-connection history off
!
crypto map VPN 1 ipsec-isakmp dynamic DMAP
!
interface TenGigabitEthernet0/0
 ip address 100.100.100.2 255.255.255.0
 ip access-group FIREWALL_IN in
 ip access-group FIREWALL_OUT out
 crypto map VPN
 ip inspect INSPECT out
!
interface TenGigabitEthernet0/1
 ip address 192.168.255.10 255.255.255.0
!
interface TenGigabitEthernet0/2
 no ip address
 shutdown
!
interface Tunnel1
 ip address 10.10.10.100 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
 tunnel key 1
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
```

```
!
!
ip route 0.0.0.0 0.0.0.0 100.100.100.1
!
!
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
30820227308201D4A003020102021058E026BFD6D625BE4582C16C6189C18330
...
E34EE9640D905032C1F1E3B4491257D5

quit
!
end
```

Настройки NHRP

Содержимое файла `/etc/opennhrp/opennhrp.conf`:

```
interface mgre_1
 multicast dynamic
 holding-time 90
 cisco-authentication secret
 non-caching
```

Настройки BGP

Содержимое файла `/etc/frr/frr.conf`:

```
frr version 9.0.2
frr defaults traditional
hostname Hub1
log syslog informational
no ipv6 forwarding
service integrated-vtysh-config
!
router bgp 65000
 bgp router-id 10.10.10.100
 bgp log-neighbor-changes
 no bgp ebgp-requires-policy
 timers bgp 3 9
 neighbor spokes-ibgp peer-group
 neighbor spokes-ibgp remote-as 65000
 neighbor 192.168.255.1 remote-as 65001
 bgp listen limit 5000
 bgp listen range 10.10.10.0/24 peer-group spokes-ibgp
 !
 address-family ipv4 unicast
  neighbor spokes-ibgp route-reflector-client
  neighbor spokes-ibgp next-hop-self
  neighbor spokes-ibgp route-map FROM-HUB1-OUT out
 exit-address-family
exit
!
route-map FROM-HUB1-OUT permit 1
 set local-preference 100
exit
!
```

```
end
```

Устройство Hub2

cisco-like консоль

```
!
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 3
crypto isakmp keepalive retry-count 3
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
hostname Hub2
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp policy 2
 encr gost
 hash gost341112-256-tc26
 authentication gost-sig
 group vko2
!
crypto isakmp profile ALGS_PROFILE
 match identity dn
 set identity mapping
 set policy 1
 set policy 2
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
 mode transport
!
crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
 mode transport
!
object-group network SELF
 description IP on WAN interface
 host 100.100.101.2
!
object-group network PRIVATE
 description private LANs
 100.64.0.0 255.192.0.0
 172.16.0.0 255.240.0.0
```

```
 192.168.0.0 255.255.0.0
 10.0.0.0 255.0.0.0
!
object-group network MULTICAST
 host 224.0.0.5
 host 224.0.0.6
 host 224.0.0.18
!
ip access-list extended FIREWALL_OUT
 permit ip object-group SELF object-group PRIVATE
 deny   ip any object-group PRIVATE
 deny   gre any any
 permit ip any any
!
ip access-list extended FIREWALL_IN
 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
!
ip access-list extended IPSEC_DMVPN
 remark special DMVPN filter
 permit gre object-group SELF each
!
ip access-list extended IPSEC_DMVPN_FOR_4.3
 remark special DMVPN filter for 4.3
 deny   icmp object-group SELF any
 deny   udp object-group SELF any
 deny   tcp object-group SELF any
 deny   ip object-group SELF object-group MULTICAST
 permit ip object-group SELF each
!
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 dynamic-map DMAP 1
 match address IPSEC_DMVPN
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile ALGS_PROFILE
 set security-association mss-fix out
 set pre-fragmentation
 set dead-connection history off
!
crypto map VPN 1 ipsec-isakmp dynamic DMAP
!
interface TenGigabitEthernet0/0
 ip address 100.100.101.2 255.255.255.0
 ip access-group FIREWALL_IN in
 ip access-group FIREWALL_OUT out
 crypto map VPN
 ip inspect INSPECT out
!
interface TenGigabitEthernet0/1
 ip address 192.168.255.11 255.255.255.0
!
interface TenGigabitEthernet0/2
```

```
 no ip address
 shutdown
!
interface Tunnel1
 ip address 10.10.10.200 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
 tunnel key 1
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
!
!
ip route 0.0.0.0 0.0.0.0 100.100.101.1
!
!
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
30820227308201D4A003020102021058E026BFD6D625BE4582C16C6189C18330
...
E34EE9640D905032C1F1E3B4491257D5

quit
!
end
```

Настройки NHRP

Содержимое файла `/etc/opennhrp/opennhrp.conf`:

```
interface mgre_1
 multicast dynamic
 holding-time 90
 cisco-authentication secret
 non-caching
```

Настройки BGP

```
frr version 9.0.2
frr defaults traditional
hostname Hub2
log syslog informational
no ipv6 forwarding
service integrated-vtysh-config
!
router bgp 65000
 bgp router-id 10.10.10.200
 bgp log-neighbor-changes
 no bgp ebgp-requires-policy
 timers bgp 3 9
 neighbor spokes-ibgp peer-group
 neighbor spokes-ibgp remote-as 65000
 neighbor 192.168.255.1 remote-as 65001
 bgp listen limit 5000
 bgp listen range 10.10.10.0/24 peer-group spokes-ibgp
 !
 address-family ipv4 unicast
```

```
  neighbor spokes-ibgp route-reflector-client
  neighbor spokes-ibgp next-hop-self
  neighbor spokes-ibgp route-map FROM-HUB2-OUT out
 exit-address-family
exit
!
route-map FROM-HUB2-OUT permit 1
 set local-preference 75
exit
!
line vty
!
end
```

Устройство Hub3

cisco-like консоль

```
!
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 3
crypto isakmp keepalive retry-count 3
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
hostname Hub3
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp policy 2
 encr gost
 hash gost341112-256-tc26
 authentication gost-sig
 group vko2
!
crypto isakmp profile ALGS_PROFILE
 match identity dn
 set identity mapping
 set policy 1
 set policy 2
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
 mode transport
```

```
!
crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
 mode transport
!
object-group network SELF
 description IP on WAN interface
 host 100.101.100.2
!
object-group network PRIVATE
 description private LANs
 100.64.0.0 255.192.0.0
 172.16.0.0 255.240.0.0
 192.168.0.0 255.255.0.0
 10.0.0.0 255.0.0.0
!
object-group network MULTICAST
 host 224.0.0.5
 host 224.0.0.6
 host 224.0.0.18
!
ip access-list extended FIREWALL_OUT
 permit ip object-group SELF object-group PRIVATE
 deny   ip any object-group PRIVATE
 deny   gre any any
 permit ip any any
!
ip access-list extended FIREWALL_IN
 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
!
ip access-list extended IPSEC_DMVPN
 remark special DMVPN filter
 permit gre object-group SELF each
!
ip access-list extended IPSEC_DMVPN_FOR_4.3
 remark special DMVPN filter for 4.3
 deny   icmp object-group SELF any
 deny   udp object-group SELF any
 deny   tcp object-group SELF any
 deny   ip object-group SELF object-group MULTICAST
 permit ip object-group SELF each
!
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 dynamic-map DMAP 1
 match address IPSEC_DMVPN
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile ALGS_PROFILE
 set security-association mss-fix out
 set pre-fragmentation
 set dead-connection history off
!
crypto map VPN 1 ipsec-isakmp dynamic DMAP
```

```
!
interface TenGigabitEthernet0/0
 ip address 100.101.100.2 255.255.255.0
 ip access-group FIREWALL_IN in
 ip access-group FIREWALL_OUT out
 crypto map VPN
 ip inspect INSPECT out
!
interface TenGigabitEthernet0/1
 ip address 192.168.255.20 255.255.255.0
!
interface TenGigabitEthernet0/2
 no ip address
 shutdown
!
interface Tunnel1
 ip address 10.10.20.100 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
 tunnel key 2
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
!
!
ip route 0.0.0.0 0.0.0.0 100.101.100.1
!
!
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
30820227308201D4A003020102021058E026BFD6D625BE4582C16C6189C18330
...
E34EE9640D905032C1F1E3B4491257D5

quit
!
end
```

Настройки NHRP

Содержимое файла `/etc/opennhrp/opennhrp.conf`:

```
interface mgre_1
 multicast dynamic
 holding-time 90
 cisco-authentication secret
 non-caching
```

Настройки BGP

```
frr version 9.0.2
frr defaults traditional
hostname Hub3
log syslog informational
no ipv6 forwarding
service integrated-vtysh-config
!
```

```
router bgp 65000
 bgp router-id 10.10.20.100
 bgp log-neighbor-changes
 no bgp ebgp-requires-policy
 timers bgp 3 9
 neighbor spokes-ibgp peer-group
 neighbor spokes-ibgp remote-as 65000
 neighbor 192.168.255.1 remote-as 65001
 bgp listen limit 5000
 bgp listen range 10.10.20.0/24 peer-group spokes-ibgp
 !
 address-family ipv4 unicast
  neighbor spokes-ibgp route-reflector-client
  neighbor spokes-ibgp next-hop-self
  neighbor spokes-ibgp route-map FROM-HUB3-OUT out
 exit-address-family
exit
!
route-map FROM-HUB3-OUT permit 1
 set local-preference 50
exit
!
line vty
!
end
```

Устройство Hub4

cisco-like консоль

```
!
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 3
crypto isakmp keepalive retry-count 3
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
hostname Hub4
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp policy 2
 encr gost
```

```
 hash gost341112-256-tc26
 authentication gost-sig
 group vko2
!
crypto isakmp profile ALGS_PROFILE
 match identity dn
 set identity mapping
 set policy 1
 set policy 2
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
 mode transport
!
crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
 mode transport
!
object-group network SELF
 description IP on WAN interface
 host 100.101.101.2
!
object-group network PRIVATE
 description private LANs
 100.64.0.0 255.192.0.0
 172.16.0.0 255.240.0.0
 192.168.0.0 255.255.0.0
 10.0.0.0 255.0.0.0
!
object-group network MULTICAST
 host 224.0.0.5
 host 224.0.0.6
 host 224.0.0.18
!
ip access-list extended FIREWALL_OUT
 permit ip object-group SELF object-group PRIVATE
 deny   ip any object-group PRIVATE
 deny   gre any any
 permit ip any any
!
ip access-list extended FIREWALL_IN
 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
!
ip access-list extended IPSEC_DMVPN
 remark special DMVPN filter
 permit gre object-group SELF each
!
ip access-list extended IPSEC_DMVPN_FOR_4.3
 remark special DMVPN filter for 4.3
 deny   icmp object-group SELF any
 deny   udp object-group SELF any
 deny   tcp object-group SELF any
 deny   ip object-group SELF object-group MULTICAST
 permit ip object-group SELF each
!
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 dynamic-map DMAP 1
 match address IPSEC_DMVPN
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile ALGS_PROFILE
 set security-association mss-fix out
 set pre-fragmentation
 set dead-connection history off
!
crypto map VPN 1 ipsec-isakmp dynamic DMAP
!
interface TenGigabitEthernet0/0
 ip address 100.101.101.2 255.255.255.0
 ip access-group FIREWALL_IN in
 ip access-group FIREWALL_OUT out
 crypto map VPN
 ip inspect INSPECT out
!
interface TenGigabitEthernet0/1
 ip address 192.168.255.21 255.255.255.0
!
interface TenGigabitEthernet0/2
 no ip address
 shutdown
!
interface Tunnel1
 ip address 10.10.20.200 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
 tunnel key 2
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
!
!
ip route 0.0.0.0 0.0.0.0 100.101.101.1
!
!
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
30820227308201D4A003020102021058E026BFD6D625BE4582C16C6189C18330
...
E34EE9640D905032C1F1E3B4491257D5

quit
!
end
```

Настройки NHRP

Содержимое файла `/etc/opennhrp/opennhrp.conf`:

```
interface mgre_1
 multicast dynamic
 holding-time 90
```

```
 cisco-authentication secret
 non-caching
```

Настройки BGP

```
frr version 9.0.2
frr defaults traditional
hostname Hub4
log syslog informational
no ipv6 forwarding
service integrated-vtysh-config
!
router bgp 65000
 bgp router-id 10.10.20.200
 bgp log-neighbor-changes
 no bgp ebgp-requires-policy
 timers bgp 3 9
 neighbor spokes-ibgp peer-group
 neighbor spokes-ibgp remote-as 65000
 neighbor 192.168.255.1 remote-as 65001
 bgp listen limit 5000
 bgp listen range 10.10.20.0/24 peer-group spokes-ibgp
 !
 address-family ipv4 unicast
  neighbor spokes-ibgp route-reflector-client
  neighbor spokes-ibgp next-hop-self
  neighbor spokes-ibgp route-map FROM-HUB4-OUT out
 exit-address-family
exit
!
route-map FROM-HUB4-OUT permit 1
 set local-preference 25
exit
!
line vty
!
end
```

Устройство Spoke1-n1

cisco-like консоль

```
!
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 3
crypto isakmp keepalive retry-count 3
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
```

```
fhrp version vrrp v3
hostname Spoke1-n1
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp policy 2
 encr gost
 hash gost341112-256-tc26
 authentication gost-sig
 group vko2
!
crypto isakmp profile ALGS_PROFILE
 match identity dn
 set identity mapping
 set policy 1
 set policy 2
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
 mode transport
!
crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
 mode transport
!
object-group network SELF
 description IP on WAN interface
 host 100.100.1.2
!
object-group network PRIVATE
 description private LANs
 100.64.0.0 255.192.0.0
 172.16.0.0 255.240.0.0
 192.168.0.0 255.255.0.0
 10.0.0.0 255.0.0.0
!
object-group network MULTICAST
 host 224.0.0.5
 host 224.0.0.6
 host 224.0.0.18
!
ip access-list extended FIREWALL_OUT
 permit ip object-group SELF object-group PRIVATE
 deny   ip any object-group PRIVATE
 deny   gre any any
 permit ip any any
!
ip access-list extended FIREWALL_IN
 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
!
ip access-list extended IPSEC_DMVPN
 remark special DMVPN filter
 permit gre object-group SELF each
!
```

```
ip access-list extended IPSEC_DMVPN_FOR_4.3
 remark special DMVPN filter for 4.3
 deny   icmp object-group SELF any
 deny   udp object-group SELF any
 deny   tcp object-group SELF any
 deny   ip object-group SELF object-group MULTICAST
 permit ip object-group SELF each
!
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 dynamic-map DMAP 1
 match address IPSEC_DMVPN
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile ALGS_PROFILE
 set security-association mss-fix out
 set pre-fragmentation
 set dead-connection history off
!
crypto map VPN 1 ipsec-isakmp dynamic DMAP
!
interface TenGigabitEthernet0/0
 ip address 100.100.1.2 255.255.255.0
 ip access-group FIREWALL_IN in
 ip access-group FIREWALL_OUT out
 crypto map VPN
 ip inspect INSPECT out
!
interface TenGigabitEthernet0/1
 ip address 192.168.1.10 255.255.255.0
 vrrp 1 address-family ipv4
  timers advertise 3
  timers garp 5
  no preempt
  track sla check_isp_link
  track interface TenGigabitEthernet0/0
  address 192.168.1.1 255.255.255.0
  exit-vrrp
!
interface TenGigabitEthernet0/2
 no ip address
 shutdown
!
interface Tunnel1
 ip address 10.10.10.1 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
 tunnel key 1
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
!
interface Tunnel2
 ip address 10.10.20.1 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
```

```
 tunnel key 2
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
!
!
ip route 0.0.0.0 0.0.0.0 100.100.1.1
!
!
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
30820227308201D4A003020102021058E026BFD6D625BE4582C16C6189C18330
...
E34EE9640D905032C1F1E3B4491257D5

quit
!
vrrp track sla check_isp_link
 host 172.16.1.1
 host 172.16.101.15
 mode icmp-echo
 reachability any
 source TenGigabitEthernet0/0
 fall 15
 rise 10
 timeout 2
 interval 3
!
vrrp notify common
end
```

Настройки NHRP

Содержимое файла `/etc/opennhrp/opennhrp.conf`:

```
interface mgre_1
 map 10.10.10.100/24 172.16.100.2 register
 map 10.10.10.200/24 172.16.101.2 register
 multicast nhs
 holding-time 90
 cisco-authentication secret
 non-caching
interface mgre_2
 map 10.10.20.100/24 172.17.100.2 register
 map 10.10.20.200/24 172.17.101.2 register
 multicast nhs
 holding-time 90
 cisco-authentication secret
 non-caching
```

Настройки BGP

```
frr version 9.0.2
frr defaults traditional
hostname Spoke1-n1
log syslog informational
no ipv6 forwarding
```

```
service integrated-vtysh-config
!
ip route 192.168.100.0/24 192.168.1.1 240 table 254
!
router bgp 65000
 bgp router-id 192.168.1.10
 bgp log-neighbor-changes
 no bgp ebgp-requires-policy
 timers bgp 3 9
 neighbor 10.10.10.100 remote-as 65000
 neighbor 10.10.10.100 timers connect 30
 neighbor 10.10.20.100 remote-as 65000
 neighbor 10.10.20.100 timers connect 30
 neighbor 10.10.10.200 remote-as 65000
 neighbor 10.10.10.200 timers connect 30
 neighbor 10.10.20.200 remote-as 65000
 neighbor 10.10.20.200 timers connect 30
 !
 address-family ipv4 unicast
  network 192.168.1.0/24
  neighbor 10.10.10.100 route-map TO-HUB1-OUT out
  neighbor 10.10.10.200 route-map TO-HUB2-OUT out
  neighbor 10.10.20.100 route-map TO-HUB3-OUT out
  neighbor 10.10.20.200 route-map TO-HUB4-OUT out
 exit-address-family
!
route-map TO-HUB2-OUT permit 1
 set local-preference 75
!
route-map TO-HUB1-OUT permit 1
 set local-preference 100
!
route-map TO-HUB4-OUT permit 1
 set local-preference 25
!
route-map TO-HUB3-OUT permit 1
 set local-preference 50
!
line vty
!
```

Настройки VRRP

Содержимое файла `/etc/keepalived/notify_common.conf`:

```
FLAG_DMVPN_SCENARIO_ON_CLUSTERED_SPOKE="true"
DMVPN_INTERNAL_ROUTING_PROTOCOL="IBGP"
DMVPN_LOCAl_ASN="65000"
DMVPN_BGP_NEIGHBORS="10.10.10.100 10.10.10.200 10.10.20.100 10.10.20.200"
```

Содержимое файла `/etc/keepalived/keepalived.conf`:

```
global_defs {
  enable_dbus
}
vrrp_sync_group 0 {
  notify "/etc/keepalived/scripts/notify_common"
  group {
    eth1_1
  }
}
vrrp_script check_isp_link {
```

```
  script "/etc/keepalived/scripts/sla check --ip-version 4 --mode icmp-echo --hosts
172.16.1.1 172.16.101.15 --reachability any --source-iface eth0 --tos 0xB8 --data-
size 56 --name check_isp_link"
  fall 15
  rise 10
  timeout 2
  interval 3
}
vrrp_instance eth1_1 {
  version 3
  interface eth1
  track_interface {
    eth0
  }
  virtual_ipaddress {
    192.168.1.1/24 label eth1:900
  }
  nopreempt
  priority 100
  advert_int 3
  garp_master_refresh 5
  virtual_router_id 1
  track_script {
    check_isp_link
  }
}
```

Устройство Spoke1-n2

cisco-like консоль

```
!
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 3
crypto isakmp keepalive retry-count 3
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
fhrp version vrrp v3
hostname Spoke1-n2
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
```

```
crypto isakmp policy 2
 encr gost
 hash gost341112-256-tc26
 authentication gost-sig
 group vko2
!
crypto isakmp profile ALGS_PROFILE
 match identity dn
 set identity mapping
 set policy 1
 set policy 2
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
 mode transport
!
crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
 mode transport
!
object-group network SELF
 description IP on WAN interface
 host 100.101.1.2
!
object-group network PRIVATE
 description private LANs
 100.64.0.0 255.192.0.0
 172.16.0.0 255.240.0.0
 192.168.0.0 255.255.0.0
 10.0.0.0 255.0.0.0
!
object-group network MULTICAST
 host 224.0.0.5
 host 224.0.0.6
 host 224.0.0.18
!
ip access-list extended FIREWALL_OUT
 permit ip object-group SELF object-group PRIVATE
 deny   ip any object-group PRIVATE
 deny   gre any any
 permit ip any any
!
ip access-list extended FIREWALL_IN
 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
!
ip access-list extended IPSEC_DMVPN
 remark special DMVPN filter
 permit gre object-group SELF each
!
ip access-list extended IPSEC_DMVPN_FOR_4.3
 remark special DMVPN filter for 4.3
 deny   icmp object-group SELF any
 deny   udp object-group SELF any
 deny   tcp object-group SELF any
 deny   ip object-group SELF object-group MULTICAST
 permit ip object-group SELF each
!
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 dynamic-map DMAP 1
 match address IPSEC_DMVPN
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile ALGS_PROFILE
 set security-association mss-fix out
 set pre-fragmentation
 set dead-connection history off
!
crypto map VPN 1 ipsec-isakmp dynamic DMAP
!
interface TenGigabitEthernet0/0
 ip address 100.101.1.2 255.255.255.0
 ip access-group FIREWALL_IN in
 ip access-group FIREWALL_OUT out
 crypto map VPN
 ip inspect INSPECT out
!
interface TenGigabitEthernet0/1
 ip address 192.168.1.20 255.255.255.0
 vrrp 1 address-family ipv4
  timers advertise 3
  timers garp 5
  no preempt
  priority 50
  track sla check_isp_link
  track interface TenGigabitEthernet0/0
  address 192.168.1.1 255.255.255.0
  exit-vrrp
!
interface TenGigabitEthernet0/2
 no ip address
 shutdown
!
interface Tunnel1
 ip address 10.10.10.2 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
 tunnel key 1
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
!
interface Tunnel2
 ip address 10.10.20.2 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
 tunnel key 2
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
!
!
ip route 0.0.0.0 0.0.0.0 100.101.1.1
!
!
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
30820227308201D4A003020102021058E026BFD6D625BE4582C16C6189C18330
...
E34EE9640D905032C1F1E3B4491257D5

quit
!
vrrp track sla check_isp_link
 host 172.16.1.1
 host 172.16.101.15
 mode icmp-echo
 reachability any
 source TenGigabitEthernet0/0
 fall 15
 rise 10
 timeout 2
 interval 3
!
vrrp notify common
end
```

Настройки NHRP

Содержимое файла `/etc/opennhrp/opennhrp.conf`:

```
interface mgre_1
 map 10.10.10.100/24 172.16.100.2 register
 map 10.10.10.200/24 172.16.101.2 register
 multicast nhs
 holding-time 90
 cisco-authentication secret
 non-caching
interface mgre_2
 map 10.10.20.100/24 172.17.100.2 register
 map 10.10.20.200/24 172.17.101.2 register
 multicast nhs
 holding-time 90
 cisco-authentication secret
 non-caching
```

Настройки BGP

```
frr version 9.0.2
frr defaults traditional
hostname Spoke1-n2
log syslog informational
no ipv6 forwarding
service integrated-vtysh-config
!
ip route 192.168.100.0/24 192.168.1.1 240 table 254
!
router bgp 65000
 bgp router-id 192.168.1.20
 bgp log-neighbor-changes
 no bgp ebgp-requires-policy
 timers bgp 3 9
 neighbor 10.10.10.100 remote-as 65000
```

```
 neighbor 10.10.10.100 timers connect 30
 neighbor 10.10.20.100 remote-as 65000
 neighbor 10.10.20.100 timers connect 30
 neighbor 10.10.10.200 remote-as 65000
 neighbor 10.10.10.200 timers connect 30
 neighbor 10.10.20.200 remote-as 65000
 neighbor 10.10.20.200 timers connect 30
 !
 address-family ipv4 unicast
  network 192.168.1.0/24
  neighbor 10.10.10.100 route-map TO-HUB1-OUT out
  neighbor 10.10.10.200 route-map TO-HUB2-OUT out
  neighbor 10.10.20.100 route-map TO-HUB3-OUT out
  neighbor 10.10.20.200 route-map TO-HUB4-OUT out
 exit-address-family
!
route-map TO-HUB2-OUT permit 1
 set local-preference 75
!
route-map TO-HUB1-OUT permit 1
 set local-preference 100
!
route-map TO-HUB4-OUT permit 1
 set local-preference 25
!
route-map TO-HUB3-OUT permit 1
 set local-preference 50
!
line vty
!
```

Настройки VRRP

Содержимое файла `/etc/keepalived/notify_common.conf`:

```
FLAG_DMVPN_SCENARIO_ON_CLUSTERED_SPOKE="true"
DMVPN_INTERNAL_ROUTING_PROTOCOL="IBGP"
DMVPN_LOCAl_ASN="65000"
DMVPN_BGP_NEIGHBORS="10.10.10.100 10.10.10.200 10.10.20.100 10.10.20.200"
```

Содержимое файла `/etc/keepalived/keepalived.conf`:

```
global_defs {
  enable_dbus
}
vrrp_sync_group 0 {
  notify "/etc/keepalived/scripts/notify_common"
  group {
    eth1_1
  }
}
vrrp_script check_isp_link {
  script "/etc/keepalived/scripts/sla check --ip-version 4 --mode icmp-echo --hosts
172.16.1.1 172.16.101.15 --reachability any --source-iface eth0 --tos 0xB8 --data-
size 56 --name check_isp_link"
  fall 15
  rise 10
  timeout 2
  interval 3
}
vrrp_instance eth1_1 {
  version 3
```

```
  interface eth1
  track_interface {
    eth0
  }
  virtual_ipaddress {
    192.168.1.1/24 label eth1:900
  }
  nopreempt
  priority 50
  advert_int 3
  garp_master_refresh 5
  virtual_router_id 1
  track_script {
    check_isp_link
  }
}
```

Устройство Spoke2

cisco-like консоль

```
!
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 3
crypto isakmp keepalive retry-count 3
username cscons privilege 15 secret 5 $6$tHtq8SR6$t3CWE6udI6L/ARr9jQowUYR7wEbOW
Zlx61OvLi7goonOFUYhNSGV49BA.RDGEZ7oKXBA1aTRi20ElR4wtMXTl0
aaa new-model
!
!
hostname Spoke2
enable secret 5 PC9d7N5HlAyLrzuA3qRJvQ==
!
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp policy 2
 encr gost
 hash gost341112-256-tc26
 authentication gost-sig
 group vko2
!
crypto isakmp profile ALGS_PROFILE
 match identity dn
 set identity mapping
 set policy 1
 set policy 2
```

```
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
 mode transport
!
crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
 mode transport
!
object-group network SELF
 description IP on WAN interface
 host 172.16.2.2
 host 172.17.2.2
!
object-group network PRIVATE
 description private LANs
 100.64.0.0 255.192.0.0
 172.16.0.0 255.240.0.0
 192.168.0.0 255.255.0.0
 10.0.0.0 255.0.0.0
!
object-group network MULTICAST
 host 224.0.0.5
 host 224.0.0.6
 host 224.0.0.18
!
ip access-list extended FIREWALL_OUT
 permit ip object-group SELF object-group PRIVATE
 deny   ip any object-group PRIVATE
 deny   gre any any
 permit ip any any
!
ip access-list extended FIREWALL_IN
 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
!
ip access-list extended IPSEC_DMVPN
 remark special DMVPN filter
 permit gre object-group SELF each
!
ip access-list extended IPSEC_DMVPN_FOR_4.3
 remark special DMVPN filter for 4.3
 deny   icmp object-group SELF any
 deny   udp object-group SELF any
 deny   tcp object-group SELF any
 deny   ip object-group SELF object-group MULTICAST
 permit ip object-group SELF each
!
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 dynamic-map DMAP_ISP1 1
 match address IPSEC_DMVPN
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile ALGS_PROFILE
 set security-association mss-fix out
```

```
 set pre-fragmentation
 set dead-connection history off
!
crypto dynamic-map DMAP_ISP2 1
 match address IPSEC_DMVPN
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile ALGS_PROFILE
 set security-association mss-fix out
 set pre-fragmentation
 set dead-connection history off
!
crypto map VPN_ISP1 1 ipsec-isakmp dynamic DMAP_ISP1
!
crypto map VPN_ISP2 1 ipsec-isakmp dynamic DMAP_ISP2
!
interface TenGigabitEthernet0/0
 ip address 172.16.2.2 255.255.255.0
 ip access-group FIREWALL_IN in
 ip access-group FIREWALL_OUT out
 crypto map VPN_ISP1
 ip inspect INSPECT out
!
interface TenGigabitEthernet0/1
 ip address 172.17.2.2 255.255.255.0
 ip access-group FIREWALL_IN in
 ip access-group FIREWALL_OUT out
 crypto map VPN_ISP2
 ip inspect INSPECT out
!
interface TenGigabitEthernet0/2
 ip address 192.168.2.1 255.255.255.0
!
interface TenGigabitEthernet0/3
 no ip address
 shutdown
!
interface Tunnel1
 ip address 10.10.10.3 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
 tunnel key 1
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
!
interface Tunnel2
 ip address 10.10.20.3 255.255.255.0
 mtu 1400
 tunnel mode gre multipoint
 tunnel key 2
 offloading tx-checksumming
 tunnel ttl 64
 tunnel tos inherit
!
!
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
```

```
30820227308201D4A003020102021058E026BFD6D625BE4582C16C6189C18330
...
E34EE9640D905032C1F1E3B4491257D5

quit
!
end
```

Настройки NHRP

Содержимое файла `/etc/opennhrp/opennhrp.conf`:

```
interface mgre_1
 map 10.10.10.100/24 172.16.100.2 register
 map 10.10.10.200/24 172.16.101.2 register
 multicast nhs
 holding-time 90
 cisco-authentication secret
 non-caching
interface mgre_2
 map 10.10.20.100/24 172.17.100.2 register
 map 10.10.20.200/24 172.17.101.2 register
 multicast nhs
 holding-time 90
 cisco-authentication secret
 non-caching
```

Настройки BGP

```
frr version 9.0.2
frr defaults traditional
hostname spoke2
log syslog informational
no ipv6 informational
service integrated-vtysh-config
!
router bgp 65000
 bgp router-id 192.168.2.1
 bgp log-neighbor-changes
 no bgp ebgp-requires-policy
 timers bgp 3 9
 neighbor 10.10.10.100 remote-as 65000
 neighbor 10.10.10.100 timers connect 30
 neighbor 10.10.20.100 remote-as 65000
 neighbor 10.10.20.100 timers connect 30
 neighbor 10.10.10.200 remote-as 65000
 neighbor 10.10.10.200 timers connect 30
 neighbor 10.10.20.200 remote-as 65000
 neighbor 10.10.20.200 timers connect 30
 !
 address-family ipv4 unicast
  network 192.168.2.0/24
  neighbor 10.10.10.100 route-map TO-HUB1-OUT out
  neighbor 10.10.10.200 route-map TO-HUB2-OUT  out
  neighbor 10.10.20.100 route-map TO-HUB3-OUT  out
  neighbor 10.10.20.200 route-map TO-HUB4-OUT  out
 exit-address-family
!
route-map TO-HUB2-OUT permit 1
 set local-preference 75
!
```

```
route-map TO-HUB1-OUT permit 1
 set local-preference 100
!
route-map TO-HUB4-OUT permit 1
 set local-preference 25
!
route-map TO-HUB3-OUT permit 1
 set local-preference 50
!
line vty
!
```

Настройки multiwan

```
config_format_version: 1
links_params:
  links:
  - name: ISP1
    iface_name: eth0
    prio: 300
    routes:
      ipv4:
        networks:
        - 0.0.0.0/0
        nexthop: 172.16.2.1
      watch_foreign_routes: true
    check:
      hosts_reachability: any
      ipv4:
        hosts:
        - 172.16.101.15
        - 172.16.1.1
      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: 172.17.2.1
      watch_foreign_routes: True
    check:
      hosts_reachability: any
      ipv4:
        hosts:
        - 172.16.101.15
        - 172.16.1.1
      method:
        icmp:
          max_rtt: 100
          interval: 3
          restore_interval: 1
```

```
        restore_count: 10
        failure_count: 10
        tos: 184
        data_size: 56
```