Введение в шифрование IPSec

Содержание

Введение

В этом документе кратко описывается технология IPsec. Здесь приводятся базовые конфигурации для обмена ключами через Интернет (IKE) с использованием предварительных общих ключей, обмена ключами через Интернет с центром сертификации и IPsec. Это неполная документация. Однако этот документ поможет вам понять все необходимые задачи, а также порядок их выполнения.

 **% Warning:** Существуют жесткие ограничения на экспорте сильной криптографии. Если вы нарушаете Федеральный закон США, то вы, не Cisco, считаетесь ответственными. При возникновении любых вопросов отнесенные к экспортному контролю, передайте и Электронная почта к export@cisco.com.

Примечание: Групповая адресация и Широковещательное сообщение не поддерживаются на обычной LAN в туннели LAN или на клиентах VPN, которые завершаются на любых устройствах. Многоадресные пакеты передаются только по туннелям GRE. Это

поддерживается только на маршрутизаторах, но не на концентраторах VPN 3000 или межсетевых экранах (ASA/PIX).

Предварительные условия

Требования

Для этого документа отсутствуют особые требования.

Используемые компоненты

Настоящий документ не имеет жесткой привязки к каким-либо конкретным версиям программного обеспечения и оборудования.

Условные обозначения

Дополнительные сведения об условных обозначениях см. в документе Условные обозначения технических терминов Cisco.

Общие сведения

IPsec является платформой шифрования сетевого уровня следующего поколения для платформ Безопасности Cisco (программное обеспечение Cisco IOS, PIX, и т.д). Первоначально описанный в RFC 1825 - 1829, которые являются теперь устаревшими, IPsec в настоящее время обсуждается во многих документах, представленных Рабочей группой IP-безопасности IETF. В настоящее время IPsec поддерживает одноадресные пакеты IP версии 4. Поддержка IPv6 и многоадресной передачи появится в будущем.

По сравнению с другими криптографическими решениями Cisco IPsec содержит следующие преимущества:

1. **Разных производителей** — Так как платформа IPsec стандартизирована, клиенты не блокированы ни в какой определенный продукт поставщика. IPsec присутствует на маршрутизаторах, межсетевых экранах и клиентских рабочих станциях (Windows, Mac и т. д.).
2. **Масштабируемость** — IPsec разработан с крупными предприятиями в памяти. Поэтому предусмотрено встроенное управление ключами.

Примечание: В то время как некоторые платформы Cisco могут использовать IPSec, этот документ ориентирован на программное обеспечение Cisco IOS.

Криптоязык (словарь)

Необходимо познакомиться с приведенной ниже терминологией, прежде чем изучать этот документ. Все акронимы, встречающиеся в документе, приведены на этой странице.

Расширенный стандарт шифрования (AES) — AES был завершен как Федеральный стандарт обработки информации (FIPS) (FIPS) - утвержденный криптографический алгоритм, который будет использоваться, для защиты передачи цифровых данных (FIPS

PUB 197). Стандарт AES основан на алгоритме Rijndael, который определяет, как следует использовать ключи длиной 128, 192 или 256 битов для шифрования блоков длиной 128, 192 или 256 битов. Доступны все девять комбинаций длины ключа и блока.

Заголовок аутентификации (AH) — Это - протокол безопасности, который предоставляет опознавательные и дополнительные сервисы определения повторной передачи. Протокол AH встраивается в защищаемые данные, например в полную IP-датаграмму. AH может использоваться отдельно или вместе с Encryption Service Payload (ESP). См. протокол RFC 2402.

Аутентификация — Это - одна из функций платформы IPsec. Аутентификация обеспечивает целостность потока данных и гарантирует его сохранность при передаче. Также она обеспечивает подтверждение происхождения потока данных.

Центр сертификации (CA) — Это - сторонний объект с обязанностью выполнить и отозвать сертификаты. Каждое устройство, имеющее собственный сертификат и открытый ключ центра сертификации, может выполнять аутентификацию любого другого устройства в данном домене центра сертификации. Этот термин также относится к серверному программному обеспечению, предоставляющему соответствующие функции.

Сертификат — криптографически объект со знаком, который содержит идентичность и открытый ключ, привязанный к этой идентичности.

Классическое шифрование — Это - механизм шифрования по собственному алгоритму Cisco, используемый в программном обеспечении Cisco IOS версии 11.2. Классическое шифрование Classic crypto доступно в ПО Cisco IOS версии 11.3. Однако IPsec не является технологией, обратно совместимой с версией Cisco IOS 11.2. Вы также можете найти сведения о classic crypto в литературе по маркетингу, в которой это шифрование называется шифрованием CET (Cisco Encryption Technology).

Список отозванных сертификатов (CRL) — Это - снабженное цифровой подписью сообщение, которое перечисляет все текущие, но отозванные сертификаты, перечисленные данным CA. Это аналогично списку украденных номеров платежных карт, который позволяет магазинам аннулировать недействительные кредитные карты.

Криптокарта — Это - объект Конфигурации программного обеспечения Cisco IOS, который выполняет две первичных функции. Во-первых, он выбирает потоки данных, безопасность которых следует обеспечить. Во-вторых, он определяет политику для этих потоков, а также криптографический одноранговый узел, для которого предназначается этот трафик.

Криптокарта применяется к интерфейсу. Концепция криптокарты была введена в классическом шифровании classic crypto, но была расширено для IPSec.

Целостность данных — Это - механизмы обеспечения целостности данных, с помощью основанного секретного ключа, или общий ключ базировал алгоритмы, которые позволяют получателю части защищенных данных, чтобы проверить, что данные не модифицировались в пути.

Конфиденциальность данных — Это - метод, где защищенными данными манипулируют так, чтобы никакой атакующий не мог считать его. Как правило, это обеспечивается за счет шифрования данных и использования ключей, которые доступны только сторонам, поддерживающим связь друг с другом.

Проверка подлинности источника данных — Это - сервис безопасности, где получатель может проверить, что защищенные данные, возможно, произошли только из отправителя. Для работы этой службы требуется служба целостности данных и механизм распределения ключей, при котором секретный ключ является общим только для отправителя и получателя.

Стандарт шифрования данных (DES) — DES был публикован в 1977 Национальным бюро стандартов и является схемой шифрования секретного ключа на основе алгоритма Люцифер от IBM. Противоположность стандарта DES — открытый ключ. В устройствах Cisco стандарт DES используется в classic crypto (40-битные и 56-битные ключи), IPsec crypto (56-битные ключи) и в сетевых экранах PIX (56-битные ключи).

Диффи-Хеллман — Это - метод установления общего ключа по незащищенным средствам. Этот метод является компонентом Oakley, который также определен в этом списке.

DSS — алгоритм цифровой подписи, разработанный Национальным институтом стандартов и технологий US (NIST) на основе шифрования с открытым ключом. Алгоритм DSS не служит для шифрования датаграмм. Он является компонентом classic crypto, а также карты Redcreek IPsec, но не входит в реализацию IPsec в ПО Cisco IOS.

Модуль сервиса шифрования (ESA) — Это - основанный ускоритель шифрования аппаратных средств, который используется в:

- Маршрутизаторы Cisco 7204 и 7206
- Многосторонний интерфейс второго поколения Processor2-40s (VIP2-40s) на всех Cisco 7500 series routers
- VIP2-40 в маршрутизаторах серии Cisco 7000, на которых установлены процессор RSP7000 и интерфейсные карты шасси RSP7000CI.

В IPsec не используется ускорение ESA, однако IPsec может работать на устройстве с картой ESA, но только на уровне программного обеспечения.

Безопасное закрытие полезной нагрузки (ESP) — протокол безопасности, который предоставляет конфиденциальности данных и защите с необязательной проверкой полдлинности и сервисами определения повторной передачи. ESP полностью инкапсулирует пользовательские данные. ESP может использоваться отдельно или вместе с протоколом AH. См. RFC 2406: Инкапсуляция защищенной полезной нагрузки (ESP) в IP.

Хэш — Это - один путь функция, которая берет входящее сообщение произвольной длины и производит дайджест фиксированной длины. В реализации среды IPsec от корпорации Cisco используется хэширование по алгоритмам Secure Hash Algorithm (SHA) и Message Digest 5 (MD5). Дополнительные сведения см. в определении HMAC.

HMAC — Это - механизм для проверки подлинности сообщений, которая использует криптографические хэши, такие как SHA и MD5. См. RFC 2104 для всестороннего обсуждения HMAC.

Протокол IKE — гибридный протокол, который использует Oakley части и часть другого набора протоколов, названного SKEME в платформе Протокола ISAKMP. IKE используется для установления совместно используемой политики безопасности и аутентифицируемых ключей для сервисов, таких как IPsec, которые требуют ключей. Перед передачей трафика IPsec каждый маршрутизатор, межсетевой экран или хост должен иметь возможность проверить идентификацию другого узла. Для этого можно вручную ввести предварительные

общие ключи на обоих хостах, можно использовать службу CA или выходящую в скором времени безопасную службу DNS (DNSSec). Это - протокол, раньше известный как ISAKMP/Oakley, и определено в RFC 2409: Обмен ключами в Интернете (IKE). К некоторой путанице может привести тот факт, что акронимы ISAKMP и IKE используются в ПО Cisco IOS как синонимы. Однако их функции немного отличаются.

Протокол ISAKMP — Это - структура протокола, которая определяет механику реализации протокола обмена ключами и согласования политики безопасности. ISAKMP определяется как протокол управления сопоставлениями безопасности и ключами в Интернете (Internet Security Association and Key Management Protocol).

Прозрачность NAT IPSec — функция прозрачности NAT IPSec представляет поддержку IP-безопасности (IPsec) трафик для перемещения через Технологию NAT или точки Переадресации точки (PAT) в сети путем адресации ко многим известным несовместимостям между NAT и IPsec. Прослеживание NAT представляет собой функцию, автоматически обнаруживаемую устройствами VPN. Нет необходимости в настройке конфигурации маршрутизаторов, работающих под управлением ПО Cisco IOS версии 12.2(13)T и более поздних версий. Если оба устройства VPN способны осуществлять NAT-T, то прозрачность NAT обнаруживается и согласовывается автоматически.

ISAKMP/Oakley — Видит IKE.

Алгоритм представления сообщения в краткой форме 5 (MD5) — Это - один путь алгоритм хеширования, который производит 128-разрядный хэш. Алгоритмы MD5 и SHA являются разновидностями алгоритма MD4, разработанные для усиления безопасности этого алгоритма хэширования. Алгоритм SHA более безопасен, чем MD4 и MD5. Устройства Cisco используют хэширование для аутентификации в среди IPsec.

Oakley — Это - протокол обмена ключами, который определяет, как получить материал для генерации аутентифицированных ключей. Основным механизмом протокола Oakley служит алгоритм обмена ключами Диффи-Хеллмана. Можно найти стандарт в RFC 2412: Протокол определения ключа OAKLEY.

Безопасная пересылка (Perfect Forward Secrecy, PFS) — безопасная пересылка (PFS) гарантирует, что данный ключ контекста безопасности IPSec не был получен на основании никакой другой тайны, как некоторые другие ключи. Другими словами, если кто-либо взламывает ключ, PFS гарантирует невозможность получения злоумышленником какого-либо другого ключа. Если функция PFS не включена, злоумышленник может взломать секретный ключ контекста безопасности IKE, скопировать все защищенные данные IPsec, а затем использовать свои знания секретного ключа контекста безопасности IKE для взлома других контекстов безопасности IPsec с помощью этого контекста безопасности IKE. При использовании PFS взлом IKE не даст злоумышленнику непосредственного доступа к IPsec. Атакующему придется взламывать каждый контекст безопасности IPsec по отдельности. Реализация IPsec в Cisco IOS IPsec по умолчанию использует PFS группы 1 (D-H 768 бит).

Обнаружение воспроизведения — Это - сервис безопасности, где получатель может отклонить старый или повторяющиеся пакеты для нанесения поражения атак с повторением пакетов. Атаки с повторением пакетов полагаются на атакующего для отсылки более старый или повторяющиеся пакеты к получателю и получателю, чтобы думать, что поддельный трафик легитимен. Обнаружение воспроизведения сделано при помощи порядковых номеров, объединенных с аутентификацией, и является стандартной характеристикой IPsec.

RSA — Это - криптографический алгоритм с открытым ключом, названный в честь его изобретателей, Rivest, Шамира и Адлемена, с переменной длиной ключа. Основным недостатком алгоритма RSA является значительно меньшая скорость обработки по сравнению с другими известными алгоритмами с использованием секретного ключа, например DES. В реализации Cisco IKE используется обмен Диффи-Хеллмана для получения секретных ключей. Аутентификация этого обмена может быть выполнена с помощью RSA или общих секретных ключей. При обмене по методу Диффи-Хеллмана ключ DES никогда не отправляется по сети даже в зашифрованном виде, что отличается от метода шифрования и подписи с использованием RSA. RSA не является свободно доступным и должен лицензироваться в RSA Data Security.

Сопоставление безопасности (SA) - экземпляр политики безопасности, и материал для кодирования применился к потоку данных. Контексты безопасности применяются в протоколах IKE и IPsec. Эти протоколы используют разные контексты безопасности. Сопоставления безопасности IPsec являются однонаправленными, при этом они уникальны в каждом протоколе безопасности. Для обеспечения безопасности магистрали требуется набор контекстов безопасности. Каждое направление и протокол должны иметь собственный контекст безопасности. Например, если имеется магистраль, участники которой поддерживают ESP, необходимо по одному контексту безопасности ESP для каждого направления. Уникальная идентификация контекста безопасности реализуется посредством адреса узла назначения (конечная точка соединения по IPsec), протокола безопасности (AH или ESP) и индекса параметра безопасности (SPI).

IKE согласовывает и создает контекст безопасности от имени IPsec. Пользователь имеет возможность создавать контексты безопасности IPsec вручную.

Контекст безопасности IKE используется только IKE. В отличие от контекста безопасности IPsec он является двусторонним.

Защищенный алгоритм хэширования (SHA) — Это - односторонний хэш, выдвинутый NIST. Алгоритм SHA смоделирован на основе MD4 и служит для создания 160-битного дайджеста. Так как SHA создает 160-битный дайджест, этот алгоритм более защищен от атак прямым перебором, чем 128-битные хэши (такие как MD5), но скорость работы этого протокола заметно снижается.

Разделенное туннелирование — Это - процесс разрешения удаленного пользователя VPN для доступа к открытой сети, обычно Интернет, в то же самое время, когда пользователю разрешают обратиться к ресурсам в удаленном офисе. Этот метод сетевого доступа позволяет пользователю получать доступ к удаленным устройствам, таким как сетевой принтер и серверы, и в то же время работать в общей сети (Интернет). Преимуществом использования раздельного туннелирования является устранение узких мест и экономия пропускной способности, так как трафик Интернета не будет проходить через VPN-сервер. Недостаток этого метода заключается в том, что VPN становится объектом атак, так как эта сеть доступна через общую, незащищенную сеть.

Преобразуйте — А, преобразовывают, описывает протокол безопасности (AH или ESP) с его соответствующими алгоритмами. Например, ESP с алгоритмом шифрования DES и HMAC-SHA для аутентификации.

Транспортный режим — Это - режим инкапсуляции для AH/ESP. Транспортный режим инкапсулирует нагрузку на верхние уровни, например использование протоколов TCP и UDP исходной IP-датаграммы. Этот режим может использоваться только, когда одноранговые узлы являются конечными точками связи. Противоположностью транспортного режима

является туннельный режим.

Tunnel mode - инкапсуляция завершенной Дейтаграммы IP для IPsec. Туннельный режим используется для защиты датаграмм, исходящих или предназначенных для систем, отличных от IPsec, например VPN.

Настройте ISAKMP

IKE существует только для создания контекстов безопасности для IPsec. До создания контекста IKE необходимо согласовать контекст безопасности с одноранговым участником. Так как IKE согласовывает собственную политику, можно настроить несколько политик с различными конфигурациями, а затем предоставить двум хостам право на согласование. ISAKMP согласовывает следующие факторы:

- **Алгоритм шифрования** — Это ограничено 56-разрядным DES только.
- **Алгоритм хеширования** — MD5 или SHA
- **Аутентификация** — сигнатуры RSA, Случаи при шифровании RSA (случайные числа) или предварительные общие ключи
- **Срок действия SA** — В секундах

В настоящее время существует два метода настройки ISAKMP:

1. Использование предварительных ключей, которые просты в настройке.
2. **Использование центра сертификации, что позволяет масштабировать решение во всей организации.**

Примечание: Ike согласование сделан на UDP 500. IPsec использует Протоколы "IP" 50 и 51. Убедитесь, что эти порты разрешены во всех списках доступа между одноранговыми участниками.

1. Предварительные совместно используемые ключи

Это простой и быстрый метод настройки IKE. Хотя конфигурация IKE простая и нет необходимости в центре сертификации, масштабируемость остается низкой.

Для настройки IKE следует выполнить следующие действия:

- Настроить пакеты защиты ISAKMP.
- Настроить ключ ISAKMP.

Настроить пакеты защиты ISAKMP

Следующая команда создает объект политики ISAKMP. Можно настроить несколько политик, однако в этом примере используется только одна:

```
dt3-45a(config)#crypto isakmp policy 1 dt3-45a(config-isakmp)#
```

С помощью команды group можно указать размер, который будет использовать в вычислениях Диффи-Хеллмана. Длинна группы 1 768 битов, группы 2 — 1024 бита. Почему можно предпочесть одну группу другой? Не все поставщики поддерживают группу 2. Также для группы 2 понадобиться значительно больше ресурсов ЦП, чем для группы 1. По этой причине не рекомендуется использовать группу 2 на бюджетных маршрутизаторах,

например на маршрутизаторах серии Cisco 2500 и ниже. **Однако группа 2 более безопасна, чем группа 1. Так как в этом примере используется Cisco 4500, будет применяться группа 2. Также следует убедиться, что на одноранговом участнике тоже настроено использование группы 2. По умолчанию используется группа 1. Если выбрать свойства по умолчанию, каналы группы 1 не будут отображены при выполнении команды write terminal.**

```
dt3-45a(config-isakmp)#group 2
```

На этом канале используется алгоритм хэширования MD5. Хотя реализация SHA и MD5 является обязательной, не все одноранговые участники могут быть настроены для согласования какого-либо из этих алгоритмов. По умолчанию в Cisco IOS используется SHA, который обеспечивает лучшую защиту, чем MD5.

```
dt3-45a(config-isakmp)#hash md5
```

В этом случае срок действия контекста безопасности составляет 500 секунд, что показано в следующей команде. Если не задать срок действия, будет использоваться значение по умолчанию, равное 86400 секунд или одному дню. По истечении срока действия в качестве защитной меры выполняется повторное согласование контекста безопасности.

```
dt3-45a(config-isakmp)#lifetime 500
```

В этой команде ключ, который следует использовать для IKE, задается вручную. **Поэтому используется команда pre-share. Два параметра, следующие за командой pre-share: rsa-encr и rsa-sig. Команда rsa-encr задает шифрованные случайные числа RSA, а команда rsa-sig настраивает подпись RSA. Rsa-encr** и **команды rsa-sig** обращены в Использовании раздел CA. **Сейчас же следует помнить, что команда rsa-sig является командой по умолчанию.**

```
dt3-45a(config-isakmp)#authentication pre-share
```

Настроить ключ ISAKMP

В этих командах задаются ключи, которые следует использовать для IKE. В этом случае участник 192.168.10.38 должен использовать тот же инструмент для производства ключей в своей конфигурации.

```
dt3-45a(config-isakmp)#exit dt3-45a(config)#crypto isakmp key Slurpee-Machine address 192.168.10.38
```

Конфигурация IKE настроена. Следующие команды предназначены для настройки IKE на одноранговом участнике. Завершенные конфигурации для обоих маршрутизаторов находятся в разделе Примеров конфигурации этого документа:

```
crypto isakmp policy 1
 hash md5
 group 2
 authentication pre-share
crypto isakmp key Slurpee-Machine address 192.168.10.66
```

2. Используйте CA

Использование центра сертификации — это сложный метод настройки IKE. Так как он очень масштабируем при использовании IPsec, следует использовать IPsec вместо классического шифрования classic crypto. С выходом Cisco IOS версии 11.3(3) многие поставщики ЦС будут поставлять этот продукт. **Изначально большинство конфигураций было основано на предварительных общих ключах.** VeriSign, Entrust, Microsoft и Netscape, а также многие другие компании работают над продуктами, предоставляющими услуги центров сертификации. В этом примере используется центр сертификации VeriSign.

Необходимо выполнить следующие действия, чтобы использовать центр сертификации:

- Создать криптографическую пару RSA для маршрутизатора.
- Запросить сертификат ЦС.
- Зарегистрировать сертификаты для клиентского маршрутизатора.
- Настроить пакеты защиты ISAKMP.

Создать криптографическую пару RSA для маршрутизатора

Команда crypto key gen rsa usage-keys может привести в замешательство. Эта команда создает две пары ключей для RSA:

- одна пара ключей для шифрования
- другая — для цифровых подписей

Пара ключей состоит из открытого ключа и соответствующего закрытого ключа. Если не указать ключи специального применения в конце этой команды, маршрутизатор создаст только одну пару ключей RSA и будет использовать ее как для шифрования, так и для цифровых подписей. Обратите внимание, что эта команда может быть использована для создания ключей DSS. Однако DSS — это часть классического шифрования classic crypto, а не IPsec.

```
dt3-45a(config)#crypto key gen rsa usage-keys The name for the keys will be: dt3-45a.cisco.com %You
already have RSA keys defined for dt3-45a.cisco.com. %Do you really want to replace them? [yes/no] yes
```

Так как в устройстве уже используются ключи RSA, появится запрос на удаление существующих ключей. Подтвердите удаление. Появятся следующие выходные данные:

```
Choose the size of the key modulus in the range of
   360 to 2048 for your Signature keys.
Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the modulus [512]: <return>
Generating RSA keys...
[OK]

Choose the size of the key modulus in the range of
   360 to 2048 for your Encryption keys.
Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the modulus [512]: <return>
Generating RSA keys...
[OK]
dt3-45a(config)#
```

Создается пара ключей RSA с размером по умолчанию 512 битов. Выйдите из режима конфигурации и выполните команду show crypto key mypubkey rsa. Теперь можно просмотреть открытые ключи RSA. Закрытый ключ из пары ключей никогда не отображается. Даже если не существовало каких-либо ранее созданных ключей, выходные данные будут аналогичны предыдущим.

Примечание: Не забудьте сохранять свою конфигурацию, как только вы генерировали свои пары согласованных ключей.

Запросите сертификат CA

Теперь нужно настроить маршрутизатор таким образом, чтобы он мог взаимодействовать с ЦС. Для этого следует выполнить несколько действий. Необходимо согласовать свои действия с администратором ЦС.

В следующих строках конфигурации имя домена прописывается в маршрутизаторе. Создается имя хоста ciscoca-ultra, задается IP-адрес и указываются серверы разрешения имен. Необходимо использовать имена хостов, определенных для ЦС, или настроить на устройстве DNS. Корпорация Cisco рекомендует настроить DNS.

```
dt3-45a(config)#ip host ciscoca-ultra 171.69.54.46 dt3-45a(config)#ip domain-name cisco.com dt3-45a(config)#ip name-server 171.692.132 dt3-45a(config)#ip name-server 198.92.30.32
```

Начните настройку параметров ЦС. verisign-ca — это заменяемое имя.

```
dt3-45a(config)#crypto ca identity verisign-ca dt3-45a(ca-identity)#
```

В этих выходных данных протокол регистрации Cisco использует HTTP для связи с ЦС. Команда dt3-45a(ca-identity)#enrollment url http://ciscoca-ultra приводит к переходу маршрутизатора по указанному URL для начала сеанса связи с ЦС. Команда dt3-45a(ca-identity)#crypto ca authenticate verisign-ca служит для получения сертификата из ЦС. Прежде чем получать сертификаты из ЦС, необходимо убедиться, что это фактический центр сертификации. Проверьте сертификат центра сертификации вместе с администратором ЦС для обеспечения аутентичности.

```
dt3-45a(ca-identity)#enrollment url http://ciscoca-ultra dt3-45a(ca-identity)#exit dt3-45a(ca-identity)#crypto ca authenticate verisign-ca
```

[Зарегистрировать сертификаты для клиентского маршрутизатора]

Выполните команду crypto ca enroll verisign-ca, чтобы начать регистрацию в ЦС. Необходимо выполнить несколько действий. Сначала необходимо проверить идентичность ЦС, затем ЦС должен проверить идентичность маршрутизатора. Если необходимо отозвать сертификат до истечения срока его действия, если было изменено количество интерфейсов на маршрутизаторе, или если есть подозрение, что сертификатом пользуются злоумышленники, необходимо предоставить пароль для администратора ЦС. Введите пароль, как показано в следующем примере. После ввода пароля маршрутизатор продолжит работу.

```
dt3-45a(config)#crypto ca enroll verisign-ca %Start certificate enrollment .. %Create a challenge password. You will need to verbally provide this password to the CA Administrator in order to revoke your certificate. For security reasons your password will not be saved in the configuration. Please make a note of it. Password: Re-enter password:
```

Теперь вы видите «отпечатки пальцев» ЦС. Проверьте эти отпечатки вместе с администратором ЦС. **Кроме того, если выполнить команду show crypto ca cert, будут отображены сертификаты ЦС помимо ваших собственных сертификатов.** Сертификаты ЦС обозначены как ожидающие подтверждения.

```
% The subject name for the keys will be: dt3-45a.cisco.com
% Include the router serial number in the subject name? [yes/no]: yes
% The serial number in the certificate will be: 01204044
% Include an IP address in the subject name? [yes/no]: yes
Interface: Ethernet 0
Request certificate from CA? [yes/no]: yes
```

Свяжитесь с администратором ЦС, так как он должен подтвердить идентичность узла перед выдачей сертификата. После выдачи сертификата центром сертификации этот сертификат становится доступным для использования. На этом завершается регистрация сертификата в ЦС. Однако еще не все сделано. Необходимо настроить объекты политики ISAKMP.

[Настроить пакеты защиты ISAKMP]

В этих выходных данных используется команда по умолчанию rsa-sig. Можно использовать

несколько пакетов защиты, однако в этом примере используется только один. В случае использования нескольких пакетов защиты политики предоставляются одноранговому участнику последовательно, и участник выбирает используемую политику. Это следует применять в том случае, если известно, что не все участники поддерживают определенные функции. Маршрутизатор не будет пытаться согласовывать функции, которые для него непонятны. **Например, если была настроена политика для rsa-sig, но отсутствуют сертификаты, маршрутизатор не станет согласовывать эту политику.**

```
dt3-45a(config)#crypto isakmp policy 1 dt3-45a(config-isakmp)#hash md5 dt3-45a(config-isakmp)#lifetime 4000 dt3-45a(config-isakmp)#exit
```

Настройте IPsec

После определения предварительных общих ключей или настройки ЦС, а также после настройки IKE необходимо настроить IPsec. Независимо от используемого метода IKE действия по конфигурации IPsec не отличаются.

Для настройки IPsec следует выполнить следующие действия:

- Создайте расширенный список ACL.
- Создайте IPsec, преобразовывают (s).
- Создание криптокарты.
- Примените криптокарту к интерфейсу.

Создайте расширенный список ACL

Следующая команда является очень простым списком ACL, который позволяет маршрутизаторам взаимодействовать, например поддерживать сеансы Telnet.

```
dt3-45a(config)#access-list 101 permit ip host 192.168.10.38 host 192.168.10.66
```

Более реалистичный ACL приведен ниже. Эта команда является обычным расширенным списком ACL, где 192.168.3.0 является подсетью, находящейся за рассматриваемым маршрутизатором, а 10.3.2.0 — это подсеть за маршрутизатором-участником. **Помните, что permit включает шифрование, а deny запрещает шифрование.**

```
dt3-45a(config)#access-list 101 permit ip 192.168.3.0 0.0.0.255 10.3.2.0 0.0.0.255
```

Создайте IPsec, преобразовывают (s)

Создайте три набора преобразований. Первый использует только ESP, второй - AH вместе с ESP и последний — только AH. Во время согласования контекста безопасности IPsec все три набора предоставляются одноранговому участнику, который выбирает один из них. **Также для всех трех наборов преобразований используется один туннельный режим.** Транспортный режим может быть использован, только когда конечные точки шифрования также являются конечными точками связи. **Транспортный режим может быть определен командой mode transport в конфигурации transform-set.** Туннельный режим используется в основном для работы с VPN. **Также обратите внимание, что esp-rfc1829 и ah-rfc1828 основаны на исходных документах RFC для этой технологии и являются устаревшими платформами, включенными для обратной совместимости. Не все поставщики поддерживают эти платформы, однако некоторое поставщики поддерживают только их.**

Наборы преобразований в этих командах не являются самыми практичными. Например, наборы PapaBear и BabyBear имеют связанные поднаборы преобразований. **Используйте esp-rfc1829 и ah-rfc1828 в одном наборе преобразований.**

```
dt3-45a(config)#crypto ipsec transform-set PapaBear esp-rfc1829 dt3-45a(cfg-crypto-trans)#exit dt3-45a(config)#crypto ipsec transform-set MamaBear ah-md5-hmac esp-des dt3-45a(cfg-crypto-trans)#exit dt3-45a(config)#crypto ipsec transform-set BabyBear ah-rfc1828 dt3-45a(cfg-crypto-trans)#exit dt3-45a(config)#
```

Создание криптокарты

Метка ipsec-isakmp оповещает маршрутизатор о том, что эта криптокарта является криптокартой IPsec. Хотя в этой криптокарте определен только один участник, можно добавить в криптокарту дополнительных участников. **Параметр session key lifetime может быть выражен в килобайтах (после указанного объема трафика следует сменить ключ) или в секундах, как показано в следующих командах.** Целью является затруднить попытки взлома. **Команда set transform-set служит для сопоставления преобразований с криптокартой.** Кроме того, большое значение имеет порядок определения преобразований. Более предпочтительным в этой конфигурации является MamaBear, а остальные в нисходящей последовательности до BabyBear. **Команда match address 101 означает использование списка доступа 101 для определения релевантности трафика.** Можно использовать несколько криптокарт с одинаковым именем (в этом примере — armadillo) и различные номера последовательности (в этом примере 10). Использование нескольких криптокарт с различными последовательными номерами позволяет сочетать классическое шифрование и IPSec. Здесь можно также изменять свою конфигурацию PFS. По умолчанию в этом примере используется PFS-группа 1. Можно задать PFS-группу 2 или отключить эту функцию вообще, что настоятельно не рекомендуется делать.

```
dt3-45a(config)#crypto map armadillo 10 ipsec-isakmp dt3-45a(config-crypto-map)#set peer 192.168.10.38
dt3-45a(config-crypto-map)#set session-key lifetime seconds 4000 dt3-45a(config-crypto-map)#set
transform-set MamaBear PapaBear BabyBear dt3-45a(config-crypto-map)#match address 101
```

Применение криптокарты к интерфейсу

Эти команды применяют криптокарту к интерфейсу. Можно назначить только один набор криптокарт для интерфейса. Если несколько криптокарт имеют одинаковое имя, но разные номера последовательностей, они считаются частью одного набора и все применяются к интерфейсу. **Устройство защиты оценивает запись криптокарты crypto map, начиная с наименьшего номера в последовательности.**

```
dt3-45a(config)#interface e0 dt3-45a(config-if)#crypto map armadillo
```

Обсуждение памяти и CPU

Пакеты IPsec обрабатываются медленнее, чем пакеты, используемые при классическом шифровании. В основе этого лежит несколько причин, которые могут вызвать серьезные проблемы с производительностью:

1. IPsec использует расширение пакетов, что, скорее всего, потребует фрагментации и последующей сборки датаграмм IPsec.
2. Скорее всего, зашифрованные пакеты проходят аутентификацию, что означает необходимость выполнения двух криптографических операций для каждого пакета.
3. Алгоритмы аутентификации работают медленно, хотя была проделана работа по ускорению расчетов Диффи-Хеллмана.

Кроме того, обмен ключами по методу Диффи-Хеллмана, используемый в IKE, является возведением в степень очень больших чисел (от 768 до 1024 байтов) и может занять до четырех секунд на Cisco 2500. Производительность RSA зависит от размера простого числа, выбранного для пары ключей RSA.

На каждом маршрутизаторе база данных контекстов безопасности занимает примерно 300 байтов и по 120 байтов каждый хранящийся в ней контекст безопасности. Если используются два контекста безопасности IPsec, один входящий и второй исходящий, в большинстве случаев необходимо 540 байтов. Каждая запись контекста безопасности IKE составляет 64 байта. Один контекст безопасности IPsec для потока данных используется в том случае, когда связь является односторонней.

При работе IPsec и IKE влияют на производительность. Метод Диффи-Хеллмана, аутентификация с использованием открытого ключа, а также шифрование и расшифровка требуют значительного количества ресурсов. Многие усилия затрачиваются на снижение этого воздействия.

Незначительное снижение производительности наблюдается для незашифрованных пакетов, которые проходят через интерфейс с поддержкой шифрования crypto. Это происходит потому, что все пакеты должны проверяться по криптокарте. Пакеты, проходящие через маршрутизатор в обход интерфейса, поддерживающего crypto, не влияют на производительность. Наибольшее воздействие оказывается на зашифрованные потоки данных.

Используйте группу 1 для обмена ключами по методу Диффи-Хеллмана в контексте IKE, используйте MD5 в качестве алгоритма хэширования и задайте более длительные сроки действия, чтобы снизить воздействие подсистемы шифрования на остальные интерфейсы маршрутизатора. Улучшив производительность, вы ослабите криптографию. В конечном итоге решение о том, какие функции использовать, а какие нет, принимает клиент исходя из собственной политики безопасности.

Выходные данные для команды "show"

Примечание: Перехваты в этих разделах взяты от другой серии тестов, чем используемые в предыдущих разделах этого документа. Следовательно, в них могут содержаться другие IP-адреса и использоваться немного другие конфигурации. Другая серия **команд показа** предоставлена в разделе Отладочной информации этого документа.

Выходные данные, связанные с IKE

Изучите эти команды, чтобы ознакомиться с регистрацией в ЦС VeriSign. В этих командах показаны открытые ключи, которые используются для шифрования и подписей RSA.

```
dt1-45a#show crypto key mypubkey rsa % Key pair was generated at: 11:31:59 PDT Apr 9 1998 Key name: dt1-45a.cisco.com Usage: Signature Key Key Data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00C11854 39A9C75C 4E34C987 B4D7F36C A058D697 13172767 192166E1 661483DD 0FDB907B F9C10B7A CB5A034F A41DF385 23BEB6A7 C14344BE E6915A12 1C86374F 83020301 0001 % Key pair was generated at: 11:32:02 PDT Apr 9 1998 Key name: dt1-45a.cisco.com Usage: Encryption Key Key Data: 305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00DCF5AC 360DD5A6 C69704CF 47B2362D 65123BD4 424B6FF6 AD10C33E 89983D08 16F1EA58 3700BCF9 1EF17E71 5931A9FC 18D60D9A E0852DDD 3F25369C F09DFB75 05020301 0001
```

В этой команде показаны сертификаты, распознаваемые маршрутизатором. **Сертификат в состоянии ожидания pending был отправлен в ЦС на утверждение.**

```
dt1-45a#show crypto ca certificates Certificate Subject Name Name: dt1-45a.cisco.com Serial Number: 01193485 Status: Available Certificate Serial Number: 650534996414E2BE701F4EF3170EDFAD Key Usage: Signature CA Certificate Status: Available Certificate Serial Number: 3051DF7169BEE31B821DFE4B3A338E5F Key Usage: Not Set Certificate Subject Name Name: dt1-45a.cisco.com Serial Number: 01193485 Status: Available Certificate Serial Number: 1e621faf3b9902bc5b49d0f99dc66d14 Key Usage: Encryption
```

В этих выходных данных показаны открытые ключи маршрутизатора и место их получения.

```
dt1-45a#show crypto key pubkey-chain rsa Codes: M - Manually configured, C - Extracted from certificate Code Usage IP-Address Name C Signing Cisco SystemsDevtestCISCOCA-ULTRA C General 172.21.30.71 dt1-7ka.cisco.com
```

Это таблица контекстов безопасности ISAKMP (IKE). Здесь можно увидеть, какие контексты безопасности существуют в настоящее время между 172.21.30.71 и 172.21.30.70. Одноранговый участник должен поддерживать контекст безопасности в таком же состоянии, которое показано в выходных данных для этого маршрутизатора.

```
dt1-7ka#show crypto isakmp sa dst src state conn-id slot 172.21.30.70 172.21.30.71 QM_IDLE 47 5
```

В этих строках показаны настроенные объекты политики. **В данном случае помимо политики по умолчанию используются политики 1, 2 и 4. Политики предлагаются одноранговому участнику по возрастанию, при этом политика 1 является самой предпочитаемой.**

```
dt1-45a#show crypto isakmp policy Protection suite of priority 1 encryption algorithm: DES - Data Encryption Standard (56 bit keys). hash algorithm: Message Digest 5 authentication method: Rivest-Shamir-Adleman Signature Diffie-Hellman group: #1 (768 bit) lifetime: 180 seconds, no volume limit Protection suite of priority 2 encryption algorithm: DES - Data Encryption Standard (56 bit keys). hash algorithm: Secure Hash Standard authentication method: Pre-Shared Key Diffie-Hellman group: #2 (1024 bit) lifetime: 180 seconds, no volume limit Protection suite of priority 4 encryption algorithm: DES - Data Encryption Standard (56 bit keys). hash algorithm: Message Digest 5 authentication method: Pre-Shared Key Diffie-Hellman group: #2 (1024 bit) lifetime: 180 seconds, no volume limit Default protection suite encryption algorithm: DES - Data Encryption Standard (56 bit keys). hash algorithm: Secure Hash Standard authentication method: Rivest-Shamir-Adleman Signature Diffie-Hellman group: #1 (768 bit) lifetime: 86400 seconds, no volume limit
```

Команды show в контексте IPSec

Эта команда служит для отображения криптокарты ToOtherRouter, списков доступа ACL и предложений преобразований, которые применимы к этой криптокарте, участникам и сроку действия ключей.

```
S3-2513-2#show crypto map Crypto Map "ToOtherRouter" 10 ipsec-isakmp Peer = 192.168.1.1 Extended IP access list 101 access-list 101 permit ip source: addr = 192.168.45.0/0.0.0.255 dest: addr = 192.168.3.0/0.0.0.255 Connection Id = UNSET (0 established, 0 failed) Current peer: 192.168.1.1 Session key lifetime: 4608000 kilobytes/3600 seconds PFS (Y/N): N Transform proposals={ Elvis, Bubba, BarneyDino, }
```

В этой конфигурации используется тот же маршрутизатор, с которого были получены предыдущие выходные данные. Однако заданы другие команды. Будут отображены все предложения преобразования, согласуемые параметры и значения по умолчанию.

```
S3-2513-2#show crypto ipsec transform-set Transform proposal Elvis: { ah-sha-hmac } supported settings = { Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel, }, { esp-des } supported settings = { Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel, }, Transform proposal Bubba: { ah-rfc1828 } supported settings = { Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel, }, { esp-des esp-md5-hmac } supported settings = { Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel, }, Transform proposal BarneyDino: { ah-md5-hmac } supported settings = { Tunnel, }, default settings = { Tunnel, }, will negotiate = { Tunnel, },
```

В этой команде показаны текущие контексты безопасности IPsec на этом маршрутизаторе. В маршрутизаторе используется один контекст безопасности AH для входящего и исходящего трафика.

```
S3-2513-2#show crypto ip session Session key lifetime: 4608000 kilobytes/3600 seconds S3-2513-2#show
crypto ipsec sa interface: Ethernet0 Crypto map tag: ToOtherRouter, local addr. 192.168.1.2 local ident
(addr/mask/prot/port): (192.168.45.0/255.255.255.0/0/0) remote ident (addr/mask/prot/port):
(192.168.3.0/255.255.255.0/0/0) current_peer: 192.168.1.1 PERMIT, flags={origin_is_acl,} #pkts encaps: 0,
#pkts encrypt: 0, #pkts digest 0 #pkts decaps: 0, #pkts decrypt: 0, #pkts verify 0 #send errors 5, #recv
errors 0 local crypto endpt.: 192.168.1.2, remote crypto endpt.: 192.168.1.1 path mtu 1500, media mtu
1500 current outbound spi: 25081A81 inbound esp sas: inbound ah sas: spi: 0x1EE91DDC(518594012)
transform: ah-md5-hmac , in use settings ={Tunnel, } slot: 0, conn id: 16, crypto map: ToOtherRouter sa
timing: remaining key lifetime (k/sec): (4608000/3423) replay detection support: Y outbound esp sas:
outbound ah sas: spi: 0x25081A81(621288065) transform: ah-md5-hmac , in use settings ={Tunnel, } slot: 0,
conn id: 17, crypto map: ToOtherRouter sa timing: remaining key lifetime (k/sec): (4608000/3424) replay
detection support: Y
```

Примеры конфигураций

В этой конфигурации используются предварительные общие ключи. Эта конфигурация маршрутизатора используется для создания выходных данных отладки, перечисленных в разделе Отладочной информации. Данная конфигурация позволяет сети под названием «X», расположенной за маршрутизатором источника, взаимодействовать с сетью под названием «Y», расположенной за одноранговым маршрутизатором. Консультируйтесь с Документацией ПО Cisco IOS для своей версии Cisco IOS или используйте Средство поиска команд Command Lookup Tool (только зарегистрированные клиенты) для получения дополнительной информации об определенной команде. Этот инструмент позволяет пользователю просматривать подробное описание определенной команды или рекомендации по конфигурации этой команды.

Схема сети

Конфигурации

- Исходный маршрутизатор
- Одноранговый маршрутизатор

Исходный маршрутизатор

```
Current configuration:
!
version 11.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname goss-e4-2513
!
enable secret 5 $1$ZuRD$YBaAh3oIv4iltIn0TMCUX1
enable password ww
!
!--- IKE configuration crypto isakmp policy 1 hash md5
authentication pre-share crypto isakmp key Slurpee-Machine
address 20.20.20.21 ! !--- IPsec configuration crypto ipsec
transform-set BearPapa esp-rfc1829 crypto ipsec transform-set
BearMama ah-md5-hmac esp-des crypto ipsec transform-set
BearBaby ah-rfc1828 ! crypto map armadillo 1 ipsec-isakmp set
peer 20.20.20.21 set security-association lifetime seconds
190 set transform-set BearPapa BearMama BearBaby !--- Traffic
to encrypt match address 101 ! interface Ethernet0 ip address
60.60.60.60 255.255.255.0 no mop enabled ! interface Serial0
ip address 20.20.20.20 255.255.255.0 no ip mroute-cache no
```

```
fair-queue crypto map armadillo ! interface Serial1 no ip
address shutdown ! interface TokenRing0 no ip address
shutdown ! ip classless ip route 0.0.0.0 0.0.0.0 20.20.20.21
!--- Traffic to encrypt access-list 101 permit ip 60.60.60.0
0.0.0.255 50.50.50.0 0.0.0.255 dialer-list 1 protocol ip
permit dialer-list 1 protocol ipx permit ! line con 0 exec-
timeout 0 0 line aux 0 line vty 0 4 password ww login ! end
```

Одноранговый маршрутизатор

```
Current configuration:
!
version 11.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname goss-c2-2513
!
enable secret 5 $1$DBTl$Wtg2eS7Eb/Cw5l.nDhkEi/
enable password ww
!
ip subnet-zero
!
!--- IKE configuration crypto isakmp policy 1 hash md5
authentication pre-share crypto isakmp key Slurpee-Machine
address 20.20.20.20 ! !--- IPsec configuration crypto ipsec
transform-set PapaBear esp-rfc1829 crypto ipsec transform-set
MamaBear ah-md5-hmac esp-des crypto ipsec transform-set
BabyBear ah-rfc1828 ! ! crypto map armadillo 1 ipsec-isakmp
set peer 20.20.20.20 set security-association lifetime
seconds 190 set transform-set MamaBear PapaBear BabyBear !---
Traffic to encrypt match address 101 ! ! ! interface
Ethernet0 ip address 50.50.50.50 255.255.255.0 no ip
directed-broadcast ! interface Serial0 ip address 20.20.20.21
255.255.255.0 no ip directed-broadcast no ip mroute-cache no
fair-queue clockrate 9600 crypto map armadillo ! interface
Serial1 no ip address no ip directed-broadcast shutdown !
interface TokenRing0 no ip address no ip directed-broadcast
shutdown ! ip classless ip route 0.0.0.0 0.0.0.0 20.20.20.20
!--- Traffic to encrypt access-list 101 permit ip 50.50.50.0
0.0.0.255 60.60.60.0 0.0.0.255 dialer-list 1 protocol ip
permit dialer-list 1 protocol ipx permit ! ! line con 0 exec-
timeout 0 0 transport input none line aux 0 line aux 0 line
vty 0 4 password ww login ! end
```

Данные отладки

В этом разделе содержатся выходные данные отладки нормального сеанса IKE/IPSec между двумя маршрутизаторами. Конфигурации взяты из раздела "Примеры конфигураций" этого документа. Маршрутизаторы используют предварительный общий ключ. **На обоих маршрутизаторах включены команды debug crypto isakmp, debug crypto ipsec и debug crypto engine.** Это было протестировано с помощью расширенной проверки подключения с помощью команды ping с интерфейса ethernet исходного маршрутизатора до интерфейса ethernet одноранговой маршрутизатора (60.60.60.60 до 50.50.50.50).

Примечание: Синий, выражения, выделенные курсивом в этом примере отладочных выходных данных являются примечаниями, чтобы помочь вам придерживаться того, что происходит, они не часть выходных данных отладки.

Исходный маршрутизатор

```
goss-e4-2513#show clock goss-e4-2513#ping Protocol [ip]:
Target IP address: 50.50.50.50 Repeat count [5]: 10 Datagram
size [100]: Timeout in seconds [2]: Extended commands [n]: y
Source address or interface: 60.60.60.60 Type of service [0]:
Set DF bit in IP header? [no]: Validate reply data? [no]:
Data pattern [0xABCD]: Loose, Strict, Record, Timestamp,
Verbose[none]: Sweep range of sizes [n]: Type escape sequence
to abort. Sending 10, 100-byte ICMP Echos to 50.50.50.50,
timeout is 2 seconds: Apr 2 12:03:55.347: IPSEC(sa_request):
, (key eng. msg.) src= 20.20.20.20, dest= 20.20.20.21,
src_proxy= 60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0x0(0), conn_id= 0, keysize= 0, flags= 0x4004 Apr 2
12:03:55.355: IPSEC(sa_request): , (key eng. msg.) src=
20.20.20.20, dest= 20.20.20.21, src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= AH,
transform= ah-md5-hmac , lifedur= 190s and 4608000kb, spi=
0x0(0), conn_id= 0, keysize= 0, flags= 0x4004 Apr 2
12:03:55.363: IPSEC(sa_request): , (key eng. msg.) src=
20.20.20.20, dest= 20.20.20.21, src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-des , lifedur= 190s and 4608000kb, spi=
0x0(0), conn_id= 0, keysi.ze= 0, flags= 0x4004 Apr 2
12:03:55.375: IPSEC(sa_request): , (key eng. msg.) src=
20.20.20.20, dest= 20.20.20.21, src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= AH,
transform= ah-rfc1828 , lifedur= 190s and 4608000kb, spi=
0x0(0), conn_id= 0, keysize= 0, flags= 0x4004 !--- Note that
the router offers to the peer all of the !--- available
transforms. Apr 2 12:03:55.391: ISAKMP (14): beginning Main
Mode exchange Apr 2 12:03:57.199: ISAKMP (14): processing SA
payload. message ID = 0 Apr 2 12:03:57.203: ISAKMP (14):
Checking ISAKMP transform 1 against priority 1 policy Apr 2
12:03:57.203: ISAKMP: encryption DES-CBC Apr 2 12:03:57.207:
ISAKMP: hash MD5 Apr 2 12:03:57.207: ISAKMP: default group 1
Apr 2 12:03:57.207: ISAKMP: auth pre-share Apr 2
12:03:57.211: ISAKMP (14): atts are acceptable. Next payload
is 0 Apr 2 12:03:57.215: Crypto engine 0: generate alg param
Apr 2 12:03:5.8.867: CRYPTO_ENGINE: Dh phase 1 status: 0 Apr
2 12:03:58.871: ISAKMP (14): SA is doing pre-shared key
authentication.. Apr 2 12:04:01.291: ISAKMP (14): processing
KE payload. message ID = 0 Apr 2 12:04:01.295: Crypto engine
0: generate alg param Apr 2 12:04:03.343: ISAKMP (14):
processing NONCE payload. message ID = 0 Apr 2 12:04:03.347:
Crypto engine 0: create ISAKMP SKEYID for conn id 14 Apr 2
12:04:03.363: ISAKMP (14): SKEYID state generated Apr 2
12:04:03.367: ISAKMP (14): processing vendor id payload Apr 2
12:04:03.371: ISAKMP (14): speaking to another IOS box! Apr 2
12:04:03.371: generate hmac context for conn id 14 Apr 2
```

```
12:04:03.615: ISAKMP (14): processing ID payload. message ID
= 0 Apr 2 12:04:03.615: ISAKMP (14): processing HASH payload.
message ID = 0 Apr 2 12:04:03.619: generate hmac context for
conn id 14 Apr 2 12:04:03.627: ISAKMP (14): SA has been
authenticated Apr 2 12:04:03.627: ISAKMP (14): beginning
Quick Mode exchange, M-ID of 1628162439 !--- These lines
represent verification that the policy !--- attributes are
fine, and the final authentication of the IKE SA. !--- Once
the IKE SA is authenticated, a valid IKE SA exists. !--- New
IKE kicks off IPsec negotiation: Apr 2 12:04:03.635:
IPSEC(key_engine): got a queue event... Apr 2 12:04:03.635:
IPSEC(spi_response): getting spi 303564824ld for SA .!!!from
20.20.20.21 to 20.20.20.20 for prot 3 Apr 2 12:04:03.639:
IPSEC(spi_response): getting spi 423956280ld for SA from
20.20.20.21 to 20.20.20.20 for prot 2 Apr 2 12:04:03.643:
IPSEC(spi_response): getting spi 415305621ld for SA from
20.20.20.21 to 20.20.20.20 for prot 3 Apr 2 12:04:03.647:
IPSEC(spi_response): getting spi 218308976ld for SA from
20.20.20.21 to 20.20.20.20 for prot 2 Apr 2 12:04:03.891:
generate hmac context for conn id 14 Apr 2 12:04:04.!!
Success rate is 50 percent (5/10), round-trip min/avg/max =
264/265/268 ms goss-e4-2513#723: generate hmac context for
conn id 14 Apr 2 12:04:04.731: ISAKMP (14): processing SA
payload. message ID = 1628162439 Apr 2 12:04:04.731: ISAKMP
(14): Checking IPSec proposal 1 Apr 2 12:04:04.735: ISAKMP:
transform 1, ESP_DES_IV64 Apr 2 12:04:04.735: ISAKMP:
attributes in transform: Apr 2 12:04:04.735: ISAKMP: encaps
is 1 Apr 2 12:04:04.739: ISAKMP: SA life type in seconds Apr
2 12:04:04.739: ISAKMP: SA life duration (basic) of 190 Apr 2
12:04:04.739: ISAKMP: SA life type in kilobytes Apr 2
12:04:04.743: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50
0x0 Apr 2 12:04:04.747: ISAKMP (14): atts are acceptable. !--
- The ISAKMP debug is listed because IKE is the !--- entity
that negotiates IPsec SAs on behalf of IPsec. Apr 2
12:04:04.747: IPSEC(validate_proposal_request): proposal part
#1, (key eng. msg.) dest= 20.20.20.21, src= 20.20.20.20,
dest_proxy= 50.50.50.0/255.255.255.0/0/0 (type=4), src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 0s and 0kb, spi= 0x0(0),
conn_id= 0, keysize= 0, flags= 0x4 Apr 2 12:04:04.759: ISAKMP
(14): processing NONCE payload. message ID = 1628162439 Apr 2
12:04:04.759: ISAKMP (14): processing ID payload. message ID
= 1628162439 Apr 2 12:04:04.763: ISAKMP (14): processing ID
payload. message ID = 1628162439 Apr 2 12:04:04.767: generate
hmac context for conn id 14 Apr 2 12:04:04.799: ISAKMP (14):
Creating IPSec SAs Apr 2 12:04:04.803: inbound SA from
20.20.20.21 to 20.20.20.20 (proxy 50.50.50.0 to 60.60.60.0)
Apr 2 12:04:04.803: has spi 303564824 and conn_id 15 and
flags 4 Apr 2 12:04:04.807: lifetime of 190 seconds Apr 2
12:04:04.807: lifetime of 4608000 kilobytes Apr 2
12:04:04.811: outbound SA from 20.20.20.20 to 20.20.20.21
(proxy 60.60.60.0 to 50.50.50.0) Apr 2 12:04:04.811: has spi
183903875 and conn_id 16 and flags 4 Apr 2 12:04:04.815:
lifetime of 190 seconds Apr 2 12:04:04.815: lifetime of
4608000 kilobytes Apr 2 12:04:04.823: IPSEC(key_engine): got
a queue event... Apr 2 12:04:04.823: IPSEC(initialize_sas): ,
(key eng. msg.) dest= 20.20.20.20, src= 20.20.20.21,
dest_proxy= 60.60.60.0/255.255.255.0/0/0 (type=4), src_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0x12180818(303564824), conn_id= 15, keysize= 0, flags= 0x4
Apr 2 12:04:04.831: IPSEC(initialize_sas): , (key eng. msg.)
src= 20.20.20.20, dest= 20.20.20.21, src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), dest_proxy=
```

```
50.50.50.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0xAF62683(183903875), conn_id= 16, keysize= 0, flags= 0x4 Apr
2 12:04:04.839: IPSEC(create_sa): sa created, (sa) sa_dest=
20.20.20.20, sa_prot= 50, sa_spi= 0x12180818(303564824),
sa_trans= esp-rfc1829 , sa_conn_id= 15 Apr 2 12:04:04.843:
IPSEC(create_sa): sa created, (sa) sa_dest= 20.20.20.21,
sa_prot= 50, sa_spi= 0xAF62683(183903875), sa_trans= esp-
rfc1829 , sa_conn_id= 16 !--- These lines show that IPsec SAs
are created and !--- encrypted traffic can now pass.
```

Выходные данные по команде "show" для маршрутизатора источника после согласования IKE/IPSec Negotiation

```
goss-e4-2513#
goss-e4-2513#show crypto isakmp sa dst src state conn-id slot
20.20.20.21 20.20.20.20 QM_IDLE 14 0 goss-e4-2513#show crypto
ipsec sa interface: Serial0 Crypto map tag: armadillo, local
addr. 20.20.20.20 local ident (addr/mask/prot/port):
(60.60.60.0/255.255.255.0/0/0) remote ident
(addr/mask/prot/port): (50.50.50.0/255.255.255.0/0/0)
current_peer: 20.20.20.21 PERMIT, flags={origin_is_acl,}
#pkts encaps: 5, #pkts encrypt: 5, #pkts digest 0 #pkts
decaps: 5, #pkts decrypt: 5, #pkts verify 0 #send errors 5,
#recv errors 0 local crypto endpt.: 20.20.20.20, remote
crypto endpt.: 20.20.20.21 path mtu 1500, media mtu 1500
current outbound spi: AF62683 inbound esp sas: spi:
0x12180818(303564824) transform: esp-rfc1829 , in use
settings ={Var len IV, Tunnel, } slot: 0, conn id: 15, crypto
map: armadillo sa timing: remaining key lifetime (k/sec):
(4607999/135) IV size: 8 bytes replay detection support: N
inbound ah sas: outbound esp sas: spi: 0xAF62683(183903875)
transform: esp-rfc1829 , in use settings ={Var len IV,
Tunnel, } slot: 0, conn id: 16, crypto map: armadillo sa
timing: remaining key lifetime (k/sec): (4607999/117) IV
size: 8 bytes replay detection support: N outbound ah sas:
goss-e4-2513#show crypto isakmp policy Protection suite of
priority 1 encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Message Digest 5
authentication method: Pre-Shared Key Diffie-Hellman group:
#1 (768 bit) lifetime: 86400 seconds, no volume limit Default
protection suite encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Secure Hash Standard
authentication method: Rivest-Shamir-Adleman Signature
Diffie-Hellman group: #1 (768 bit) lifetime: 86400 seconds,
no volume limit goss-e4-2513#show crypto map Crypto Map
"armadillo" 1 ipsec-isakmp Peer = 20.20.20.21 Extended IP
access list 101 access-list 101 permit ip 60.60.60.0
0.0.0.255 50.50.50.0 0.0.0.255 Current peer: 20.20.20.21
Security association lifetime: 4608000 kilobytes/190 seconds
PFS (Y/N): N Transform sets={ BearPapa, BearMama, BearBaby, }
```

Одноранговый маршрутизатор с той же последовательностью проверок связи, как и у другой стороны

```
goss-c2-2513#show debug Cryptographic Subsystem: Crypto
ISAKMP debugging is on Crypto Engine debugging is on Crypto
IPSEC debugging is on goss-c2-2513# Apr 2 12:03:55.107:
ISAKMP (14): processing SA payload. message ID = 0 Apr 2
12:03:55.111: ISAKMP (14): Checking ISAKMP transform 1
against priority 1 policy Apr 2 12:03:55.111: ISAKMP:
encryption DES-CBC Apr 2 12:03:55.111: ISAKMP: hash MD5 Apr 2
12:03:55.115: ISAKMP: default group 1 Apr 2 12:03:55.115:
```

```
ISAKMP: auth pre-share Apr 2 12:03:55.115: ISAKMP (14): atts
are acceptable. Next payload is 0 !--- IKE performs its
operation, and then kicks off IPsec. Apr 2 12:03:55.119:
Crypto engine 0: generate alg param Apr 2 12:03:56.707:
CRYPTO_ENGINE: Dh phase 1 status: 0 Apr 2 12:03:56.711:
ISAKMP (14): SA is doing pre-shared key authentication Apr 2
12:03:58.667: ISAKMP (14): processing KE payload. message ID
= 0 Apr 2 12:03:58.671: Crypto engine 0: generate alg param
Apr 2 12:04:00.687: ISAKMP (14): processing NONCE payload.
message ID = 0 Apr 2 12:04:00.695: Crypto engine 0: create
ISAKMP SKEYID for conn id 14 Apr 2 12:04:00.707: ISAKMP (14):
SKEYID state generated Apr 2 12:04:00.711: ISAKMP (14):
processing vendor id payload Apr 2 12:04:00.715: ISAKMP (14):
speaking to another IOS box! Apr 2 12:04:03.095: ISAKMP (14):
processing ID payload. message ID = 0 Apr 2 12:04:03.095:
ISAKMP (14): processing HASH payload. message ID = 0 Apr 2
12:04:03.099: generate hmac context for conn id 14 Apr 2
12:04:03.107: ISAKMP (14): SA has been authenticated Apr 2
12:04:03.111: generate hmac context for conn id 14 Apr 2
12:04:03.835: generate hmac context for conn id 14 Apr 2
12:04:03.839: ISAKMP (14): processing SA payload. message ID
= 1628162439 Apr 2 12:04:03.843: ISAKMP (14): Checking IPSec
proposal 1 Apr 2 12:04:03.843: ISAKMP: transform 1,
ESP_DES_IV64 Apr 2 12:04:03.847: ISAKMP: attributes in
transform: Apr 2 12:04:03.847: ISAKMP: encaps is 1 Apr 2
12:04:03.847: ISAKMP: SA life type in seconds Apr 2
12:04:03.851: ISAKMP: SA life duration (basic) of 190 Apr 2
12:04:03.851: ISAKMP: SA life type in kilobytes Apr 2
12:04:03.855: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50
0x0 Apr 2 12:04:03.855: ISAKMP (14): atts are acceptable. Apr
2 12:04:03.859: IPSEC(validate_proposal_request): proposal
part #1, (key eng. msg.) dest= 20.20.20.21, src= 20.20.20.20,
dest_proxy= 50.50.50.0/255.255.255.0/0/0 (type=4), src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 0s and 0kb, spi= 0x0(0),
conn_id= 0, keysize= 0, flags= 0x4 Apr 2 12:04:03.867: ISAKMP
(14): processing NONCE payload. message ID = 1628162439 Apr 2
12:04:03.871: ISAKMP (14): processing ID payload. message ID
= 1628162439 Apr 2 12:04:03.871: ISAKMP (14): processing ID
payload. message ID = 1628162439 Apr 2 12:04:03.879:
IPSEC(key_engine): got a queue event... Apr 2 12:04:03.879:
IPSEC(spi_response): getting spi 183903875ld for SA from
20.20.20.20 to 20.20.20.21 for prot 3 Apr 2 12:04:04.131:
generate hmac context for conn id 14 Apr 2 12:04:04.547:
generate hmac context for conn id 14 Apr 2 12:04:04.579:
ISAKMP (14): Creating IPSec SAs Apr 2 12:04:04.579: inbound
SA from 20.20.20.20 to 20.20.20.21 (proxy 60.60.60.0 to
50.50.50.0) Apr 2 12:04:04.583: has spi 183903875 and conn_id
15 and flags 4 Apr 2 12:04:04.583: lifetime of 190 seconds
Apr 2 12:04:04.587: lifetime of 4608000 kilobytes Apr 2
12:04:04.587: outbound SA from 20.20.20.21 to 20.20.20.20
(proxy 50.50.50.0 to 60.60.60.0) Apr 2 12:04:04.591: has spi
303564824 and conn_id 16 and flags 4 Apr 2 12:04:04.591:
lifetime of 190 seconds Apr 2 12:04:04.595: lifetime of
4608000 kilobytes Apr 2 12:04:04.599: IPSEC(key_engine): got
a queue event... Apr 2 12:04:04.599: IPSEC(initialize_sas): ,
(key eng. msg.) dest= 20.20.20.21, src= 20.20.20.20,
dest_proxy= 50.50.50.0/255.255.255.0/0/0 (type=4), src_proxy=
60.60.60.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0xAF62683(183903875), conn_id= 15, keysize= 0, flags= 0x4 Apr
2 12:04:04.607: IPSEC(initialize_sas): , (key eng. msg.) src=
20.20.20.21, dest= 20.20.20.20, src_proxy=
50.50.50.0/255.255.255.0/0/0 (type=4), dest_proxy=
```

```
60.60.60.0/255.255.255.0/0/0 (type=4), protocol= ESP,
transform= esp-rfc1829 , lifedur= 190s and 4608000kb, spi=
0x12180818(303564824), conn_id= 16, keysize= 0, flags= 0x4
Apr 2 12:04:04.615: IPSEC(create_sa): sa created, (sa)
sa_dest= 20.20.20.21, sa_prot= 50, sa_spi=
0xAF62683(183903875), sa_trans= esp-rfc1829 , sa_conn_id= 15
Apr 2 12:04:04.619: IPSEC(create_sa): sa created, (sa)
sa_dest= 20.20.20.20, sa_prot= 50, sa_spi=
0x12180818(303564824), sa_trans= esp-rfc1829 , sa_conn_id= 16
!--- The IPsec SAs are created, and ICMP traffic can flow.
```

Команды show на одноранговом маршрутизаторе

```
!--- This illustrates a series of show command output after
!--- IKE/IPsec negotiation takes place. goss-c2-2513#show
crypto isakmp sa dst src state conn-id slot 20.20.20.21
20.20.20.20 QM_IDLE 14 0 goss-c2-2513#show crypto ipsec sa
interface: Serial0 Crypto map tag: armadillo, local addr.
20.20.20.21 local ident (addr/mask/prot/port):
(50.50.50.0/255.255.255.0/0/0) remote ident
(addr/mask/prot/port): (60.60.60.0/255.255.255.0/0/0)
current_peer: 20.20.20.20 PERMIT, flags={origin_is_acl,}
#pkts encaps: 5, #pkts encrypt: 5, #pkts digest 0 #pkts
decaps: 5, #pkts decrypt: 5, #pkts verify 0 #send errors 0,
#recv errors 0 local crypto endpt.: 20.20.20.21, remote
crypto endpt.: 20.20.20.20 path mtu 1500, media mtu 1500
current outbound spi: 12180818 inbound esp sas: spi:
0xAF62683(183903875) transform: esp-rfc1829 , in use settings
={Var len IV, Tunnel, } slot: 0, conn id: 15, crypto map:
armadillo sa timing: remaining key lifetime (k/sec):
(4607999/118) IV size: 8 bytes replay detection support: N
inbound ah sas: outbound esp sas: spi: 0x12180818(303564824)
transform: esp-rfc1829 , in use settings ={Var len IV,
Tunnel, } slot: 0, conn id: 16, crypto map: armadillo sa
timing: remaining key lifetime (k/sec): (4607999/109) IV
size: 8 bytes replay detection support: N outbound ah sas:
goss-c2-2513#show crypto isakmp policy Protection suite of
priority 1 encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Message Digest 5
authentication method: Pre-Shared Key Diffie-Hellman group:
#1 (768 bit) lifetime: 86400 seconds, no volume limit Default
protection suite encryption algorithm: DES - Data Encryption
Standard (56 bit keys). hash algorithm: Secure Hash Standard
authentication method: Rivest-Shamir-Adleman Signature
Diffie-Hellman group: #1 (768 bit) lifetime: 86400 seconds,
no volume limit goss-c2-2513#show crypto map Crypto Map
"armadillo" 1 ipsec-isakmp Peer = 20.20.20.20 Extended IP
access list 101 access-list 101 permit ip 50.50.50.0
0.0.0.255 60.60.60.0 0.0.0.255 Current peer: 20.20.20.20
Security association lifetime: 4608000 kilobytes/190 seconds
PFS (Y/N): N Transform sets={ MamaBear, PapaBear, BabyBear, }
```

Реализация подсказок для IPSec

Ниже приведены некоторые советы по реализации IPsec:

- Убедитесь, что существует подключение между двумя конечными точками, прежде чем настраивать шифрование.
- Убедитесь, что DNS работает на маршрутизаторе или имя хоста ЦС указано вручную, если используется ЦС.

- IPsec использует IP-протоколы 50 и 51, а трафик IKE передается по протоколу 17, использующему порт 500 (UDP 500). Убедитесь, что эти порты разрешены.
- Убедитесь, что в списке ACL не используется слово any. Это вызывает проблемы. См. Руководства по использованию для **access-list** в Справочнике по командам PIX для получения дополнительной информации.
- Рекомендуются следующие комбинации преобразований:`esp-des and esp-sha-hmac`
  `ah-sha-hmac and esp-des`
- Помните, что AH — это всего лишь аутентифицированный заголовок. Фактический поток данных пользователя не зашифрован. Необходимо использовать протокол ESP для шифрования потока данных. Если используется только AH и по сети передается открытый текст, не удивляйтесь. Также используйте ESP при использовании AH. Обратите внимание, что ESP также может выполнять аутентификацию. **Следовательно, можно использовать комбинацию преобразования, такую как esp-des и esp-sha-hmac.**
- **ah-rfc1828 и esp-rfc1829 являются устаревшими платформами, включенными для обратной совместимости с более старыми реализациями IPsec.** Если одноранговый участник не поддерживает новые преобразование, попробуйте следующее.
- SHA более медленный и более безопасный алгоритм, чем MD5. Некоторые специалисты не чувствуют достаточной защищенности при использовании MD5.
- Если есть сомнения, используйте туннельный режим. По умолчанию используется туннельный режим, с тем же успехом может использоваться транспортный режим по причине его возможностей VPN.
- Пользователям классического шифрования classic crypto, которые обновляются до ПО Cisco IOS версии 11.3, следует обратить внимание, что методы хранения команд crypto в этой конфигурации изменились, чтобы разрешить использование IPsec. Следовательно, если пользователи, использовавшие классическое шифрование classic crypto, перейдут обратно на версию Cisco IOS 11.2, им потребуется повторно указать необходимые конфигурации шифрования.
- Если выполнить проверку связи с помощью команды ping по зашифрованному каналу после настройки конфигурации, процесс согласования может занять некоторое время: около шести секунд на Cisco 4500 и 20 секунд на Cisco 2500, так как контексты безопасности еще не согласованы. Даже если все настроено успешно, проверка связи может завершиться со сбоем. **Команды debug crypto ipsec и debug crypto isakmp показывают, что происходит. После завершения шифрования потоков данных команды проверки связи начинают успешно выполняться.**
- **Если произошла неполадка при согласовании и внесении изменений в конфигурацию, используйте команды clear crypto is и clear crypto sa, чтобы очистить базы данных перед повторной попыткой.** Это приводит к запуску нового согласования без использования старого. **Очень полезными окажутся команды clear cryptois и clear cry sa.**

Справка и соответствующие ссылки

Информация по IPSec

- Страница поддержки IPSec
- Политика шифрования ECRA и Процедуры — Посылают Электронное письмо export@cisco.com

Больше примеров конфигурации для IPsec

- Настройка и шифрование сетевого уровня Cisco устранения проблем: IPSec и ISAKMP
- Обзор сетевой безопасности IPSec
- Документации по конфигурации IPsec на межсетевых экранах PIXPIX 5.1PIX 5.2PIX 5.3PIX 6.0PIX 6.1PIX 6.2PIX 6.3

Свяжитесь с технической поддержкой Cisco в (800) 553-24HR, (408) 526-7209, или передайте и Электронная почта к tac@cisco.com при требовании дальнейшей поддержки с IPsec.

Ссылки

Harkins, D. *Функциональная спецификация программного блока Функции Протокола ISAKMP/Oakley.* ENG-0000 Rev A. Cisco Systems.

Мэдсон, C. *Версия F IPSec Software Unit Functional Specification ENG-17610.* C i s c o S y s t e m s.

Кауфман, C. Перлман P. и Спенсер, M. *Безопасность сети: Частная связь в открытом мире.* Prentice Hall, 1995.

Шнайер, B. *Прикладная криптография: Протоколы, алгоритмы и исходный код в криптографии*. Second Ed. John Wiley & Sons, Inc.

Различные рабочие проекты IP-безопасности IETF

Дополнительные сведения

- Страница поддержки IPSec
- Принципы работы частных виртуальных сетей
- Устранение наиболее распространенных проблем удаленных VPN-подключений и VPN-туннелей LAN — LAN на базе протокола IPSec
- Cisco Systems – техническая поддержка и документация