**VPN**

**GRE**

**Generic Routing Encapsulation** – очень простой протокол туннелирования. Грубо говоря, это означает, что берутся ваши изначальные данные вместе со служебными заголовками (как правило, это IP, но может быть и Ethernet и ATM), упаковываются в пакет и передаются по публичной сети. На конечном узле заголовки нового пакета снимаются, а ваши данные в исходном виде продолжают своё путешествие.

Настраивается GRE-туннель следующим образом:

*interface Tunnel0*

*ip address [address mask]*

Поскольку туннель является виртуальным L3 интерфейсом, через который у нас будет происходить маршрутизация, ему должен быть назначен IP-адрес, который выбирается согласно вашему IP-плану, вероятно, из приватной сети.

В качестве адреса источника можно выбрать как IP-адрес выходного интерфейса (белый адрес, предоставленный провайдером), так и его имя.

*tunnel source [IP-адрес выходного интерфейса]*

Адрес destination – публичный адрес удалённой стороны:

*tunnel destination [публичный адрес удалённой стороны]*

По умолчанию GRE не проверяет доступность адреса назначения и сразу отправляет туннель в Up. Но стоит только добавить в туннельный интерфейс команду keepalive X, как маршрутизатор начинает отсылать кипалайвы и не поднимется, пока не будет ответа.

GRE-туннели являются однонаправленными, и обычно подразумевается наличие обратного туннеля на другой стороне, хотя вообще говоря, это необязательно.

GRE туннель представляет собой соединение точка - точка, его можно считать одной из разновидностей VPN туннеля, без шифрования. Основное достоинство GRE это возможность передавать широковещательный трафик, что позволяет пропускать через такой туннель протоколы маршрутизации использующие его, IPSec туннели в чистом виде этого не могут. Причин для организации GRE туннеля может быть множество от банальной необходимости пробросить свою сеть через чужое IP пространство до использования протоколов OSPF, RIPv2, EGRP совместно с IPSec. Так же GRE, в отличии от IPIP, может помочь пробросить немаршрутизируюмые протоколы, такие как NetBios, IPX, AppleTalk.

Туннелирование увеличивает нагрузку на систему и сеть, потому что добавляются дополнительные IP-заголовки. Таким образом, если обычный размер пакета (MTU) в сети равен 1500 байтам, то при пересылке по туннелю, пакет будет меньше, 1476 байт для GRE и 1480 байт для IPIP. Задать MTU можно вручную или с помощью PMTUD (path MTU discovery). Основная проблема в ручной настройке MTU и/или MSS состоит в том, что по пути между вашими площадками может оказаться линк с MTU, скажем, 1300. Тут на помощь может прийти PMTUD. Протокол целиком и полностью полагается на ICMP протокол диагностики перегрузки сети unreachable messages, которые должны быть разрешены на всем пути между соседями. Cisco рекомендует устанавливать MTU в 1400 байт вне зависимости от того работает GRE поверх IPSec в туннельном или в транспортном режиме.

Туннелирование подразумевает три протокола:

инкапсулированный протокол

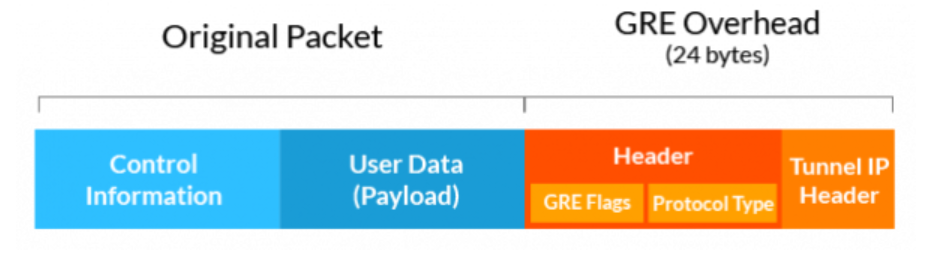
протокол инкапсуляции (GRE)

транспортный протокол (IP)

**Инкапсуляция заголовка GRE в IP-пакет**

GRE-заголовок накладывается «поверх» стандартного IP-пакета. При этом в самом GRE-заголовке содержится так называемый Tunnel IP Header. Именно в нем содержится информация о tunnel source и tunnel destination.

Данные адреса вкладываются в основной пакет, когда он отправляется в публичную сеть. В поле Control Information оригинального IP-пакета содержатся исходные IP-адреса источника и назначения. Таким образом, локальные серые IP-адреса скрыты в пакете, а в маршрутизации участвуют только те адреса, которые мы указали в **tunnel source** и **tunnel destination**. При передаче пакета в локальную сеть GRE-заголовок отбрасывается и остается «чистый» IP-пакет.



Недостатки

* Безопасность. Данные, инкапсулированные в GRE, передаются тем не менее в открытом виде.
* Сложность масштабирования. Если у вас 5-7 филиалов, обслуживание такого количества туннелей ещё кажется возможным, а если их 50? Причём туннелирование трафика зачастую производится на CPU, особенно на младшей и средней линейках, поэтому это лишняя нагрузка на процессор.
* Все филиалы будут взаимодействовать друг с другом через центральный узел, хотя могли бы напрямую.

**IPIP**

Различия между туннель GRE или IPIP:

IPIP — инкапсулирует только unicast IPv4-трафик

GRE — IPv4/IPv6 unicast/multicast трафик

**IPSec**

Сейчас для организации шифрованного VPN-канала используются преимущественно следующие технологии: IPSec (IP Security), OpenVPN и PPTP (Point-to-Point Tunneling Protocol).

**IPSec** – это не протокол, это стандарт, включающий в себя целых три протокола, каждый со своими функциями:

**ESP** (**Encapsulating Security Payload** – безопасная инкапсуляция полезной нагрузки) занимается непосредственно шифрованием данных, а также может обеспечивать аутентификацию источника и проверку целостности данных

**AH** (**Authentication Header** – заголовок аутентификации) отвечает за аутентификацию источника и проверку целостности данных

**IKE** (**Internet Key Exchange protocol** – протокол обмена ключами) используется для формирования IPSec SA (Security Association), проще говоря, согласования работы участников защищенного соединения. Используя этот протокол, участники договариваются, какой алгоритм шифрования будет использоваться, по какому алгоритму будет производиться (и будет ли вообще) проверка целостности, как аутентифицировать друг друга

Прежде чем переходить дальше, разберемся с термином **SA – Security Association**. SA в общем смысле представляет собой набор параметров защищенного соединения (например, алгоритм шифрования, ключ шифрования), который может использоваться обеими сторонами соединения. У каждого соединения есть ассоциированный с ним SA.

**Как создается защищенное соединение в IPSec:**

* Для начала, участникам надо договориться, какие алгоритмы/механизмы защиты они будут использовать для своего защищенного соединения, поэтому в дело вступает **IKE**. Процесс состоит из двух фаз:

1. Фаза первая: участники аутентифицируют друг друга и договариваются о параметрах установки специального соединения (тоже защищенного), предназначенного только для обмена информацией о желаемых/поддерживаемых алгоритмах шифрования и прочих деталях будущего IPSec-туннеля. Параметры этого мини-туннеля (правильно он называется **ISAKMP Tunnel**) определяются политикой ISAKMP, в режим редактирования которой мы можем попасть из конфигурационного режима командой crypto isakmp policy номер\_политики. Если стороны пришли к соглашению, устанавливается ISAKMP туннель (его наличие можно посмотреть командой show crypto isakmp sa), по которому уже проходит вторая фаза IKE.
2. Фаза вторая: уже доверяющие друг другу участники договариваются о том, как строить основной туннель для данных. Они по очереди предлагают друг другу варианты, указанные в команде crypto ipsec transform-set, и, если приходят к согласию, поднимают основной туннель. Нужно сказать, что, после его установления, вспомогательный ISAKMP туннель никуда не пропадает – он используется для обновления SA основного. Дело в том, что ключи, выбираемые для шифрования информации в IPSec-туннеле, имеют некоторое “время жизни” (может выражаться как в количестве байт, так и в секундах – что первое достигнет порогового значения), по истечение которого должны быть заменены. Это как пароль, который вы меняете раз в час (по умолчанию lifetime IPSec SA составляет 4608000 килобайт/3600 секунд).

* Участники получили шифрованный туннель с параметрами, которые их всех устраивают, и направляют туда потоки данных, подлежащие шифрованию, т.е., подпадающие под указанный в crypto map аксесс-лист.
* Периодически, в соответствии с настроенным lifetime, обновляются ключи шифрования для основного туннеля: участники вновь связываются по ISAKMP-туннелю, проходят вторую фазу и устанавливают новые SA.

Теперь немного о трансформ-сете и чем отличается ESP от AH. Как будут шифроваться наши данные, идущие через туннель, определяет команда *crypto ipsec transform-set имя\_сета*, после которой идет название протокола, который будет использован (ESP или AH) + алгоритм, по которому будет работать протокол. Например, команда *crypto ipsec transform-set SET1 esp-aes* даст понять роутеру, что трансформ-сет с именем “SET1”, если он будет применен, будет работать только по протоколу ESP c шифрованием алгоритмом AES.

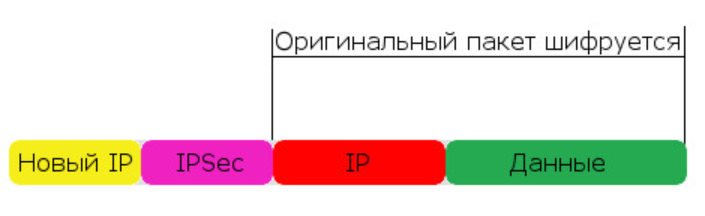
Ну если с ESP все более-менее понятно, его дело-шифровать (обеспечивать конфиденциальность), то что такое AH и зачем он вообще нужен? AH обеспечивает аутентификацию данных, то есть дает уверенность, что эти данные пришли именно от того, с кем мы установили связь, и не были изменены по дороге. Если не углубляться в подробности, работает это так: в каждый пакет между заголовком IP и заголовком транспортного уровня вставляется заголовок AH, в котором присутствует:

* информация, по которой получатель может понять, к какой SA относится данный пакет (т.е., в том числе, по какому алгоритму ему считать хеш для сравнения – MD5 или SHA)
* так называемый ICV (Integrity Check Value), представляющий собой хеш от пакета (на самом деле, не всего пакета, а неизменяемых в процессе путешествия полей), который позволяет однозначно убедиться получателю, что этот пакет не изменялся по дороге, путем вычисления хеша от той же информации и сравнения результата со значением этого поля.

IPSec может работать в двух режимах: туннельном и транспортном.

**Туннельный режим работы IPSec**

В этом режиме берётся изначальный IP-пакет, шифруется полностью, вместе с заголовком IP, добавляется служебная информация IPSec и новый заголовок IP:



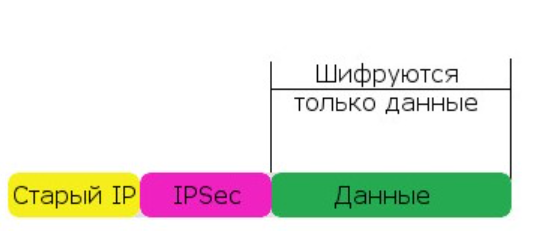
При создании isakmp sa, та сторона, которая инициирует соединение, отправляет все локально настроенные политики isakmp.

Принимающая сторона просматривает по очереди, в порядке приоритетности свои локально настроенные политики. Первая же политика, для которой найдено совпадение, будет использоваться.

Шифрование, происходит практически в самую последнюю очередь, после маршрутизации. И это, кстати, очень важный момент. Вам недостаточно маршрута до публичного адреса пира. Нужен маршрут до приватного, пусть даже он дефолтный. Иначе пакет будет отброшен в соответствии с обычными правилами маршрутизации.

**Транспортный режим работы IPSec**

Он много чем отличается от туннельного, но самое важное – это метод инкапсуляции.



Туннельный режим шифрует изначальный пакет полностью и добавляет новый заголовок IP. Транспортный же шифрует всё, что выше уровня IP, а заголовок IP оставляет без изменений.

Грубо говоря, туннельный режим вы используете для того, чтобы связать две приватные сети через публичную, обеспечив при этом шифрование (Что-то вроде безопасного GRE). Транспортный же актуален тогда, когда IP-связность уже достигнута, но трафик между узлами нужно шифровать.

Удачным примером применения транспортного режима может быть схема сервер-клиент. Например, работа клиент-банка. Сервер и так уже доступен, но трафик нужно зашифровать.

**GRE over IPSec**

Обычный режим, который применяется в подавляющем большинстве случаев, – это GRE over IPSec, то есть данные GRE инкапсулируются заголовками ESP или AH



А IPSec over GRE означает, наоборот, что внутри будут зашифрованные данные IPSec, а сверху заголовки GRE/IP. Они будут не зашифрованы:



Такой вариант возможен, например, если шифрование у вас происходит на отдельном устройстве перед туннелированием.

GRE over IPSec

1 Настраивается GRE туннель

2 Настраивается IPSec

Что при этом меняется в настройке IPSec?

Во-первых, поскольку туннель уже существует (GRE), нет нужды делать его ещё и средствами IPSec – можно перевести его в транспортный режим, тем самым, сэкономив 20 байтов на лишнем IP-заголовке:

*crypto ipsec transform-set AES128-SHA esp-aes esp-sha-hmac*

*mode transport*

Во-вторых, шифроваться должен весь трафик между филиалами, то есть тот, который идёт через туннель, соответственно, нет необходимости прописывать все сети в ACL

**DMVPN**

DMVPN (Dymamic Multipoint VPN) – пропиетарная цисковская.

Решает проблему масштабируемости.

Выбирается центральная точка Hub (или несколько). Она будет сервером, к которому будут подключаться клиенты (Spoke) и получать всю необходимую информацию. При этом:

1) Данные будут зашифрованы IPSec

2) Клиенты могут передавать трафик непосредственно друг другу в обход центрального узла

3) Только на центральном узле необходим статический публичный IP-адрес. Удалённые узлы могут иметь динамический адрес и находиться даже за NATом, используя адреса из частных диапазонов (Технология NAT Traversal ). Но при этом возникают ограничения по части динамических туннелей.