Построение отказоустойчивого решения на базе протокола VRRP с динамической маршрутизацией GRE/OSPF

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

Введение

Данный сценарий содержит пример настройки «С-Терра Шлюз» для построения отказоустойчивого решения на базе протокола VRRP с динамической маршрутизацией GRE/OSPF.

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

Партнеры по IPsec, в настоящем документе – криптошлюзы, для аутентификации будут использовать сертификаты в формате x509. В качестве криптопровайдера будет использована криптографическая библиотека, разработанная компанией «С-Терра СиЭсПи». Криптошлюзы в центральном офисе и филиале версии 5.0.

В данном сценарии между криптошлюзом центрального офиса, захватившим роль Master, и криптошлюзом филиала строится GRE-туннель, по которому происходит обмен маршрутами до защищаемых подсетей с помощью протокола OSPF. Пользовательский трафик также инкапсулируется в GRE и шифруется.

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, VRRP, NAT, Firewall, routing, switching, GRE, OSPF.

Администратор должен быть ознакомлен с базовым сценарием «Организация защищенного взаимодействия между подсетями центрального офиса и филиалов посредством IPsec VPN туннеля (протокол IPv4)».

Требования к инфраструктуре

1. Требования к устройствам.

1.1. «С-Терра Шлюз».

1.1.1. Устройства «С-Терра Шлюз» должны быть инициализированы (подробнее 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 http://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), либо утилиту `ca-tool`, входящую в состав пакета auxiliary-tools (пакет auxiliary-tools предназначен для установки на «С-Терра Шлюз» 5.0, документация на ca-tool приведена в инструкции «Центр выпуска тестовых ГОСТ сертификатов для IPsec», документацию и пакет можно загрузить на http://doc.s-terra.ru).

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 (СОС). Устройство с именем CRL_distribution_point на схеме. Если по объективным причинам использование СОС не представляется возможным или не требуется, то проверку СОС можно отключить в CLI «С-Терра Шлюз» (при соблюдении правил пользования).

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

2. Требования к сетевому взаимодействию.

2.1. Между устройствами стенда должна быть обеспечена IP связность.

Схема взаимодействия

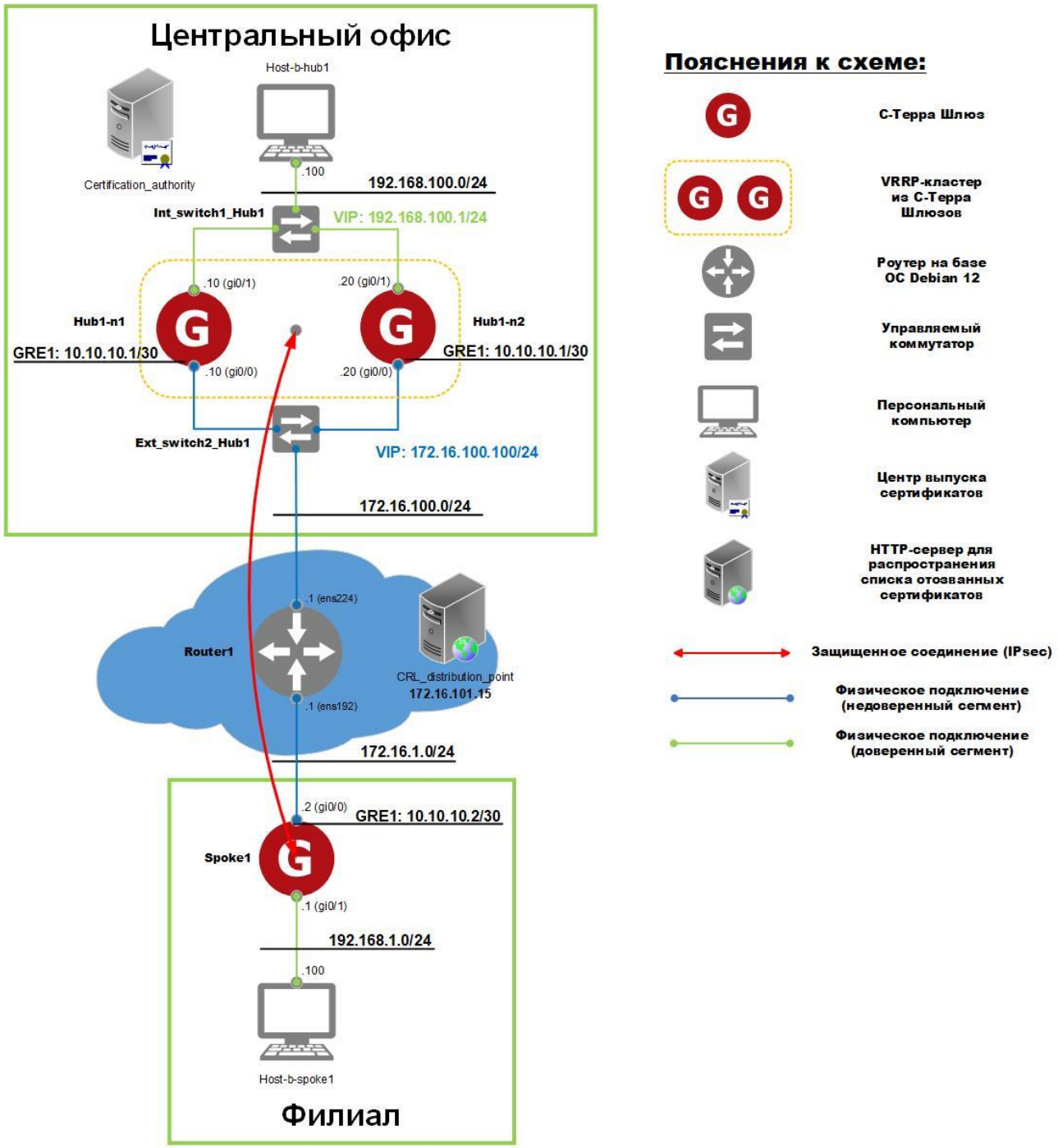


Рисунок 1. Схема взаимодействия

Общая логика работы

1. Размещение устройств.

   1.1. В центральном офисе размещаются: центр выпуска сертификатов (Certification_authority), два криптошлюза «С-Терра Шлюз» (Hub1-n1 и Hub1-n2), два коммутатора (Int_switch1_Hub1 и Ext_switch2_Hub1) и персональный компьютер (Host-b-hub1).

   1.2. В филиале размещаются: криптошлюз «С-Терра Шлюз» (Spoke1) и персональный компьютер (Host-b-spoke1).

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

2. Подключение к сети Интернет.

   В данном сценарии для эмуляции сети Интернет используется маршрутизатор Router1.

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

   2.1. Криптошлюзы Hub1-n1 и Hub1-n2 подключается к сети Интернет с помощью статической маршрутизации (маршрут по умолчанию через маршрутизатор Router1).

   Если на кластер выделен только один публичный IP адрес, то используйте разные подсети для кластерного IP адреса и основного IP адреса на интерфейсе, где включен VRRP.

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

3. Отказоустойчивость.

   Криптошлюзы Hub1-n1 и Hub1-n2 объединяются в отказоустойчивый кластер, работающий на базе протокола VRRP в режиме Master/Backup. За реализацию протокола VRRP отвечает ПО «keepalived». Криптошлюзы Hub1-n1 и Hub1-n2 являются нодами кластера.

   Криптошлюз Hub1-n1 считается основной нодой кластера, криптошлюз Hub1-n2 - резервной. Приоритет VRRP на интерфейсах резервной ноды ниже, чем на интерфейсах основной ноды.

   Конфигурация между нодами кластера не синхронизируется. При изменении настроек на одной ноде, нужно выполнить аналогичные настройки на другой. Ноды должны отличаться настройками отслеживаемых интерфейсов, приоритетом и маршрутом по умолчанию, используемым нодой при переходе в режим Backup/False.

   Нода, находящаяся в состоянии Master, владеет кластерными адресами в доверенном и недоверенном сегментах и, соответственно, обрабатывает трафик. Нода, находящаяся в состоянии Backup или Fault, не владеет кластерными адресами (VIP адреса на схеме).

   По умолчанию все VRRP интерфейсы синхронизированы в группу номер ноль. Соответственно, они одновременно меняют свое состояние (данное поведение отличается от Cisco IOS, где по умолчанию VRRP интерфейсы не синхронизированы).

   Обнаружение отказа происходит благодаря служебным пакетам протокола VRRP (VRRP Advertisement messages). Служебные пакеты отправляются только нодой, находящейся в состоянии Master (адрес источника пакетов – основной IP адрес на интерфейсе, где включен VRRP; IP адрес назначения - 224.0.0.18). Если резервная нода в состоянии Backup перестает получать на всех синхронизированных интерфейсах пакеты VRRP Advertisement messages от основной ноды, то она захватывает роль Master.

   На нодах есть маршрут по умолчанию через адрес доверенного сегмента другой ноды (для доступа резервной ноды в сеть интернет). Метрика данного маршрута должна быть больше, чем метрика маршрута, добавляемая при переходе ноды в состоянии Master.

   Кластер обрабатывает следующие типы отказов:

   - отключение питания;
   - выход из строя аппаратной платформы;
   - отказ сетевого порта на аппаратной платформе;
   - отказ сетевого порта на коммутационном оборудовании;
   - отказ сервиса `vpngate`, обеспечивающего построение IPsec туннелей.

   Среднее время переключения составляет 15 секунд.

4. Параметры шифрования IPsec.

   Внутри GRE-туннелей проходит пользовательский трафик между защищаемыми подсетями, а также пакеты динамической маршрутизации OSPF. Весь GRE трафик между криптошлюзами центрального офиса (Hub1-n1, Hub1-n2) и криптошлюзом филиала (Spoke1) защищается с использованием алгоритмов ГОСТ и протокола IPsec в туннельном режиме.

   Инициировать защищенное соединение между криптошлюзом центрального офиса в состоянии Master и криптошлюзом Spoke1 может трафик из подсети филиала (192.168.1.0/24) в подсеть центрального офиса (192.168.100.0/24) и наоборот из центра в филиал.

   GRE-туннель строится между внешним VIP адресом кластера (172.16.100.100/24) и адресом внешнего интерфейса филиала (172.16.1.2/24). В GRE-туннеле происходит обмен информации о защищённых сетях. За реализацию протокола OSPF отвечает ПО «FFRouting». При отказе GRE-туннеля или протокола OSPF, трафик до защищаемой подсети партнера будет блокироваться с помощью межсетевого экрана.

   4.1. Параметры протокола IKE:

      4.1.1. 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).

      4.1.2. 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).

   4.2. Параметры протокола ESP для версии 5.0:

      4.2.1. 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).

      4.2.2. Параметры протокола ESP для совместимости с версией 4.3:

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

5. Параметры межсетевого экранирования.

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

   5.1. На LAN интерфейсах IPv4 трафик не контролируется.

   5.2. На LAN интерфейсах IPv6 трафик заблокирован.

   5.3. На WAN интерфейсах разрешен входящий IPv4 трафик:

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

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

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

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

      5.3.5. VRRP трафик – для работы VRRP.

5.4. На WAN интерфейсах IPv6 трафик заблокирован.

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

5.6. На WAN интерфейсах запрещен исходящий IPv4 трафик, направленный из локальной защищаемой сети в защищаемую сеть партнера. Правило служит для предотвращения утечки трафика в случае падения IPsec.

Настройка стенда

Настройка устройства Router1

1. Настройте IP адрес – 172.16.1.1 и маску – 255.255.255.0 на сетевом интерфейсе ens224.
2. Настройте IP адрес – 172.16.100.1 и маску – 255.255.255.0 на сетевом интерфейсе ens192.
3. Разрешите прохождение IP трафика (если заблокировано).

Настройка устройства Host-b-hub1

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

Настройка устройства Host-b-spoke1

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

Настройка криптошлюза Hub1-n1

Настройка будет происходить локально при помощи консольного подключения.

Настройка может осуществляться и удаленно (по SSH), но исключительно по доверенному каналу связи. Доверенным каналом связи может считаться канал в пределах контролируемой зоны в случае отсутствия в нем нарушителя (в нашем примере это подсети 192.168.100.0/24, 192.168.1.0/24). 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.

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.

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 криптошлюз и импортируйте их в базу Продукта.

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-n1
```

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

```
Hub1-n1(config)#interface GigabitEthernet 0/0
Hub1-n1(config-if)#ip address 172.16.100.10 255.255.255.0
Hub1-n1(config-if)#no shutdown
Hub1-n1(config-if)#exit
Hub1-n1(config)#interface GigabitEthernet 0/1
Hub1-n1(config-if)#no shutdown
Hub1-n1(config-if)#ip address 192.168.100.10 255.255.255.0
Hub1-n1(config-if)#exit
```

3. Задайте маршрут по умолчанию через LAN интерфейс второй ноды (Hub1-n2) с метрикой 100:

```
Hub1-n1(config)#ip route 0.0.0.0 0.0.0.0 192.168.100.20 100
```

Данный маршрут будет использоваться нодой, когда она переходит в состояние Backup/Fault.

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка шифрования

1. Параметры IKE.

1.1. Задайте общие параметры IKE:

```
Hub1-n1(config)#crypto isakmp identity dn
Hub1-n1(config)#crypto isakmp keepalive 3 2
Hub1-n1(config)#crypto isakmp keepalive retry-count 5
Hub1-n1(config)#crypto isakmp fragmentation
Hub1-n1(config)#crypto isakmp security-association lifetime delta 50
Hub1-n1(config)#crypto isakmp initiator-sessions-max 100
Hub1-n1(config)#crypto isakmp responder-sessions-max 100
```

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

```
Hub1-n1(config)#crypto isakmp policy 1
Hub1-n1(config-isakmp)#encryption magma
Hub1-n1(config-isakmp)#hash stribog-256
Hub1-n1(config-isakmp)#authentication gost-sig
Hub1-n1(config-isakmp)#group vko2
Hub1-n1(config-isakmp)#exit
```

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

```
Hub1-n1(config)#crypto isakmp policy 2
```

```
Hub1-n1(config-isakmp)#encryption gost
Hub1-n1(config-isakmp)#hash gost341112-256-tc26
Hub1-n1(config-isakmp)#authentication gost-sig
Hub1-n1(config-isakmp)#group vko2
Hub1-n1(config-isakmp)#exit
```

1.4. Создайте профиль настроек ISAKMP под именем `NEW_ALGS`:

```
Hub1-n1(config)#crypto isakmp profile NEW_ALGS
Hub1-n1(conf-isa-prof)#match identity dn
Hub1-n1(conf-isa-prof)#set identity mapping
Hub1-n1(conf-isa-prof)#set policy 1
Hub1-n1(conf-isa-prof)#exit
```

1.5. Опционально: для совместимости с версией 4.3 создайте профиль настроек ISAKMP (алгоритм IKE) под именем `OLD_ALGS`, чтобы далее его прикрепить к соответствующему разделу крипто-карты:

```
Hub1-n1(config)#crypto isakmp profile OLD_ALGS
Hub1-n1(conf-isa-prof)#match identity dn
Hub1-n1(conf-isa-prof)#set identity mapping
Hub1-n1(conf-isa-prof)#set policy 2
Hub1-n1(conf-isa-prof)#exit
```

2. Параметры IPsec.

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

```
Hub1-n1(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-n1(config)# crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
Hub1-n1(cfg-crypto-trans)#exit
```

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-n1(config)# crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
Hub1-n1(cfg-crypto-trans)#exit
```

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

```
Hub1-n1(config)#ip access-list extended IPSEC_ACl_HUB1_AND_SPOKE1
Hub1-n1(config-ext-nacl)# permit gre host 172.16.100.100 host 172.16.1.2
Hub1-n1(config-ext-nacl)#exit
```

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

```
Hub1-n1(config)#crypto map VPN 1 ipsec-isakmp
Hub1-n1(config-crypto-map)#match address IPSEC_ACl_HUB1_AND_SPOKE1
Hub1-n1(config-crypto-map)#set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
Hub1-n1(config-crypto-map)#set peer 172.16.1.2
Hub1-n1(config-crypto-map)#set isakmp-profile NEW_ALGS
Hub1-n1(config-crypto-map)#set pre-fragmentation
Hub1-n1(config-crypto-map)#set dead-connection history off
Hub1-n1(config-crypto-map)#set security-association mss-fix out
Hub1-n1(config-crypto-map)#exit
```

2.6. Опционально: для совместимости с 4.3 используйте криптокарту со следующими параметрами:

```
Hub1-n1(config)#crypto map VPN 1 ipsec-isakmp
Hub1-n1(config-crypto-map)#match address IPSEC_ACl_HUB1_AND_SPOKE1
Hub1-n1(config-crypto-map)#set transform-set GOST_ENCRYPT_AND_INTEGRITY
```

```
Hub1-n1(config-crypto-map)#set peer 172.16.1.2
Hub1-n1(config-crypto-map)#set isakmp-profile OLD_ALGS
Hub1-n1(config-crypto-map)#set dead-connection history off
Hub1-n1(config-crypto-map)#set pre-fragmentation
Hub1-n1(config-crypto-map)#set security-association mss-fix out
Hub1-n1(config-crypto-map)#exit
```

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

```
Hub1-n1(config)#interface GigabitEthernet0/0
Hub1-n1(config-if)# crypto map VPN
```

3. Примените настройки:

```
Hub1-n1(config-if)#end
```

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка политики обработки СОС (CRL)

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

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

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка межсетевого экрана

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

```
Hub1-n1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Hub1-n1(config)#object-group network SELF
Hub1-n1(config-network-group)#remark WAN IP
Hub1-n1(config-network-group)#host 172.16.100.100
Hub1-n1(config-network-group)#host 172.16.100.10
Hub1-n1(config-network-group)#exit
```

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

```
Hub1-n1(config)#ip access-list extended FW_INPUT_WAN
Hub1-n1(config-ext-nacl)#permit udp any object-group SELF eq isakmp
Hub1-n1(config-ext-nacl)#permit udp any object-group SELF eq non500-isakmp
Hub1-n1(config-ext-nacl)#permit icmp any object-group SELF
Hub1-n1(config-ext-nacl)#permit esp any object-group SELF
Hub1-n1(config-ext-nacl)#permit vrrp any host 224.0.0.18
Hub1-n1(config-ext-nacl)#exit
```

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

```
Hub1-n1(config)#ip inspect name INSPECT tcp
Hub1-n1(config)#ip inspect name INSPECT udp
Hub1-n1(config)#ip inspect tcp sessions-max 256000
Hub1-n1(config)#ip inspect udp sessions-max 65536
```

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

```
Hub1-n1(config)#interface GigabitEthernet0/0
Hub1-n1(config-if)#ip access-group FW_INPUT_WAN in
Hub1-n1(config-if)#ip inspect INSPECT out
Hub1-n1(config-if)#exit
```

5. Примените настройки:

```
Hub1-n1(config)#end
```

Настройка списков доступа для предотвращения утечки трафика

В случае отсутствия IPsec соединения остановится обмен OSPF пакетами, на криптошлюзах будет удален маршрут до защищаемой сети партнера. Это приведет к тому, что трафик между подсетью центрального офиса и филиалом не будет инкапсулирован в GRE и будет отправляться по маршрутам по умолчанию. В целях безопасности, на криптошлюзах следует заблокировать прохождение этого трафика. Сделать это можно с помощью описания расширенных списков доступа. При этом, при наличии NAT необходимо использовать следующую логику: блокировать весь трафик, у которого исходный адрес соответствует сети доверенного сегмента (192.168.100.0/24), а целевой адрес – сети доверенного сегмента партнера (192.168.1.0/24).

Для устройства Hub1-n1 настройка будет следующей:

1. Создайте список доступа для защищаемого трафика:

```
Hub1-n1(config)# ip access-list extended ACL_TRAFFIC_LEAK
Hub1-n1(config-ext-nacl)# deny gre any any
Hub1-n1(config-ext-nacl)# deny   ip 192.168.100.0 0.0.0.255 192.168.1.0 0.0.0.255
Hub1-n1(config-ext-nacl)# permit ip any any
```

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

```
Hub1-n1(config)# interface GigabitEthernet0/0
Hub1-n1(config-if)# ip access-group ACL_TRAFFIC_LEAK out
Hub1-n1(config-if)# exit
```

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

Подробная настройка с описанием команд приведена в базовом сценарии.

Настройка VRRP

Для «С-Терра Шлюз» версии 5.0 в данном сценарии настройка VRRP будет происходить только из cisco-like консоли. Настройка notify-common скрипта не нужна из-за возможности указать в cisco-like консоли метрику для маршрутов (появилось в 5.0). Маршрут по умолчанию, добавленный с помощью VRRP, будет более приоритетным, чем маршрут по умолчанию с метрикой, добавленный с помощью cisco-like консоли. Нода в режиме Master будет использовать маршрут по умолчанию, добавленный с помощью VRRP. Нода в режиме Backup/Fault будет использовать маршрут по умолчанию, добавленный в cisco-like консоли.

1. Настройте VRRP в соответствии со схемой стенда на внешнем `GigabitEthernet0/0`:

> Для каждого интерфейса используется свой номер VRRP instance. По умолчанию все VRRP интерфейсы синхронизированы в группу номер ноль.
> Для стабильности работы кластера не рекомендуется уменьшать `advertisement` интервал (команда: `vrrp <VRID> timers advertise <SECONDS>`) и отключать периодическую отсылку gARP пакетов (команда: `vrrp <VRID> timers garp <SECONDS>`).

1.1. Включите использование vrrp v3:

```
Hub1-n1#configure terminal
Hub1-n1(config)# fhrp version vrrp v3
```

1.2. Включите VRRP на внешнем интерфейсе (создает VRRP инстанс):

```
Hub1-n1(config)#interface GigabitEthernet0/0
Hub1-n1(config-if)# vrrp 1 address-family ipv4
Hub1-n1(config-if-vrrp)#
```

1.3. Задайте кластерный IPv4 адрес:

```
Hub1-n1(config-if-vrrp)# address 172.16.100.100 255.255.255.0
```

1.4. Установите интервал отправки advertisement-сообщений:

```
Hub1-n1(config-if-vrrp)# timers advertise 3
```

Примечание - если за время, равное трём интервалам между отправкой VRRP advertise пакетов, резервная нода не будет получать VRRP advertise пакеты, то резервная нода перейдёт в состояние Master. При большом размере конфигурационного файла время загрузки конфигурации может превысить три интервала между отправкой VRRP advertise пакетов. Например, для LSP конфигурации размером от 100000 строк время загрузки может составить 9 секунд. Во время загрузки будет использоваться политика драйвера по умолчанию (DDP), которая в данном сценарии блокирует весь трафик. VRRP advertise пакеты не будут отсылаться и резервная нода может перейти в состояние Master. После загрузки конфигурации обе ноды будут в состоянии Master и будут иметь кластерные адреса.

Для отображения количества строк в используемой LSP конфигурации используете команду “`lsp_mgr show | wc -l`”.

1.5. Установите интервал отправки gratuitous ARP-сообщений:

```
Hub1-n1(config-if-vrrp)# timers garp 5
```

1.6. Установите приоритет для VRRP инстанса:

```
Hub1-n1(config-if-vrrp)# priority 100
```

По умолчанию приоритет 100 и данная команда не будет отображаться в настройках консоли cisco-like.

1.7. Отключите режим перехвата роли Master в зависимости от приоритета:

```
Hub1-n1(config-if-vrrp)# no preempt
Hub1-n1(config-if-vrrp)#exit
```

2. Настройте VRRP в соответствии со схемой стенда на внешнем `GigabitEthernet0/1`:

```
Hub1-n1(config)#interface GigabitEthernet0/1
Hub1-n1(config-if)# vrrp 2 address-family ipv4
Hub1-n1(config-if-vrrp)# timers advertise 3
Hub1-n1(config-if-vrrp)# timers garp 5
Hub1-n1(config-if-vrrp)# no preempt
Hub1-n1(config-if-vrrp)# priority 100
Hub1-n1(config-if-vrrp)# address 192.168.100.1 255.255.255.0
Hub1-n1(config-if-vrrp)#end
```

3. Настройте VRRP маршрут, который будет добавляться в таблицу маршрутизации, когда нода будет переходить в состояние Master (в качестве src нужно указать кластерный IP адрес внешнего интерфейса):

```
Hub1-n1#configure terminal
Hub1-n1(config)#vrrp ip route 0.0.0.0 0.0.0.0 172.16.100.1 src 172.16.100.100
```

VRRP маршрут будет использоваться нодой, когда она переходит в состояние Master. При переходе ноды в состояние Backup/Fault, VRRP маршрут будет удаляться из системы. Нода будет использовать маршрут с большей метрикой через адрес 192.168.100.20. Данный маршрут был добавлен ранее в главе «Настройка сетевых параметров».

4. Примените настройки (настройки VRRP и IPsec применяются по выходу из режима конфигурирования):

```
Hub1-n1(config)#end
Hub1-n1#
```

5. Проверьте состояние VRRP:

```
Hub1-n1#show vrrp
```

```
Interface                      VRID  State   AddrFamily
-------------------------------------------------------
GigabitEthernet0/0             1     Master  IPv4
GigabitEthernet0/1             2     Master  IPv4
```

6. Убедитесь, что маршрут по умолчанию добавлен через 172.16.100.1:

```
Hub1-n1#show ip route
```

```
Codes: C - connected, S - stash runtic, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 172.16.100.1 to network 0.0.0.0

S*   0.0.0.0/0 [0/0] via 172.16.100.1, Source is 172.16.100.100
               [100/0] via 192.168.100.20
     172.16.0.0/24 is subnetted, 1 subnets
C         172.16.100.0 [0/0] is directly connected, GigabitEthernet0/0, Source is
172.16.100.10
C     192.168.100.0/24 [0/0] is directly connected, GigabitEthernet0/1, Source is
192.168.100.10
```

7. Чтобы посмотреть получившийся конфигурационный файл `/etc/keepalived/keepalived.conf` сервиса `keepalived` выполните:

При настройке VRRP через cisco-like консоль менять файл `/etc/keepalived/keepalived.conf` вручную категорически запрещено, это может привести к неработоспособности кластера.

```
Hub1-n1#run cat /etc/keepalived/keepalived.conf
```

```
global_defs {
  enable_dbus
}
vrrp_sync_group 0 {
  group {
    eth0_1
    eth1_2
  }
}
vrrp_instance eth0_1 {
  version 3
  interface eth0
  virtual_routes {
    0.0.0.0/0 via 172.16.100.1 src 172.16.100.100 dev eth0
  }
  virtual_ipaddress {
    172.16.100.100/24 label eth0:900
  }
  nopreempt
  priority 100
  advert_int 3
  garp_master_refresh 5
  virtual_router_id 1
}
vrrp_instance eth1_2 {
  version 3
  interface eth1
  virtual_ipaddress {
    192.168.100.1/24 label eth1:900
  }
  nopreempt
  priority 100
  advert_int 3
```

```
  garp_master_refresh 5
  virtual_router_id 2
}
```

Настройка трансляции адресов источника (source NAT)

Настройка sourse NAT выполняется, если устройствам из защищаемой сети или ноде в состоянии Backup/Fault необходим выход в Интернет, например, для загрузки СОС.

Трансляция адресов источника осуществляется в кластерный адрес 172.16.100.100 для трафика, выходящего с внешнего интерфейса `GigabitEthernet0/0`, который в данном документе соответствует eth0 (соответствие между именем интерфейса в нотации cisco-like и linux можно посмотреть в файле `/etc/ifaliases.cf`).

Важно:
1) порядок обработки исходящих с интерфейса пакетов: сначала source NAT, потом IPsec (зашифрование);
2) если требуется использовать метки (marks) в `iptables`, то на криптошлюзе запрещено изменять старшие 16 бит значения метки, так как они используются для внутренних нужд Продукта. Для работы с метками используйте маску (подробнее см. документ «Инструкция по настройке межсетевого экрана на «С-Терра Шлюз» при помощи утилиты «iptables»).

1. Посмотрите содержимое карты интерфейсов и найдите интерфейс в нотации linux для внешнего интерфейса `GigabitEthernet0/0`:

```
root@Hub1-n1:~# cat /etc/ifaliases.cf
```

```
interface (name="GigabitEthernet0/0" pattern="eth0")
interface (name="GigabitEthernet0/1" pattern="eth1")
interface (name="GigabitEthernet0/2" pattern="eth2")
interface (name="default" pattern="*")
```

   Видно, что интерфейсу `GigabitEthernet0/0` соответствует eth0.

2. Очистите настройки `iptables`:

```
root@Hub1-n1:~# netfilter-persistent flush
```

```
run-parts: executing /usr/share/netfilter-persistent/plugins.d/10-ipset flush
run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables flush
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables flush
run-parts: executing /usr/share/netfilter-persistent/plugins.d/35-ebtables flush
run-parts: executing /usr/share/netfilter-persistent/plugins.d/40-ipset flush
```

3. Добавьте две цепочки в таблицу NAT:

```
root@Hub1-n1:~# iptables -N NO_SOURCE_NAT -t nat
root@Hub1-n1:~# iptables -N SOURCE_NAT -t nat
```

   В цепочку NO_SOURCE_NAT будет направляться трафик, для которого **не нужно** делать source NAT. В цепочку SOURCE_NAT будет направляться трафик, для которого **нужно** делать source NAT.

4. Добавьте правила в цепочку POSTROUTING для перенаправления пакетов в цепочки NO_SOURCE_NAT, SOURCE_NAT:

```
root@Hub1-n1:~# iptables -I POSTROUTING -t nat -j NO_SOURCE_NAT -m comment --
comment 'pass traffic to chain to bypass source NAT'
root@Hub1-n1:~# iptables -A POSTROUTING -t nat -j SOURCE_NAT -m comment --comment
'pass traffic to chain to do source NAT'
```

   Логика следующая: все пакеты из цепочки POSTROUTING сначала направляются в цепочку NO_SOURCE_NAT, в которой определяется трафик, для которого **не нужно** делать source NAT. После того как трафик был определен, он прекращает свою обработку (действие ACCEPT) и, соответственно, не попадает в цепочку SOURCE_NAT. То есть в цепочку SOURCE_NAT попадает все, что не «отсеялось» в цепочке NO_SOURCE_NAT.

5. Добавьте `iptables` правила в цепочку NO_SOURCE_NAT для трафика, для которого не нужно делать source NAT:

   5.1. Выключите source NAT для локальных IKE/ESP/NAT-T пакетов:

   ```
   root@Hub1-n1:~# iptables -t nat -A NO_SOURCE_NAT -m mark --mark
   0x8000000/0x8000000 -j ACCEPT -m comment --comment 'local IKE packets'
   root@Hub1-n1:~# iptables -t nat -A NO_SOURCE_NAT -m mark --mark
   0x40000000/0x40000000 -j ACCEPT -m comment --comment 'local ESP/NAT-T packets'
   ```

   Метки 0x8000000 и 0x40000000 являются внутренними метками Продукта для локальных IKE и ESP/NAT-T пакетов соответственно.

   5.2. Выключите source NAT для пакетов протокола VRPP:

   ```
   root@Hub1-n1:~# iptables -t nat -A NO_SOURCE_NAT -p 112 -j ACCEPT -m comment --
   comment 'local VRRP packets'
   ```

   5.3. Выключите source NAT для пользовательского трафика между защищаемыми сетями:

   ```
   root@Hub1-n1:~# iptables -t nat -A NO_SOURCE_NAT -s 192.168.100.0/24 -d
   192.168.1.0/24 -j ACCEPT -m comment --comment 'traffic which must be encapsulated'
   ```

   5.4. Выключите source NAT для трафика, который должен шифроваться:

   ```
   root@Hub1-n1:~# iptables -t nat -A NO_SOURCE_NAT -p 47 -s 172.16.100.0/24 -d
   172.16.1.0/24 -j ACCEPT -m comment --comment 'traffic which must be protected by
   IPsec'
   ```

6. Добавьте `iptables` правила в цепочку SOURCE_NAT для трафика, для которого нужно делать source NAT:

   6.1. Включите source NAT для транзитного IPsec трафика, использующего порты 500/4500 протокола UDP:

   ```
   root@Hub1-n1:~# iptables -t nat -A SOURCE_NAT -p udp --sport 500 -o eth0 -j SNAT -
   -to-source 172.16.100.100:40500-40540 -m comment --comment 'transit IKE packets'
   root@Hub1-n1:~# iptables -t nat -A SOURCE_NAT -p udp --sport 4500 -o eth0 -j SNAT
   --to-source 172.16.100.100:44500-44540 -m comment --comment 'transit ESP/NAT-T
   packets'
   ```

   Нельзя, чтобы транзитный IPsec трафик использовал те же source UDP порты (500/4500), что и Продукт, так как в этом случае Продукт будет считать входящие транзитные IPsec пакеты за свои.

   6.2. Включите source NAT для трафика из подсети 192.168.100.0/24:

   ```
   root@Hub1-n1:~# iptables -t nat -A SOURCE_NAT -s 192.168.100.0/24 -o eth0 -j SNAT
   --to-source 172.16.100.100
   ```

7. Сохраните добавленные `iptables` правила (правила сохраняются в файл `/etc/iptables/rules.v4`):

   ```
   root@Hub1-n1:~# netfilter-persistent save
   ```

   ```
   run-parts: executing /usr/share/netfilter-persistent/plugins.d/10-ipset save
   run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables save
   run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables save
   run-parts: executing /usr/share/netfilter-persistent/plugins.d/35-ebtables save
   run-parts: executing /usr/share/netfilter-persistent/plugins.d/40-ipset save
   ```

8. Добавьте сервис `netfilter-persistent.service` в автозапуск, чтобы правила `iptables` применялись во время старта криптошлюза:

   ```
   root@Hub1-n1:~# systemctl enable netfilter-persistent.service
   ```

   ```
   Synchronizing state of netfilter-persistent.service with SysV service script with
   /lib/systemd/systemd-sysv-install.
   ```

```
Executing: /lib/systemd/systemd-sysv-install enable netfilter-persistent
```

9. Для просмотра текущих NAT правил воспользуйтесь командой:

```
root@Hub1-n1:~# iptables -L -n -v -t nat
```

10. Для очистки текущей таблицы трансляций сетевых соединений воспользуйтесь командой:

```
root@Hub1-n1:~# conntrack -D conntrack
```

При изменении NAT правил во время работы могут использоваться старые NAT правила. Очистка таблицы трансляций сетевых соединений позволит использовать новые правила.

11. Для просмотра текущей таблицы NAT трансляций воспользуйтесь командой:

```
root@Hub1-n1:~# conntrack -L
```

Настройки GRE-интерфейса

1. Войдите в cisco-like консоль из CLI разграничения доступа и перейдите в режим конфигурирования:

```
administrator@ Hub1-n1] configure
sterragate login: cscons
```

```
Password:
S-Terra Gate 5.0.XXXXX (amd64)
Hub1-n1#
```

2. Перейдите в режим конфигурирования:

```
Hub1-n1#configure terminal
```

```
Hub1-n1 (config)#
```

3. Создайте и настройте point-to-point GRE интерфейс `Tunnel 0`:

```
Hub1-n1(config)#interface Tunnel 0
Hub1-n1(config-if)#ip address 10.10.10.1 255.255.255.252
Hub1-n1(config-if)#tunnel source 172.16.100.100
Hub1-n1(config-if)#tunnel destination 172.16.1.2
Hub1-n1(config-if)#no tunnel path-mtu-discovery
Hub1-n1(config-if)#tunnel ignore-df
Hub1-n1(config-if)#tunnel ttl 255
Hub1-n1(config-if)#tunnel tos inherit
Hub1-n1(config-if)#no shutdown
Hub1-n1(config-if)#end
```

4. Просмотрите текущую конфигурацию созданного интерфейса:

```
Hub1-n1#show running-config
```

```
...
interface Tunnel0
 ip address 10.10.10.1 255.255.255.252
 mtu 1400
 tunnel source 172.16.100.100
 tunnel destination 172.16.1.2
 offloading tx-checksumming
 no tunnel path-mtu-discovery
 tunnel ignore-df
 tunnel ttl 255
 tunnel tos inherit
...
```

5. Убедитесь, что настройки интерфейса автоматически прописались в файл `/etc/network/interfaces`:

```
Hub1-n1#run cat /etc/network/interfaces
```

```
...
auto gre_0
```

```
iface gre_0 inet manual
st_type GRE
st_state up
st_local 172.16.100.100
st_remote 172.16.1.2
st_txchecksumoffload enabled
st_gre_ttl 255
st_gre_tos inherit
st_gre_pmtudisc off
st_gre_ignore_df on
st_mtu 1400
address 10.10.10.1
netmask 255.255.255.252
broadcast 10.10.10.3
...
```

6. Убедитесь, что в системе появился соответствующий сетевой интерфейс:

```
Hub1-n1#run ip -d link show dev gre_0
```

```
8: gre_0@NONE: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1400 qdisc noqueue state
UNKNOWN mode DEFAULT group default qlen 10000
    link/gre 172.16.100.100 peer 172.16.1.2 promiscuity 0  allmulti 0 minmtu 0
maxmtu 0
    gre remote 172.16.1.2 local 172.16.100.100 ttl 255 tos inherit nopmtudisc
ignore-df addrgenmode random numtxqueues 1 numrxqueues 1 gso_max_size 65536
gso_max_segs 65535 tso_max_size 65536 tso_max_segs 65535 gro_max_size 65536
```

Подробная настройка GRE интерфейса через cisco-like консоль приведена в инструкции Сетевые интерфейсы в «С-Терра Шлюз» в главе «GRE интерфейсы»

Настройка динамической маршрутизации OSPF

Настройка FRR

1. Перейдите в linux bash консоль:

```
Hub1-n1# exit
```

```
root@Hub1-n1:~#
```

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

```
ospfd=yes
```

3. Перезапустите сервис FRR:

```
root@Hub1-n1:~# service frr restart
```

4. Убедитесь, что демон `ospfd` запущен:

```
root@Hub1-n1:~# ps -ef | grep "ospfd"
```

```
root        484      1  0 12:34 ?        00:00:00 /usr/lib/frr/watchfrr -d -F
traditional zebra ospfd staticd
frr         543      1  0 12:34 ?        00:00:00 /usr/lib/frr/ospfd -d -F
traditional -A 127.0.0.1
```

Настройка OSPF

1. Войдите в консоль FRR:

```
root@Hub1-n1:~# vtysh
```

```
Hello, this is FRRouting (version 9.0.4).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

Hub1-n1#
```

2. Перейдите в режим конфигурации:

```
Hub1-n1# configure terminal
```

```
Hub1-n1(config)#
```

3. Включите процесс OSPF:

```
Hub1-n1(config)# router ospf
```

```
Hub1-n1(config-router)#
```

4. Задайте `router-id`:

```
Hub1-n1(config-router)# ospf router-id 172.16.100.100
```

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

```
Hub1-n1(config-router)# passive-interface default
```

6. Включите OSPF для подсети центрального офиса (192.168.100.0/24) и для подсети GRE туннелей (10.10.10.0/30):

```
Hub1-n1(config-router)# network 192.168.100.0/24 area 0
Hub1-n1(config-router)# network 10.10.10.0/30 area 0
```

7. Выйдите из режима конфигурации OSPF:

```
Hub1-n1(config-router)# exit
```

```
Hub1-n1(config)#
```

8. Включите отправку Hello пакетов на интерфейсе gre_0 и укажите тип сети:

```
Hub1-n1(config)# interface gre_0
Hub1-n1(config-if)# no ip ospf passive
Hub1-n1(config-if)# ip ospf network point-to-point
Hub1-n1(config-if)# exit
```

9. Завершите процедуру конфигурации:

```
Hub1-n1(config)# end
```

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

```
Hub1-n1# write memory
```

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

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

```
Hub1-n1# exit
root@Hub1-n1:~#
```

12. В случаях, когда OSPF задействован на большом количестве интерфейсов (10 или более), В Продукте может происходить достижение лимита MULTICAST записей на один сокет, заданного в системной переменной `net.ipv4.igmp_max_memberships` (значение по умолчанию – 20). В этих случаях может наблюдаться нарушение работоспособности OSPF (потери OSPF соседства), и в логе `/var/log/frr/frr.log` появляются сообщения об ошибке, например:

```
Mar 31 10:01:32 Hub1 ospfd[4174]: [V5G42-FE94B][EC 100663299] can't setsockopt
IP_ADD_MEMBERSHIP (fd 15, addr 10.10.9.1, ifindex 20, AllSPFRouters): No buffer
space available; perhaps a kernel limit on # of multicast group memberships has
been exceeded?
```

В таком случае следует в конфигурационный файл `/etc/sysctl.conf` добавить эту переменную с увеличенным значением, например:

```
net.ipv4.igmp_max_memberships=40
```

Для применения изменений следует выполнить следующую команду из Linux консоли криптошлюза:

```
root@Hub1-n1:~# sysctl -p
```

В общем случае, значение следует выбирать в соответствии с количеством интерфейсов, задействованных в OSPF, умноженном на два. Рекомендуемое максимальное значение для данного параметра – до 150.

Посмотреть количество текущих MULTICAST записей (в контексте OSPF) на всех интерфейсах можно следующей командой в Linux консоли криптошлюза:

```
root@Hub1-n1:~#ip -4 -o maddress | grep "224.0.0.5\|224.0.0.6" | wc -l
```

Настройка криптошлюза Hub1-n1 завершена.

В Приложении представлены тексты конфигураций для криптошлюза Hub1-n1:

- текст консоли cisco-like;
- текст LSP;
- конфигурация FRR;
- правила iptables.

Настройка криптошлюза Hub1-n2

Настройка криптошлюза Hub1-n2 производится по аналогии с Hub1-n1 с учетом соответствующих IP-адресов (см. Рисунок 1) и приоритета VRRP на интерфейсах.

В Приложении представлены тексты конфигураций для криптошлюза Hub1-n2:

- текст консоли cisco-like;
- текст LSP;
- конфигурация FRR;
- правила iptables.

Настройка криптошлюза Spoke1

Настройка криптошлюза Spoke1 производится по аналогии с Hub1-n1, за исключением настройки трансляции адресов источника и настройки VRRP.

В Приложении представлены тексты конфигураций для криптошлюза Spoke1:

- текст консоли cisco-like;
- текст LSP;
- конфигурация FRR.

Проверка работоспособности стенда

Проверка состояния кластера

Убедитесь, что основная нода Hub1-n1 находится в состоянии Master, а резервная Hub1-n2 в Backup, для этого выполните следующие команды из консоли cisco-like:

```
Hub1-n1#show vrrp
```

```
Interface                       VRID  State   AddrFamily
------------------------------------------------------
GigabitEthernet0/0              1     Master  IPv4
GigabitEthernet0/1              2     Master  IPv4
```

```
Hub1-n2#show vrrp
```

```
Interface                       VRID  State   AddrFamily
------------------------------------------------------
GigabitEthernet0/0              1     Backup  IPv4
GigabitEthernet0/1              2     Backup  IPv4
```

Видно, что нода Hub1-n1 находится в состоянии Master, нода Hub-n2 находится в состоянии Backup

Проверка IP связности

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

```
Hub1-n1#ping 172.16.100.1
```

```
PING 172.16.100.1 (172.16.100.1) 100(128) bytes of data.
108 bytes from 172.16.100.1: icmp_seq=1 ttl=64 time=1.10 ms
108 bytes from 172.16.100.1: icmp_seq=2 ttl=64 time=0.304 ms
108 bytes from 172.16.100.1: icmp_seq=3 ttl=64 time=0.547 ms
108 bytes from 172.16.100.1: icmp_seq=4 ttl=64 time=0.390 ms
108 bytes from 172.16.100.1: icmp_seq=5 ttl=64 time=0.319 ms

--- 172.16.100.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4066ms
rtt min/avg/max/mdev = 0.304/0.532/1.101/0.297 ms
```

```
Hub1-n2#ping 172.16.100.1
```

```
PING 172.16.100.1 (172.16.100.1) 100(128) bytes of data.
108 bytes from 172.16.100.1: icmp_seq=1 ttl=64 time=1.33 ms
108 bytes from 172.16.100.1: icmp_seq=2 ttl=64 time=0.322 ms
108 bytes from 172.16.100.1: icmp_seq=3 ttl=64 time=0.298 ms
108 bytes from 172.16.100.1: icmp_seq=4 ttl=64 time=0.477 ms
108 bytes from 172.16.100.1: icmp_seq=5 ttl=64 time=0.347 ms

--- 172.16.100.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4070ms
rtt min/avg/max/mdev = 0.298/0.553/1.325/0.390 ms
```

```
Spoke1#ping 172.16.1.1
```

```
PING 172.16.1.1 (172.16.1.1) 100(128) bytes of data.
108 bytes from 172.16.1.1: icmp_seq=1 ttl=64 time=0.771 ms
108 bytes from 172.16.1.1: icmp_seq=2 ttl=64 time=0.282 ms
108 bytes from 172.16.1.1: icmp_seq=3 ttl=64 time=0.276 ms
108 bytes from 172.16.1.1: icmp_seq=4 ttl=64 time=0.268 ms
108 bytes from 172.16.1.1: icmp_seq=5 ttl=64 time=0.244 ms

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

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

2. Проверьте, что с защищаемых устройств Host-b-hub1, Host-b-spoke1 доступны по ICMP соответствующие криптошлюзы. Для этого выполните команду `ping` из linux bash консоли защищаемых устройств.

```
root@Host-b-hub1:~# ping 192.168.100.1 -c 5
```

```
PING 192.168.100.1 (192.168.100.1) 56(84) bytes of data.
64 bytes from 192.168.100.1: icmp_seq=1 ttl=64 time=0.449 ms
64 bytes from 192.168.100.1: icmp_seq=2 ttl=64 time=0.430 ms
64 bytes from 192.168.100.1: icmp_seq=3 ttl=64 time=0.488 ms
64 bytes from 192.168.100.1: icmp_seq=4 ttl=64 time=0.435 ms
64 bytes from 192.168.100.1: icmp_seq=5 ttl=64 time=0.371 ms

--- 192.168.100.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 0.371/0.434/0.488/0.037 ms
```

```
root@Host-b-hub1:~# ping 192.168.100.10 -c 5
```

```
PING 192.168.100.10 (192.168.100.10) 56(84) bytes of data.
64 bytes from 192.168.100.10: icmp_seq=1 ttl=64 time=0.635 ms
64 bytes from 192.168.100.10: icmp_seq=2 ttl=64 time=0.377 ms
64 bytes from 192.168.100.10: icmp_seq=3 ttl=64 time=0.318 ms
64 bytes from 192.168.100.10: icmp_seq=4 ttl=64 time=0.410 ms
64 bytes from 192.168.100.10: icmp_seq=5 ttl=64 time=0.434 ms

--- 192.168.100.10 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 0.318/0.434/0.635/0.107 ms
```

```
root@Host-b-hub1:~# ping 192.168.100.20 -c 5
```

```
PING 192.168.100.20 (192.168.100.20) 56(84) bytes of data.
64 bytes from 192.168.100.20: icmp_seq=1 ttl=64 time=0.906 ms
64 bytes from 192.168.100.20: icmp_seq=2 ttl=64 time=0.585 ms
64 bytes from 192.168.100.20: icmp_seq=3 ttl=64 time=0.330 ms
64 bytes from 192.168.100.20: icmp_seq=4 ttl=64 time=0.456 ms
64 bytes from 192.168.100.20: icmp_seq=5 ttl=64 time=0.406 ms

--- 192.168.100.20 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4010ms
rtt min/avg/max/mdev = 0.330/0.536/0.906/0.202 ms
```

```
root@Host-b-spoke1:~# ping 192.168.1.1 -c 5
```

```
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.473 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.385 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.415 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=0.405 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=0.471 ms

--- 192.168.1.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 0.385/0.429/0.473/0.035 ms
```

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

3. Проверьте, что с криптошлюза Hub1-n1 (нода в состоянии Master) доступен по ICMP криптошлюз Spoke1 (по внешним интерфейсам). Для этого выполните команду `ping` из cisco-like консоли соответствующего криптошлюза.

```
Hub1-n1#ping 172.16.1.2
```

```
PING 172.16.1.2 (172.16.1.2) 100(128) bytes of data.
108 bytes from 172.16.1.2: icmp_seq=1 ttl=63 time=0.712 ms
```

```
108 bytes from 172.16.1.2: icmp_seq=2 ttl=63 time=0.728 ms
108 bytes from 172.16.1.2: icmp_seq=3 ttl=63 time=0.723 ms
108 bytes from 172.16.1.2: icmp_seq=4 ttl=63 time=0.760 ms
108 bytes from 172.16.1.2: icmp_seq=5 ttl=63 time=0.801 ms

--- 172.16.1.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4095ms
rtt min/avg/max/mdev = 0.712/0.744/0.801/0.032 ms
```

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

Проверка PKI

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

   ```
   Hub1-n1#run cert_mgr show
   ```

   ```
   Found 3 certificates. Found 1 CRL.
   1 Status: local   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Hub1-n1
   2 Status: trusted C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root
   CA
   3 Status: remote  C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Spoke1
   4 CRL: C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root CA
   ```

   ```
   Hub1-n2#run cert_mgr show
   ```

   ```
   Found 2 certificates. Found 1 CRL.
   1 Status: local   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Hub1-n2
   2 Status: trusted C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root
   CA
   3 CRL: C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root CA
   ```

   ```
   Spoke1#run cert_mgr show
   ```

   ```
   Found 3 certificates. Found 1 CRL.
   1 Status: local   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Spoke1
   2 Status: trusted C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root
   CA
   3 Status: remote  C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Hub1-n1
   4 CRL: C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root CA
   ```

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

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

   ```
   Hub1-n1#run cert_mgr check
   ```

   ```
   1 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Hub1-n1
   2 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root
   CA
   3 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Spoke1
   ```

   ```
   Hub1-n2#run cert_mgr check
   ```

   ```
   1 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Hub1-n2
   2 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root
   CA
   ```

   ```
   Spoke1#run cert_mgr check
   ```

   ```
   1 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Spoke1
   2 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=S-Terra CSP Test Root
   CA
   3 State: Active   C=RU,L=Zelenograd,O=S-Terra CSP,OU=RnD,CN=Hub1-n1
   ```

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

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

Проверка наличия маршрутов

1. Убедитесь в том, что на основной ноде Hub1-n1 маршрут по умолчанию через Router1, а на резервной ноде Hub1-n2 – через LAN адрес Hub1-n1. Так же убедитесь, что был получен маршрут до защищенной подсети филиала Spoke1 через gre интерфейс:

```
Hub1-n1#show ip route
```

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 172.16.100.1 to network 0.0.0.0

S*   0.0.0.0/0 [0/0] via 172.16.100.1, Source is 172.16.100.100
               [100/0] via 192.168.100.20
     10.0.0.0/30 is subnetted, 1 subnets
C       10.10.10.0 [0/0] is directly connected, Tunnel0, Source is 10.10.10.1
     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.100.0 [0/0] is directly connected, GigabitEthernet0/0, Source is
 172.16.100.10
S    192.168.1.0/24 [20/0] via 10.10.10.2
C     192.168.100.0/24 [0/0] is directly connected, GigabitEthernet0/1, Source is
192.168.100.10
```

```
Hub1-n2#show ip route
```

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.168.100.10 to network 0.0.0.0

S*   0.0.0.0/0 [100/0] via 192.168.100.10
     10.0.0.0/30 is subnetted, 1 subnets
C       10.10.10.0 [0/0] is directly connected, Tunnel0, Source is 10.10.10.1
     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.100.0 [0/0] is directly connected, GigabitEthernet0/0, Source is
 172.16.100.20
C     192.168.100.0/24 [0/0] is directly connected, GigabitEthernet0/1, Source is
192.168.100.20
```

2. Убедитесь, что на устройствах Hub1-n1 и Spoke1 маршрут до защищаемой подсети был получен при помощи протокола OSPF с помощью команды `run ip route`:

```
Hub1-n1#run ip route
```

```
default via 172.16.100.1 dev eth0 proto keepalived src 172.16.100.100
default via 192.168.100.20 dev eth1 proto st_policy metric 100
10.10.10.0/30 dev gre_0 proto kernel scope link src 10.10.10.1
172.16.100.0/24 dev eth0 proto kernel scope link src 172.16.100.10
192.168.1.0/24 nhid 24 via 10.10.10.2 dev gre_0 proto ospf metric 20
```

```
192.168.100.0/24 dev eth1 proto kernel scope link src 192.168.100.10
```

```
Spoke1#run ip route
```

```
default via 172.16.1.1 dev eth0 proto st_policy
10.10.10.0/30 dev gre_0 proto kernel scope link src 10.10.10.2
172.16.1.0/24 dev eth0 proto kernel scope link src 172.16.1.2
192.168.1.0/24 dev eth1 proto kernel scope link src 192.168.1.1
192.168.100.0/24 nhid 20 via 10.10.10.1 dev gre_0 proto ospf metric 20
```

Видно, что маршрут до защищаемой подсети 192.168.1.0/24 присутствует на Master ноде Hub1-n1, а маршрут до защищаемой подсети 192.168.100.0/24 на криптошлюзе филиала Spoke1.

Проверка построения защищенного соединения

1. Проверьте наличие IPsec туннелей на криптошлюзах Hub1 и Spoke1 с помощью команды `run sa_mgr show` из cisco-like консоли криптошлюзов. Обратите внимание на номер протокола в разделе IPsec connections, номер 47 соответствует протоколу GRE.

   Эти IPsec туннели должны появиться сразу после настройки криптошлюзов, так как построение защищенного соединения будет инициировано GRE трафиком между в связи с началом обмена информацией по протоколу OSPF.

   ```
   Hub1-n1#run sa_mgr show
   ```

   ```
   ISAKMP sessions: 0 initiated, 0 responded

   ISAKMP connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
   1 1 (172.16.100.100,500)-(172.16.1.2,500) active 1920 1952

   IPsec connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
   1 1 (172.16.100.100,*)-(172.16.1.2,*) 47 ESP tunn 192 192
   ```

   ```
   Hub1-n2#run sa_mgr show
   ```

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

   ```
   Spoke1#run sa_mgr show
   ```

   ```
   ISAKMP sessions: 0 initiated, 0 responded

   ISAKMP connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
   1 4 (172.16.1.2,500)-(172.16.100.100,500) active 2152 2120

   IPsec connections:
   Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
   1 8 (172.16.1.2,*)-(172.16.100.100,*) 47 ESP tunn 576 576
   ```

2. Инициируйте построение защищенного соединения при помощи ICMP трафика, посылаемого с устройства Host-b-spoke1 на Host-b-hub1 (можно и наоборот). Для этого выполните команду `ping` из linux bash консоли защищаемого устройства Host-b-spoke1:

   ```
   root@Host-b-spoke1:~# ping 192.168.100.100 -c 5
   ```

   ```
   PING 192.168.100.100 (192.168.100.100) 56(84) bytes of data.
   64 bytes from 192.168.100.100: icmp_seq=1 ttl=62 time=1.63 ms
   64 bytes from 192.168.100.100: icmp_seq=2 ttl=62 time=1.54 ms
   64 bytes from 192.168.100.100: icmp_seq=3 ttl=62 time=1.46 ms
   64 bytes from 192.168.100.100: icmp_seq=4 ttl=62 time=2.04 ms
   64 bytes from 192.168.100.100: icmp_seq=5 ttl=62 time=1.52 ms

   --- 192.168.100.100 ping statistics ---
   5 packets transmitted, 5 received, 0% packet loss, time 4006ms
   rtt min/avg/max/mdev = 1.464/1.638/2.041/0.208 ms
   ```

Видно, что между криптошлюзами Hub1-n1 и Spoke1 установлено защищенное IPsec соединение. Если этого не произошло, то проверьте файл журнала (команда `run less /var/log/cspvpngate.log`). При необходимости увеличьте уровень логирования (команда `logging trap debugging` в консоли cisco-like)

Проверка работы межсетевого экрана

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

```
Hub1-n1#run fwconn_mgr show
Hub1-n1#run klogview -f FW,drop
```

Проверка отказоустойчивости

Файлы журналов:

- сервис `keepalived` - `/tmp/keepalived.log`, `/tmp/keepalived_vrrp.log` (также содержимое данных лог файлов присутствует в `/var/log/cspvpngate.log`, префикс Keepalived).
- сервис `vpngate` (IKE/IPsec) - `/var/log/cspvpngate.log` (для включения расширенных лог сообщений выполните в консоли cisco-like команду `logging trap debugging`).

1. Смоделируйте отказ на основной ноде Hub1-n1, например отключите сетевой кабель от интерфейса `GigabitEthernet0/0` и убедитесь, что роль мастера перешла к Hub1-n2:

```
Hub1-n1#show interfaces GigabitEthernet 0/0
```

```
GigabitEthernet0/0 is down, line protocol is down
  Hardware address is 0050.569e.7767
  Internet address is 172.16.100.10/24
  MTU 1500 bytes
```

```
Hub1-n1#show vrrp
```

```
Interface                      VRID  State   AddrFamily
-------------------------------------------------------
GigabitEthernet0/0             1     Fault   IPv4
GigabitEthernet0/1             2     Fault   IPv4
```

```
Hub1-n2#show vrrp
```

```
Interface                      VRID  State   AddrFamily
-------------------------------------------------------
GigabitEthernet0/0             1     Master  IPv4
GigabitEthernet0/1             2     Master  IPv4
```

2. Убедитесь, что IPsec туннель между резервной нодой кластера Hub1-n2 и криптошлюзом в филиале Spoke1 успешно устанавливается. Для этого выполните следующие действия:

   2.1. Пустите трафик с устройства Host-b-spoke1 на устройство Host-b-hub1, который должен шифроваться, например:

```
root@Host-b-spoke1:~# ping 192.168.100.100 -c 5
```

```
PING 192.168.100.100 (192.168.100.100) 56(84) bytes of data.

--- 192.168.100.100 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4086ms
```

```
root@Host-b-spoke1:~# ping 192.168.100.100 -c 5
```

```
PING 192.168.100.100 (192.168.100.100) 56(84) bytes of data.
64 bytes from 192.168.100.100: icmp_seq=1 ttl=62 time=442 ms
64 bytes from 192.168.100.100: icmp_seq=2 ttl=62 time=0.811 ms
64 bytes from 192.168.100.100: icmp_seq=3 ttl=62 time=0.750 ms
64 bytes from 192.168.100.100: icmp_seq=4 ttl=62 time=0.756 ms
64 bytes from 192.168.100.100: icmp_seq=5 ttl=62 time=0.822 ms
```

```
--- 192.168.100.100 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4056ms
rtt min/avg/max/mdev = 0.750/89.065/442.187/176.561 ms
```

До момента получения на резервной ноде OSPF маршрута до сети филиала пакеты будут теряться. После получения маршрута связь восстановится.

2.2. Проверьте наличие IPsec туннелей на Hub1-n2 и Spoke1 (на Hub1-n1 IPsec туннелей не должно быть):

```
Hub1-n1#run sa_mgr show
```

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

```
Hub1-n2#run sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 17 (172.16.100.100,500)-(172.16.1.2,500) active 2252 2156

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 7 (172.16.100.100,*)-(172.16.1.2,*) 47 ESP tunn 3872 3872
```

```
Spoke1#run sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 41 (172.16.1.2,500)-(172.16.100.100,500) active 1952 1920

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 49 (172.16.1.2,*)-(172.16.100.100,*) 47 ESP tunn 3872 3872
```

3. Исправьте причину отказа на основной ноде Hub1-n1 и убедитесь, что она перешла из состояния Fault в Backup, а также проверьте наличие маршрута по умолчанию через LAN адрес Hub1-n2:

Нода Hub1-n1 не захватит роль Master (пока нода Hub1-n2 работоспособна), так как выключен `preempt mode`.

```
Hub1-n1#show interfaces GigabitEthernet 0/0
```

```
GigabitEthernet0/0 is up, line protocol is up
  Hardware address is 0050.569e.7767
  Internet address is 172.16.100.10/24
  MTU 1500 bytes
```

```
Hub1-n1#show vrrp
```

```
Interface                     VRID  State   AddrFamily
-------------------------------------------------------
GigabitEthernet0/0            1     Backup  IPv4
GigabitEthernet0/1            2     Backup  IPv4
```

```
Hub1-n1#show ip route
```

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
```

```
Gateway of last resort is 192.168.100.20 to network 0.0.0.0

S*   0.0.0.0/0 [100/0] via 192.168.100.20
     10.0.0.0/30 is subnetted, 1 subnets
C       10.10.10.0 [0/0] is directly connected, Tunnel0, Source is 10.10.10.1
     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.100.0 [0/0] is directly connected, GigabitEthernet0/0, Source is
 172.16.100.10
C     192.168.100.0/24 [0/0] is directly connected, GigabitEthernet0/1, Source is
192.168.100.10
```

Видно, что переключение с основной ноды на резервную отрабатывает корректно.

Приложение

Конфигурации криптошлюза Hub1-n1

1. Конфигурация cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity dn
crypto isakmp fragmentation
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto ipsec security-association lifetime kilobytes unlimited
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $6$RiWz4IVm$qMp6sbpYnTfSUxuaJPZ9UCq4yy8/x
heIEbSmbTfMzmbQqepMD6R1b18vHTdFQt949xNXzpRXWA9dZ/gCK70z./
aaa new-model
!
fhrp version vrrp v3
hostname Hub1-n1
enable secret 5 HGJcyG+CRmCjINGFkW48VQ==
!
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 NEW_ALGS
 match identity dn
 set identity mapping
 set policy 1
!
crypto isakmp profile OLD_ALGS
 match identity dn
 set identity mapping
 set policy 2
!
crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
object-group network SELF
 remark WAN IP
 host 172.16.100.100
 host 172.16.100.10
!
ip access-list extended FW_INPUT_WAN
 permit udp any object-group SELF eq isakmp
 permit udp any object-group SELF eq non500-isakmp
```

```
 permit icmp any object-group SELF
 permit esp any object-group SELF
 permit vrrp any host 224.0.0.18
!
ip access-list extended IPSEC_ACL_HUB1_AND_SPOKE1
 permit gre host 172.16.100.100 host 172.16.1.2
!
ip access-list extended ACL_TRAFFIC_LEAK
 deny   gre any any
 deny   ip 192.168.100.0 0.0.0.255 192.168.1.0 0.0.0.255
 permit ip any any
!
ipv6 default filtering deny
!
!
ip inspect tcp sessions-max 256000
ip inspect udp sessions-max 65536
ip inspect name INSPECT tcp
ip inspect name INSPECT udp
!
crypto map VPN 1 ipsec-isakmp
 match address IPSEC_ACL_HUB1_AND_SPOKE1
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set security-association mss-fix out
 set peer 172.16.1.2
 set pre-fragmentation
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 172.16.100.10 255.255.255.0
 ip access-group FW_INPUT_WAN in
 ip access-group ACL_TRAFFIC_LEAK out
 crypto map VPN
 ip inspect INSPECT out
 vrrp 1 address-family ipv4
  timers advertise 3
  timers garp 5
  no preempt
  priority 100
  address 172.16.100.100 255.255.255.0
  exit-vrrp
!
interface GigabitEthernet0/1
 ip address 192.168.100.10 255.255.255.0
 vrrp 2 address-family ipv4
  timers advertise 3
  timers garp 5
  no preempt
  priority 100
  address 192.168.100.1 255.255.255.0
  exit-vrrp
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
interface Tunnel0
 ip address 10.10.10.1 255.255.255.252
 mtu 1400
 tunnel source 172.16.100.100
 tunnel destination 172.16.1.2
```

```
 offloading tx-checksumming
 no tunnel path-mtu-discovery
 tunnel ignore-df
 tunnel ttl 255
 tunnel tos inherit
!
!
ip route 0.0.0.0 0.0.0.0 192.168.100.20 100
!
!
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
...
AD4F8901771632E0A0AF83

quit
!
vrrp ip route 0.0.0.0 0.0.0.0 172.16.100.1 src 172.16.100.100
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Tue Jul 15 17:45:06 2025

GlobalParameters(
    Title                          = "This LSP was automatically generated by CSP
Converter at Tue Jul 15 17:45:06 2025 (user: cscons)"
    Version                     = LSP_5_0
    CRLHandlingMode             = ENABLE
    PreserveIPsecSA             = FALSE
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 192.168.100.20
            Metric = 100
        )

)

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

```
    UDPSessionsMax = 65536
)

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

IKETransform crypto:isakmp:policy:2
(
    CipherAlg   = "G2814789CPRO1-K256-CBC-65534"
    HashAlg     = "GR341112_256TC26-65128"
    GroupID     = VKO2_1B
    RestrictAuthenticationTo = GOST_SIGN
    LifetimeSeconds = 86400
)

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

FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.100.100
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.10
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.100
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.10
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.100
        ProtocolID = 1
```

```
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.10
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.100
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.10
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 224.0.0.18
        ProtocolID = 112
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = DROP
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

FilterChain FilterChain:ACL_TRAFFIC_LEAK (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 47
        Action = DROP
        LogEventID = "ACL_TRAFFIC_LEAK"
    ),
    Filter (
        SourceIP = 192.168.100.0/24
        DestinationIP = 192.168.1.0/24
        Action = DROP
        LogEventID = "ACL_TRAFFIC_LEAK"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = "INSPECT"
        LogEventID = "ACL_TRAFFIC_LEAK"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = DROP
    ),
    Filter (
        Label = "INSPECT"
        DestinationIP = 0.0.0.0/0
        ProtocolID = 6
```

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

IdentityEntry Remote:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

IdentityEntry Local:Cert:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

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

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

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

IKERule IKERule:VPN:1
(
    IKEPeerIPFilter = 172.16.1.2
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
```

```
    Priority            = 10
)

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

FilterChain IPsecPolicy:VPN (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 172.16.100.100
        DestinationIP = 172.16.1.2
        ProtocolID = 47
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_ACL_HUB1_AND_SPOKE1"
    )
)

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

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

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

NetworkInterface (
    LogicalName = "Tunnel0"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
```

3. Конфигурация FRR:

```
frr version 9.0.4
frr defaults traditional
hostname Hub1-n1
log syslog informational
```

```
service integrated-vtysh-config
!
interface gre_0
 no ip ospf passive
 ip ospf network point-to-point
exit
!
router ospf
 ospf router-id 172.16.100.100
 passive-interface default
 network 10.10.10.0/30 area 0
 network 192.168.100.0/24 area 0
exit
!
end
```

4. Правила iptables:

```
iptables -N NO_SOURCE_NAT -t nat
iptables -N SOURCE_NAT -t nat

iptables -I POSTROUTING -t nat -j NO_SOURCE_NAT -m comment --comment 'pass traffic
to chain to bypass source NAT'
iptables -A POSTROUTING -t nat -j SOURCE_NAT -m comment --comment 'pass traffic to
chain to do source NAT'

iptables -t nat -A NO_SOURCE_NAT -m mark --mark 0x8000000/0x8000000 -j ACCEPT -m
comment --comment 'local IKE packets'
iptables -t nat -A NO_SOURCE_NAT -m mark --mark 0x40000000/0x40000000 -j ACCEPT -m
comment --comment 'local ESP/NAT-T packets'
iptables -t nat -A NO_SOURCE_NAT -p 112 -j ACCEPT -m comment --comment 'local VRRP
packets'
iptables -t nat -A NO_SOURCE_NAT -s 192.168.100.0/24 -d 192.168.1.0/24 -j ACCEPT -
m comment --comment 'traffic which must be encapsulated'
iptables -t nat -A NO_SOURCE_NAT -p 47 -s 172.16.100.0/24 -d 172.16.1.0/24 -j
ACCEPT -m comment --comment 'traffic which must be protected by IPsec'

iptables -t nat -A SOURCE_NAT -p udp --sport 500 -o eth0 -j SNAT --to-source
172.16.100.100:40500-40540 -m comment --comment 'transit IKE packets'
iptables -t nat -A SOURCE_NAT -p udp --sport 4500 -o eth0 -j SNAT --to-source
172.16.100.100:44500-44540 -m comment --comment 'transit ESP/NAT-T packets'
iptables -t nat -A SOURCE_NAT -s 192.168.100.0/24 -o eth0 -j SNAT --to-source
172.16.100.100
```

Конфигурации криптошлюза Hub1-n2

1. Конфигурация cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity dn
crypto isakmp fragmentation
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto ipsec security-association lifetime kilobytes unlimited
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $6$RiWz4IVm$qMp6sbpYnTfSUxuaJPZ9UCq4yy8/x
heIEbSmbTfMzmbQqepMD6R1b18vHTdFQt949xNXzpRXWA9dZ/gCK70z./
```

```
aaa new-model
!
fhrp version vrrp v3
hostname Hub1-n2
enable secret 5 HGJcyG+CRmCjINGFkW48VQ==
!
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 NEW_ALGS
 match identity dn
 set identity mapping
 set policy 1
!
crypto isakmp profile OLD_ALGS
 match identity dn
 set identity mapping
 set policy 2
!
crypto ipsec transform-set GOST_ENCRYPT_AND_INTEGRITY esp-gost28147-4m-imit
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
object-group network SELF
 remark WAN IP
 host 172.16.100.100
 host 172.16.100.20
!
ip access-list extended FW_INPUT_WAN
 permit udp any object-group SELF eq isakmp
 permit udp any object-group SELF eq non500-isakmp
 permit icmp any object-group SELF
 permit esp any object-group SELF
 permit vrrp any host 224.0.0.18
!
ip access-list extended IPSEC_ACL_HUB1_AND_SPOKE1
 permit gre host 172.16.100.100 host 172.16.1.2
!
ip access-list extended ACL_TRAFFIC_LEAK
 deny   gre any any
 deny   ip 192.168.100.0 0.0.0.255 192.168.1.0 0.0.0.255
 permit ip any any
!
ipv6 default filtering deny
!
!
ip inspect tcp sessions-max 256000
ip inspect udp sessions-max 65536
ip inspect name INSPECT tcp
ip inspect name INSPECT udp
!
crypto map VPN 1 ipsec-isakmp
 match address IPSEC_ACL_HUB1_AND_SPOKE1
```

```
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set security-association mss-fix out
 set peer 172.16.1.2
 set pre-fragmentation
 set dead-connection history off
!
interface GigabitEthernet0/0
 no shutdown
 ip address 172.16.100.20 255.255.255.0
 ip access-group FW_INPUT_WAN in
 ip access-group ACL_TRAFFIC_LEAK out
 vrrp 1 address-family ipv4
  timers advertise 3
  timers garp 5
  no preempt
  priority 50
  address 172.16.100.100 255.255.255.0
  exit-vrrp
 crypto map VPN
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 no shutdown
 ip address 192.168.100.20 255.255.255.0
 vrrp 2 address-family ipv4
  timers advertise 3
  timers garp 5
  no preempt
  priority 50
  address 192.168.100.1 255.255.255.0
  exit-vrrp
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
interface Tunnel0
 ip address 10.10.10.1 255.255.255.252
 mtu 1400
 tunnel source 172.16.100.100
 tunnel destination 172.16.1.2
 offloading tx-checksumming
 no tunnel path-mtu-discovery
 tunnel ignore-df
 tunnel ttl 255
 tunnel tos inherit
!
ip route 0.0.0.0 0.0.0.0 192.168.100.10 100
!
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
!
vrrp ip route 0.0.0.0 0.0.0.0 172.16.100.1 src 172.16.100.100
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Tue Jul 15 17:47:23 2025
```

```
GlobalParameters(
    Title                          = "This LSP was automatically generated by CSP
Converter at Tue Jul 15 17:47:23 2025 (user: cscons)"
    Version                    = LSP_5_0
    CRLHandlingMode            = ENABLE
    PreserveIPsecSA            = FALSE
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 192.168.100.10
            Metric = 100
        )

)

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

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

IKETransform crypto:isakmp:policy:2
(
    CipherAlg   = "G2814789CPRO1-K256-CBC-65534"
    HashAlg     = "GR341112_256TC26-65128"
    GroupID     = VKO2_1B
    RestrictAuthenticationTo = GOST_SIGN
    LifetimeSeconds = 86400
)

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

```
FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.100.100
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.20
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.100
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.20
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.100
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.20
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.100
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.100.20
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 224.0.0.18
        ProtocolID = 112
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
```

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

FilterChain FilterChain:ACL_TRAFFIC_LEAK (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 47
        Action = DROP
        LogEventID = "ACL_TRAFFIC_LEAK"
    ),
    Filter (
        SourceIP = 192.168.100.0/24
        DestinationIP = 192.168.1.0/24
        Action = DROP
        LogEventID = "ACL_TRAFFIC_LEAK"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = "INSPECT"
        LogEventID = "ACL_TRAFFIC_LEAK"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = DROP
    ),
    Filter (
        Label = "INSPECT"
        DestinationIP = 0.0.0.0/0
        ProtocolID = 6
        Action = PASS
        ExtendedAction = inspect_tcp<>
        LogEventID = "Inspect:INSPECT:tcp"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 17
        Action = PASS
        ExtendedAction = inspect_udp<>
        LogEventID = "Inspect:INSPECT:udp"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = PASS
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

IdentityEntry Remote:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

IdentityEntry Local:Cert:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

FilterChain IPv6Default (
```

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

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

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

IKERule IKERule:VPN:1
(
    IKEPeerIPFilter = 172.16.1.2
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

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

FilterChain IPsecPolicy:VPN (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 172.16.100.100
        DestinationIP = 172.16.1.2
        ProtocolID = 47
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
```

```
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_ACL_HUB1_AND_SPOKE1"
    )
)

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

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

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

NetworkInterface (
    LogicalName = "Tunnel0"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)
```

3. Конфигурация FRR:

```
frr version 9.0.4
frr defaults traditional
hostname Hub1-n2
log syslog informational
service integrated-vtysh-config
!
interface gre_0
 no ip ospf passive
 ip ospf network point-to-point
exit
!
router ospf
 ospf router-id 172.16.100.100
 passive-interface default
 network 10.10.10.0/30 area 0
 network 192.168.100.0/24 area 0
exit
!
end
```

4. Правила iptables:

```
iptables -N NO_SOURCE_NAT -t nat
iptables -N SOURCE_NAT -t nat

iptables -I POSTROUTING -t nat -j NO_SOURCE_NAT -m comment --comment 'pass traffic
to chain to bypass source NAT'
iptables -A POSTROUTING -t nat -j SOURCE_NAT -m comment --comment 'pass traffic to
chain to do source NAT'

iptables -t nat -A NO_SOURCE_NAT -m mark --mark 0x8000000/0x8000000 -j ACCEPT -m
comment --comment 'local IKE packets'
```

```
iptables -t nat -A NO_SOURCE_NAT -m mark --mark 0x40000000/0x40000000 -j ACCEPT -m
comment --comment 'local ESP/NAT-T packets'
iptables -t nat -A NO_SOURCE_NAT -p 112 -j ACCEPT -m comment --comment 'local VRRP
packets'
iptables -t nat -A NO_SOURCE_NAT -s 192.168.100.0/24 -d 192.168.1.0/24 -j ACCEPT -
m comment --comment 'traffic which must be encapsulated'
iptables -t nat -A NO_SOURCE_NAT -p 47 -s 172.16.100.0/24 -d 172.16.1.0/24 -j
ACCEPT -m comment --comment 'traffic which must be protected by IPsec'

iptables -t nat -A SOURCE_NAT -p udp --sport 500 -o eth0 -j SNAT --to-source
172.16.100.100:40500-40540 -m comment --comment 'transit IKE packets'
iptables -t nat -A SOURCE_NAT -p udp --sport 4500 -o eth0 -j SNAT --to-source
172.16.100.100:44500-44540 -m comment --comment 'transit ESP/NAT-T packets'
iptables -t nat -A SOURCE_NAT -s 192.168.100.0/24 -o eth0 -j SNAT --to-source
172.16.100.100
```

Конфигурации криптошлюза Spoke1

1. Конфигурация cisco-like:

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity dn
crypto isakmp fragmentation
crypto isakmp security-association lifetime delta 50
crypto isakmp initiator-sessions-max 100
crypto isakmp responder-sessions-max 100
crypto ipsec security-association lifetime kilobytes unlimited
crypto isakmp keepalive 3
crypto isakmp keepalive retry-count 5
username cscons privilege 15 secret 5 $6$RiWz4IVm$qMp6sbpYnTfSUxuaJPZ9UCq4yy8/x
heIEbSmbTfMzmbQqepMD6R1b18vHTdFQt949xNXzpRXWA9dZ/gCK70z./
aaa new-model
!
hostname Spoke1
enable secret 5 HGJcyG+CRmCjINGFkW48VQ==
!
crypto isakmp policy 1
 encr magma
 hash stribog-256
 authentication gost-sig
 group vko2
!
crypto isakmp profile NEW_ALGS
 match identity dn
 set identity mapping
 set policy 1
!
crypto ipsec transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY esp-magma-mgm
!
object-group network SELF
 remark WAN IP
 host 172.16.1.2
!
ip access-list extended FW_INPUT_WAN
 permit udp any object-group SELF eq isakmp
 permit udp any object-group SELF eq non500-isakmp
 permit icmp any object-group SELF
 permit esp any object-group SELF
```

```
!
ip access-list extended IPSEC_ACL_HUB1_AND_SPOKE1
 permit gre host 172.16.1.2 host 172.16.100.100
!
ip access-list extended ACL_TRAFFIC_LEAK
 deny   gre any any
 deny   ip 192.168.1.0 0.0.0.255 192.168.100.0 0.0.0.255
 permit ip any any
!
ipv6 default filtering deny
!
!
ip inspect tcp sessions-max 256000
ip inspect udp sessions-max 65536
ip inspect name INSPECT tcp
ip inspect name INSPECT udp
!
crypto map VPN 1 ipsec-isakmp
 match address IPSEC_ACL_HUB1_AND_SPOKE1
 set transform-set MAGMA_GOST_ENCRYPT_AND_INTEGRITY
 set isakmp-profile NEW_ALGS
 set security-association mss-fix out
 set peer 172.16.100.100
 set pre-fragmentation
 set dead-connection history off
!
interface GigabitEthernet0/0
 ip address 172.16.1.2 255.255.255.0
 ip access-group FW_INPUT_WAN in
 ip access-group ACL_TRAFFIC_LEAK out
 crypto map VPN
 ip inspect INSPECT out
!
interface GigabitEthernet0/1
 ip address 192.168.1.1 255.255.255.0
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
interface Tunnel0
 ip address 10.10.10.2 255.255.255.252
 mtu 1400
 tunnel source 172.16.1.2
 tunnel destination 172.16.100.100
 offloading tx-checksumming
 no tunnel path-mtu-discovery
 tunnel ignore-df
 tunnel ttl 255
 tunnel tos inherit
!
!
ip route 0.0.0.0 0.0.0.0 172.16.1.1
!
!
crypto pki trustpoint s-terra_technological_trustpoint
 revocation-check crl
 crl download group ROOTCA http://172.16.101.15/certcrl.crl
 crl download time 60
crypto pki certificate chain s-terra_technological_trustpoint
certificate 58E026BFD6D625BE4582C16C6189C183
30820227308201D4A003020102021058E026BFD6D625BE4582C16C6189C18330
```

```
...
AD4F8901771632E0A0AF83

quit
!
end
```

2. Конфигурация LSP:

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Tue Jul 15 17:49:33 2025

GlobalParameters(
    Title                          = "This LSP was automatically generated by CSP
Converter at Tue Jul 15 17:49:33 2025 (user: cscons)"
    Version                     = LSP_5_0
    CRLHandlingMode             = ENABLE
    PreserveIPsecSA             = FALSE
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 172.16.1.1
        )

)

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

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

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

```
)

FilterChain FilterChain:FW_INPUT_WAN (
    Filters = Filter (
        DestinationIP = 172.16.1.2
        ProtocolID = 17
        DestinationPort = 500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.2
        ProtocolID = 17
        DestinationPort = 4500
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.2
        ProtocolID = 1
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 172.16.1.2
        ProtocolID = 50
        Action = PASS
        LogEventID = "FW_INPUT_WAN"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = DROP
    )
    IPv6Filters = Filter (
        DestinationIP = ::/0
        Action = DROP
    ))

FilterChain FilterChain:ACL_TRAFFIC_LEAK (
    Filters = Filter (
        DestinationIP = 0.0.0.0/0
        ProtocolID = 47
        Action = DROP
        LogEventID = "ACL_TRAFFIC_LEAK"
    ),
    Filter (
        SourceIP = 192.168.1.0/24
        DestinationIP = 192.168.100.0/24
        Action = DROP
        LogEventID = "ACL_TRAFFIC_LEAK"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = "INSPECT"
        LogEventID = "ACL_TRAFFIC_LEAK"
    ),
    Filter (
        DestinationIP = 0.0.0.0/0
        Action = DROP
    ),
    Filter (
        Label = "INSPECT"
```

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

IdentityEntry Remote:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

IdentityEntry Local:Cert:NEW_ALGS(
    DistinguishedName = USER_SPECIFIC_DATA
)

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

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

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

IKERule IKERule:VPN:1
(
    IKEPeerIPFilter = 172.16.100.100
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign:NEW_ALGS
    MainModeAuthMethod  = GOST:Sign:NEW_ALGS
    DPDIdleDuration     = 3
```

```
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 10
)

IPsecAction IPsecAction:VPN:1
(
    TunnelingParameters = TunnelEntry(
        PeerAddress = 172.16.100.100
        Prefragmentation = ENABLE
    )
    ContainedProposals = ( MAGMA_GOST_ENCRYPT_AND_INTEGRITY:ESP )
    NoDeadConnectionHistory = TRUE
    MSSFixOutgoing = TRUE
    DFHandling = COPY
    IKERule = IKERule:VPN:1
)

FilterChain IPsecPolicy:VPN (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 172.16.1.2
        DestinationIP = 172.16.100.100
        ProtocolID = 47
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:VPN:1 >
        LogEventID = "IPsec:Protect:VPN:1:IPSEC_ACL_HUB1_AND_SPOKE1"
    )
)

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

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

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

NetworkInterface (
    LogicalName = "Tunnel0"
    InputFilter = IPv6Default
    OutputFilter = IPv6Default
)
```

3. Конфигурация FRR:

```
!
```

```
frr version 9.0.4
frr defaults traditional
hostname Spoke1
log syslog informational
service integrated-vtysh-config
!
interface gre_0
 no ip ospf passive
 ip ospf network point-to-point
exit
!
router ospf
 ospf router-id 172.16.1.2
 passive-interface default
 network 10.10.10.0/30 area 0
 network 192.168.1.0/24 area 0
exit
!
end
```