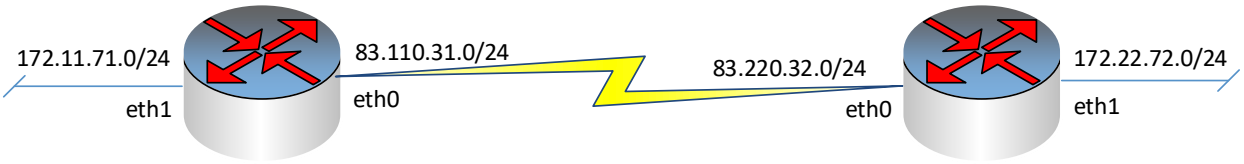
Настройка IPSec-соединения типа «Subnet-to-Subnet» по протоколу IKEv1 с взаимной двусторонней аутентификацией по сертификатам X.509

Рассмотрим простой пример настройки IPSec-соединения типа «подсеть-подсеть» по протоколу IKEv1 с взаимной аутентификацией по сертификатам X.509.

Допустим имеются два узла Dionis-NX с IP-адресами 83.110.31.11 (узел 1) и 83.220.32.22 (узел 2), а также защищаемые подсети 172.11.71.0/24 (подключена к узлу 1) и 172.22.72.0/24 (подключена к узлу 2).

IPSec-туннель «Subnet-to-Subnet»
с аутентификацией
по сертификатам X.509



Dionis-NX 1

```
!
ip mangle-group mss local-in
ip mangle-group mss local-out
ip mangle-list mss
 1 mangle adjust-mss 1400 tcp
!
interface ethernet 0
  ip address 83.110.31.11/24
 enable
!
interface ethernet 1
 ip address 172.11.71.11/24
enable
!
ip route 0.0.0.0/0 83.110.31.254
!
crypto ike config
 anti-replay window 32
 auth-log all
 crl cache
 crl fetch interval 10
 crl fetch margin 100
 crl policy strict
 debug control
 esp sync
 nat keep-alive interval 300
 nat traversal
 no unique ids
!
crypto ike enable
!
crypto ike conn t1
 auth pubkey
 auto listen
 dpd
  action close
```

Dionis-NX 2

```
!
ip mangle-group mss local-in
ip mangle-group mss local-out
ip mangle-list mss
 1 mangle adjust-mss 1400 tcp
!
interface ethernet 0
  ip address 83.220.32.22/24
 enable
!
interface ethernet 1
 ip address 172.22.72.22/24
enable
!
ip route 0.0.0.0/0 83.220.32.254
!
crypto ike config
 anti-replay window 32
 auth-log all
 crl cache
 crl fetch interval 10
 crl fetch margin 100
 crl policy strict
 debug control
 esp sync
 nat keep-alive interval 300
 nat traversal
 no unique ids
!
crypto ike enable
!
crypto ike conn t1
 auth pubkey
 auto initiate
 dpd
  action initiate
```

```
 interval 10
 timeout 30
keying tries 1
local cert router1.cer
local ip 83.110.31.11
local subnet 172.11.71.0/24
modeconfig mode pull
native-policy
no rekey
pfs mode force
ph margin fuzz 5
ph margin time 180
ph1 life time 3600
ph1 transforms
 no strict
ph2 life time 1200
ph2 transforms
 no strict
remote cert router2.cer
remote id from cert router2.cer
remote ip 83.220.32.22
remote subnet 172.22.72.0/24
send cert ifasked
type tunnel
!
crypto ike enable conn t1
!
```

```
 interval 10
 timeout 30
keying tries forever
local cert router2.cer
local ip 83.220.32.22
local subnet 172.22.72.0/24
modeconfig mode pull
native-policy
rekey
pfs mode force
ph margin fuzz 5
ph margin time 180
ph1 life time 3600
ph1 transforms
 1 add magma gost2012-512-vko-a
 strict
ph2 life time 1200
ph2 transforms
 1 add magma-4m-imit
 strict
remote cert router1.cer
remote id from cert router1.cer
remote ip 83.110.31.11
remote subnet 172.11.71.0/24
send cert always
type tunnel
!
crypto ike enable conn t1
!
```

Для организации IPSec-соединения типа «Subnet-to-Subnet» по протоколу IKEv1 с взаимной двусторонней аутентификацией по сертификатам X.509, выполним настройку Узла 1 для работы в режиме Сервера (в роли Ответчика при установлении IPSec-туннеля) и Узла 2 для работы в режиме Клиента (в роли Инициатора установления IPSec-туннеля):

Настройка узла 1.

Выполним настройку узла 1 для работы в режиме Сервера:

Импортируем сертификат удостоверяющего центра, список отозванных сертификатов, закрытый ключ для пользовательского сертификата узла 1 и соответствующий ему пользовательский сертификат узла 1, а также пользовательский сертификат узла 2 с внешнего носителя (внешнего флэш-носителя flashN[.N] или USB-токена/смарт-карты token):

```
# crypto pki import root ca cert from flash0:/certs/ca.cer
# crypto pki import crl from flash0:/certs/ca.crl
# crypto pki import key from flash0:/keys/router1.p15
# crypto pki import cert from flash0:/certs/router1.cer
# crypto pki import cert from flash0:/certs/router2.cer
```

Входим в режим конфигурации, настраиваем и запускаем службу IKE:

```
# configure terminal
(config)# crypto ike config
(config-ike)# anti-replay window 32
(config-ike)# auth-log all
(config-ike)# crl cache
(config-ike)# crl fetch interval 10
(config-ike)# crl fetch margin 100
(config-ike)# crl policy strict
(config-ike)# debug control
(config-ike)# esp sync
(config-ike)# nat keep-alive interval 300
(config-ike)# nat traversal
(config-ike)# no unique ids
(config-ike)# crypto ike enable
```

Выполним настройку соединения для работы в роли Ответчика при установлении IPSec-туннеля, назовём его «t1»:

```
(config)# crypto ike conn t1
(config-ike-conn-t1)#
```

Вид строки приглашения говорит о том, система находится в режиме редактирования настроек соединения «t1».

Задаём «слушающее» состояние соединения для работы в роли ответчика:

```
(config-ike-conn-t1)# auto listen
```

Задаём IP-адреса концов туннеля - локального и удалённого:

```
(config-ike-conn-t1)# local ip 83.110.31.11
(config-ike-conn-t1)# remote ip 83.220.32.22
```

Задаём правила отбора пакетов в IPsec-туннель локальной и удалённой защищаемых подсетей для создания IPsec-соединения типа «подсеть-подсеть»:

```
(config-ike-conn-t1)# local subnet 172.11.71.0/24
(config-ike-conn-t1)# remote subnet 172.22.72.0/24
```

Также можем включить упрощённый механизм отбора пакетов в IPsec-туннель для более оптимальной обработки трафика (по умолчанию данный механизм отключен, что эквивалентно опции «no native-policy»; использование опции «native-policy» допускается только том случае, когда не требуется выполнение фильтрации пакетов):

```
(config-ike-conn-t1)# native-policy
```

По умолчанию действует режим аутентификации по сертификатам X.509, что эквивалентно опции «auth pubkey»:

```
(config-ike-conn-t1)# auth pubkey
```

Задаём имя используемого локального сертификата:

```
(config-ike-conn-t1)# local cert router1.cer
```

Задаём имя сертификата нашего оппонента и идентификатор оппонента:

```
(config-ike-conn-t1)# remote cert router2.cer
(config-ike-conn-t1)# remote id from cert router2.cer
```

Задаём политику пересылки своего сертификата «только по требованию оппонента» для работы в роли ответчика:

```
(config-ike-conn-t1)# send cert ifasked
```

Задаём режим «Запрос/Ответ» работы фазы ModeConfig:

```
(config-ike-conn-t1)# modeconfig mode pull
```

Выполняем настройку криптопараметров IKE для работы в роли ответчика, – выключаем действующую по умолчанию строгую политику выбора криптопараметров Фазы 1 и Фазы 2 IKE опцией «no strict»:

```
(config-ike-conn-t1)# ph1 transforms
(config-ike-conn-t1-ph1)# no strict
(config-ike-conn-t1-ph1)# exit

(config-ike-conn-t1)# ph2 transforms
(config-ike-conn-t1-ph2)# no strict
(config-ike-conn-t1-ph2)# exit
```

Задаём режим совершенной прямой секретности для выработки более криптостойкого ключевого материала на фазе 2:

```
(config-ike-conn-t1)# pfs mode force
```

Задаём времена жизни туннелей/фаз IKE (при необходимости использования значений, отличных от действующих по умолчанию):

```
(config-ike-conn-t1)# ph1 life time 3600
(config-ike-conn-t1)# ph2 life time 1200
```

Отключаем продление туннеля при истечении времени жизни Фазы 1 и Фазы 2 IKE для работы в роли ответчика:

```
(config-ike-conn-t1)# no rekey
```

Настраиваем заблаговременное установление нового туннеля (при необходимости использования значений, отличных от действующих по умолчанию):

```
(config-ike-conn-t1)# ph margin fuzz 5
(config-ike-conn-t1)# ph margin time 180
```

Задаём число циклов попыток установления соединения (выполнения инициации соединения с самого начала Фазы 1 IKE) для работы в роли ответчика:

```
(config-ike-conn-t1)# keying tries 1
```

Настраиваем механизм обнаружения «умерших» оппонентов (DPD) для работы в роли ответчика:

```
(config-ike-conn-t1)# dpd
(config-ike-conn-t1-dpd)# action close
(config-ike-conn-t1-dpd)# interval 10
(config-ike-conn-t1-dpd)# timeout 30
(config-ike-conn-t1-dpd)# exit
```

По умолчанию IPsec- соединение будет работать в туннельном режиме, что эквивалентно опции «type tunnel»:

```
(config-ike-conn-t1)# type tunnel
```

Новые созданные соединения изначально находятся в выключенном состоянии. Чтобы наше соединение смогло стать активным, его необходимо включить:

```
(config)# crypto ike enable conn t1
```

Теперь на узле 1 соединение включено, находится в «слушающем» состоянии и готово к установлению IPsec-туннеля в роли ответчика.
Установление туннеля с ответчиком (сервером) на узле 1 **может быть инициировано** оппонентом (**клиентом**) на узле 2.

Настройка узла 2.

Выполним настройку узла 2 для работы в режиме Клиента:

Импортируем сертификат удостоверяющего центра, список отозванных сертификатов, закрытый ключ для пользовательского сертификата узла 2 и соответствующий ему пользовательский сертификат узла 2, а также пользовательский сертификат узла 1 с внешнего носителя (внешнего флэш-носителя flashN[.N] или USB-токена/смарт-карты token):

```
# crypto pki import root ca cert from flash0:/certs/ca.cer
# crypto pki import crl from flash0:/certs/ca.crl
# crypto pki import key from flash0:/keys/router2.p15
# crypto pki import cert from flash0:/certs/router2.cer
# crypto pki import cert from flash0:/certs/router1.cer
```

Входим в режим конфигурации, настраиваем и запускаем службу IKE:

```
# configure terminal
(config)# crypto ike config
(config-ike)# anti-replay window 32
(config-ike)# auth-log all
(config-ike)# crl cache
(config-ike)# crl fetch interval 10
(config-ike)# crl fetch margin 100
(config-ike)# crl policy strict
(config-ike)# debug control
(config-ike)# esp sync
(config-ike)# nat keep-alive interval 300
(config-ike)# nat traversal
(config-ike)# no unique ids
(config-ike)# crypto ike enable
```

Выполним настройку соединения в роли Инициатора установления IPSec-туннеля, назовём его «t1»:

```
(config)# crypto ike conn t1
(config-ike-conn-t1)#
```

Вид строки приглашения говорит о том, система находится в режиме редактирования настроек соединения «t1».

Задаём «инициирующее» состояние соединения для работы в роли инициатора:

```
(config-ike-conn-t1)# auto initiate
```

Задаём IP-адреса концов туннеля - локального и удалённого:

```
(config-ike-conn-t1)# local ip 83.220.32.22
(config-ike-conn-t1)# remote ip 83.110.31.11
```

Задаём правила отбора пакетов в IPsec-туннель локальной и удалённой защищаемых подсетей для создания IPsec-соединения типа «подсеть-подсеть»:

```
(config-ike-conn-t1)# local subnet 172.22.72.0/24
(config-ike-conn-t1)# remote subnet 172.11.71.0/24
```

Также можем включить упрощённый механизм отбора пакетов в IPsec-туннель для более оптимальной обработки трафика (по умолчанию данный механизм отключен, что эквивалентно опции «no native-policy»; использование опции «native-policy» допускается только том случае, когда не требуется выполнение фильтрации пакетов):

```
(config-ike-conn-t1)# native-policy
```

По умолчанию действует режим аутентификации по сертификатам X.509, что эквивалентно опции «auth pubkey»:

```
(config-ike-conn-t1)# auth pubkey
```

Задаём имя используемого локального сертификата:

```
(config-ike-conn-t1)# local cert router2.cer
```

Задаём имя сертификата нашего оппонента и идентификатор оппонента:

```
(config-ike-conn-t1)# remote cert router1.cer
(config-ike-conn-t1)# remote id from cert router1.cer
```

Задаём политику пересылки своего сертификата «всегда пересылать» для работы в роли инициатора:

```
(config-ike-conn-t1)# send cert always
```

Задаём режим «Запрос/Ответ» работы фазы ModeConfig:

```
(config-ike-conn-t1)# modeconfig mode pull
```

Выполняем настройку криптопараметров IKE для работы в роли инициатора, – по умолчанию действует строгая политика выбора криптопараметров фазы 1 и фазы 2, что эквивалентно опции «strict»:

Для Фазы 1 IKE зададим алгоритм шифрования (например, «MAGMA» по ГОСТ Р 34.12-2015), и алгоритм выработки общего секрета (например, «gost2012-512-vko-a» по ГОСТ Р 34.10-2012):

```
(config-ike-conn-t1)# ph1 transforms
(config-ike-conn-t1-ph1)# add magma gost2012-512-vko-a
(config-ike-conn-t1-ph1)# strict
(config-ike-conn-t1-ph1)# exit
```

Для Фазы 2 IKE зададим алгоритм шифрования для ESP (например, «magma-4m-imit» по ГОСТ Р 34.12-2015):

```
(config-ike-conn-t1)# ph2 transforms
(config-ike-conn-t1-ph2)# add magma-4m-imit
(config-ike-conn-t1-ph2)# strict
(config-ike-conn-t1-ph2)# exit
```

Задаём режим совершенной прямой секретности для выработки более криптостойкого ключевого материала на фазе 2:

```
(config-ike-conn-t1)# pfs mode force
```

Задаём времена жизни туннелей/фаз IKE (при необходимости использования значений, отличных от действующих по умолчанию):

```
(config-ike-conn-t1)# ph1 life time 3600
(config-ike-conn-t1)# ph2 life time 1200
```

Настраиваем продление туннеля при истечении времени жизни Фазы 1 и Фазы 2 IKE для работы в роли инициатора, – по умолчанию для соединения включено продление туннеля, что эквивалентно опции «rekey»:

```
(config-ike-conn-t1)# rekey
```

Настраиваем заблаговременное установление нового туннеля (при необходимости использования значений, отличных от действующих по умолчанию):

```
(config-ike-conn-t1)# ph margin fuzz 5
(config-ike-conn-t1)# ph margin time 180
```

Задаём число циклов попыток установления соединения (выполнения инициации соединения с самого начала Фазы 1 IKE) для работы в роли инициатора, – по умолчанию количество циклов не ограничено, что эквивалентно опции «keying tries forever»:

```
(config-ike-conn-t1)# keying tries forever
```

Настраиваем механизм обнаружения «умерших» оппонентов (DPD) для работы в роли инициатора:

```
(config-ike-conn-t1)# dpd
(config-ike-conn-t1-dpd)# action initiate
(config-ike-conn-t1-dpd)# interval 10
(config-ike-conn-t1-dpd)# timeout 30
(config-ike-conn-t1-dpd)# exit
```

По умолчанию IPsec-соединение будет работать в туннельном режиме, что эквивалентно опции «type tunnel»:

```
(config-ike-conn-t1)# type tunnel
```

Новые созданные соединения изначально находятся в выключенном состоянии. Чтобы наше соединение смогло стать активным, его необходимо включить:

```
(config)# crypto ike enable conn t1
```

Теперь на узле 2 соединение включено, находится в «инициирующем» состоянии и готово к установлению IPsec-туннеля в роли инициатора с ответчиком на узле 1.
Установление туннеля инициируется клиентом.

Убедиться в том, что IPSec-туннель установился успешно, можно с помощью команды «show crypto ike conn t1»:

На узле 2:

```
# show crypto ike conn t1 stats
------------------------------------------------------------------------------
t1            online
uptime: 1d 1h 5m 45s, tx: 0 B, rx: 0 B, opponent subnet: 172.11.71.0/24
------------------------------------------------------------------------------
```

На узле 1:

```
# show crypto ike conn t1 stats
------------------------------------------------------------------------------t1
online E=router2.@factor-ts.ru, O=Testing, C=RU, CN=router2
uptime: 1d 1h 5m 45s, tx: 0 B, rx: 0 B, opponent subnet: 172.22.72.0/24
------------------------------------------------------------------------------
```

Примечания:
1. При этом весь трафик из локальной и удалённой защищаемых подсетей через IPSec-соединение типа «подсеть-подсеть» между узлами 1 и 2 будет инкапсулироваться в протокол ESP.

2. В том случае, если например к узлу 1 подключены другие сети, то проходящий трафик через узел 1 к узлу 2 из этих сетей НЕ будет отбираться в IPSec-туннель и будет идти в открытом виде.