



**Мультипротокольные  
маршрутизаторы  
NSG  
Программное обеспечение NSG Linux**

**Руководство пользователя  
Часть 5  
Туннелирование  
и виртуальные частные сети (VPN)**

Версия программного обеспечения 1.0 build 3

Обновлено 05.07.2008

Москва 2008

## АННОТАЦИЯ

Данный документ содержит руководство по настройке и применению мультипротокольных маршрутизаторов NSG, оснащенных программным обеспечением NSG Linux. Руководства по применению других продуктов NSG, а также базового программного обеспечения NSG для серий NPS-7e, NSG-500, NX-300 и NSG-800 содержатся в отдельных документах.

Документ состоит из следующих разделов:

- Часть 1. Общесистемная конфигурация.
- Часть 2. Физические порты.
- Часть 3. Протоколы канального уровня. Коммутация пакетов.
- Часть 4. Маршрутизация и службы IP.
- Часть 5. Туннелирование и виртуальные частные сети (VPN).
- Часть 6. Основные команды и утилиты NSG Linux.

Пятая часть документа посвящена построению виртуальных частных сетей (VPN) на основе различных технологий и спецификаций. В устройствах NSG реализовано большое число мультипротокольных инкапсуляций, туннелей и VPN, включая современные методы обеспечения целостности данных, защиты от несанкционированного доступа, автоматическое и ручное создание безопасных туннелей через IP-сети общего пользования. Все реализации совместимы с оборудованием других производителей.

Общее описание системы, описание общесистемных параметров и командного языка системы приведены в [Части 1](#). Настройка физических интерфейсов различного типа представлена в [Части 2](#). Настройка протоколов канального уровня (в т.ч. VLAN, организация сеансового доступа средствами PPP и доступа к асинхронным портам средствами Reverse Telnet), коммутация пакетов на втором уровне (Ethernet bridging, Frame Relay) и коммутация пакетов X.25 рассмотрены в [Части 3](#). Настройка IP-маршрутизации и связанных с ней служб, а также механизмов управления IP-трафиком и обеспечения QoS, описана в [Части 4](#). В [Части 6](#) изложены начала работы с ОС Linux в объеме, желательном для администрирования и отладки сетей на основе оборудования NSG с использованием расширенных возможностей системы.

**ВНИМАНИЕ** Продукция компании непрерывно совершенствуется, в связи с чем возможны изменения отдельных аппаратных и программных характеристик по сравнению с настоящим описанием. Сведения о последних изменениях приведены в файлах README.TXT, CHANGES, а также в документации на отдельные устройства.

Замечания и комментарии по документации NSG принимаются по адресу: [doc@nsg.net.ru](mailto:doc@nsg.net.ru).

## § СОДЕРЖАНИЕ §

### Часть 5. Туннелирование и виртуальные частные сети (VPN)

§5.1. Туннелирование протоколов через сети IP .....	4
§5.1.1. Туннели GRE и IP-in-IP .....	4
§5.1.2. Механизм <i>keepalive</i> для туннелей GRE .....	6
§5.1.3. Пример построения туннеля IP-over-GRE .....	8
§5.2. Сети VPDN второго уровня .....	9
§5.2.1. Общие замечания о VPDN .....	9
§5.2.2. Клиент PPPoE .....	9
§5.2.3. Сервер PPPoE .....	11
§5.2.6. Клиент PPTP .....	11
§5.3. Общие сведения о виртуальных частных сетях третьего уровня .....	13
§5.3.1. Технология построения VPN на базе IPsec .....	13
§5.3.2. Поддерживаемые стандарты и спецификации .....	14
§5.4. Настройка IPsec .....	15
§5.4.1. Определение трафика, который должен быть защищен ( <i>access list</i> ) .....	15
§5.4.2. Определение правила преобразования трафика ( <i>transform set</i> ) .....	16
§5.4.3. Настройка туннеля ( <i>crypto map, crypto isakmp</i> ) .....	16
§5.4.4. Включение и выключение режима туннелирования на интерфейсе .....	18
§5.4.5. Просмотр информации о туннелях .....	18
§5.4.6. Особенности реализации IPsec в устройствах Cisco Systems .....	19
§5.4.7. Особенности настройки IPsec в продуктах Майкрософт .....	20
§5.4.8. Совместное использование IPsec и NAT .....	20
§5.6. Мультипротокольные инкапсуляции X.25 .....	21
§5.6.1. Инкапсуляция X.25-over-TCP/IP .....	21
§5.6.2. Инкапсуляция IP-over-X.25 .....	21
Приложение 5–А. Примеры настройки туннелей и VPN .....	23
§5–А.1. Подключение устройства NSG к серверам PPPoE .....	23
§5–А.2. Подключение клиентов PPPoE к устройству NSG .....	24
§5–А.3. Настройка туннеля PPTP .....	25
§5–А.4. Настройка статического туннеля IPsec между NSG и Cisco .....	26
§5–А.5. Настройка динамического туннеля IPsec (IKE) между NSG и Cisco .....	27
§5–А.6. Настройка динамического туннеля IPsec (IKE) между NSG и Windows .....	29
§5–А.7. Настройка динамического туннеля IPsec (IKE) между NSG и IPsec-клиентами для Windows .....	37

## §5.1. Туннелирование протоколов через сети IP

Программное обеспечение NSG Linux позволяет организовывать туннели для передачи пакетов с одним протоколом через сеть с таким же или другим протоколом. Как правило, в прикладных задачах трафик корпоративной сети туннелируется через сеть общего пользования. Данная версия NSG Linux поддерживает следующие типы туннелей:

- IP-over-IP (GRE)
- IP-in-IP (реализация, совместимая с Linux)
- PPTP (клиент, с поддержкой MPPE) см. п.5.2.6
- PPP-over-Ethernet (клиент и сервер, с поддержкой MPPE) см. пп.5.2.2, 5.2.3
- Ethernet bridge-over-IP (GRE)
- Frame Relay-over-IP (GRE)
- Generic HDLC-over-IP (расширение GRE)
- IP-over-X.25 см. п.5.6.1
- X.25-over-IP см. п.5.6.2

### §5.1.1. Туннели GRE и IP-in-IP

Управление туннелями GRE и IP-in-IP производится в меню (config-nsg)# следующими командами:

```
tunnel ip <1...255>
```

```
no tunnel ip <1...255>
```

Создание/изменение и удаление туннеля с указанным номером, соответственно. Номер туннеля используется только как локальный идентификатор в устройстве NSG и никак не связан с номером, присвоенным этому туннелю на удаленной стороне.

**ПРИМЕЧАНИЕ** Протокол GRE является самостоятельным протоколом транспортного уровня (в терминах модели OSI), работающим не поверх общеизвестных протоколов TCP или UDP, а параллельно с ними. Идентификатор протокола GRE, указываемый в заголовке IP-пакетов — 47. Для нормальной работы GRE-туннелей в системах с брандмауэрами и фильтрами необходимо разрешить прохождение пакетов IP с данным идентификатором.

Для каждого создаваемого туннеля в системе создается IP-интерфейс с именем вида tuniN. Дальнейшая настройка производится меню туннеля (config-tunnel-N)#. Меню содержит следующие команды:

```
description "<комментарий>"
```

Административное описание данного туннеля. Если строка содержит пробелы, она должна быть заключена в кавычки. Максимальная длина описания — 255 символов.

```
adm-state { up | down }
```

Административное состояние интерфейса.

```
destination-ip <ip-адрес>
```

Адрес удаленной стороны туннеля. В некоторых специальных случаях (при использовании параметра device) адрес может быть не указан, т.е. установлен в значение 0.0.0.0. Это же значение устанавливается по умолчанию.

```
source-ip <ip-адрес>
```

IP-адрес, который будет указываться в качестве источника в пакетах, отправляемых в сеть общего пользования. Этот же адрес должен быть указан в качестве назначения в пакетах, получаемых из сети общего пользования и относящихся к данному туннелю.

Если адрес не указан (значение 0.0.0.0, оно же установлено по умолчанию), то в исходящих пакетах в качестве источника указывается адрес того IP-интерфейса, с которого отправляются пакеты. Во входящих пакетах в этом случае в качестве назначения может быть указан IP-адрес любого IP-интерфейса данного устройства NSG; принадлежность пакета тому или иному туннелю (если их несколько) определяется при помощи ключа.

```
device use {<интерфейс> | none }
```

Жесткая привязка туннеля к IP-интерфейсу с указанным именем. При этом, если указанный интерфейс отсутствует или находится в состоянии down, то туннель работать не будет.

Если для туннеля не задан destination-ip, то интерфейс должен иметь тип "точка-точка".

Если для туннеля установлено значение device none (оно же установлено по умолчанию), то для него необходимо указать destination-ip. В этом случае выходной интерфейс выбирается согласно текущей таблице маршрутизации.

**ПРИМЕЧАНИЕ** В качестве транспорта для туннеля может использоваться любой IP-интерфейс, в т.ч. другой туннельный интерфейс, виртуальный многоканальный интерфейс teql, и т.п.

key { use <0...4294967295> | A.B.C.D | none }

Ключ туннеля — число длиной 32 бита. Для удобства ввода ключ может быть задан как в виде обычного десятичного числа, так и в десятично-точечной нотации. При помощи ключа реализуется слабый метод защиты туннеля, а также выбор туннеля, если их в данном устройстве несколько с совпадающими (или не определенными) IP-адресами.

По умолчанию ключ не задан (установлен в значение none).

Как можно видеть, в общем случае для туннеля должно быть установлено хотя бы одно из двух значений destination-ip или device. Если туннелей более одного, то должно быть также установлено хотя бы одно из двух значений source-ip или key.

Для туннелей с инкапсуляцией eth-br-over-gre (см. ниже), а также для работы механизма keepalive (см. след. параграф) обязательно указывать source-ip или device.

**ПРИМЕЧАНИЕ** В ряде программных и аппаратных продуктов, в т.ч. в некоторых продуктах Cisco Systems, возможна работа только одного туннеля с одинаковыми IP-адресами сторон. В программном обеспечении NSG Linux такая ситуация допускается, а туннели в этом случае различаются по ключу.

checksum { no | yes }

Обработка контрольной суммы. При установленном значении yes контрольная сумма исходящих пакетов вычисляется и указывается в заголовке GRE. Во входящих пакетах проверяется контрольная сумма, поврежденные пакеты уничтожаются.

По умолчанию проверка отключена.

pmtudisc { yes | no }

Включение/выключение функции Path MTU Discovery в туннеле. По умолчанию, а также при использовании параметра TTL, данная функция включена.

sequence-datagrams { yes | no }

Включение/выключение контроля последовательности пакетов. Данный механизм предназначен для уничтожения всех "запоздавших" пакетов. Если он включен, то в исходящие GRE пакеты добавляются порядковые номера, а во входящих — проверяются их номер и пакеты, выпадающие из последовательности, уничтожаются. Например, если пакеты получены в такой последовательности:

1 2 3 5 6 4 7 8

то пакет 4 будет удален.

По умолчанию контроль последовательности выключен. Если он включен, то включать его необходимо на обеих сторонах туннеля.

tos use { <0...255> | none }

Установка поля Type of Service для пакетов GRE-туннеля в сети общего пользования. По умолчанию принудительная установка поля TOS выключена; в этом случае поле TOS туннеля берется из туннелируемых пакетов корпоративной сети.

ttl use { <0...255> | none }

Установка поля TTL для туннелируемых пакетов корпоративной сети. Поскольку туннель сокращает количество шагов маршрутизации (*hops*) для этих пакетов, рекомендуется использовать небольшие значения — обычно 64. По умолчанию принудительная установка TTL выключена.

encapsulation { ip-over-gre | ip-over-ip | eth-br-over-ip | hdlc-over-gre }

Выбор инкапсулируемого протокола, т.е. "полезной нагрузки" туннеля, и способа инкапсуляции. В данной версии NSG Linux поддерживаются следующие варианты туннелей:

ip-over-gre	Инкапсуляция GRE, передаются пакеты IP. Туннель участвует в IP-маршрутизации наравне с остальными IP-интерфейсами.
ip-over-ip	Простая инкапсуляция IP-over-IP, используемая в Linux. (Не тождественна GRE!) Совместима с другими программными и аппаратными Linux-системами. Туннель участвует в IP-маршрутизации наравне с остальными IP-интерфейсами.
eth-br-over-gre	Инкапсуляция GRE, передаются пакеты Ethernet. Туннель должен быть включен в состав Bridge Group.
hdlc-over-gre	Инкапсуляция GRE, передаются пакеты HDLC общего вида. Как частные случаи, это могут быть пакеты Frame Relay, X.25, PPP или Cisco-HDLC.

По умолчанию используется инкапсуляция IP-over-GRE.

keepalive { <1...32000> retry <1...32000> | no }

Настройка механизма *keepalive* для туннеля. Об особенностях реализации данного механизма см. следующий параграф. Первый параметр определяет периодичность отсылки запросов *keepalive*, второй — максимально допустимое число неудачных попыток.

Далее для туннеля с инкапсуляцией `ip-over-gre` или `ip-over-ip` в меню включаются команды, общие для всех IP-интерфейсов (физических портов, Frame Relay DLCI, Ethernet VLAN и т.п.). В данном случае они определяют характеристики интерфейса, представляющего туннель с точки зрения наложенной сети:

[no] access-group ...

Настройка фильтрации IP-пакетов на данном интерфейсе. Подробно см. [Часть 4](#).

[no] crypto ...

Настройка защищенных туннелей VPN, создаваемых на данном IP-интерфейсе. В данном случае трафик частной сети IP передается внутри защищенного туннеля VPN, который, в свою очередь, проходит по туннелю IP-over-IP. Подробно о настройке VPN см. раздел 5.4.

[no] ip ... Настройка параметров протокола IP. Подробно см. [Часть 4](#).

mtu <64...18000>

Установка размера MTU для IP-пакета. Подробно см. [Часть 4](#).

nat ... Настройка трансляции сетевых IP-адресов (NAT) для данного интерфейса. Подробно см. [Часть 4](#).

[no] service-policy ...

Выбор и настройка политики управления IP-трафиком для данного интерфейса. Подробно см. [Часть 4](#).

show ...

Просмотр состояния и статистики интерфейса. Подробно см. [Часть 4](#).

Для туннеля с инкапсуляцией `eth-br-over-gre` добавляется пункт для включения его в состав Ethernet-моста:

bridge-group { <номер> | no }

Включение данного туннеля в программный мост (*bridge group*) Ethernet и исключение из него. Помимо туннелей GRE, в состав моста могут входить физические порты Ethernet, VLAN и виртуальные каналы Frame Relay. Подробно об использовании Bridge Group см. [Часть 3](#).

Для туннеля с инкапсуляцией `hdlc-over-gre` создается отдельный настраиваемый объект — виртуальный порт с именем вида `tN`, где `N` — номер туннеля. Данный объект обладает всеми протокольными параметрами, присущими синхронному порту. В частности, ему может быть назначена инкапсуляция Frame Relay, X.25, PPP Cisco-HDLC, либо Raw-HDLC. Для туннеля Frame Relay-over-IP создаются виртуальные каналы Frame Relay и т.п. В целом настройка такого виртуального порта полностью аналогична протокольной настройке физического синхронного порта, описанной в [Части 3](#).

**ПРИМЕЧАНИЕ** Туннель Generic HDLC-over-GRE представляет собой фирменное расширение стандартной инкапсуляции Frame Relay-over-GRE, состоящее в том, что для всех протоколов HDLC-семейства в заголовке пакета GRE указывается идентификатор протокола Frame Relay (по причине отсутствия других стандартных идентификаторов). В части Frame Relay-over-IP, данная реализация в устройствах NSG совместима с продуктами других производителей.

### §5.1.2. Механизм *keepalive* для туннелей GRE

Протокол GRE не предусматривает встроенного механизма *keepalive*, однако у различных производителей имеются собственные реализации этого механизма. В программном обеспечении NSG Linux используется механизм, предложенный компанией Cisco Systems; подробное описание этого алгоритма приведено в документе Cisco Systems: *GRE Tunnel Keepalives* (Document ID: 64565) и доступно по адресу: [http://www.cisco.com/en/US/tech/tk827/tk369/technologies\\_tech\\_note09186a008048cffc.shtml](http://www.cisco.com/en/US/tech/tk827/tk369/technologies_tech_note09186a008048cffc.shtml).

Суть данного механизма состоит в том, что на удаленную сторону посылается специально сформированный пакет GRE с инкапсулируемым протоколом IP. Внутри него, однако, находится не просто IP-пакет, а еще один GRE-пакет, имеющий адресом назначения IP-адрес системы-инициатора запроса. Такая конструкция не противоречит спецификации GRE, поскольку пакет GRE является частным случаем IP-пакета. В качестве идентификатора протокола в этом пакете указан 0, что позволяет отличить его от остальных пакетов GRE-туннеля.

Инициатор посылает запросы GRE *keepalive* через установленные промежутки времени. Удаленная сторона туннеля разбирает внешний пакет GRE, извлекает из него вложенный пакет и обрабатывает его в соответствии со своей таблицей маршрутизации. Поскольку этот пакет представляет собой готовый пакет GRE-туннеля, он маршрутизируется обратно инициатору. Тот, получив пакет, разбирает его заголовок, по идентификатору протокола определяет, что это не пакет с полезными данными, а ответ на *keepalive*, и считает запрос ответственным.

При отключенном механизме *keepalive* интерфейс посылает данные в туннель "наугад", не имея никакой информации о доступности и работоспособности удаленной стороны. В этом случае реализация GRE совместима с продуктами любых других сторонних производителей. Однако, чтобы туннель не превратился в "черную дыру", для контроля целостности данных следует использовать механизмы вложенных протоколов.

**ВНИМАНИЕ** Для работы механизма *keepalive* обязательно должен быть указыван `source-ip` или `device`.

**ПРИМЕЧАНИЕ** Механизм *keepalive* не реализован для туннелей типа IP-in-IP (Linux).

Пример конфигурации.

```
!  
nsg  
  tunnel ip 1  
    destination-ip 10.0.52.34  
    source-ip 10.0.52.33  
    keepalive 3 retry 5  
    encapsulation hdlc-over-gre  
  exit  
!
```

В данном случае запрос посылается каждые 3 секунды. В случае 5 неудачных запросов интерфейс `tun1` переходит в состояние DOWN. При этом удаляются маршруты через этот интерфейс и др. Пакеты с данными, поступающие от удаленной стороны туннеля, сбрасываются с сообщением "proto unreachable" (как если бы туннеля не было вовсе).

Независимо от состояния интерфейса запросы продолжают посылаться; при получении первого ответа, т.е. при восстановлении работоспособности туннеля, интерфейс переходит в состояние UP, для него восстанавливаются все маршруты и дополнительные службы.

Из сути данного механизма следует, что запрос *keepalive* формируется в рамках туннеля и ответить на него может только вторая сторона туннеля, в рамках которого он создан. При этом наличие потока данных (в любую сторону) никак не влияет на алгоритм поднятия и опускания туннеля, т.е. если ответы на *keepalive* не приходят, то интерфейс перейдет в состояние DOWN независимо от того, что данные вроде как идут.

В частности, для того, чтобы система отвечала на запросы удаленной стороны, в ней должен быть создан туннельный интерфейс, для него указано административное состояние UP и указан адрес удаленной стороны. На входящие запросы туннель отвечает всегда, даже если сам он находится в состоянии DOWN по причине неполучения ответов *keepalive* от удаленной стороны, или на нём выставлен флаг DOWN вручную (командой `ifconfig tun1 down` в командной оболочке Linux). Кроме того, прохождение пакетов GRE *keepalive* должно быть не запрещено фильтрами на обеих сторонах:

```
add 1 permit 47 <источник> <назначение>
```

**ВНИМАНИЕ**

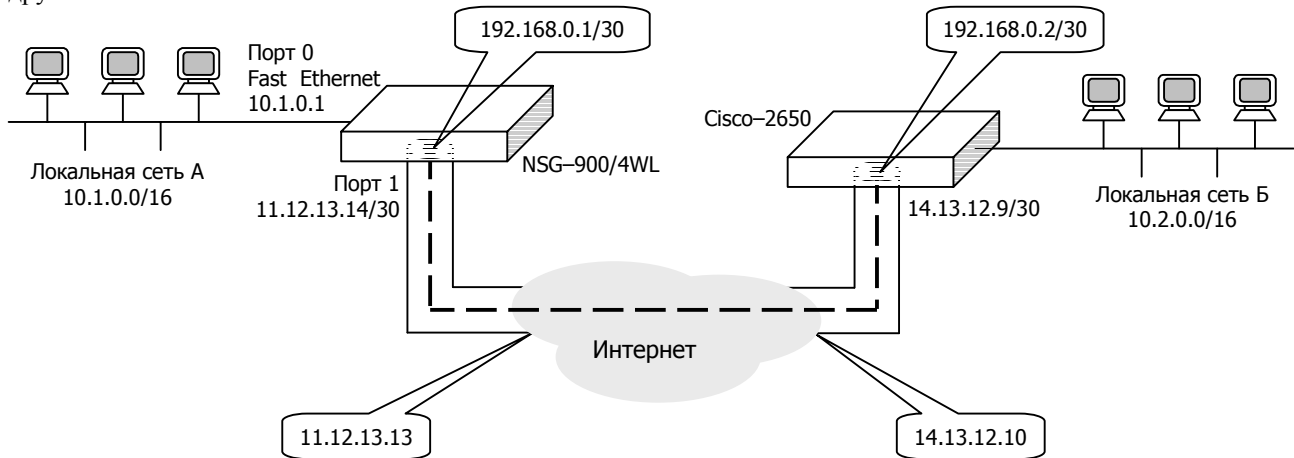
Обычные Linux-системы не поддерживают *keepalive* и не отвечают на него, несмотря на то, что туннели поддерживаются. Это следует из общего правила, гласящего, что система не может принимать пакеты, которые якобы исходят от неё самой (а именно такой прием используется в данном случае).

Маршрутизаторы Cisco Systems поддерживают ответ на *keepalive* даже в том случае, если сами не умеют его посылать, но умеют создавать туннели.

Для продуктов других производителей возможна одна из двух вышеописанных ситуаций, в также иные фирменные реализации GRE *keepalive*.

### §5.1.3. Пример построения туннеля IP-over-GRE

Схема стенда показана на рисунке. Стенд состоит из двух пограничных маршрутизаторов, соединенных через сеть общего пользования. Для наглядности на одной стороне используется устройство NSG-900, на другой — Cisco-2650.



Через маршрутизаторы связаны две приватные сети 10.1.0.0/16 и 10.2.0.0/16. Трафик этих сетей передается в туннеле между интерфейсами пограничных маршрутизаторов NSG-900 (11.12.13.14) и Cisco-2650 (14.13.12.9). При этом весь IP-пакет приватной сети, включая заголовок, передается как данные в новом пакете между двумя маршрутизаторами. На противоположной стороне пакеты корпоративной сети извлекаются из пакетов туннеля и передаются в приватную сеть. Для простоты приведены минимальные настройки, необходимые для работы туннеля. (Параметры, не относящиеся непосредственно к туннелю, опущены.)

#### Настройка NSG-900

```
!
nsg
  port eth0
    ip address 10.1.0.1/16
  tunnel ip 1
    description "tunnel NSG - CISCO"
    adm-state up
    destination-ip 14.13.12.9
    source-ip 11.12.13.14
    ip address 192.168.0.1/30
    exit
  exit
!
ip route 10.2.0.0/16 tuni1
ip route 14.13.12.9/32 11.12.13.13
!
```

#### Настройка Cisco-2650

```
!
interface FastEthernet0/0
  ip address 10.2.0.1 255.255.0.0
!
interface tunnel 0
  description "tunnel CISCO - NSG"
  tunnel mode gre ip
  tunnel destination 11.12.13.14
  tunnel source 14.13.12.9
  ip address 192.168.0.2/30
!
ip route 10.1.0.0/16 192.168.0.1
ip route 11.12.13.14 255.255.255.255 14.13.12.10
!
```



## §5.2. Сети VPDN второго уровня

### §5.2.1. Общие замечания о VPDN

Виртуальные частные сети второго уровня в данной версии NSG Linux представлены соединениями PPP-over-Ethernet (PPPoE) и Point-to-Point Tunneling Protocol (PPTP). Оба эти протокола представляют собой расширения PPP, позволяющие устанавливать соединения "точка-точка", вместо физической среды, через сети Ethernet и IP, соответственно. Оба протокола имеют ряд свойств, унаследованных от PPP:

- Соединения имеют сеансовый, а не постоянный характер.
- При установке PPP-соединения может быть выполнена аутентификация и авторизация (односторонняя или взаимная), а по мере работы — учёт работы пользователей. Для этих целей могут быть использованы локальная таблица пользователей или централизованные сервера TACACS+ (в данной версии NSG Linux не реализовано) и RADIUS.
- При установке IP-соединения могут быть согласованы параметры IP.
- При передаче трафика может использоваться сжатие с использованием различных методов, а также шифрование по протоколу MPPE (Microsoft Point-to-Point Encryption).

Второй и четвёртый пункт из этого списка полностью решают задачи, определяющие сущность VPN: аутентификацию сторон, обеспечение целостности трафика и защиту от несанкционированного доступа к данным. С учетом сеансового характера соединений, такие сети называются *Virtual Private Dial-up Networks* — VPDN.

Для организации соединений PPPoE и PPTP, как и для PPP, используется шаблон — *virtual-template*. Он содержит полную информацию о настройках виртуального интерфейса: IP-адреса и способ их назначения, способ аутентификации и т.п. Подробно о *virtual-template* см. [Часть 3](#).

Существенное отличие PPPoE и PPTP от PPP заключается в том, что эти два протокола — асимметричные, т.е. основаны на чётком разделении ролей клиента и сервера.

### §5.2.2. Клиент PPPoE

Клиент PPPoE может работать поверх как физической, так и виртуальной сети Ethernet. Меню портов Ethernet с инкапсуляцией ethernet и меню суб-интерфейсов VLAN на портах Ethernet с инкапсуляцией vlan содержит команды:

- no pppoe      Полное отключение клиента PPPoE. При этом разрывается соединение, если оно существует в данный момент, и все параметры клиента принимают значения по умолчанию (в т.ч. client disable).
- pppoe        Переход в подменю настройки клиента PPPoE.

Дальнейшая настройка производится в меню (config-pppoe)#, которое содержит следующие пункты:

- client { enable | disable }
- Разрешить клиенту PPPoE на данном порту подключиться к серверу, или запретить подключение. По умолчанию соединение запрещено. Если для клиента настроены некоторые параметры, отличные от значений по умолчанию, то при установке client disable соединения запрещаются (и разрывается существующее), но настроенные значения параметров сохраняются — в отличие от команды no pppoe в вышестоящем меню.

Если клиент PPPoE запущен, в системе создается интерфейс с именем родительского интерфейса и суффиксом ".0" например:

- eth0.0      PPPoE клиент на порту eth0
- s1.0        PPPoE клиент на порту s1 с установленным модулем расширения IM-ET10F
- eth0.101.0   PPPoE клиент на VLAN 101, работающей на физическом порту eth0

На этом же родительском интерфейсе, параллельно с пакетами PPPoE, могут отправляться и приниматься обычные пакеты Ethernet или VLAN, соответственно. Формат, в котором будет отправлен пакет ("обычный" или PPPoE), целиком определяется маршрутизацией.

В системе могут одновременно работать несколько клиентов PPPoE, при условии, что каждый из них работает на отдельном интерфейсе Ethernet или VLAN.

**ПРИМЕЧАНИЕ** В данной версии NSG Linux клиенты PPPoE могут работать на любом числе интерфейсов Ethernet, но только на одном суб-интерфейсе VLAN.

**server <имя>**

Указание имени PPPoE-сервера, к которому нужно подключиться. Имя может состоять из букв и цифр и вводится в двойных кавычках или без них. Большие и маленькие буквы в имени различаются. По умолчанию имя сервера не указано (указана пустая строка " "). В этом случае клиенту разрешается подключиться к любому серверу, доступному в данной локальной сети.

**virtual-template <номер>**

Указатель на шаблон интерфейса (virtual-template). Подробно о *virtual-template* см. [Часть 3](#).

**ppp-log { previous | current }**

Просмотр журнала сеанса PPPoE. Ключевое слово *previous* выводит журнал последней завершенной попытки, *current* — текущей попытки. Во втором случае, повторяя ввод команды, можно проследить ход сеанса по мере его выполнения.

**discovery**

Поиск доступных серверов PPPoE в данной локальной сети. Команда выполняется в течении 5 секунд. Пример вывода:

```
nsg(config-port-eth0)# pppoe discovery
```

```
Access-Concentrator: LinuxRH9
```

```
Got a cookie: 43 19 cf a8 b3 0b 07 2f ce ea 76 00 e1 14 d5 1e db 20 00 00
```

```
AC-Ethernet-Address: 00:0c:6e:41:51:b5
```

```
-----
```

```
Access-Concentrator: NSGbasicSW
```

```
Got a cookie: e8 49 c1 49 5b 94 0a d3 8c 21 d8 ef 99 5c b4 95
```

```
AC-Ethernet-Address: 00:09:56:10:05:97
```

```
-----
```

```
Access-Concentrator: CISCO
```

```
Got a cookie: cb 13 b7 11 c0 6a 92 ff fb bf 42 ea 3f 8f 5f 03
```

```
AC-Ethernet-Address: 00:02:16:66:7b:40
```

```
-----
```

```
Access-Concentrator: NSGLinux
```

```
Got a cookie: 96 79 58 ff 45 3e 8f 1b f8 af 75 3e 5b 20 90 b6 00 00 00 76
```

```
AC-Ethernet-Address: 00:09:56:12:00:fe
```

Здесь в первой строке указаны имена серверов (Access-Concentrator), в третьей строке — их MAC-адреса. В случае, если ни один из серверов не отозвался на процедуру *discovery*, будет выведено сообщение:

```
pppoe: Timeout waiting for PADO packets
```

Параметр *discovery* является командой и не сохраняется в энергонезависимой памяти.

**show**

Просмотр статуса и статистики соединения PPPoE. Пример вывода:

```
nsg(config-port-s1)# pppoe show
```

```
15: s1.0: <POINTOPOINT,MULTICAST,NOARP,UP> mtu 1500 qdisc pfifo_fast qlen 3
```

```
link/ppp
```

```
inet 17.0.0.2 peer 17.0.0.1/32 scope global s1.0
```

```
RX: bytes  packets  errors  dropped  overrun  mcast
```

```
1062997  2715    0       0        0        0
```

```
RX errors: length  crc    frame  fifo    missed
```

```
0          0      0      0      0      0
```

```
TX: bytes  packets  errors  dropped  carrier  collsns
```

```
758       19      0       0        0        0
```

```
TX errors: aborted  fifo    window  heartbeat
```

```
0          0      0       0        0
```

Здесь в первой строке важно состояние интерфейса (ключевое слово "UP"). В третьей показаны адреса — локальный (*inet*) и удаленный (*peer*) — полученные от сервера в результате согласования. Подробнее о команде *show* см. [Часть 4](#).

**ПРИМЕЧАНИЕ** При работе с PPTP-серверами компании Cisco Systems рекомендуется использовать на них следующие настройки для корректного согласования размера MTU:

```
vpdn-group <номер>
```

```
ip mtu adjust
```

а также, в некоторых случаях, для корректной обработки фрагментированных пакетов

```
no ip cef
```

### §5.2.3. Сервер PPPoE

Для работы устройства NSG в качестве сервера PPPoE (PPP-over-Ethernet) необходимо определить виртуальную частную сеть доступа (Virtual Private Dialup Network, VPDN). В данной версии NSG Linux такие сети создаются и удаляются в меню NSG следующим образом:

```
(config-nsg)# vpdn-group <1...25>
(config-nsg)# no vpdn-group <номер>
```

Первая команда создает шаблон с указанным номером, если он не существует, и осуществляет вход в меню редактирования шаблона. Одновременно в устройстве могут работать несколько серверов PPPoE, если каждый из них относится к своему порту Ethernet (или эквивалентному ему объекту).

Настройка сервера VPDN производится в меню (config-*vpdn-group*)#.

protocol pppoe

Установить протокол туннелирования PPPoE. Другие названия протоколов, которые могут упоминаться в подсказке, зарезервированы для дальнейшего использования.

pppoe name <имя>

Установить имя сервера PPPoE. По умолчанию данный параметр имеет пустое значение; в этом случае созданный сервер получает имя, указанное в параметре *hostname* устройства.

pppoe limit <число>

Установить максимальное число одновременных сеансов работы пользователей.

virtual-template <номер>

Указатель на шаблон интерфейса (*virtual-template*). Подробно о *virtual-template* см. [Часть 3](#).

Запуск и останов сервера PPPoE производится в меню порта, имеющего инкапсуляцию Ethernet (или эквивалентного ему объекта):

```
vpdn-group { <номер> | no }
```

Запустить/остановить сервер PPPoE с указанным номером на данном порту.

**ПРИМЕЧАНИЕ** Для корректной работы с удалёнными клиентами PPPoE некоторых производителей следует принудительно указать в *virtual-template* максимальный размер блока данных, принимаемый устройством NSG:

```
ppp options "mru 1492"
```

### §5.2.6. Клиент PPTP

Протокол PPTP предназначен для передачи пакетов PPP через сеть IP при помощи общего механизма GRE. Для работы этого протокола, помимо потока датаграмм, содержащих "полезную нагрузку", организуется управляющее соединение, устанавливаемое от клиента к серверу на номер порта TCP 1723. Для работы PPTP необходимо, чтобы на стороне сервера было разрешено принимать входящие TCP-пакеты и запросы на установление соединений по данному порту.

Создание туннелей PPTP производится в меню (config-nsg)# следующими командами:

```
tunnel pptp <1...255>
```

```
no tunnel pptp <1...255>
```

Создание/изменение и удаление туннеля с указанным номером, соответственно. Номер туннеля используется только как локальный идентификатор в устройстве NSG и никак не связан с номером, присвоенным этому туннелю на удаленной стороне.

Для каждого создаваемого туннеля в системе создается IP-интерфейс с именем вида *pptpN*. Дальнейшая настройка производится меню туннеля (config-tunnel-N)#. Меню содержит следующие команды:

```
description "<комментарий>"
```

Административное описание данного туннеля. Если строка содержит пробелы, она должна быть заключена в кавычки. Максимальная длина описания — 255 символов.

```
adm-state { up | down }
```

Административное состояние интерфейса.

```
server-address <ip-адрес>
```

Адрес удаленной стороны (сервера PPTP). Параметр обязательный.

**ВНИМАНИЕ** При настройке сервера PPTP необходимо обратить внимание на случай, когда туннельный интерфейс сервера является нумерованным (*numbered*). При такой конфигурации он не должен использовать IP-адрес от того внешнего интерфейса, через который входит туннель.

**source-address <ip-адрес>**

Адрес, указываемый в качестве адреса источника в пакетах, относящихся к данному туннелю.

Если параметр не задан (0.0.0.0 — значение по умолчанию), в качестве адреса источника указывается адрес того IP-интерфейса, через который пакеты уходят в сеть общего пользования. В специфических сетевых решениях, требующих некоторого определенного значения адреса источника (например, для фильтрации), данный параметр позволяет принудительно установить любой IP-адрес, принадлежащий какому-либо интерфейсу устройства.

**virtual-template <1...25>**

Указатель на шаблон интерфейса. Подробно о *virtual-template* см. [Часть 3](#).

**ppp-log { previous | current }**

Просмотр журнала сеанса РРТР. Ключевое слово *previous* выводит журнал последней завершенной попытки, *current* — текущей попытки. Во втором случае, повторяя ввод команды, можно проследить ход сеанса по мере его выполнения.

**keepalive { no | <0...3600> [retry {<1...100> | no } ] }**

Проверка целостности управляющего TCP-соединения с помощью механизма Echo Request/Reply. Первый параметр определяет интервал (в секундах) между посылкой контрольных пакетов; если значение параметра равно нулю или *no*, запросы не посылаются. При этом ответы на приходящие запросы отсылаются в любом случае (в т.ч. и при *keepalive no*).

Второй параметр устанавливает максимальное количество запросов. Если на указанное число запросов подряд не получено ни одного ответа, соединение разрывается. Суммарное время, по истечении которого интерфейс сочтет соединение неработающим и рестартует, равно произведению этих двух параметров. Значение *retry no* показывает, что разрыв соединения не производится, независимо от отсутствия ответов на запросы; такая установка целесообразна, например, если пакеты *keepalive* посылаются с единственной целью предотвратить разрыв соединения на физическом уровне из-за отсутствия трафика (переход сотовых модемов в "спящий" режим и т.п.).

При изменении параметра *keepalive* параметр *retry* автоматически принимает значение *no*. Таким образом, чтобы использовать механизм зондирования и разрыва соединения, данную команду необходимо вводить полностью.

По умолчанию установлены следующие значения параметров: *keepalive no retry no*.

**show ...** Просмотр состояния и статистики интерфейса. Подробно см. [Часть 4](#).

Особо стоит остановиться на контроле целостности соединения РРТР, поскольку он может осуществляться в трех местах: на уровне несущего соединения PPP или SLIP в сети общего пользования, на уровне соединения РРТР и в управляющем соединении РРТР. Рекомендуется использовать контроль только на одном объекте, представляющем собой наиболее слабое звено стека, а именно:

- Для соединений через сотовые сети, коммутируемые модемные линии и другие типы подключений, которые могут быть потенциально ненадежны и неустойчивы — в шаблоне несущего соединения PPP.
- Для соединений через сети Ethernet, IP-over-X.25 и т.п. надежные среды — в управляющем соединении РРТР.

**ПРИМЕЧАНИЕ** При работе с РРТР-серверами компании Cisco Systems рекомендуется использовать на них следующие настройки для корректного согласования размера MTU и фрагментации пакетов:

```
no ip cef
vpdn-group <номер>
ip mtu adjust
```

## §5.3. Общие сведения о виртуальных частных сетях третьего уровня

### §5.3.1. Технология построения VPN на базе IPsec

Программное обеспечение NSG Linux поддерживает построение виртуальных частных сетей (VPN) при помощи IP-туннелирования на основе спецификации IPsec. Настройка VPN складывается из следующих этапов:

- Определение трафика, который должен быть защищен. Производится с помощью списков доступа (*access lists*), каждый из которых имеет свой уникальный номер.
- Определение правила преобразования для этого трафика. Производится с помощью описаний правил (*transform sets*), каждое из которых также имеет уникальный номер.
- Настройка туннеля для передачи указанного трафика с указанным преобразованием. Совокупность параметров туннеля (*crypto map*) включает номер списка доступа и номер правила преобразования, а также параметры, относящиеся к собственно соединению между двумя шлюзами VPN. Туннель может быть создан одним из двух способов:
  - Постоянно существующий туннель. Для создания постоянного туннеля используется ручное согласование ключей, используемых обоими шлюзами.
  - Динамически создаваемый туннель. Для создания такого туннеля используется процедура автоматического согласования ключей, определенная протоколом IKE (Internet Key Exchange).
- Подключение созданного туннеля к одному из IP-интерфейсов маршрутизатора.

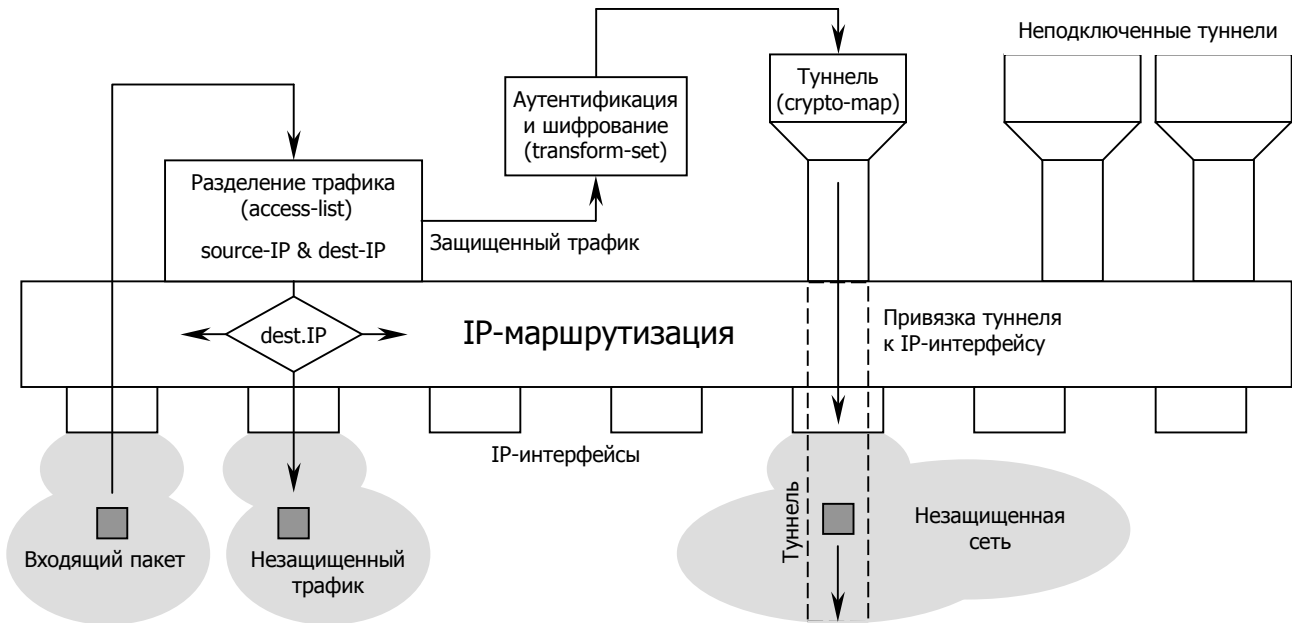
Постоянный и динамически создаваемый туннели имеют следующие отличия:

Постоянный	Динамический
Создается в результате ручного конфигурирования и существует до тех пор, пока не будет удален из конфигурации устройства.	Подготавливается (!) в результате конфигурирования устройства. Создается автоматически по инициативе какой-либо из участвующих сторон (VPN-шлюзов), если на этом устройстве имеется пакет, который должен быть направлен в данный туннель.
Ключи аутентификации и/или шифрования, индекс туннеля (SPI) и правило преобразования выбираются администратором и должны быть строго согласованы (одинаковы) с удаленной стороной.	Ключи аутентификации и/или шифрования, индекс туннеля (SPI) и конкретное правило преобразования выбираются VPN-шлюзами автоматически.
Удаляется из конфигурации устройства вручную администратором.  Никакие способы автоматического разрыва туннеля или изменения его параметров (ключи аутентификации и/или шифрования, индекс туннеля) в процессе его работы невозможны.	Созданный туннель может быть разорван любой из участвующих сторон в произвольный момент времени. Разрыв туннеля может происходить, в частности, в следующих случаях: <ul style="list-style-type: none"> <li>— Истечение установленного времени неактивности;</li> <li>— Истечение установленного срока жизни (по времени или объему переданного трафика);</li> <li>— Вручную по инициативе администратора одного из VPN-шлюзов, при помощи соответствующей команды.</li> </ul> После разрыва туннель остается в состоянии готовности и может быть автоматически создан снова по инициативе одной из сторон. После восстановления туннель будет обладать другими параметрами (ключи аутентификации и/или шифрования, индекс туннеля).

Этапы обработки обычного и защищенного трафика в VPN-маршрутизаторе показаны на рисунке на следующей странице. Уместно заметить, что с теоретической точки зрения работа VPN представляет собой расширенный, двухступенчатый вариант маршрутизации, внутрь которого дополнительно вставлено шифрование. Именно, разделение пакетов на трафик VPN и обычный трафик можно рассматривать как статическую маршрутизацию по совокупности IP-адресов источника и назначения; если пакет не подпадает под правила, определенные в *access-list*, то далее к нему применяется обычная маршрутизация по IP-адресу назначения. Защищенный трафик шифруется и передается на вход туннеля; привязка туннеля к IP-интерфейсу есть не что иное, как совокупность двух правил маршрутизации: для исходящих пакетов — фиксированной, для входящих — по номеру интерфейса и IP-адресу источника. Эти два правила являются статическими для постоянных туннелей и динамическими — для динамических туннелей.

**ВНИМАНИЕ**

Весь механизм VPN и структура команд VPN существенно переработаны, начиная с версии NSG Linux 1.0.0 beta 1. Пользователям более ранних версий NSG Linux настоятельно рекомендуется перейти на актуальную версию и при настройке руководствоваться данным документом.



Этапы обработки трафика в VPN-маршрутизаторе

### §5.3.2. Поддерживаемые стандарты и спецификации

Технология VPN реализована в устройствах NSG на основании следующих документов IETF:

Архитектура безопасности для протокола IP:

Security Architecture for the Internet Protocol RFC 2401 (RFC 1825)

Описание протокола AH (Authentication Header):

IP Authentication Header RFC 2402 (RFC 1826)

Вспомогательные рекомендации:

IP Authentication using Keyed MD5 RFC 1828

The Use of HMAC-MD5-96 within ESP and AH RFC 2403

The Use of HMAC-SHA-1-96 within ESP and AH RFC 2404

HMAC: Keyed-Hashing for Message Authentication RFC 2104

HMAC-MD5 IP Authentication with Replay Prevention RFC 2085

The Use of HMAC-RIPEMD-160-96 within ESP and AH RFC 2857

Описание протокола ESP (Encapsulating Security Payload):

IP Encapsulating Security Payload (ESP) RFC 2406 (RFC 1827)

Вспомогательные рекомендации:

The ESP DES-CBC Transform RFC 1829

The ESP DES-CBC Cipher Algorithm With Explicit IV RFC 2405

The ESP CBC-Mode Cipher Algorithms RFC 2451

The NULL Encryption Algorithm and Its Use With IPsec RFC 2410

Описание протокола IKE Internet Key Exchange):

The Internet Key Exchange (IKE) RFC 2409

Вспомогательные рекомендации:

The Internet IP Security Domain of Interpretation for ISAKMP RFC 2407

Internet Security Association and Key Management Protocol (ISAKMP) RFC 2408

The OAKLEY Key Determination Protocol RFC 2412

Взаимосвязь между стандартами и спецификациями:

IP Security Document Roadmap RFC 2411

Реализация VPN в устройствах NSG совместима с аппаратными и программными решениями других производителей, соответствующей указанным стандартам.

## §5.4. Настройка IPsec

### §5.4.1. Определение трафика, который должен быть защищен (*access list*)

Пакеты направляются в безопасный туннель в том случае, если их IP-адреса источника и назначения находятся в заданных диапазонах. Для такой сортировки трафика в NSG Linux используются расширенные списки доступа (*access lists*) NSG, которые назначаются и удаляются при помощи следующих команд в меню (config-nsg)#:

```
access-list ext-ip <номер>
no access-list ext-ip <номер>
```

Создать *access list* и войти в меню его редактирования, и удалить *access list*, соответственно. Номер расширенного *access list* может находиться в диапазоне 100...199 или 2000...2699.

Редактирование списка производится в меню (config-access-...-NN)# при помощи команд:

```
add <приоритет> { permit | deny } ip <источник> <назначение> ...
```

Добавить запись в *access list*. Запись относится ко всем IP-пакетам независимо от протокола, инкапсулируемого в IP. При этом параметры <источник> и <назначение> определяют два множества IP-адресов (*source addresses* и *destination addresses*, соответственно), на которые должна действовать данная запись. Оба эти параметра могут быть записаны в одном из следующих форматов:

<ip-адрес> <шаблон>	совокупность базового IP-адреса и шаблона ( <i>wildcard bits</i> )
host <ip-адрес>	одиночный IP-адрес
any	любой IP-адрес

```
delete <приоритет>
```

Удалить запись с заданным номером из *access list*.

```
description <комментарий>
```

Ввести текстовое описание (строка) для данного *access list*. Если строка содержит пробелы, она должна быть заключена в кавычки. Максимальная длина описания — 255 символов.

**ПРИМЕЧАНИЕ** Шаблон адреса (*wildcard bits*) содержит единицы в тех битах, значения которых могут варьироваться. Иногда он называется также *инверсией маски*; однако в общем случае это более широкое понятие, поскольку нулевые/ненулевые биты могут чередоваться в нем произвольным образом. Примеры:

- шаблон 0.0.0.7 соответствует маске длиной 29 бит (255.255.255.248);
- шаблон 0.0.0.3 соответствует маске длиной 30 бит (255.255.255.252);
- шаблон 0.0.0.5 не может быть описан никакой маской (в последнем байте допускается изменение 0 и 2 битов, но не допускается изменение битов 1 и 3...7).

Подробнее о списках доступа см. [Часть 4](#) данного руководства. Пример:

```
access-list ext-ip 151
add 1 permit ip 11.0.0.0 0.255.255.255 12.0.0.0 0.255.255.255
```

Под данное правило (номер 151) подпадает любой трафик, посылаемый из сети 11.0.0.0 с маской 255.0.0.0 в сеть 12.0.0.0 с маской 255.0.0.0.

**ВНИМАНИЕ** *Access-list*, предназначенный для отбора трафика в безопасный туннель, может содержать только одну запись. Это принципиальная особенность реализации VPN в NSG Linux. Если требуется защитить трафик для нескольких пар "источник — назначение", то для каждой из них следует создать отдельный *access-list* и отдельную *crypto map* (см. ниже.)

### §5.4.2. Определение правила преобразования трафика (*transform set*)

На данном шаге устанавливается тип протокола, который применяется для организации туннеля, вариант используемой аутентификации и размер ключа шифрования. В данной версии NSG Linux реализован только протокол ESP с шифрованием 3DES (с аутентификацией либо без нее). Правила преобразования трафика создаются и удаляются в меню `(config-nsg)#` следующими командами:

```
crypto transform-set <имя> esp { 3des-md5-hmac | 3des-sha-hmac }
no crypto transform-set <имя>
```

Имя правила преобразования — текстовая строка длиной до 15 символов. `esp` является в данной реализации ключевым словом, поскольку это единственный выбор. Последний параметр определяет только алгоритм аутентификации:

<code>3des-md5-hmac</code>	Шифрование Triple DES (168 бит) с аутентификацией MD5
<code>3des-sha-hmac</code>	Шифрование Triple DES (168 бит) с аутентификацией SHA

Другие алгоритмы туннелирования, предусмотренные IPsec, не поддерживаются ввиду их неактуальности.

### §5.4.3. Настройка туннеля (*crypto map, crypto isakmp*)

Процедура создания постоянного туннеля включает в себя: назначение индекса туннеля (SPI) и определение секретного ключа; установку ссылок на соответствующие правило преобразования и диапазон адресов защищаемого трафика; определение IP-адреса конечной точки туннеля (интерфейса маршрутизатора, работающего в паре с данным) и маршрута к нему. Описание туннеля создается и удаляется в меню `(config-nsg)#` следующими командами:

```
crypto map <имя> <1...10>
no crypto map <имя> <1...10>
```

Создать или подготовить туннель с указанными именем и приоритетом (от 1 до 10) и войти в меню его настройки, и удалить туннель, соответственно.

**ПРИМЕЧАНИЕ** Если некоторый интерфейс устройства является точкой начала нескольких туннелей, то все описания этих туннелей должны иметь одинаковое имя. Все описания туннелей рассматриваются в порядке убывания приоритета. Меньший номер соответствует более высокому приоритету.

**ПРИМЕЧАНИЕ** Данная версия NSG Linux содержит 4 интерфейса IPsec, на каждом из которых возможно создать до 10 туннелей. Таким образом, максимальное число удаленных узлов, которые могут быть обслужены одним устройством, составляет 40, причем через различные физические интерфейсы или суб-интерфейсы (DLCI, VLAN). Данные ограничения являются временными и могут быть изменены в последующих версиях NSG Linux.

Дальнейшая настройка туннеля осуществляется в подменю `(config-crypto-map-XXX)#`.

```
method { ipsec-manual | ipsec-isakmp }
```

Установить тип создаваемого туннеля:

<code>ipsec-manual</code>	Постоянный туннель с ручным назначением ключей (устанавливается по умолчанию).
<code>ipsec-isakmp</code>	Динамически создаваемый туннель с автоматическим согласованием ключей.

```
match address <номер access-list>
```

Определить закрываемый трафик при помощи соответствующего *access-list* (см. п.5.4.1).

```
set peer <ip-адрес>
```

Установить IP-адрес маршрутизатора, работающего в паре с данным. Между двумя пограничными маршрутизаторами образуется безопасный туннель через сеть общего пользования, по которому передается закрываемый трафик. Чтобы отменить установленный адрес парного маршрутизатора, следует установить его в значение 0.0.0.0.

```
set transform-set <имя правила>
```

Для постоянного туннеля — установить правило преобразования трафика с заданным номером (протокол ESP, шифрование 3DES, аутентификация MD5, SHA либо отсутствует).

Для динамически создаваемого туннеля — установить набор алгоритмов, которые будут предлагаться удаленной стороне (или выбираться из предложенных) при создании туннелей. При этом, если задан туннель без аутентификации, то только такой туннель и может быть установлен. Если задана аутентификация, то два VPN-шлюза выбирают алгоритм MD5 или SHA, исходя из перечней разрешенных алгоритмов и их приоритетов, установленных на каждой из сторон. Если в двух этих списках не найдется ни одного алгоритма, который было бы разрешено использовать обоим устройствам, туннель не будет создан.



**set nexthop <ip-адрес>**

Установить/отменить IP-адрес следующего маршрутизатора в открытой сети на пути следования туннеля. Именно через него будут отправляться пакеты, принадлежащие к данному туннелю (де-факто при этом создается маршрут к удаленной стороне туннеля с длиной маски 32 бита). Чтобы удалить этот маршрут, следует установить `nexthop` равным 0.0.0.0.

Параметр является обязательным, если создаваемый туннель должен работать через широкополосный интерфейс (Ethernet, VLAN и т.п.), а удаленный VPN-шлюз находится в другой сети за одним или несколькими маршрутизаторами.

Если удаленный VPN-шлюз расположен в непосредственно подключенной сети, то указание `nexthop` не требуется.

Если создаваемый туннель должен работать на интерфейсе типа "точка-точка" (например, на PPP-интерфейсе сотового соединения CDMA или GPRS), то `nexthop` должен быть установлен в значение 0.0.0.0. Вместо него безусловно используется адрес удаленной стороны этого соединения.

**set lifetime <1200...28800>**

Только для динамически создаваемого туннеля: установить ограничение на время жизни туннеля, в секундах. По истечении указанного времени туннель принудительно разрывается и может быть установлен заново по инициативе любой из сторон. После переустановления туннель будет использовать другие ключи шифрования и может получить другой индекс.

**set pfs { yes | no }**

Только для динамически создаваемого туннеля: включить режим повторного согласования длины ключа на этапе Quick Mode. Поддерживаются группы Диффи-Хеллмана 2 либо 5 (1024 и 1536 бит, соответственно). Необходимо для совместимости с IPsec-шлюзами других производителей, если они требуют такого согласования. (Подробнее см. п.5.4.6). По умолчанию установлено значение `no` (не согласовывать). В предыдущих версиях NSG Linux согласование также отключено.

**set session-key esp <256...4294967295> cipher <ключ-С> authenticator <ключ-А>**

Только для постоянного туннеля: установить параметры для протокола ESP с аутентификацией (т.е. 3des-md5-hmac или 3des-sha-hmac):

<256...4294967295> Индекс туннеля (Security Parameter Index, SPI)

<ключ-С> Ключ, используемый для шифрования

<ключ-А> Ключ, используемый для аутентификации

Все три указанные параметра должны быть установлены одинаковыми на обеих сторонах туннеля.

Правила назначения ключей для постоянного туннеля:

- SPI и каждый из ключей устанавливаются в одинаковое значение на обоих концах туннеля.
- Для задания ключа используются шестнадцатеричные цифры 0..9, A..F.
- Размер ключа, используемого для вычисления хэш-функции при аутентификации (<ключ-А>):
  - для варианта MD5 — 32 шестнадцатеричные цифры (128 бит)
  - для варианта SHA-1 — 40 шестнадцатеричных цифр (160 бит)
- Размер ключа, используемого для шифрования (<ключ-С>) — 48 шестнадцатеричных цифр (используется 168 бит)

**ПРИМЕЧАНИЕ** Если какой-либо из вышеперечисленных параметров описания туннеля (за исключением `set nexthop`) не определен, то данное описание туннеля будет неработоспособно, о чем будет сообщено при включении на интерфейсе режима туннелирования (см. следующий параграф).

Для динамически создаваемого туннеля, вместо назначения ключей, необходимо определить динамически же создаваемую ассоциацию безопасности — *security association* (SA). В рамках этой ассоциации осуществляется согласование SPI, ключей, алгоритмов и других параметров создаваемого туннеля между пограничными маршрутизаторами (шлюзами). Ассоциация описывается следующими атрибутами:

- Способ взаимной аутентификации двух сторон. В данной версии NSG Linux поддерживается только механизм разделяемого секрета — *preshared key* (PSK).
- Стойкость шифра для обмена ключами (группа Диффи-Хеллмана) — 2 либо 5
- Шифрование — обязательное, 3DES
- Аутентификация — обязательная, SHA-1 либо MD5

При этом единственный параметр, задаваемый на устройстве NSG административно — это собственно PSK (разделяемый секрет). Команда определения PSK для конкретной SA находится в меню конфигурирования (`config-nsg`)# и имеет вид:

```
crypto isakmp key <psk> address <удаленный_ip> <локальный_ip>
no crypto isakmp key <psk>
```

Создать/удалить PSK со следующими параметрами:

<psk>	Произвольный набор символов (строка) — разделяемый секрет. Он должен быть установлен одинаковым на обоих маршрутизаторах.
<удаленный_ip>	IP-адрес интерфейса удаленного маршрутизатора, с которым образуется SA.
<локальный_ip>	IP-адрес интерфейса данного маршрутизатора, который будет участвовать в создании SA.

#### §5.4.4. Включение и выключение режима туннелирования на интерфейсе

После того, как определены параметры туннеля (постоянного) или правила для его создания (динамического), туннель может быть подключен к некоторому интерфейсу маршрутизатора. Для включения/выключения туннеля необходимо войти в подменю порта, VLAN или DLCI, на котором начинается туннель.

```
crypto map <имя>
```

Включить на интерфейсе режим туннелирования и определить для него все туннели с указанным именем *crypto map*. При этом будут созданы все туннели, определяемые *crypto maps* типа *ipsec-manual*. Все туннели, определяемые *crypto maps* типа *ipsec-isakmp*, окончательно подготовлены для создания и будут реально создаваться либо при поступлении данных, предназначенных для отправки в этот туннель, либо по инициативе удаленной стороны.

```
no crypto map
```

Выключить на интерфейсе режим туннелирования. При этом разрываются все существующие на нем туннели — как статические, так и динамические. Кроме того, от интерфейса отключаются также все определенные для него правила создания динамических туннелей. (Однако сами правила при этом сохраняются в конфигурации устройства.)

```
crypto show
```

Показать состояние и статистику всех туннелей (как статических, так и динамических) и SA, установленных на данном порту, VLAN или DLCI.

```
crypto clear
```

Разорвать все SA, установленные на данном порту, VLAN или DLCI. Динамически созданная безопасная ассоциация (SA) может быть разорвана по инициативе любой из участвующих сторон. В дальнейшем она может быть создана вновь. Данная команда разрыва воздействует только на динамически созданные SA. Туннели, созданные вручную (*ipsec-manual*) сохраняются до тех пор, пока не будут удалены вручную же командой *no crypto map*.

Де-факто при подключении туннелей выполняется следующая процедура. Все туннели с одним именем подключены к внутреннему служебному интерфейсу IPsec (таких в системе 4 с именами *ipsec0*, ...). Обратное, один интерфейс IPsec позволяет определить только одну *crypto map*. Сами по себе эти интерфейсы являются служебными и напрямую средствами основной командной оболочки не настраиваются, но их можно просмотреть командой *show interface* или средствами ОС Linux. Команда *crypto map* привязывает интерфейс IPsec к выбранному небезопасному IP-интерфейсу для подключения к внешней сети. Таким образом, одно устройство NSG может обслуживать до 40 удаленных VPN-шлюзов через 4 различных физических интерфейса или суб-интерфейса (DLCI, VLAN) по 10 на каждом.

#### §5.4.5. Просмотр информации о туннелях

Для просмотра сводной информации обо всех туннелях и безопасных ассоциациях, определенных в устройстве, предусмотрена следующая команда в меню (*config-nsg*)#:

```
crypto show
```

Вход в меню просмотра информации IPsec.

Внутри данного узла меню имеются команды:

<i>running</i>	Вывести информацию обо всех существующих статических и динамических туннелях и безопасных ассоциациях, в т.ч. о тех SA, в которых в данный момент нет действующих туннелей.
<i>eroute</i>	Вывести информацию о маршрутах, уходящих в удаленные сети через безопасные туннели.
<i>spi</i>	Вывести информацию о реально существующих туннелях (статических и динамических) и их индексы (SPI).
<i>tncfg</i>	Вывести информацию о соответствии туннелей реальным физическим интерфейсам.

Все перечисленные команды не имеют параметров.

### §5.4.6. Особенности реализации IPsec в устройствах Cisco Systems

Стандарты и спецификации VPN допускают неоднозначное толкование некоторых деталей. Кроме того, они не определяют некоторые смежные вопросы функционирования устройства. Различные производители могут по-разному интерпретировать эти моменты, и возникающие отличия следует взаимно учитывать при установлении туннелей между их устройствами. В частности, в устройствах NSG имеются следующие особенности по сравнению с реализацией, предлагаемой компанией Cisco Systems.

#### а) Маршрутизация при использовании туннелей

Данная особенность относится как к статическим, так и к динамически создаваемым туннелям. В маршрутизаторах Cisco наличие туннеля само по себе не оказывает никакого влияния на таблицу маршрутизации. Иначе говоря, помимо создания туннеля, необходимо вручную сконфигурировать маршрут в удаленную сеть, находящуюся на другой стороне туннеля (обычный статический маршрут). Пример конфигурации (фрагмент, непосредственно связанный с маршрутизацией):

```
!
interface FastEthernet0/0
ip address 10.0.0.31 255.0.0.0
!
crypto map test1 1 ipsec-manual
set peer 10.0.2.11
.....
!
ip route 192.168.1.0 255.255.255.0 10.0.2.11
!
```

Здесь 10.0.0.31 — IP-адрес интерфейса Cisco, 10.0.2.11 — IP-адрес удаленного маршрутизатора. Последняя строка означает, что неизвестная для данного устройства сеть 192.168.1.0 с маской 255.255.255.0 находится за точкой 10.0.2.11.

При удалении туннеля (или правила для установления динамического туннеля) следует удалить и статические маршруты, проходящие через этот туннель.

В устройствах NSG, напротив, настройка статических маршрутов в удаленные сегменты приватной сети не является обязательной. При установлении и удалении туннелей автоматически создаются/удаляются и соответствующие записи в таблице маршрутизации.

#### б) Организация туннеля ISAKMP SA (стадия MAIN MODE) между пограничными маршрутизаторами

1. При инициализации туннеля со стороны NSG предлагается сразу весь (!) пакет предложений, в следующем составе:
  - PSK, group5 (1536), 3DES, SHA
  - PSK, group5 (1536), 3DES, MD5
  - PSK, group2 (1024), 3DES, SHA
  - PSK, group2 (1024), 3DES, MD5
2. При получении запроса на установление туннеля список предложений, поступивший от удаленной стороны, поочередно сравнивается в приведенном выше списке. Совпавший вариант отсылается в качестве подтверждения (выбора).
3. В Cisco все варианты туннелей ISAKMP образуют приоритетное множество предложений (*policies*), которые при посылке отсылаются в порядке, определяемом приоритетом *policy*, а при приеме предложений начинают проверяться в соответствии с приоритетом.

#### в) Организация туннеля ISAKMP SA (стадия QUICK MODE) для защищенной передачи трафика

1. При инициализации туннеля со стороны NSG предлагается (или выбирается из предложенных) пакет из двух предложений с использованием Encapsulation Secure Payload (ESP) и обязательной аутентификацией. Выбор варианта MD5 либо SHA-1 оставляется на усмотрение удаленной стороны.
2. В маршрутизаторах Cisco конкретное множество правил преобразования и их приоритет определяются в самом описании *crypto-map* (тип *ipsec-isakmp*). Это устанавливается перечислением в параметре
 

```
(config-crypto-map)# transform-set <предложение_1> <предложение_2> <предложение_3> ...
```

Если инициатором соединения был удаленный маршрутизатор, то для устройства NSG предпочтительным является алгоритм ESP\_3DES + SHA, а если он не предложен удаленной стороной — тогда ESP\_3DES + MD5.

#### г) Согласование длины ключа на стадии QUICK MODE

Длина ключа безопасной пересылки (PFS) может согласовываться как на стадии Main Mode, так и на стадии Quick Mode. В устройствах NSG поддерживаются группы Диффи-Хеллмана 2 и 5 (длина ключа 1024 и 1536 бит, соответственно). Группа 1 (768 бит) не поддерживается ввиду её недостаточной криптостойкости при современных вычислительных ресурсах.

На стадии Main Mode согласование длины ключа на устройствах NSG включено безусловно, причём посылается сразу весь пакет предложений (см. выше). На маршрутизаторах Cisco предложения, посылаемые на этом этапе, определяются политиками (*policies*).

На стадии Quick Mode согласование длины ключа устанавливается на устройстве NSG следующим параметром в описании crypto map:

```
set pfs { yes | no }
```

а на устройствах Cisco — параметрами

```
set pfs group { 1 | 2 | 5 }
no set pfs
```

Совместная работа возможна только в следующих случаях:

- На обеих сторонах эти параметры установлены в no. (На устройствах под управлением NSG Linux 1.0 *build 1.1* и ранее согласование PFS на этом этапе отключено всегда.)
- На устройстве NSG согласование PFS включено, а на устройстве Cisco выбрана группа 2 или 5.

При всех других возможных сочетаниях настроек туннель установлен не будет.

### §5.4.7. Особенности настройки IPsec в продуктах Майкрософт

Если на другой стороне туннеля IPsec используется программный шлюз под управлением одной из ОС семейства Windows, то при его настройке необходимо обратить внимание на следующие моменты:

- Перед началом настройки IPsec следует настроить обычную маршрутизацию на обеих сторонах и убедиться в нормальном прохождении пакетов из одного сегмента защищаемой сети в другой. Настройку маршрутизации в продуктах Майкрософт удобнее всего производить из командной строки при помощи команды `route -p add ...` (для справки см. `route help`), но можно сделать это и путём кликания мышкой в окошках.
- Настройка производится в оснастке "Локальная политика безопасности". Для работы IPsec необходимо создать *политику*, содержащую два *правила безопасности IP* — для трафика в одну и в другую сторону.
- Флаг PFS в окне "Параметры обмена ключами" необходимо снять.

Пример настройки см. в Приложении 5–А.6.

### §5.4.8. Совместное использование IPsec и NAT

При совместном использовании VPN и NAT на одном интерфейсе возникает конфликт между двумя этими механизмами. Типичным примером является подключение офисной сети к Интернет, когда один и тот же интерфейс с единственным реальным IP-адресом используется и для доступа к хостам Интернет, и для установления защищенного соединения с удалённым офисом. Для нормальной работы необходимо исключить пакеты VPN (с номером протокола 50) из числа пакетов, подлежащих преобразованию адресов.

Для Source NAT (IP-маскарадинга) корректная настройка в этом случае имеет вид:

```
!
nsg
  access-list ext-ip 100
    add 1 deny 50 <ip-адрес интерфейса> <ip-адрес удалённого шлюза>
    add 2 permit ip any any
  exit
port s1
  nat source access-list 100 masquerade
exit
!
```

Настройка в аналогичной задаче для Destination NAT (виртуальных серверов):

```
!
nsg
  access-list ext-ip 101
    add 1 deny 50 <ip-адрес удалённого шлюза> <ip-адрес интерфейса>
    add 2 permit ip any any
  exit
port s1
  nat destination access-list 101 <ip-адрес сервера во внутренней сети>
exit
!
```

## §5.6. Мультипротокольные инкапсуляции X.25

### §5.6.1. Инкапсуляция X.25-over-TCP/IP

Служба X.25-over-TCP/IP (ХОТ) позволяет использовать в качестве транспорта для коммутируемых виртуальных каналов X.25 сеть IP и, как ее частные случаи — безопасный туннель VPN через сеть общего пользования, PPP-соединение по асинхронной модемной линии, каналу GSM, сети GPRS, и т.п. Служба ХОТ реализована в соответствии со стандартом IETF RFC–1613 и совместима с продуктами других производителей.

Серверная часть ХОТ включена постоянно и принимает входящие пакеты на порту TCP 1998. Из полученных пакетов IP извлекаются пакеты X.25, которые передаются коммутатору X.25. Клиент ХОТ активируется в том случае, если получен пакет CALL, для которого в таблице маршрутизации X.25 задан маршрут через ХОТ. В этом случае он инкапсулирует пакеты X.25 в оболочку IP и отправляет их по указанному IP-адресу. Подробно о маршрутизации вызовов X.25 см. [Часть 3](#).

В качестве IP-адреса источника (*source address*) для пакетов ХОТ может использоваться некоторый адрес, установленный пользователем принудительно, например:

```
(config-nsg)# x25 route add prio 2 destination 987654 xot 123.145.167.189 xot-source 198.176.154.132
```

Если адрес источника для пакетов ХОТ явным образом не указан, то используется IP-адрес интерфейса, через который отправляется данный пакет.

**ВНИМАНИЕ** В качестве *xot-source* могут использоваться только адреса, принадлежащие какому-либо из IP-интерфейсов данного устройства. Если требуемый адрес не назначен ни одному интерфейсу, необходимо предварительно назначить его фиктивному интерфейсу *dummy0*:

```
!
interface dummy0
 ip address 10.1.1.1/32
!
nsg
x25 route add prio 1 destination 777666100 xot 10.2.2.2 xot-source 10.1.1.1
```

Тонкая настройка остальных параметров службы ХОТ в данной версии NSG Linux не предусмотрена. Для всех пакетов используются следующие параметры по умолчанию:

- Максимальная длина поля данных пакетного уровня — 128 байт.
- Величина окна пакетного уровня для входящих и исходящих пакетов — 2.
- Время ожидания установки TCP-соединения — определяется параметрами протокола TCP (около 3 минут).

### §5.6.2. Инкапсуляция IP-over-X.25

Для передачи пакетов IP через сети X.25 используется инкапсуляция IP-over-X.25. При этом создается туннель через сеть X.25 посредством установки SVC с аналогичной службой на удаленной стороне. Конечная точка туннеля описывается в сети X.25 некоторым X.121 адресом и представляет собой IP-интерфейс с присущим ему набором параметров. Имена создаваемых интерфейсов — *tunxN*, где N — номер туннеля.

Сервис работает и как сервер, и как клиент. Если установлено коммутируемое виртуальное соединение (SVC) с удаленной стороной, то по туннелю будет идти обмен пакетами. Если в очередь к интерфейсу становится IP-пакет, но SVC в данный момент не установлено, то интерфейс устанавливает SVC к заданному хосту X.25. Если в течение некоторого времени никакого трафика по туннелю нет, то SVC разрывается.

Создание, настройка и удаление туннелей IP-over-X.25 производится в меню `(config-nsg)#` командами:

```
tunnel x25 <1...255>
no tunnel x25 <1...255>
```

Создание/изменение и удаление туннеля, соответственно. Номер туннеля используется только как локальный идентификатор в устройстве NSG и никак не связан с номером, присвоенным этому туннелю на удаленной стороне (если таковой имеется).

Дальнейшая настройка производится в меню `(config-tunnel-N)#`. Меню содержит следующие команды (помимо общих для всех подменю узла *nsg*):

```
local-x121-address {<адрес X.121> | "" }
```

Адрес X.121, который будет указываться в качестве вызывающего (Calling Address) в пакетах CALL, посылаемых данным интерфейсом для установления соединения X.25. Этот же адрес ожидается в качестве вызываемого (Called Address) в пакетах CALL, получаемых от удаленной стороны. Максимальная длина адреса — 15 цифр. Аналогом данного параметра в базовом ПО NSG является LADR.

```
remote-x121-address {<адрес X.121> | "" }
```

Адрес X.121, который будет указываться в качестве вызываемого (Called Address) в пакетах CALL, посылаемых данным интерфейсом для установления соединения X.25. Этот же адрес ожидается в

качестве вызывающего (Calling Address) в пакетах CALL, получаемых от удаленной стороны. Максимальная длина адреса — 15 цифр. Аналогом данного параметра в базовом ПО NSG является XADR.

По умолчанию оба параметра `local-x121-address` и `remote-x121-address` имеют пустое значение. Для установки пустого адреса следует ввести пару двойных кавычек.

Значение `remote-x121-address` должно быть уникальным для каждого интерфейса IP-over-X.25, созданного в устройстве. При установлении входящих соединений X.25 автоматически выбирается интерфейс, у которого этот параметр точно соответствует полю Calling Address полученного пакета CALL. Если такой интерфейс не найден, то будет установлено соединение с локальным интерфейсом командной строки (см. [Часть 1](#)), которое по очевидным причинам окажется неработоспособным.

`window <1...7>`

Размер окна пакетного уровня X.25. (Аналог параметров `win`, `wout` для физического порта.) Значение по умолчанию — 2.

`packet_size { 16 | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 | 4098 }`

Максимальный размер пакетов 3 уровня X.25, в байтах. (Аналог параметров `x25 ips`, `x25 ops` для физического порта.) Значение по умолчанию — 128.

`timeout <1...2147483647>`

Таймаут неактивности для разрыва X.25 SVC, в секундах. Значение по умолчанию — 300 сек. Аналогом данного параметра в базовом ПО NSG является KEEF.

Остальные команды являются общими для всех подузлов меню `nsg`, представляющих собой IP-интерфейсы (физических портов, Frame Relay DLCI, Ethernet VLAN):

`adm-state { up | down }`

Разрешение и запрещение работы туннеля. По умолчанию, все создаваемые туннели находятся в состоянии `up`.

`description "<комментарий>"`

Административное описание данного интерфейса.

`[no] access-group ...`

Настройка фильтрации IP-пакетов на данном интерфейсе. Подробно см. [Часть 4](#).

`[no] crypto ...`

Настройка защищенных туннелей VPN, создаваемых на данном IP-интерфейсе. В данном случае трафик частной сети IP передается внутри защищенного туннеля VPN, который, в свою очередь, проходит по туннелю IP через сеть X.25. Подробно о настройке VPN см. [Часть 5](#).

`[no] ip ...`

Настройка параметров протокола IP. Подробно см. [Часть 4](#).

`mtu <64...18000>`

Установка размера MTU для IP-пакета. Подробно см. [Часть 4](#).

`nat ...`

Настройка трансляции сетевых IP-адресов (NAT) для данного интерфейса. Подробно см. [Часть 4](#).

`[no] service-policy ...`

Выбор и настройка политики управления IP-трафиком для данного интерфейса. Подробно см. [Часть 4](#).

`show ...`

Просмотр состояния и статистики интерфейса. Подробно см. [Часть 4](#).

Для работы туннеля IP-over-X.25 в таблице маршрутизации X.25 должна быть сделана соответствующая запись с назначением `local`, например:

```
tunnel x25 1
  local-x121-address 1234567890
  .....
  exit
x25 route add destination 1234567890 local
```

#### Пример конфигурации:

```
card s4 im-v24
port s4
  encapsulation x25
  x25 htc 16
  exit
tunnel x25 7
  local-x121-address 88
  remote-x121-address 77
  ip address 12.0.0.2/8 anycast 0.0.0.0
  exit
x25 route add prio 1 destination 77 port s4
x25 route add prio 2 destination 88 local
```

## Приложение 5–А. Примеры настройки туннелей и VPN

### §5–А.1. Подключение устройства NSG к серверам PPPoE

Настройка устройства NSG в качестве клиента PPPoE. IP-адрес назначается динамически, аутентификация в данных примерах не используется.

```
!
nsg
  port eth0
  pppoe client enable
  pppoe server <имя_сервера>
  exit
```

Настройка устройства Cisco в качестве сервера PPPoE:

```
!
vpdn enable
!
vpdn-group 1
  accept-dialin
  protocol pppoe
  virtual-template 2
  pppoe limit per-mac 20
  local name CISCO
!
interface FastEthernet0/0
  pppoe enable
!
interface Virtual-Template 2
  ip address 16.0.0.1 255.0.0.0
  peer default ip address pool TEST
!
ip local pool TEST 16.0.0.2 16.0.0.20
```

Настройка устройства NSG с ПО NSG Linux в качестве сервера PPPoE:

```
!
nsg
  virtual-template 1
    peer ip address 19.0.0.2
    ip address 19.0.0.1
    exit
  vpdn-group 1
    pppoe limit per-mac 20
    protocol pppoe
    virtual-template 1
    local name NSGLinux
  port eth0
    vpdn-group 1
    exit
  exit
```

Настройка устройства NSG с базовым ПО в качестве сервера PPPoE:

```
S P PO:00 TY:ETH ADM:UP IF:TP MODE:AUTO SP:100000000 NAME:"" ADDR:00.09.56.10.05.97.
S P ET:01 PO:0 TY:PPP NAME:"RTEMS" IP:ALL
S P IP:00 ADM:UP NUM:01 NAME:""
S P IP:01 ADM:UP NAME:"NSGbasic" IADR:17.0.0.1 MASK:255.0.0.0 TY:PPP PO:AUTO RADR:17.0.0.2
```

Настройка Linux Red Hat 9 (пакет Roaring Penguin PPPoE Version 3.5) в качестве сервера PPPoE:

```
pppoe-server -I eth0 -C Linux -L 18.0.0.1 -R 18.0.0.2
Содержимое обязательного файла /etc/ppp/pppoe-server-options
# PPP options for the PPPoE server
# LIC: GPL
lcp-echo-interval 10
lcp-echo-failure 2
```

## §5–А.2. Подключение клиентов PPPoE к устройству NSG

Устройство под управлением NSG Linux используется в качестве сервера PPPoE.

Пример 1. Подключение клиентов без аутентификации.

```
!  
nsg  
  virtual-template 1  
    ip address 14.0.0.1  
    peer ip address 14.0.0.10  
  exit  
  vpdn-group 1  
protocol pppoe  
pppoe limit 120  
pppoe name mike2  
virtual-template 1  
exit  
  port eth0  
vpdn-group 1  
!
```

Пример 2. Подключение клиентов с аутентификацией PAP (локально)

```
!  
nsg  
  username Zorro password Zorro  
  username user1 password pass1  
  username mike password pmike  
  virtual-template 1  
    ip address 14.0.0.1  
    peer ip address 14.0.0.10  
    ppp authentication pap local  
  exit  
  vpdn-group 1  
protocol pppoe  
pppoe limit 120  
pppoe name mike2  
virtual-template 1  
  exit  
  port eth0  
vpdn-group 1  
!
```

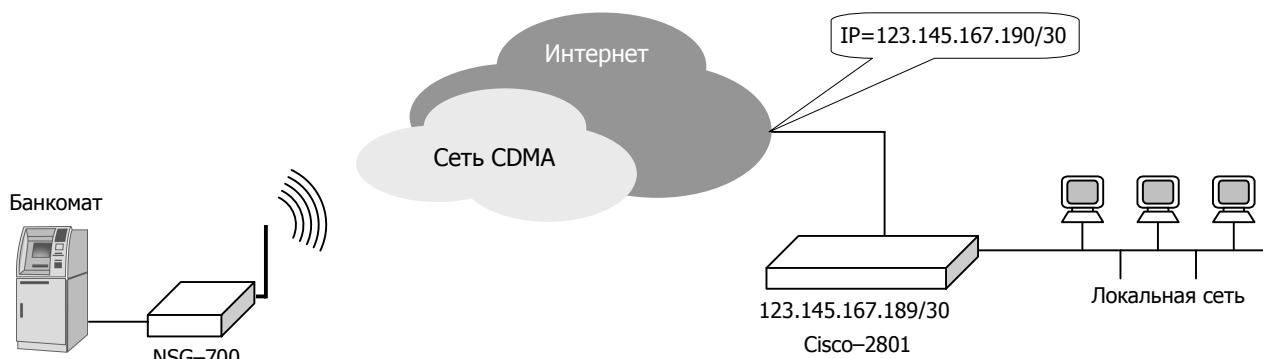
Пример 3. Подключение клиентов с аутентификацией CHAP на удаленном RADIUS-сервере.

```
!  
nsg  
  radius host 10.0.0.2  
  radius auth-port 1812  
  radius acct-port 1813  
  radius key nsg  
  radius timeout 10  
  radius retry 3  
  virtual-template 1  
    ip address 14.0.0.1  
    peer ip address 14.0.0.10  
    ppp authentication chap radius  
  exit  
  vpdn-group 1  
protocol pppoe  
pppoe limit 120  
pppoe name mike2  
virtual-template 1  
  exit  
  port eth0  
vpdn-group 1  
!
```



### §5–А.3. Настройка туннеля PPTP

Пример настройки соединения PPTP через сотовую сеть CDMA (SkyLink). Используется устройство NSG–700/4AU с интерфейсным модулем UIM–EVDO. На центральном узле корпоративной VPN в качестве сервера PPTP используется устройство Cisco. Туннель защищён с помощью MPPE (в результате согласования будет выбрано 128 бит) и MS–CHAP v2.



#### Настройка NSG–700

```

!
nsg
  virtual-template 2
    keepalive no retry no
    ppp ipcp accept-address yes
    ppp set-default-route yes
    ppp sent-username basile
    ppp encrypt-mppe auto
  exit
  virtual-template 1
    keepalive 10 retry 3
    ppp ipcp accept-address yes
    ppp sent-username mobile
  exit
  tunnel pptp 1
    server-address 123.145.167.189
    virtual-template 2
  exit
  chat-script CDMA "TIMEOUT 10 XXX-AT-OK ATD#777 CONNECT ' '"
  card s1 uim-cdma
  port eth0
    ip address 10.0.2.16/8 anycast 0.0.0.0
  exit
  port s1
    encapsulation ppp
    virtual-template 1
    chat-script CDMA
  exit
!
username basile password P0uPkiNe
username mobile password internet
!
ip route 123.145.167.189/32 s1
!

```

Здесь `mobile` и `internet` — имя и пароль для доступа к услуге CDMA, `123.145.167.189` — адрес удаленного сервера в Интернет, `basile P0uPkiNe` — имя и пароль для PPTP-соединения с этим сервером. В случае некорректного отсоединения от сети CDMA (пропадание сигнала и т.п.) отказ будет детектирован через 30 сек (3 попытки по 10 сек). После этого PPP-интерфейс рестартует и попытается снова установить соединение с сетью. Если соединение будет восстановлено успешно и с прежним IP-адресом, а по туннелю в течение всего этого времени никакие данные не посылались, то туннель PPTP продолжит работу, так что переустановка физического соединения и PPP-соединения останется незамеченным для пользователей сети. Если же соединение не восстановлено, а PPTP-интерфейс попытается передать данные по туннелю, то управляющее соединение обнаружит, что интерфейс сети общего пользования находится в состоянии DOWN (или снова в UP, но с иным IP-адресом), и туннель будет разорван.

Настройка PPTP для Cisco 2801:

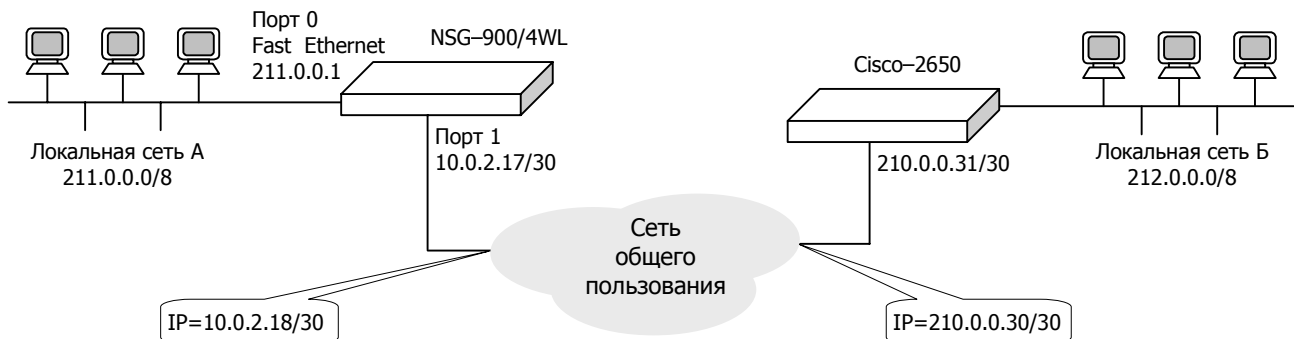
```

!
aaa new-model
aaa authentication ppp default local
no ip cef
vpdn enable
vpdn-group 1
    accept-dialin
    protocol pptp
    virtual-template 1
    ip mtu adjust
!
username basile password 0 P0uPkiNe
!
interface FastEthernet0/0
    ip address 123.145.167.189 255.255.255.252
!
interface Virtual-Template1
    ip address 172.16.0.1 255.255.255.0
    peer default ip address pool APOOL
    keepalive 11 3
    ppp encrypt mppe auto
    ppp authentication ms-chap-v2
!
ip local pool APOOL 172.16.0.2 172.16.0.20
ip route 0.0.0.0 0.0.0.0 FastEthernet0/0 123.145.167.190
!

```

**§5–А.4. Настройка статического туннеля IPSec между NSG и Cisco**

Схема стенда показана на рисунке. Стенд состоит из двух пограничных маршрутизаторов, соединенных через сеть общего пользования. Интерфейсы соседних маршрутизаторов в этой сети имеют IP-адреса 10.0.2.18 и 210.0.0.30. Для наглядности на одной стороне используется устройство NSG–900, на другой — Cisco–2650.



Через маршрутизаторы связаны две приватные сети 211.0.0.0/8 и 212.0.0.0/8. Трафик этих сетей передается в безопасном туннеле между интерфейсами пограничных маршрутизаторов NSG–900 (10.0.2.17) и Cisco–2650 (210.0.0.31). При этом весь пакет, включая заголовок, шифруется по алгоритму 3DES (длина ключа 168 бит) и передается как данные в IP-пакете между двумя маршрутизаторами. Дополнительно передается аутентификационный заголовок (вариант MD5), обеспечивающий аутентичность и целостность. На противоположной стороне туннеля данные пакета расшифровываются и передаются в приватную сеть. Весь остальной трафик принимается и отсылается указанными интерфейсами без какой-либо обработки.

Настройка NSG–900

Конфигурирование диапазона адресов, трафик которых нужно отсылать в защищенном туннеле:

```

!
nsg
    access-list ext-ip 154
        add 1 permit ip 211.0.0.0 0.255.255.255 212.0.0.0 0.255.255.255
    exit

```

Создание правила преобразования трафика, направляемого и получаемого из туннеля. Выбор механизма аутентификации MD5:

```
crypto transform-set tun4 esp 3des-md5-hmac
```

Описание туннеля. Назначение индекса (SPI) и определение секретного ключа. Установка ссылок на соответствующие правило преобразования и диапазон адресов защищаемого трафика. Определение IP-адреса конечной точки туннеля (интерфейс маршрутизатора, работающего в паре с данным):

```

crypto map tunnel_nsg 1
  method ipsec-manual
  match address 154
  set transform-set tun4
  set peer 210.0.0.31
  set nexthop 10.0.2.18
  set session-key esp 4000 cipher 112233445566778899001122334455667788990011223344
authenticator 1122334455667788990011223344556677889900
exit

```

Включение механизма туннелирования на интерфейсе, от которого начинается защищенный туннель:

```

port s1
  ip address 10.0.2.17/30
  crypto map tunnel_nsg
  exit
exit
!

```

#### Настройка Cisco-2650

```

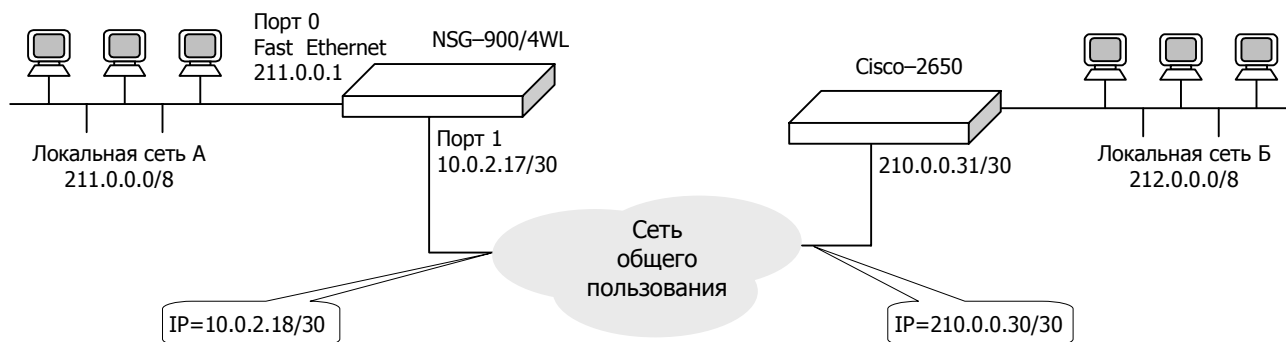
!
crypto ipsec transform-set ts4 esp-3des esp-md5-hmac
!
crypto map tunnel_cisco 40 ipsec-manual
  set peer 10.0.2.17
  set session-key inbound esp 4000 cipher 112233445566778899001122334455667788990011223344
authenticator 1122334455667788990011223344556677889900
  set session-key outbound esp 4000 cipher 112233445566778899001122334455667788990011223344
authenticator 1122334455667788990011223344556677889900
  set transform-set ts4
  match address 154
!
access-list 154 permit ip 212.0.0.0 0.255.255.255 211.0.0.0 0.255.255.255
!
interface FastEthernet0/0
  ip address 210.0.0.31 255.255.255.252
  crypto map tunnel_cisco
!
ip route 211.0.0.0 255.0.0.0 10.0.2.17
ip route 10.0.2.17 255.255.255.255 210.0.0.30
!

```

(Строки, начинающиеся от левого поля, являются продолжением предыдущей строки.)

### §5–А.5. Настройка динамического туннеля IPSec (IKE) между NSG и Cisco

Схема стенда показана на рисунке. Стенд состоит из двух пограничных маршрутизаторов, соединенных через сеть общего пользования. Интерфейсы соседних маршрутизаторов в этой сети имеют IP-адреса 10.0.2.18 и 210.0.0.30. Для наглядности на одной стороне используется устройство NSG-900, на другой — Cisco-2650.



Через маршрутизаторы связаны две приватные сети 211.0.0.0/8 и 212.0.0.0/8. Трафик этих сетей передается в безопасном туннеле между интерфейсами пограничных маршрутизаторов NSG-900 (10.0.2.17) и Cisco-2650 (210.0.0.31). При этом весь пакет, включая заголовок, шифруется по алгоритму 3DES (длина ключа 168 бит) и передается как данные в IP-пакете между двумя маршрутизаторами. Дополнительно передается аутентификационный заголовок (вариант SHA-1), обеспечивающий аутентичность и целостность. На противоположной стороне туннеля данные пакета расшифровываются и передаются в приватную сеть. Весь остальной трафик принимается и отсылается указанными интерфейсами без какой-либо обработки.

Настройка NSG-900

Конфигурирование диапазона адресов, трафик которых нужно отсылать в защищенном туннеле:

```
!
nsg
access-list ext-ip 153
add 1 permit ip 211.0.0.0 0.255.255.255 212.0.0.0 0.255.255.255
exit
```

Создание правила преобразования трафика, направляемого и получаемого из туннеля:

```
crypto transform-set tun3 esp 3des-sha-hmac
```

Описание туннеля. Установка ссылки на соответствующее правило преобразования и диапазон адресов защищаемого трафика. Определение IP-адресов конечной точки туннеля (интерфейс маршрутизатора Cisco) и следующего шлюза на этом маршруте (узел 10.0.2.16):

```
crypto map tunnel_nsg 1
method ipsec-isakmp
match address 153
set transform-set tun3
set peer 210.0.0.31
set nexthop 10.0.2.18
set lifetime 3600
exit
```

Определение разделяемого секрета — *preshared key* (PSK). В данном примере разделяемый секрет — строка из двух символов *aa*.

```
crypto isakmp key aa address 210.0.0.31 10.0.2.17
```

Включение механизма туннелирования на интерфейсе, от которого начинается защищенный туннель:

```
port s1
ip address 10.0.2.17/30
crypto map tunnel_nsg
exit
exit
!
```

Настройка Cisco-2650

```
access-list 153 permit ip 212.0.0.0 0.255.255.255 211.0.0.0 0.255.255.255
crypto ipsec transform-set ts3 esp-3des esp-sha-hmac
!
crypto map M1 3 ipsec-isakmp
set peer 10.0.2.17
set transform-set ts3
match address 153
!
crypto isakmp key aa address 10.0.2.17
!
```

Дополнительно требуется определить *policy* с указанием использования механизма PreShared Key (поскольку *default policy* использует RSA):

```
!
crypto isakmp policy 1
encr 3des
authentication pre-share
group 2
lifetime 3600
!
```

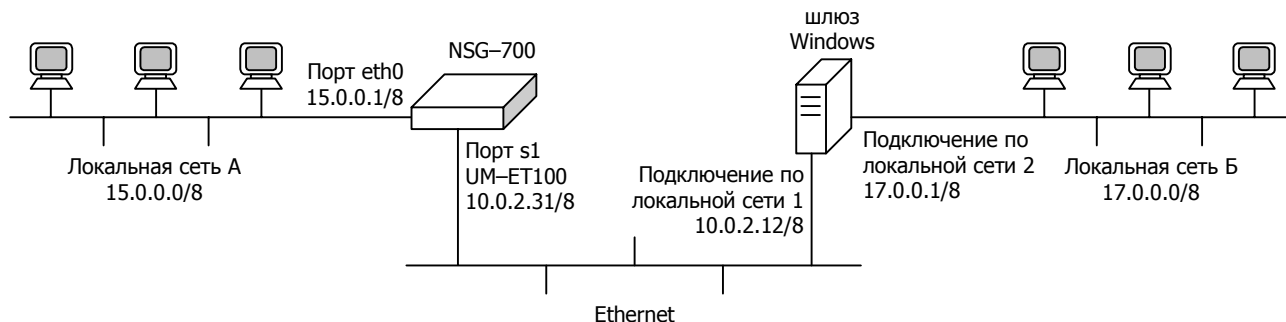
В заключение конфигурируется IP-интерфейс и маршрутизация:

```
!
interface FastEthernet0/0
ip address 210.0.0.31 255.255.255.252
crypto map M1
!
ip route 211.0.0.0 255.0.0.0 10.0.2.17
ip route 10.0.2.17 255.255.255.255 210.0.0.30
!
```

## §5–А.6. Настройка динамического туннеля IPsec (IKE) между NSG и Windows

Имеются две локальные сети, одна из которых подключена к устройству NSG–700, другая — к некоторому программному шлюзу, на котором установлена операционная система Windows 2000 или старше корпорации Майкрософт. Требуется организовать безопасный туннель IPsec между этими сетями. Для большей ясности задачи будем предполагать, что два шлюза соединены между собой просто сетью Ethernet.

**ВНИМАНИЕ** Предполагается, что до начала настройки IPsec на обоих шлюзах настроена маршрутизация, так что хосты из одной сети успешно обмениваются IP-пакетами с хостами из другой сети, и наоборот.

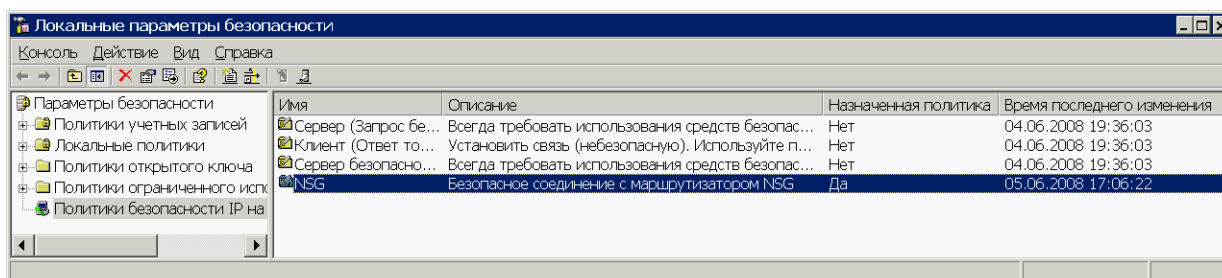


Конфигурация устройства NSG:

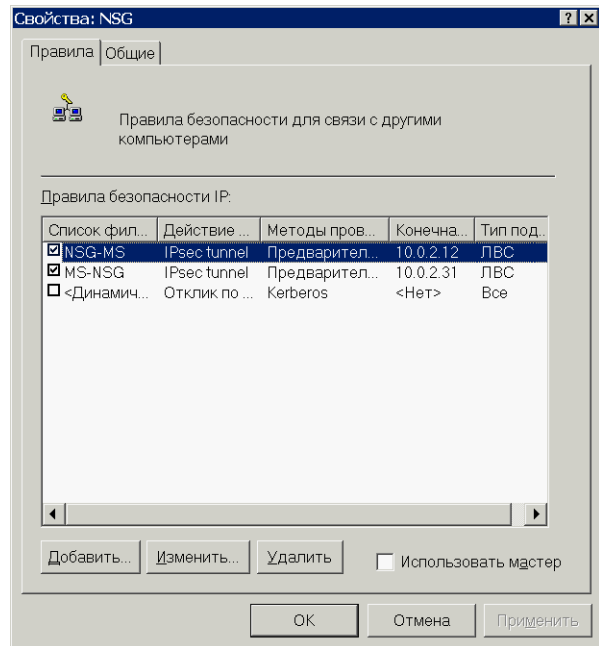
```
!
nsg
  access-list ext-ip 151
    add 1 permit ip 15.0.0.0 0.255.255.255 17.0.0.0 0.255.255.255
  exit
  crypto transform-set ts1 esp 3des-sha-hmac
  crypto isakmp key 12345678 address 10.0.2.12 10.0.2.31
  crypto map CM1
    1
      method ipsec-isakmp
      set transform-set ts1
      set peer 10.0.2.12
      match address 151
    exit
  exit
  card s1 um-et100
  port s1
    ip address 10.0.2.31/8
    crypto map CM1
  exit
  port eth0
    ip address 15.0.0.1/8
  exit
  exit
  ip route 17.0.0.0/8 10.0.2.12  (только для проверки маршрутизации; после настройки VPN удалить)
!
```

Настройка Microsoft Windows:

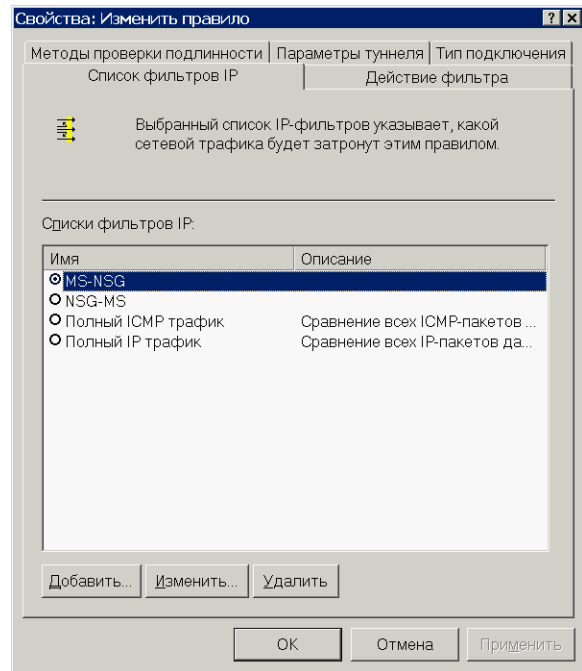
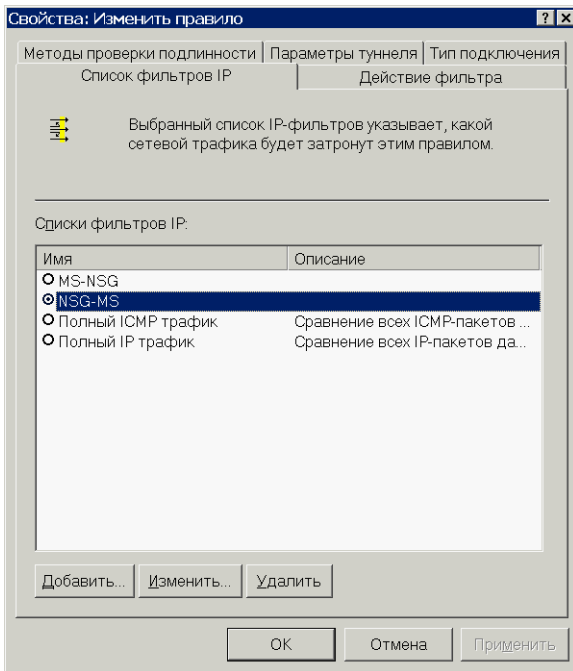
1. Запустить консоль локальной политики безопасности: "Пуск" — "Настройка" — "Панель управления" — "Администрирование" — "Локальная политика безопасности" (или "Пуск" — "Программы" — "Администрирование" — ...). В левой части консоли выбрать "Политики безопасности IP...". В правой части консоли создать новую политику, например, под названием NSG.



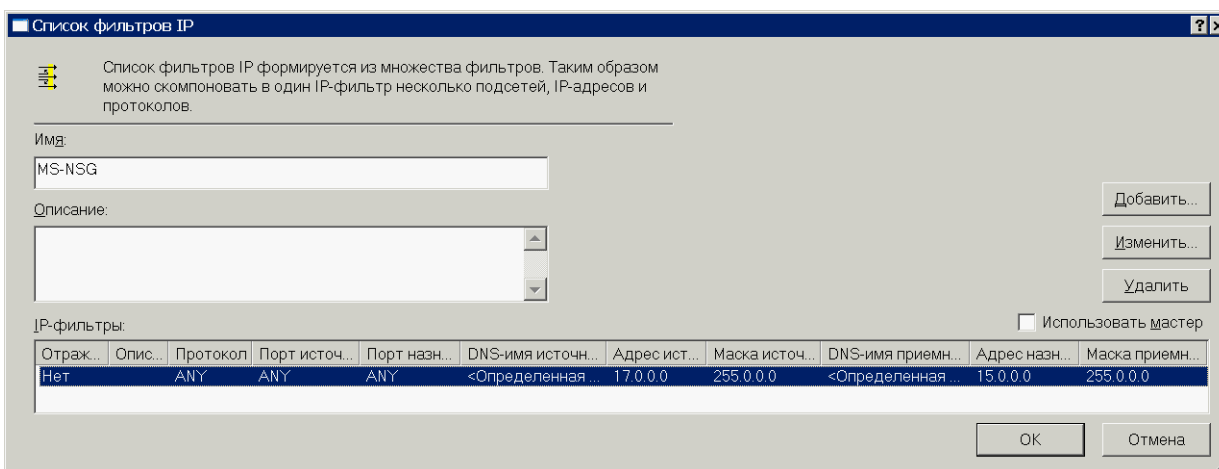
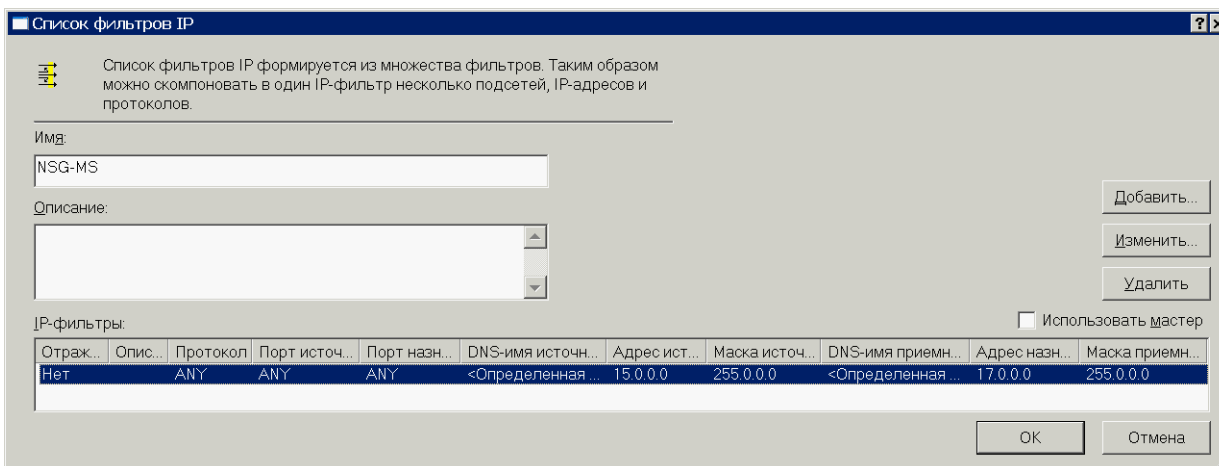
2. Открыть свойства созданной политики. Для облегчения работы рекомендуется создать политику с минимальными настройками, которые требует Мастер Создания Политик, а затем отключить на всех вкладках опцию "Использовать Мастер" и настроить нужные параметры самостоятельно. Необходимо создать и активировать (поставить отметку слева) два правила, описывающие, насколько можно понять новояз корпорации Майкрософт, передачу трафика между двумя шлюзами во встречных направлениях. Имя и описание правил существенной роли не играют; будем называть их NSG–MS и MS–NSG, соответственно. Процедура создания правил описана ниже. Остальные правила, существующие по умолчанию, следует удалить или отключить. Для создания и редактирования объектов в данном окне используются кнопки "Добавить" и "Изменить".



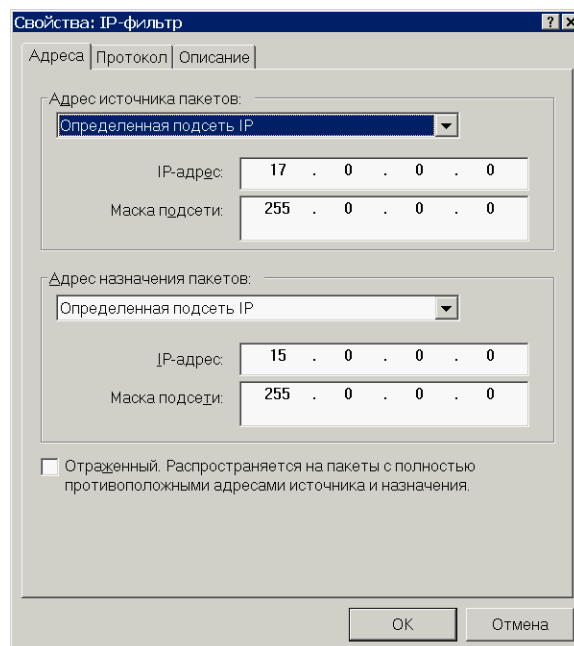
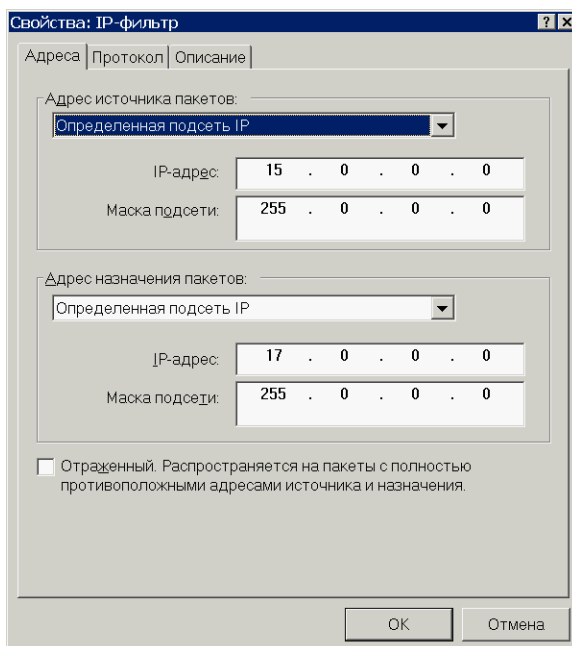
3. Для создания нового правила нажать кнопку "Добавить". Откроется окно с 5 вкладками. На вкладке "Список фильтров IP" нужно создать 2 фильтра и для каждого из правил выбрать соответствующий фильтр. Создаваемое таким образом правило получает имя от используемого в нём фильтра. На рисунках показан вид окна после создания фильтров. Фильтры, существующие по умолчанию, следует отключить или удалить.

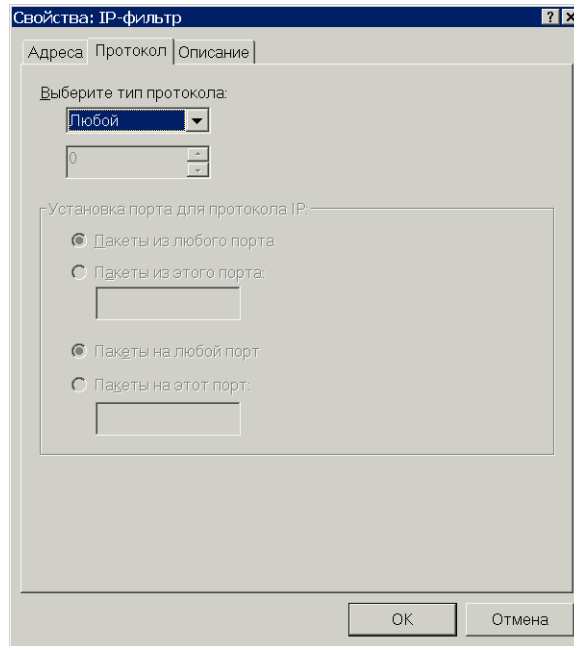


3.1. В окне редактирования свойств фильтра следует ввести имя, которое будет присвоено фильтру и правилу, и нажать кнопку "Добавить". На рисунке показан вид данного окна для того и для другого фильтров после добавления записей.

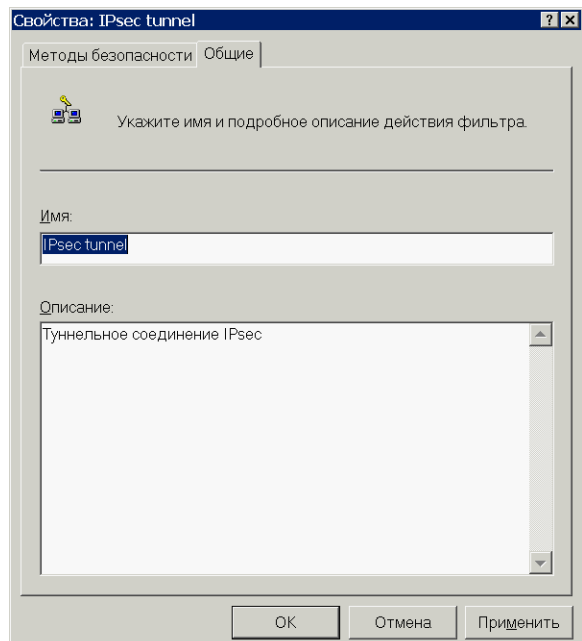
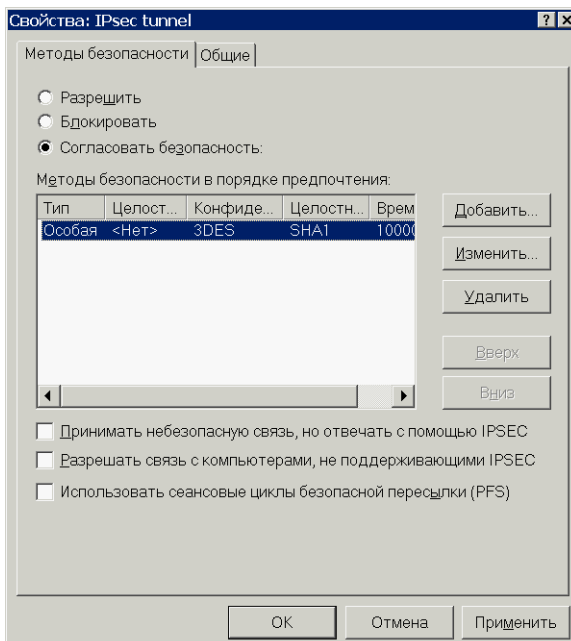
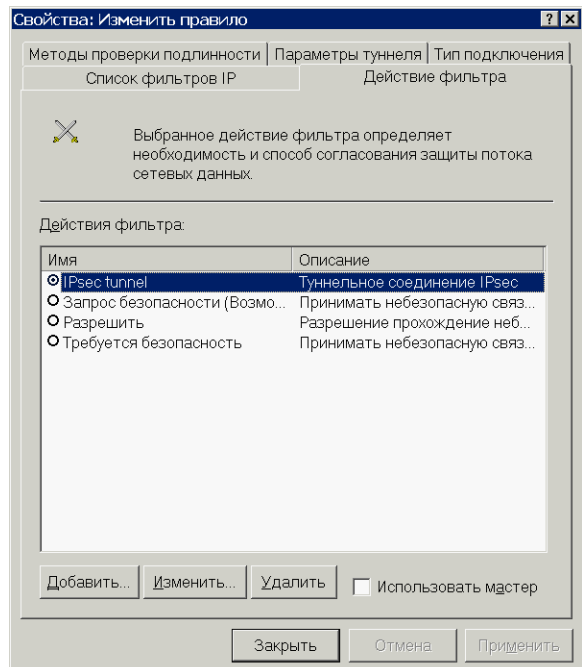


3.1.1. В окне добавления/изменения фильтра на вкладке "Адреса" выбрать для источника и назначения тип "Определённая подсеть IP" и указать адреса и маски обеих сетей (в противоположном порядке для двух фильтров). Опция "Отражённый" никакого влияния на работу системы не оказывает, её назначение неясно. При необходимости можно указать конкретные типы протоколов и номера портов TCP и UDP источника и назначения, чтобы направлять в защищённый туннель только специфические пакеты. По завершении настроек нажать кнопку "ОК", убедиться, что окно "Список фильтров IP" приняло вид, изображённый выше, и закрыть это окно.



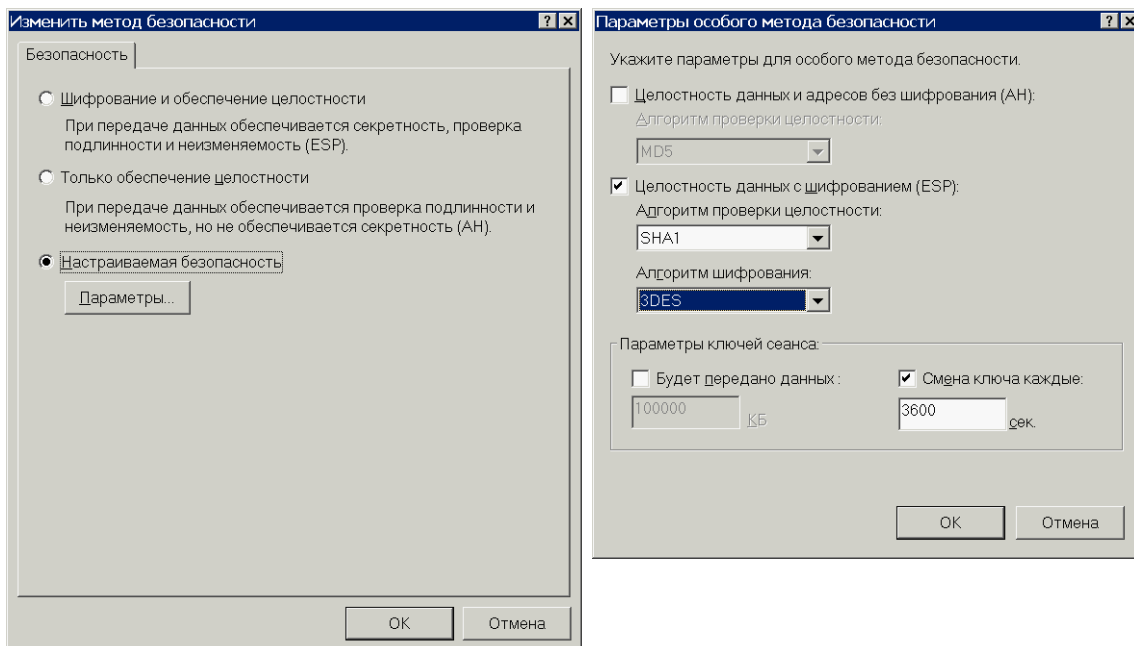


- 3.2. На вкладке "Действия фильтра" создать и выбрать действие "IPsec tunnel". В окне редактирования действия задаются его имя и описание (вкладка "Общие"). На вкладке "Методы безопасности" выбрать опцию "Согласовывать безопасность" и нажать кнопку "Добавить". На рисунке показан вид данного окна после добавления записей. Действие создаётся одинаковым для обоих фильтров.

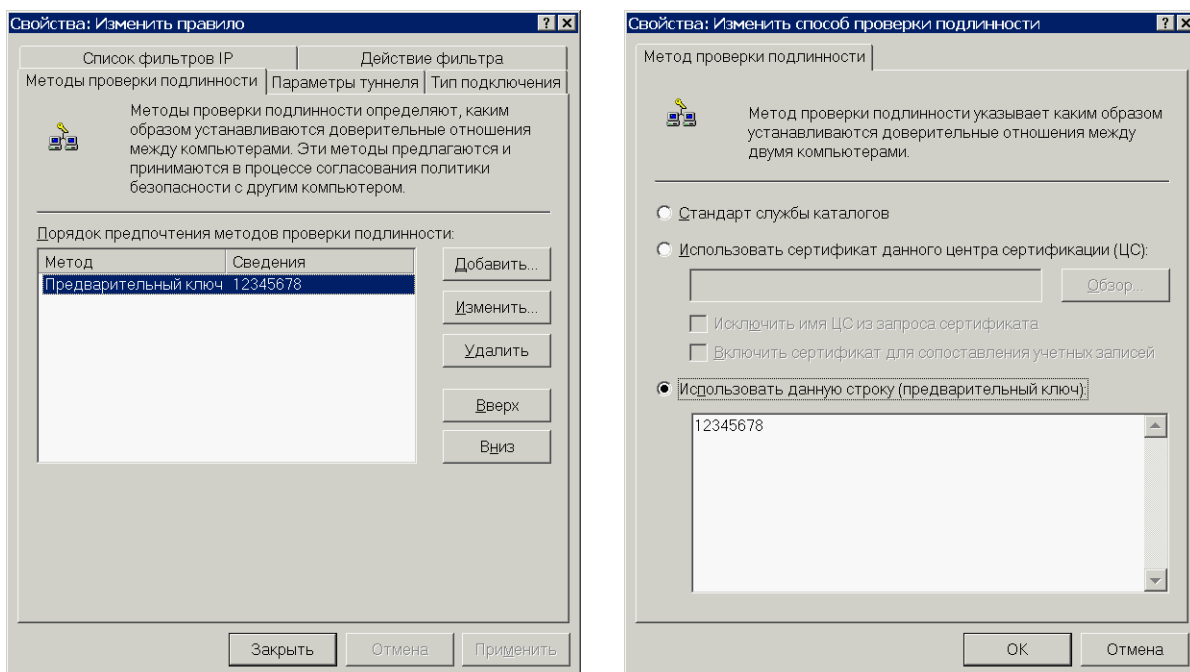




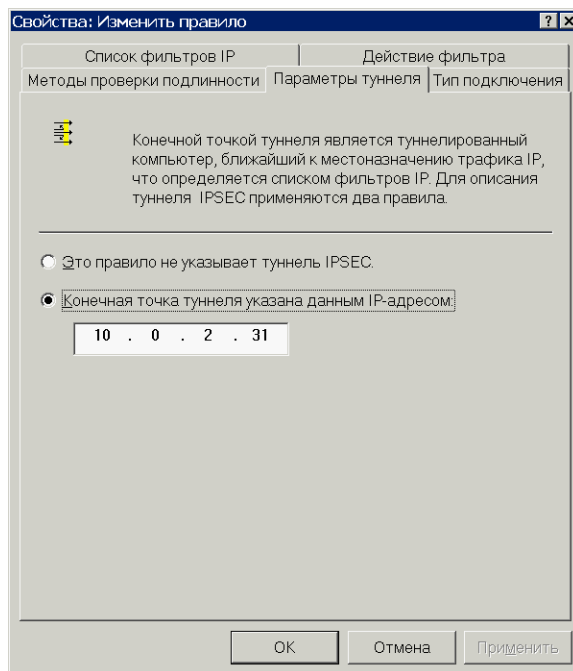
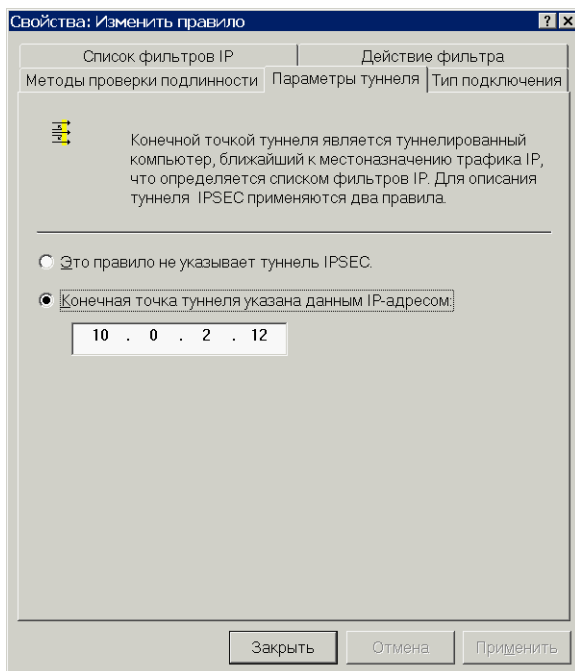
- 3.2.1. В окне "Изменить метод безопасности" выбрать опцию "Настраиваемая безопасность" и нажать кнопку "Параметры". В окне "Параметры особого метода безопасности" выбрать опцию "Целостность данных с шифрованием (ESP)", алгоритм проверки целостности SHA1, алгоритм шифрования 3DES. Рекомендуется включить опцию регулярной смены ключей. По завершении настроек нажать кнопку "OK", убедиться, что окно "Свойства: IPsec tunnel" приняло вид, изображённый выше, и закрыть это окно.



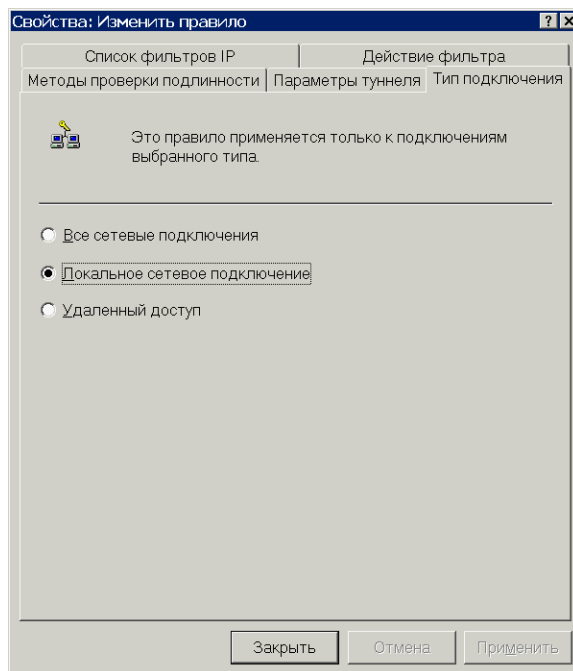
- 3.3. На вкладке "Методы проверки подлинности" создать или изменить единственный метод — с использованием предварительного ключа. В окне свойств метода выбрать опцию "Использовать данную строку (предварительный ключ)" и ввести ключ, установленный на устройстве NSG командой `crypto isakmp key`. Методы, существующие по умолчанию, удалить или передвинуть в конец списка. Данная настройка одинакова для обоих фильтров.



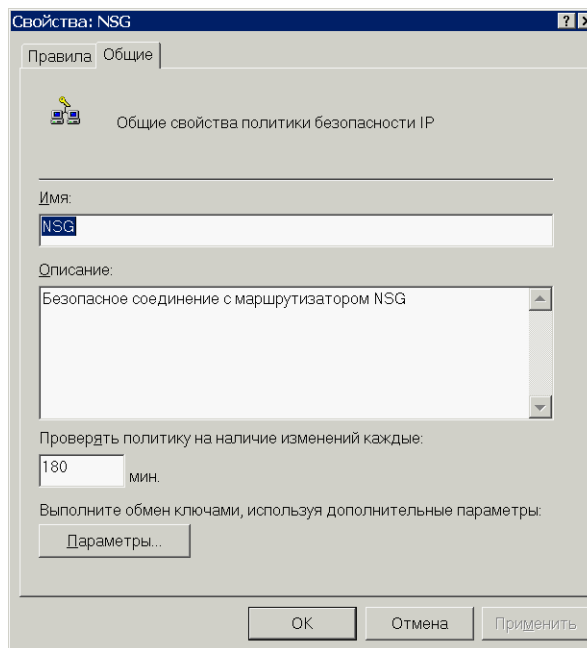
- 3.4. На вкладке "Параметры туннеля" для каждого из фильтров выбрать опцию "Конечная точка туннеля указана данным IP-адресом" и указать адрес внешнего (открытого) IP-интерфейса удалённой стороны. На рисунке слева фильтр NSG–MS, справа MS–NSG.



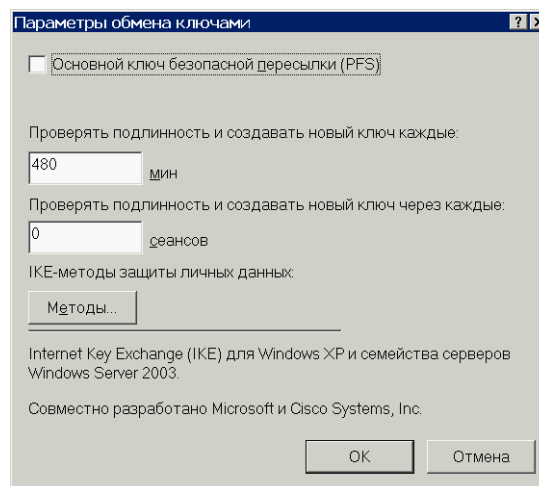
- 3.5. На вкладке "Тип подключения" выбрать опцию "Локальное сетевое подключение" для обоих фильтров. Закрывать окно свойств фильтра.



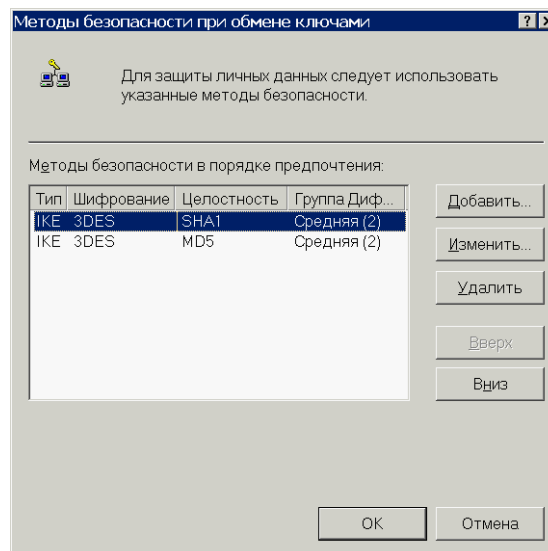
4. На вкладке "Общие" окна "Свойства политики" установить имя политики и её описание для административных целей. То и другое может быть произвольным, удобным администратору. Нажать кнопку "Параметры".



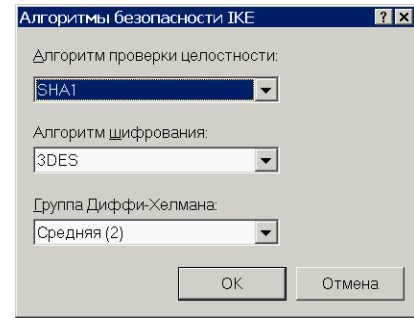
- 4.1. В открывшемся окне "Параметры обмена ключами" отключить опцию "Основной ключ безопасной пересылки (PFS)". Нажать кнопку "Методы".



- 4.1.1. В открывшемся окне "Методы безопасности при обмене ключами" оставить только один или два метода IKE: шифрование — 3DES, целостность — SHA1 или MD5, группа Диффи-Хеллмана — средняя (2).



4.1.1.1. Для создания и редактирования методов используются кнопки "Добавить" и "Изменить", которые открывают окно "Алгоритмы безопасности". Здесь следует выбрать необходимые параметры, а затем последовательно закрыть все окна свойств.



5. В консоли локальной политики безопасности выбрать политику "NSG" и совершить над ней действие "Назначить". Убедиться, что в колонке "Назначенная политика" для данной политике стоит значение "Да" (см. рис. выше в п.1).
6. Убедиться, что хосты из обеих сетей доступны друг для друга.
7. Убедиться, что трафик действительно передаётся по защищённому туннелю, с помощью команды `crypto show running` на устройстве NSG. (Подробно о данной команде см. п.5.4.5)

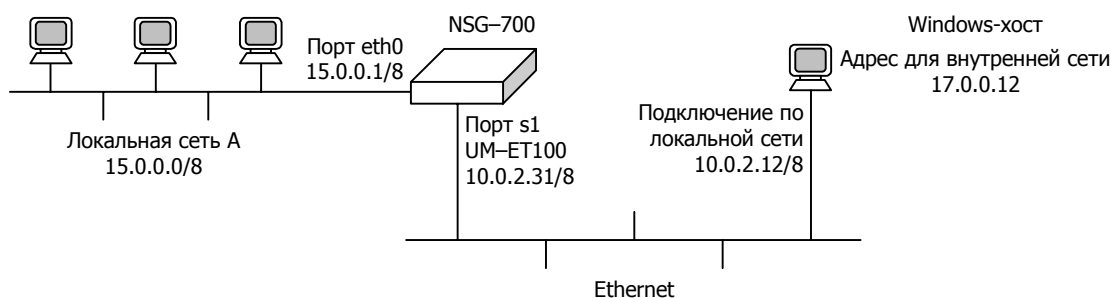
**ПРИМЕЧАНИЕ** Как утверждает корпорация Майкрософт (см. статью [252735 Базы Знаний](#)), её продукты не поддерживают подключение в транспортном режиме IPsec (т.е. в качестве конечного хоста) через удалённый шлюз к сети, находящейся за этим шлюзом. При необходимости подключения удалённых пользователей по такой схеме используется инкапсуляция IPsec-over-L2TP. Поскольку данная версия NSG Linux не поддерживает L2TP, то включение по такой схеме невозможно. Как альтернативу, и альтернативу предпочтительную, корпорация Майкрософт рекомендует использовать туннели PPTP с шифрованием MPPE, реализующие схожую схему инкапсуляции IP (через шифрование и протокол второго уровня снова в IP). Функциональность сервера PPTP планируется к реализации в последующих версиях NSG Linux.

## §5–А.7. Настройка динамического туннеля IPsec (IKE) между NSG и IPsec-клиентами для Windows

Собственно операционные системы Windows поддерживают работу в туннельном режиме только в качестве промежуточного шлюза (см. п.5–А.6). По этой причине напрямую соединить одиночный Windows-хост с устройством NSG при помощи туннеля IPsec невозможно. Однако существует значительное число IPsec-клиентов сторонних разработчиков, позволяющих решить эту задачу. Все они, по существу, организуют в системе внутренний шлюз IPsec, за которым оказывается основной стек IP, доступный приложениям и имеющий отдельный IP-адрес. Между этим шлюзом и устройством NSG организуется туннель, внутри которого передаётся трафик между приложениями Windows и сетью, расположенной за устройством NSG.

Схема стенда показана на рисунке. Для большей ясности задачи будем предполагать, что два шлюза соединены между собой просто сетью Ethernet.

**ВНИМАНИЕ** Предполагается, что до начала настройки IPsec на обоих шлюзах настроена маршрутизация, так что хосты из защищённой сети успешно обмениваются IP-пакетами с Windows-хостом, и наоборот.



Конфигурация устройства NSG:

```
!
nsg
  access-list ext-ip 152
    add 1 permit ip 15.0.0.0 0.255.255.255 host 17.0.0.12
  exit
  crypto transform-set ts1 esp 3des-sha-hmac
  crypto isakmp key 12345678 address 10.0.2.12 10.0.2.31
  crypto map CM1
    2
      method ipsec-isakmp
      set transform-set ts1
      set peer 10.0.2.12
      match address 152
    exit
  exit
  card s1 um-et100
  port s1
    ip address 10.0.2.31/8
    crypto map CM1
  exit
  port eth0
    ip address 15.0.0.1/8
  exit
  exit
  ip route 17.0.0.0/8 10.0.2.12 (только для проверки маршрутизации; после настройки VPN удалить)
!
```

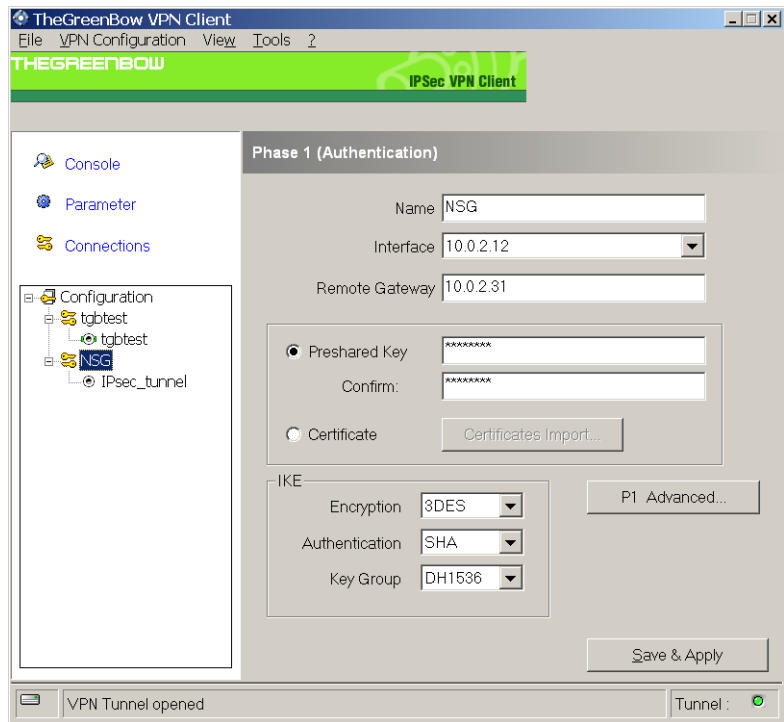
**Настройка TheGreenBow VPN Client (<http://www.thegreenbow.com>):**

Вся конфигурация укладывается в двух компактных окнах. В окне Phase 1 описываются имя и параметры туннеля, в окне Phase 2 — параметры закрываемого трафика. Оба набора параметров создаются при помощи щелчка правой клавишей мыши и пункта Add ... в выпадающем меню. Настройка показана на рисунках, существенные параметры перечислены ниже.

Encryption — 3DES

Authentication — SHA или MD5

Key Group — DH1024 или DH1536



Encryption — 3DES

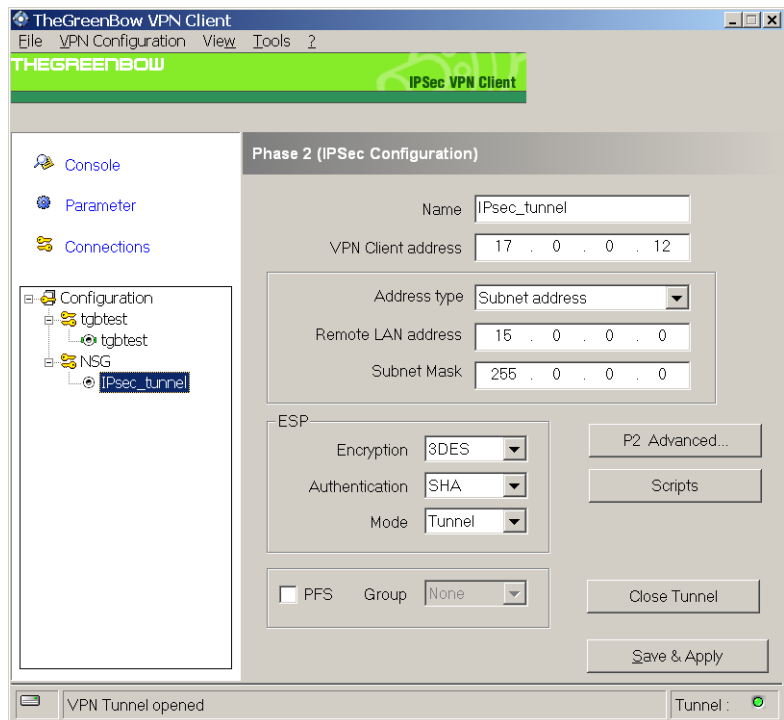
Authentication — SHA или MD5

Mode — Tunnel

PFS — отключено

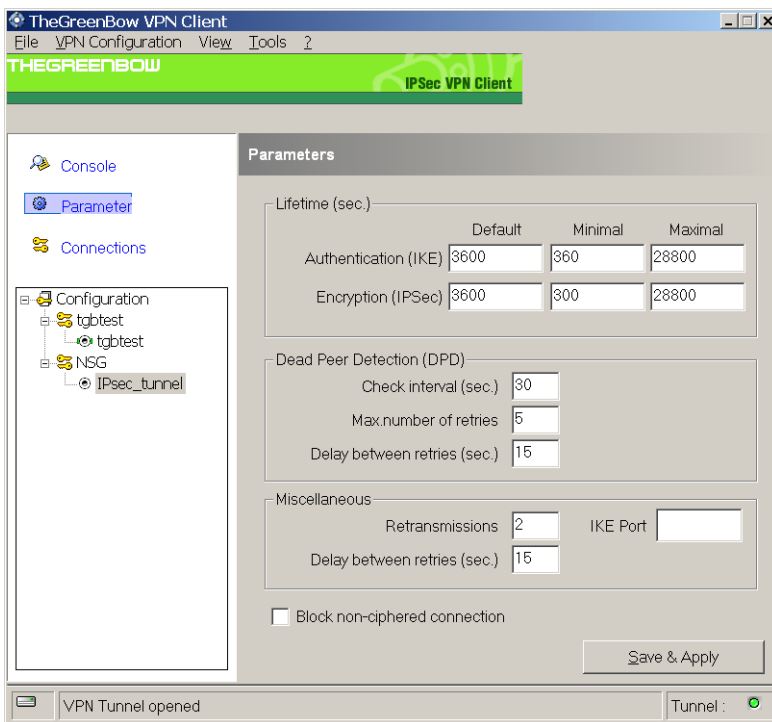
**ПРИМЕЧАНИЕ**

Внутренний адрес IP-клиента может как совпадать с внешним (10.0.2.12), так и отличаться от него (17.0.0.12), при соответствующих настройках *access-list* на устройстве NSG.

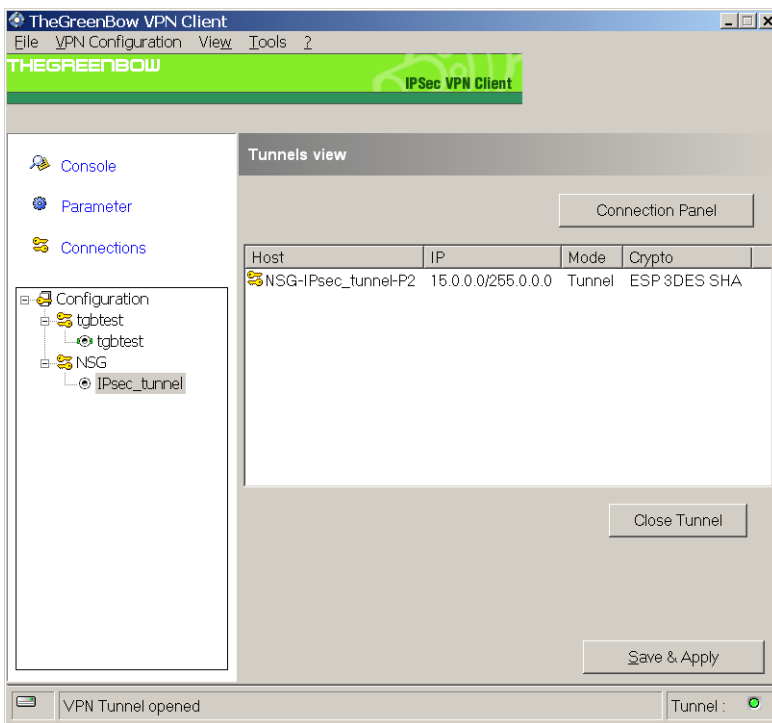


Каждый туннель устанавливается и разрывается вручную по нажатию кнопки Open/Close Tunnel в окне Phase 2.

Дополнительные параметры для всех туннелей, такие как время жизни ключей, можно установить в окне Parameter.



Установленные туннели можно посмотреть в окне Connections.



### Настройка Linsys IPsec Tool (бесплатный, [http://sourceforge.net/project/showfiles.php?group\\_id=139613](http://sourceforge.net/project/showfiles.php?group_id=139613))

Данный клиент может работать как в режиме одиночного хоста, так и в режиме шлюза, за которым расположен закрываемая подсеть (при соответствующих настройках *access-list* на устройстве NSG). При этом в режиме одиночного хоста внутренний IP-адрес должен совпадать с внешним (см. рисунок).

Конфигурация устройства NSG:

```
!
nsg
access-list ext-ip 153
  add 1 permit ip 15.0.0.0 0.255.255.255 host 10.0.2.12
  exit
crypto transform-set ts1 esp 3des-sha-hmac
crypto isakmp key 12345678 address 10.0.2.12 10.0.2.31
crypto map CM1
  3
    method ipsec-isakmp
    set transform-set ts1
    set peer 10.0.2.12
    match address 153
  exit
card s1 um-et100
port s1
  ip address 10.0.2.31/8
  crypto map CM1
  exit
port eth0
  ip address 15.0.0.1/8
  exit
exit
!
```

Если в поле Local Address/Network Mask и в *access-list* указана какая-либо другая подсеть (подключённая к другому интерфейсу хоста), то пакеты из этой сети будут направляться в туннель. Однако пакеты, отправляемые с самого Windows-хоста, будут иметь адрес источника, равный внешнему IP-адресу (10.0.2.12) и, соответственно, передаваться вне туннеля.

