Network Working Group
Request for Comments: 4301
Obsoletes: 2401
Category: Standards Track

S. Kent
K. Seo
BBN Technologies

December 2005

Архитектура защиты для протокола IP

Security Architecture for the Internet Protocol

Статус документа

В этом документе содержится спецификация протокола, предложенного сообществу Internet. Документ служит приглашением к дискуссии в целях развития и совершенствования протокола. Текущее состояние стандартизации протокола вы можете узнать из документа Internet Official Protocol Standards (STD 1). Документ может распространяться без ограничений.

Авторские права

Аннотация

Документ является обновлённой версией Security Architecture for IP и посвящён способам защиты трафика на уровне IP. Данный документ заменяет собой RFC 2401 (ноябрь 1998).

Посвящение

Этот документ посвящён памяти Charlie Lynn, который долгие годы работал в BBN и внёс важный вклад в подготовку документов по IPsec.

Оглавление

1. Введение

1.1. Обзор содержимого документа

Этот документ описывает базовую архитектуру IPsec-совместимых систем. Здесь описано, как обеспечить службы защиты трафика на уровне IP, как в среде IPv4 [Pos81a], так и для IPv6 [DH98]. Документ описывает требования к системам, реализующим IPsec, фундаментальные элементы таких систем и объединение таких элементов в среде IP. Документ также описывает средства защиты, обеспечиваемые протоколами IPsec, и способы развёртывания соответствующих служб в среде IP. Документ не рассматривает всех аспектов архитектуры IPsec. Имеются другие документы, посвящённые дополнительным деталям архитектуры при использовании в специализированных средах (например, использование IPsec с среде NAT[1] или более полная поддержка групповой адресации IP). Фундаментальные компоненты архитектуры защиты IPsec обсуждаются в терминах обеспечиваемой ими требуемой функциональности. Имеются дополнительные документы RFC (см. параграф 1.3, где указаны ссылки на эти документы) для протоколов (a), (c) и (d).

(a) протоколы защиты - AH[2] и ESP[3];

(b) защищённые связи (Security Association) - что это, как они работают, как управляются, связанные с ними операции;

(c) управление ключами - ручное или автоматизированное (IKE[4]);

(d) криптографические алгоритмы для аутентификации и шифрования.

Этот документ не описывает архитектуру безопасности[5] для Internet, он посвящён лишь защите на уровне IP, обеспечиваемой за счёт использования комбинации криптографических и протокольных механизмов защиты.

Написание IPsec является предпочтительным и используется во всех, связанных с IPsec стандартах. Использование других вариантов написания IPsec (например, IPSEC, IPSec, ipsec) является некорректным. Однако последовательность символов IPsec с любой комбинацией строчных и прописных букв следует относить к протоколам IPsec.

Ключевые слова **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не следует** (SHALL NOT), **следует** (SHOULD), **не нужно** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе интерпретируются в соответствии с RFC 2119 [Bra97].

[1]Network Address Translation - трансляция сетевых адресов.
[2]Authentication Header - аутентификационный заголовок.
[3]Encapsulating Security Payload - инкапсуляция защищённых данных.
[4]Internet Key Exchange - обмен ключами в Internet.
[5]Security Architecture - архитектура защиты.

1.2. Аудитория

Этот документ адресован прежде всего тем, кто реализует эту технологию защиты IP или разрабатывает системы, которые будут использовать такую технологию. Технически грамотные пользователи данной технологии (конечные пользователи и системные администраторы) также являются целевой аудиторией. В Приложении A приведён глоссарий терминов, который поможет заполнить пробелы в понимании основ и терминологии. Документ рассчитан на читателей, хорошо знакомых с протоколом IP, связанными с ним технологиями межсетевого взаимодействия, а также с общими концепциями и терминологией в области защиты информационных систем.

1.3. Связанные документы

Как было отмечено выше, существует ещё ряд документов, содержащих детальное определение некоторых компонент IPsec и отношений между компонентами. К таким документам относятся RFC по следующим темам:

a. **протоколы защиты** - RFC, описывающие протокол Authentication Header (AH) [Ken05b] и Encapsulating Security Payload (ESP) [Ken05a];

b. **криптографические алгоритмы для шифрования и обеспечения целостности** - один документ RFC, который определяет обязательный, используемый по умолчанию с AH и ESP алгоритм [Eas05], аналогичный документ RFC, определяющий обязательные алгоритмы для использования с IKEv2 [Sch05] и отдельные документы RFC для каждого криптографического алгоритма.

c. **автоматическое управление ключами** - документы RFC по протоколу IKEv2 [Kau05] и Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2) [Sch05].

2. Задачи протокола

2.1. Цели, задачи, требования, описание проблемы

Задачей IPsec является создание интероперабельной, высококачественной, основанной на криптографии системы защиты для IPv4 и IPv6. Набор обеспечиваемых услуг защиты включает контроль доступа, обеспечение целостности без организации прямых соединений (connectionless integrity), аутентификацию источника данных (data origin authentication), детектирование и отклонение попыток повторного использования сохранённой информации (replay), как форма сохранения целостности порядка, обеспечение сохранности тайны (confidentiality) путём шифрования, а также ограниченное сохранение конфиденциальности потоков трафика (limited traffic flow confidentiality). Эти услуги предоставляются на уровне IP, обеспечивая стандартные способы защиты для всех протоколов, которые могут передаваться через IP (включая и сам протокол IP).

IPsec включает спецификацию минимальной функциональности межсетевого экрана (firewall), поскольку такие экраны являются важным элементом контроля доступа на уровне IP. Реализации могут поддерживать более изощрённые механизмы межсетевого экранирования и реализовать обязательные функции IPsec с использованием таких механизмов[1]. Функция межсетевого экранирования IPsec делает использование криптографической аутентификации и защиты целостности, обеспечиваемые для всего трафика IPsec, более эффективным средством контроля доступа, нежели это возможно при использовании отдельного межсетевого экрана (не знающего внутренних параметров IPsec) в комбинации с отдельным механизмом криптографической защиты.

Большинство механизмов защиты обеспечивается за счёт использования двух протоколов защиты трафика - Authentication Header (AH) и Encapsulating Security Payload (ESP), а также криптографических процедур и протоколов управления ключами. Протоколы IPsec развёртываются в определённой среде (контексте) и способы использования протоколов определяются администраторами/пользователями в этом же контексте. Задачей архитектуры IPsec является обеспечение совместимых реализаций, включающих службы и интерфейсы управления, требуемые для удовлетворения потребностей в обеспечении безопасности большого числа пользователей.

При корректной реализации и развёртывании IPsec не оказывает негативного воздействия на работу пользователей, хостов и других компонент Internet, которые не используют IPsec для защиты трафика. Протоколы защиты IPsec (AH и ESP, а также в несколько меньшей степени IKE) разработаны так, чтобы обеспечивалась независимость от криптографических алгоритмов. Модульная структура позволяет выбирать различные наборы криптоалгоритмов в соответствии с реальными потребностями и независимо от других частей реализации. Например, различные группы пользователей могут выбрать разные наборы криптоалгоритмов, если это нужно.

Для повышения глобального уровня взаимодействия Internet задан набор криптоалгоритмов, используемых по умолчанию с AH и ESP [Eas05], а также набор обязательных к реализации алгоритмов для IKEv2 [Sch05]. Документы [Eas05] и [Sch05] будут периодически обновляться с учётом роста вычислительных мощностей и разработок в области криптографии. Задание этих алгоритмов в документах, отдельных от спецификаций AH, ESP и IKEv2, позволяет заменять эти алгоритмы без влияния на процесс стандартизации остального набора документов IPsec. Использование этих криптоалгоритмов в комбинации со средствами защиты трафика IPsec и протоколами управления ключами предназначено для обеспечения разработчикам систем и приложений развёртывать высококачественную технологию криптографической защиты на уровне Internet.

2.2. Допущения и предостережения

Набор протоколов IPsec и связанные с ними (используемые по умолчанию) криптоалгоритмы предназначены для обеспечения высокого уровня защиты трафика Internet. Однако обеспечиваемая протоколами защита зависит от качества реализации и этот вопрос выходит за рамки данного документа. Более того, безопасность компьютерной системы или сети зависит от множества факторов, включая персонал, физическую защиту, процедуры, уровень электромагнитного излучения и защиту компьютеров. Таким образом, IPsec является лишь частью комплексной архитектуры защиты систем.

[1]Отметим, что могут возникать проблемы взаимодействия, если межсетевой экран накладывает дополнительные ограничения на поток трафика, вносимые реализацией IPsec, но не обеспечивается согласование возможностей на основе функции селекции трафика, определённых в данном документе и согласуемых через IKEv2.

В конечном итоге, защита, обеспечиваемая при использовании IPsec очень сильно зависит от множества параметров рабочей чреды, в которой выполняется реализация IPsec. Например, дефекты в защите OS, низкое качество генераторов случайных чисел, ненадёжные протоколы сетевого управления, а также небрежная практика управления и т. п. могут существенно снизить уровень защиты, обеспечиваемой IPsec. Как было отмечено выше, ни один из этих атрибутов среды не относится к зоне ответственности этого или других стандартов IPsec.

3. Обзор системы

В этой главе даётся концептуальное описание работы IPsec, компонент системы и их совместного использования для обеспечения упомянутых выше средств защиты. Целью этого описания является представление читателю “картины” системы и процессов, их места в среде IP и основы для понимания следующих разделов документа, где более детально описываются все компоненты системы.

Реализация IPsec работает на хосте, используемом как защитный шлюз (SG[1]), или на отдельном устройстве, обеспечивая защиту трафика IP. Более подробное описание этих классов реализации приводится ниже, в параграфе 3.3. Обеспечиваемая IPsec защита основывается на требованиях базы правил безопасности (SPD[2]), разработанной и поддерживаемой пользователем, системным администратором приложением, работающим по заданной пользователем или администратором схеме. В общем случае для пакетов выбирается один из трёх вариантов обработки в зависимости от информации в заголовке IP и следующего уровня ("Селекторы", параграф 4.4.1.1) в соответствии с правилами SPD. Каждый пакет **защищается** (PROTECT) с использованием IPsec, **отбрасывается** (DISCARD) или **пропускается в обход** (BYPASS) защиты IPsec, в зависимости от правил SPD, определяемых селекторами.

3.1. Что делает IPsec

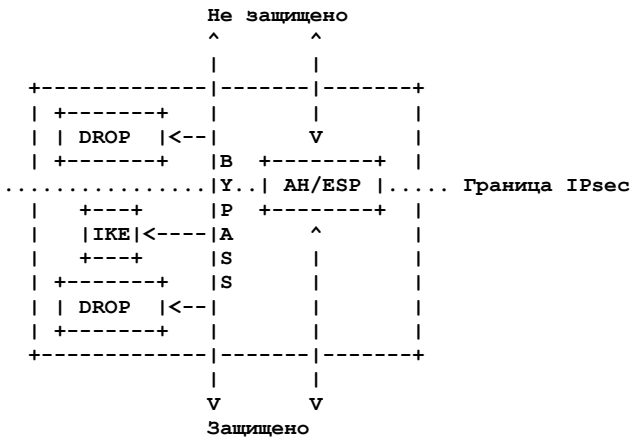


Рисунок 1. Верхний уровень рабочей модели IPsec.

IPsec создаёт границу между защищённым и незащищённым интерфейсами для хоста или сети (см. рисунок 1). К проходящему через границу трафику применяются правила контроля доступа, заданные пользователем или администратором, отвечающим за конфигурацию. Эти правила контроля определяют проходит пакет через границу беспрепятственно, к нему применяются механизмы защиты на основе AH или ESP или пакет отбрасывается.

Механизмы защиты IPsec предоставляются на уровне IP путём выбора подходящих протоколов защиты, криптографических алгоритмов и ключей. IPsec можно использовать для защиты одного или множества “путей” между (a) парой хостов, (b) парой шлюзов безопасности или (c) хостом и шлюзом безопасности. Соответствующая требованиям реализация для хоста **должна** поддерживать (a) и (c), а реализация для шлюза безопасности - все три варианта соединений, поскольку при определённых обстоятельствах шлюз безопасности действует как хост.

Незащищённый (unprotected) интерфейс называют также «чёрным» (black) или зашифрованным (ciphertext). Защищённый (protected) интерфейс называют ещё красным (red) или текстовым (plaintext). Защищённый интерфейс может быть внутренним (например, в реализации IPsec для хоста) или может быть соединён с интерфейсом уровня сокетов, предоставленным OS. В этом документе термин «входящий» (inbound) относится к трафику, входящему в реализацию IPsec через незащищённый интерфейс, или выдаваемому реализацией на незащищённую сторону границы и направленному в сторону защищённого интерфейса. Термин «исходящий» (outbound) относится к трафику, входящему в реализацию через защищённый интерфейс, или выдаваемому реализацией на защищённую сторону границы и направленному в сторону незащищённого интерфейса. Реализация IPsec может поддерживать более одного интерфейса по одну или обе сторону границы.

Обратите внимание на средства отбрасывания трафика (DROP) по обе стороны границы IPsec, средства обхода (BYPASS), позволяющие пропускать трафик через границу без криптозащиты и IKE, как средство управления ключами на защищённой стороне и управления защитой.

IPsec может также поддерживать согласование компрессии IP [SMPT01] - это обусловлено тем, что используемое в IPsec шифрование препятствует сжатию данных протоколами нижележащих уровней.

3.2. Как работает IPsec

IPsec использует два протокола для обеспечения защиты - Authentication Header (AH) и Encapsulating Security Payload (ESP). Оба протокола подробно описаны в соответствующих RFC [Ken05b, Ken05a]. Реализация IPsec **должна** поддерживать ESP и **может** поддерживать AH[3].

[1]Security gateway - защитный шлюз - промежуточная система, реализующая IPsec (например, межсетевой экран или маршрутизатор с поддержкой IPsec).
[2]Security Policy Database - база правил (политики) безопасности.

- Протокол AH [Ken05b] обеспечивает защиту целостности и аутентификацию источника данных, а также может (по усмотрению получателя) выполнять функции предотвращения повторного использования пакетов (anti-replay).

- Протокол ESP [Ken05a] обеспечивает такой же набор функций и в дополнение поддерживает средства обеспечения конфиденциальности. Использование ESP для защиты конфиденциальности без обеспечения целостности **не рекомендуется**. При использовании ESP со включённой защитой конфиденциальности имеются возможности ограниченной защиты конфиденциальности потока (т. е., сокрытия размера пакетов, а также активной генерации и отбрасывания «мусорных» пакетов). Эти возможности нужны прежде всего в виртуальных частных сетях (VPN) и в контексте перекрывающихся сетей.

- Оба протокола AH и ESP обеспечивают контроль доступа, реализуемый путём распределения криптоключей и управления потоками трафика в соответствии с базой правил SPD2 (см. параграф 4.4.1).

Эти протоколы могут использоваться вместе или по отдельности для обеспечения защиты IPv4 и IPv6. Однако большинство требований защиты можно выполнить при использовании одного протокола ESP. Каждый из протоколов поддерживает два режима работы - транспортный и туннельный. В транспортном режиме AH и ESP обеспечивают защиту прежде всего для протоколов следующего уровня, а в туннельном режиме AH и ESP применяются к туннелируемым пакетам IP. Различия между этими режимами обсуждаются в параграфе 4.1.

IPsec позволяет пользователю (или системному администратору) контролировать гранулярность услуг защиты. Например, можно создать один шифрованный туннель для передачи всего трафика между парой шлюзов безопасности или раздельные шифрованные туннели для каждого соединения TCP между каждой парой хостов, обменивающихся данными через эти шлюзы. IPsec с помощью парадигмы управления SPD обеспечивает возможность задания:

  – используемого протокола (AH или ESP), режима (транспортный или туннельный), опций защиты, используемых криптоалгоритмов, а также комбинаций протоколов и служб;

  – гранулярности применения защиты.

Поскольку большинство обеспечиваемых IPsec средств защиты требует использования криптографических ключей, IPsec полагается на отдельный набор механизмов для размещения ключей в нужных местах. Этот документ требует поддержки как ручного, так и автоматического распределения ключей. Для автоматического управления ключами документ задаёт основанный на использовании открытых ключей метод IKEv2 [Kau05], но **могут** использоваться и другие механизмы автоматического распространения ключей.

Примечание: Этот документ требует поддержки нескольких функций, которые поддерживаются в IKEv2, но отсутствуют в IKEv1 (например, согласования представляющих SA локальных и удалённых портов или согласование множества SA с одинаковыми селекторами). Следовательно, этот документ предполагает использование IKEv2 или системы управления ключами и защищёнными связями со сравнимыми возможностями.

3.3. Где можно реализовать IPsec

Существует множество вариантов реализации IPsec на хостах, в комбинации с маршрутизаторами или межсетевыми экранами для создания защитных шлюзов, а также на независимых устройствах защиты.

a. IPsec можно интегрировать непосредственно в стек IP. Это требует доступа к исходному коду IP и может применяться как для хостов, так и для защитных шлюзов, хотя хостовые реализации более подходят для этой стратегии, как показано ниже (параграф 4.4.1, глава 6, последний абзац параграфа 4.4.1.1).

b. В BITS[1]-варианте IPsec реализуется “ниже” имеющейся реализации стека IP между исходным IP и драйверами сетевых устройств. Доступ к исходному коду стека IP не требуется в таком контексте и делает данный вариант подходящим для реализации в старых системах. Обычно такие варианты реализуются на хостах.

c. Использование выделенного специализированного процессора для протоколов защиты широко распространено в военных системах, а также в некоторых коммерческих системах. Этот вариант иногда называют BITW[2]-реализацией. Такие реализации могут разрабатываться для хостов или шлюзов. Обычно устройство BITW само по себе имеет адрес IP. При поддержке одного хоста такой вариант похож на BITS-реализацию, но на маршрутизаторах и межсетевых экранах он действует подобно защитному шлюзу.

В этом документе часто говорится о реализации IPsec на хостах или защитных шлюзах без конкретизации варианта (в стеке, BITS или BITW). В тех случаях, когда такое различие имеет существенное значение, в документе указывается конкретный вариант реализации.

Хостовые реализации IPsec могут встречаться в устройствах, которые не совсем похожи на хост. Например, маршрутизатор может использовать IPsec для защиты протоколов маршрутизации (скажем, BGP) и функций управления (типа, Telnet), не воздействуя на проходящий через маршрутизатор пользовательский трафик. Защитный шлюз может использовать раздельные реализации IPsec для защиты пользовательского трафика и управляющей информации. Описанная в этом документе архитектура является очень гибкой. Например, компьютер с полнофункциональной, соответствующей спецификации реализацией IPsec в OS должен настраиваться на защиту работающих на нем приложений (хост) и защиту проходящего через этот компьютер трафика (защитный шлюз). Такая конфигурация будет использовать таблицы пересылки и функции SPD, описанные в параграфах 5.1 и 5.2.

4. Защищённые связи (SA)

В этой главе описываются требования к управлению защищёнными связями для всех реализаций IPv6 и тех реализаций IPv4, которые поддерживают протокол AH, ESP или оба. Концепция защищённых связей (Security Association или SA) является фундаментальной для IPsec. Оба протокола AH и ESP используют SA, а основной

[3]Уровень требования по поддержке AH был снижен до “**может**” потому, что опыт показывает, что существует весьма незначительное число ситуаций, в которых ESP не может обеспечить требуемой защиты. Отметим, что протокол ESP можно использовать в режиме защиты только целостности данных (integrity) без обеспечения сохранности тайны (confidentiality), что делает его сравнимым в AH в большинстве контекстов.

[1]Bump-in-the-stack утолщение в стеке.

[2]Bump-in-the-wire - утолщение в проводе.

функцией IKE является организация и поддержка SA. Все реализации AH и ESP **должны** поддерживать концепцию SA в соответствии с приведённым ниже описанием. Остальная часть этой главы описывает различные аспекты управления SA, определения требуемых характеристик для управления политикой SA и методов управления SA.

4.1. Определение и сфера применения

SA представляет собой симплексное соединение, которое обеспечивает защиту для передаваемого через него трафика. Предоставляемая SA защита основана на использовании протокола AH, ESP или обоих. Если к потоку трафика применяются оба протокола AH и ESP, должны создаваться две связи SA, которые координируются для обеспечения пошагового (согласованного) применения протоколов защиты. Для защиты типового двухстороннего соединения между двумя поддерживающими IPsec системами требуется пара SA (по одной связи для каждого направления). Протокол IKE явно создаёт пары SA, учитывая общность этого требования.

Для SA, используемых для передачи unicast-трафика, достаточно индекса SPI[1] для указания SA. Однако реализация может локально выбрать вариант использования SPI в комбинации с типом протокола IPsec (AH или ESP) для аутентификации SA. Если реализация IPsec поддерживает групповую адресацию (multicast), она **должна** поддерживать групповые (multicast) SA, используя описанный ниже алгоритм для отображения входящих дейтаграмм IPsec на связи SA. Реализация, поддерживающая только индивидуальный (unicast) трафик, не обязана реализовать этот алгоритм демультиплексирования.

Во многих архитектурах защиты с групповой адресацией (например, [RFC3740]) центральный контроллер группы/сервер обмена ключами (Group Controller/Key Server) обязательно привязывается к GSA[2] SPI. Этот индекс SPI не согласуется и не координируется с системой управления ключами (например, IKE), которая остаётся на отдельных конечных системах, формирующих группу. Следовательно, возможно одновременное использование одного индекса SPI для индивидуальной (unicast) связи SA и групповой связи GSA. Поддерживающие групповую адресацию реализации IPsec **должны** корректно демультиплексировать входящий трафик даже при возникновении конфликта SPI.

Каждая запись в базе данных SA (SAD[3], см. параграф 4.4.2) должна показывать используется ли при поиске SA IP-адрес получателя или адреса отправителя и получателя в дополнение к SPI. Для групповых SA, поле протокола не используется для поиска SA. Для каждого входящего пакета, защищённого с помощью IPsec, реализация должна проводить поиск в SAD так, чтобы найденный результат соответствовал «наиболее длинному» идентификатору SA. В этом контексте при соответствии двух и более записей SAD значению SPI для определения более длинного соответствия проводится также сравнение адреса получателя или адресов отправителя и получателя (как указано в записи SAD). Это определяет логический порядок поиска в SAD:

1. В базе SAD ищется соответствие для SPI и адресов получателя и отправителя. При нахождении записи, входящий пакет обрабатывается в соответствии с ней. Если запись не найдена, выполняется п. 2.
2. В базе SAD ищется соответствие для SPI и адреса получателя. При нахождении записи входящий пакет обрабатывается в соответствии с ней. Если запись не найдена, выполняется п. 3.
3. В базе SAD ведётся поиск только по одному значению SPI, если получатель выбрал поддержку единого пространства SPI для AH и ESP, или по обоим SPI и протоколу, в противном случае. Если запись найдена, входящий пакет обрабатывается в соответствии в ней. В противном случае пакет отбрасывается, а в системный журнал вносится запись об этом.

На практике реализация может выбрать любой метод (или не использовать никакого) ускорения этого поиска, хотя видимое извне поведение поиска **должно** быть функционально эквивалентным поиску по SAD в рассмотренном выше порядке. Например, программная реализация может помещать SPI в хэш-таблицы. Записи SAD в связном списке каждой хэш-таблицы могут сохраняться отсортированными так, чтобы записи SAD с наиболее длинными идентификаторами SA находились ближе к началу связанных списков. Записи SAD с самыми короткими идентификаторами SA могут сортироваться так, чтобы они были последними записями в связанном списке. Аппаратные реализации могут могут находить самый длинный идентификатор с помощью общедоступного механизма TCAM[4].

Индикация того, какой адрес (отправителя или получателя) используется для отображения входящего трафика IPsec на связи SA, **должна** устанавливаться при настройке конфигурации SA вручную или путём согласования использования протокола управления SA (например, IKE или GDOI[5] [RFC3547]). Обычно группы SSM[6] [HC03] используют 3-компонентный идентификатор SA, включающий SPI, групповой адрес получателя и адрес отправителя. Для групп Any-Source Multicast требуется только SPI и групповой адрес получателя.

Если при передаче различных классов трафика (отличаются битами DSCP[7]) [NiBlBaBL98], [Gro02]) в одну SA получатель использует необязательную функцию anti-replay[8], поддерживаемую AH и ESP, это может приводить к недопустимому отбрасыванию пакетов с низким приоритетом, обусловленному используемым этой функцией оконным механизмом. Поэтому отправителю **следует** помещать трафик разных классов с одинаковым значением селектора в разные SA для поддержки различных уровней качества обслуживания (QoS[9]). Для решения этой задачи реализация IPsec **должна** обеспечивать возможность организации и поддержки множества SA с одинаковым селектором между данным отправителем и получателем. Распределение трафика между этими параллельными SA для поддержки QoS задаётся локально отправителем и не согласуется через IKE. Получатель **должен** беспристрастно обрабатывать пакеты из разных SA. Эти требования применимы как к транспортным, так и к туннельным SA. Для SA в туннельном режиме нужные значения DSCP находятся во внутренних заголовках IP. Для транспортного режима значение DSCP может измениться по пути, но это не должно вызывать проблем при обработке IPsec, поскольку это значение не

[1] Security Parameters Index - индекс параметров безопасности. Информация о SPI содержится в Приложении A и спецификациях протоколов AH и ESP [Ken05b, Ken05a].
[2] Group Security Association - групповая защищённая связь.
[3] SA Database - база данных о защищённых связях (SA).
[4] Ternary Content-Addressable Memory - триадная память с адресацией по содержимому.
[5] Group Domain of Interpretation.
[6] Source-Specific Multicast - заданная отправителем групповая адресация.
[7] Differentiated Services Code Point.
[8] Предотвращение повторного использования собранных ранее пакетов.
[9] Quality of Service.

используется для выбора SA и **недопустимо** проверять это значение в процессе проверки пакетов и SA. Однако, если происходит существенное нарушение порядка доставки пакетов (например, в результате изменения значений DSCP в пути), это может включать на приёмной стороне механизм отбрасывания пакетов в результате применения механизма предотвращения повторного использования пакетов (anti-replay).

Обсуждение: Хотя поля DSCP [NiBlBaBL98, Gro02] и ECN[1] [RaFlBl01] не являются “селекторами” в понимании описываемой архитектуры, на передающей стороне требуется механизм, позволяющий направлять пакеты с данным значением (набором значений) DSCP в подходящую SA. Этот механизм можно назвать классификатором (classifier).

Как было отмечено выше, определены два типа SA - транспортный режим и туннельный режим. IKE создаёт пары SA, поэтому для простоты мы будем требовать, чтобы обе связи SA в паре были однотипными (транспортными или туннельными).

Транспортный режим SA обычно используется при связи между парой хостов для обеспечения сквозной защиты. Когда нужно обеспечение защиты между парой промежуточных систем на пути (в отличие от сквозного использования IPsec), **можно** использовать транспортный режим между защитными шлюзами или между хостом и защитным шлюзом. В тех случаях, когда используется транспортный режим между парой защитных шлюзов или шлюзом и хостом, транспортный режим можно использовать для поддержки туннелирования в IP (например, IP-in-IP [Per96], GRE[2] [FaLiHaMeTr00] или динамическая маршрутизация [ToEgWa04]) через транспортные SA. Для большей ясности отметим, что использование транспортного режима промежуточными системами (например, защитными шлюзами) допустимо лишь применительно к пакетам, в которых адреса отправителей (для исходящих пакетов) или получателей (для входящих пакетов) относятся к самой промежуточной системе. Функции контроля доступа, являющиеся важной частью IPsec, существенно ограничены в таком контексте, поскольку они не могут применяться к “сквозным” заголовкам пакетов, проходящим через SA в транспортном режиме. Таким образом, использование транспортного режима следует аккуратно оценивать с учётом конкретного контекста.

Для IPv4 заголовок протокола защиты в транспортном режиме располагается сразу же после заголовка и опций IP перед заголовком протокола следующего уровня (например, TCP или UDP). Для IPv6 заголовок протокола защиты появляется после основного заголовка IP и выбранных расширенных заголовков, но он может помещаться до опций получателя (destination options) или перед ними; этот заголовок **должен** располагаться до заголовка протоколов следующего уровня (например, TCP, UDP, SCTP[3]). В случае ESP связи SA транспортного режима обеспечивают защиту только для протоколов следующего уровня, но не для заголовков IP или любых расширенных заголовков, появляющихся перед заголовком ESP. В случае AH защита обеспечивается также для выбранной части предшествующего заголовка, выбранных частей расширенных заголовков и выбранных опций (содержащихся в заголовке IPv4, расширенных заголовках IPv6 Hop-by-Hop или IPv6 Destination). Более детальное описание защищаемых AH областей приведено в спецификации AH [Ken05b].

Связи SA в туннельном режиме используются с туннелями IP и контроль доступа применяется к заголовкам всего трафика внутри туннеля. Два хоста **могут** организовать между собой SA в туннельном режиме. За исключением двух рассмотренных ниже ситуаций связи SA **должны** работать в туннельном режиме, если любая из сторон является защитным шлюзом. Таким образом, SA между двумя защитными шлюзами обычно работает в туннельном режиме, равно как и SA между хостом и защитным шлюзом. Из этого правила есть два исключения.

- Когда трафик адресован защитному шлюзу (например, команды SNMP[4]), шлюз действует как хост и можно использовать транспортный режим. В этом случае SA завершается управляющей функцией на хосте (которым является защитный шлюз) и это меняет дело.
- Как было отмечено выше, защитные шлюзы **могут** поддерживать транспортный режим SA для обеспечения защиты трафика IP между двумя промежуточными системами на пути (например, между хостом и защитным шлюзом или парой защитных шлюзов).

Существует несколько причин использования туннельного режима для SA, включающих защитные шлюзы. Например, если имеется множество путей (скажем, через различные защитные шлюзы) к одному адресату, находящемуся за защитным шлюзом, важно чтобы пакеты IPsec передавались тому шлюзу, с которым была согласована связь SA. Аналогичная ситуация возникает и для фрагментов - если на пути возможна фрагментация пакетов, все фрагменты должны доставляться одному экземпляру IPsec для их сборки до выполнения криптографических операций. К тому же, когда фрагмент обработан IPsec, передан и подвергнут дополнительной фрагментации, очень важно иметь внешний и внутренний заголовки, чтобы сохранить состояние фрагментации для пакетов до и после операций IPsec. Следовательно, имеется множество причин использования туннельного режима SA в тех случаях, когда любая из конечных точек является защитным шлюзом. (Использование туннеля IP-in-IP в комбинации с транспортным режимом позволяет решить проблему фрагментации. Однако такая конфигурация ограничивает возможности использования IPsec для реализации политики контроля доступа.)

Примечание: Протоколы AH и ESP не могут применяться в транспортном режиме к пакетам IPv4, являющимся фрагментами. В таких случаях можно использовать только туннельный режим. Для IPv6 можно передавать незашифрованные (plaintext0 фрагменты в транспортном режиме SA; однако для простоты упомянутое ограничение сохраняется и для пакетов IPv6. Более детальное описание работы с незашифрованными фрагментами на защищённой стороне границы IPsec приведено в главе 7.

Для туннельного режима SA существует “внешний” заголовок IP, указывающий отправителя и получателя IPsec, и “внутренний” заголовок IP, который (явно) задаёт первичного отправителя и получателя для пакета. Заголовки протокола защиты размещаются после внешнего заголовка IP, не перед внутренним заголовком IP. Если в туннельном режиме используется протокол AH, некоторые части внешнего заголовка IP также защищаются (как было отмечено выше) вместе с туннелируемым пакетом IP (т. е., внутренний заголовок IP, а также протоколы следующих уровней защищены полностью). При использовании ESP защита обеспечивается только для туннелируемых пакетов, но не для внешнего заголовка.

---

[1] Explicit Congestion Notification - явное уведомление о насыщении.
[2] Generic Routing Encapsulation - базовая инкапсуляция маршрутных данных.
[3] Stream Control Transmission Protocol - протокол управления потоковой передачей.
[4] Simple Network Management Protocol - простой протокол управления сетью.

Требования к поддержке режимов можно резюмировать следующим образом:

a) Хостовая реализация IPsec **должна** поддерживать транспортный и туннельный режим. Это требование относится к реализации в стеке (native), а также к вариантам BITS и BITW.

b) Защитные шлюзы **должны** поддерживать туннельный режим и **могут** поддерживать транспортный режим. При поддержке транспортного режима следует использовать его только в тех случаях, когда защитный шлюз действует как хост (например, для управления шлюзом или обеспечения защиты между парой промежуточных систем на пути).

4.2. Функциональность SA

Набор защитных функций, предлагаемых SA, зависит от выбранного протокола защиты, режима SA, конечных точек SA и выбора дополнительных функций в рамках используемого протокола.

Например, оба протокола AH и ESP обеспечивают аутентификацию и защиту целостности, но набор функций для этих протоколов отличается и зависит от выбранного режима. Если требуется защита опций IPv4 или расширенных заголовков IPv6 на пути между отправителем и получателем, протокол AH может обеспечить такой сервис за исключением того, что заголовок IP или расширенный заголовок может быть изменён непредсказуемо для отправителя.

Однако такую же защиту в некоторых вариантах контекста можно обеспечить путём использования ESP для передающего пакеты туннеля.

Гранулярность обеспечиваемого контроля доступа определяется выбором селекторов, которые определяют каждую связь SA. Более того, способы аутентификации, задействованные партнёрами IPsec (например, IKE SA) также воздействуют на гранулярность контроля доступа.

При выборе защиты конфиденциальности ESP SA в туннельном режиме между двумя защитными шлюзами может обеспечить также некоторую защиту конфиденциальности данных о трафике. Использование туннельного режима позволяет шифровать внутренние заголовки IP, скрывая исходных отправителя и получателя трафика. Более того, может также использоваться заполнение (padding) данных ESP для сокрытия размера пакетов, обеспечивающее дополнительное сокрытие характеристик трафика. Подобная защита сведений о трафике может предлагаться в тех случаях, когда мобильный пользователь с динамическим адресом IP в контексте коммутируемого доступа организует ESP SA в туннельном режиме для соединения с корпоративным МСЭ[1], действующим в качестве защитного шлюза. Отметим, что SA с хорошей гранулярностью в общем случае более уязвимы с точки зрения анализа трафика, нежели менее гранулярные SA, используемые для передачи трафика множества абонентов.

Примечание: Для соответствующих спецификациям реализаций **недопустимо** разрешать организацию ESP SA без шифрования (NULL encryption0 и проверки целостности одновременно. Попытка организации такой связи SA проверяема как со стороны инициатора, так и с отвечающей стороны. В запись журнала системного аудита **следует** включать текущей значения времени и даты, а также локальный и удалённый IP-адреса IKE. Инициатору **следует** включать в журнал соответствующую запись SPD.

4.3. Комбинированные связи SA

Данный документ не требует поддержки вложенных защищённых связей или пакетов SA[2], определённых в RFC 2401 [RFC2401]. На эти возможности по-прежнему оказывает влияние конфигурация SPD и локальных функций пересылки (для входящего и исходящего трафика), но они находятся за пределами модуля IPsec и поэтому не включены в данную спецификацию. Управление вложенными и сгруппированными (nested/bundled) связями SA потенциально более сложно и менее надёжно, нежели модель, предполагаемая RFC 2401 [RFC2401]. Реализации, поддерживающей вложенные связи SA, **следует** обеспечивать интерфейс управления, который позволяет пользователю или администратору выражать требование вложенности и тогда создавать подходящие записи SPD и таблицы пересылки для обеспечения требуемой обработки (пример конфигурации вложенных SA см. в Приложении E).

4.4. Основные базы данных IPsec

Многие параметры, связанные с обработкой трафика IP в реализации IPsec, являются в значительное мере локальными и не задаются стандартами. Однако некоторые внешние аспекты обработки должны быть стандартизованы для обеспечения взаимодействия и поддержки минимального набора функций управления, требуемого для работы IPsec. В этом параграфе описана общая модель обработки трафика IP в контексте функциональности IPsec и поддержки взаимодействия. Описанная ниже модель является концептуальной и реализации не обязаны следовать ей в деталях, но внешнее поведение реализации **должно** соответствовать наблюдаемым извне характеристикам описанной модели.

В этой модели существует три номинальных базы данных - SPD, SAD и PAD[3]. Первая база задаёт правила для входящего и исходящего трафика хоста или защитного шлюза (параграф 4.4.1). Вторая база содержит параметры, относящиеся к каждой действующей связи SA (параграф 4.4.2). Третья база (PAD) обеспечивает связь между протоколом управления SA (таким, как IKE) и базой правил SPD (параграф 4.4.3).

Множество раздельных вариантов контекста IPsec

Если реализация IPsec используется в качестве защитного шлюза для множества абонентов, она **может** поддерживать множество раздельных вариантов контекст IPsec. Каждый контекст **может** иметь и использовать множество полностью независимых идентификаторов, правил, связей управления[4] SA и/или IPsec SA. По большей части это определяется реализацией, однако требуется способ связывания входящих вызовов (SA) с локальным контекстом. Если поддерживается использование протокола управления ключами, идентификаторы контекста **могут** передаваться от инициатора отвечающей стороне в сигнальных сообщениях. В результате этого создаются IPsec

[1] Межсетевой экран. *Прим. перев.*
[2] SA bundle.
[3] Peer Authorization Database - база данных о проверке полномочий партнёров.
[4] Key management SA.

SA с привязкой к определённому контексту. Например, защитный шлюз, предоставляющий VPN-сервис множеству пользователей, будет способен связать трафик каждого пользователя с корректной VPN.

Решения о пересылке и безопасности

Описываемая здесь модель IPsec реализует явное разделение решений о пересылке (маршрутизации) и безопасности для обеспечения возможности работы IPsec в различных вариантах контекста. Пересылка может быть тривиальной в случае использования лишь пары интерфейсов, а может быть сложной, если контекст, в котором реализуется IPsec, использует сложные функции пересылки. IPsec предполагает, что лишь трафик, прошедший обработку IPsec, пересылается в манере, соответствующей контексту использования IPsec. Поддержка вложенных SA является необязательной. Если такая поддержка нужна, требуется обеспечит координацию между таблицей пересылки и записями SPD, чтобы заставить пакеты проходить через границу IPsec более одного раза.

Локальные и удалённые адреса и порты

В этом документе применительно к адресам IP и номерам портов используются термины “локальный” и “удалённый” для выражения правил защиты. Локальными называются адреса и номера портов для объектов, защищённых реализацией IPsec (т. е., адреса и номера портов отправителей для исходящих пакетов и получателей для входящих). Удалёнными будут называться адреса и номера портов на противоположной стороне. Термины отправитель (source) и получатель (destination) используются для полей заголовков пакетов.

Начальные фрагменты

В этом документе фраза «отличный от начального фрагмент[1]» используется для обозначения фрагментов, не содержащих всех значений селекторов, которые могут потребоваться для контроля доступа (например, фрагмент может не содержать поля указания следующего уровня[2], номеров порта для отправителя и получателя, типа или кода ICMP, типа Mobility Header). Термин «начальный фрагмент[3]» используется для обозначения фрагментов, содержащий все значения селекторов, которые могут потребоваться для контроля доступа. Однако следует отметить, что для IPv6 фрагмент, содержащий идентификатор протокола следующего уровня и номера портов (тип/код ICMP или тип Mobility Header [Mobip]) может определяться типом и числом присутствующих расширений заголовка. Поэтому в таком контексте начальный фрагмент может быть не первым.

4.4.1. База правил защиты (SPD)

SA представляет собой управляющую конструкцию, используемую для применения правил защиты к трафику, проходящему через границу IPsec. Таким образом, существенным элементом обработки SA является лежащая в основе база правил защиты SPD, которая задаёт типы сервиса, обеспечиваемого для дейтаграмм IP, и способ реализации защиты. Формат базы данных и её интерфейс выходят за пределы данной спецификации. Однако в данном параграфе определена минимальная функциональность, которая должна обеспечиваться для того, чтобы позволить системному администратору или пользователю определять какие функции IPsec и каким способом применяются к трафику, передаваемому или принимаемому хостом, или проходящему через защитный шлюз. База данных SPD (или соответствующий кэш этой базы) должна использоваться при обработке всего (входящего и исходящего) трафика (включая трафик, для которого не используется защита IPsec), проходящего через границу IPsec. Сюда включается и трафик управления IPsec (такой, как IKE). реализация IPsec **должна** иметь по крайней мере одну базу SPD и **может** поддерживать множество SPD, если это подходит для контекста применения реализации IPsec. Не вводится требования поддержки SPD для каждого интерфейса, как было задано в RFC 2401 [RFC2401]. Однако, если реализация поддерживает множество SPD, она **должна** включать функцию явного выбора SPD, которая служит для указания конкретной базы SPD, используемой для обработки исходящего трафика. Аргументами этой функции могут являться параметры исходящих пакетов и любые локальные метаданные (например, интерфейс, через который получен пакет), требуемые для выбора подходящей базы SPD. Выходными данными функции является идентификатор SPD (SPD-ID).

SPD является упорядоченной базой данных, совместимой со списками контроля доступа (ACL[4]) и пакетными фильтрами в МСЭ, маршрутизаторах и т. п. Требование упорядоченности связано с тем, что записи базы часто будут перекрываться в силу присутствия (непустых) диапазонов в качестве значений селекторов. Поэтому пользователю или администратору **должна** обеспечиться возможность упорядочивания записей для выражения политики контроля доступа. В общем случае не существует способа задать канонический порядок записей SPD, поскольку допускается использование шаблонов (wildcard) для значений селекторов и сами селекторы могут быть различных типов без иерархических отношений между собой.

Варианты обработки - DISCARD, BYPASS, PROTECT

В SPD должны учитываться различия между трафиком, для которого обеспечивается защита IPsec, и трафиком, которому разрешено идти в обход IPsec. Это относится к защите IPsec, используемой отправителем, и к защите IPsec, которая должна присутствовать на приёмной стороне. Для всех входящих и исходящих дейтаграмм возможны три варианта обработки - DISCARD (отбрасывание), BYPASS (обход IPsec) и PROTECT (защита с использованием IPsec). Первый вариант относится к трафику, который не разрешается пропускать через границу IPsec (в заданном направлении). Второй вариант относится к трафику, который может проходить через границу IPsec без использования защиты IPsec. Третий вариант относится к трафику, защищаемому IPsec, и для такого трафика база SPD должна задавать используемые протоколы защиты, их режим, опции защиты, а также используемые криптоалгоритмы.

SPD-S, SPD-I, SPD-O

База SPD логически делится на три части. SPD-S (защищённый трафик) содержит записи для всего трафика, которому обеспечивается защита IPsec. SPD-O (исходящий трафик) содержит записи для всего исходящего трафика, который передаётся в обход или отбрасывается. SPD-I (входящий трафик) применяется к входящему трафику, который отбрасывается или передаётся в обход. Все три части базы могут быть декоррелированы (за

[1] Non-initial fragment.
[2] Next Layer Protocol.
[3] Initial fragment.
[4] Access Control List.

упомянутым выше исключением для реализации IPsec в стеке хоста) для облегчения кэширования. Если реализация IPsec поддерживает только одну базу SPD, эта SPD включает все три части. При поддержке множества SPD некоторые из баз могут быть неполными. Например, некоторые SPD могут включать только SPD-I для независимого контроля за передаваемым в обход входящим трафиком на каждом интерфейсе. Расщепление позволяет использовать SPD-I для входящего трафика без обращений к базе SPD-S для такого трафика. Поскольку SPD-I является лишь частью SPD, если пакет не соответствует ни одной из записей SPD-I, такой пакет **должен** отбрасываться. Отметим, что для исходящего трафика отсутствие записи в SPD-S ведёт к проверке SPD-O для аутентификации трафика, передаваемого в обход. Если же сначала проверяется SPD-O, то при отсутствии соответствий должна потом просматриваться база SPD-S. В упорядоченных SPD без декорреляции записи для SPD-S, SPD-I и SPD-O чередуются, поэтому возможен однократный поиск в SPD.

Записи SPD

Каждая запись SPD задаёт судьбу пакета - BYPASS, DISCARD или PROTECT. Запись снабжается списком из одного или множества селекторов. База SPD содержит упорядоченный набор таких записей. Обязательные типы селекторов определены в параграфе 4.4.1.1. Эти селекторы служат для задания гранулярности SA, создаваемых в ответ на исходящий пакет или по запросу от партнёра. Детальное описание структуры записей SPD приведено в параграфе 4.4.1.2. Каждой базе SPD **следует** иметь номинальную конечную запись, которая соответствует всему, что не нашло соответствия в предыдущих записях, и обеспечивает отбрасывание таких пакетов. База SPD **должна** обеспечивать пользователю или администратору возможность задания политики записей, как показано ниже:

- SPD-I: для входящего трафика, который передаётся в обход или отбрасывается; запись содержит значения селекторов, определяющих трафик для обхода или отбрасывания;
- SPD-O: для исходящего трафика, который передаётся в обход или отбрасывается; запись содержит значения селекторов, определяющих трафик для обхода или отбрасывания;
- SPD-S: для трафика, защищаемого с использованием IPsec; запись содержит значения селекторов, определяющих трафик для защиты с использованием AH или ESP, контролирует создание SA на основе этих селекторов и параметры, требуемые для обеспечения защиты (например, алгоритмы, режимы и т. п.). Отметим, что записи SPD-S содержат такую информацию, как флаг PFP[1] (см. ниже параграф «Как получить значения для записи SAD«) и биты, показывающие используются ли при поиске SA локальные и удалённые адреса IP в дополнение к SPI (см. спецификации AH [Ken05b] и ESP [Ken05a]).

Задание направления в записях SPD

Для трафика, защищаемого IPsec, локальные и удалённые адреса и номера портов меняются местами в записи SPD для задания направления в соответствии с соглашениями IKE. В общем случае протоколы, с которыми имеет дело IPsec, требуют организации симметричных SA с поменянными местами локальными и удалёнными адресами IP. Однако для ICMP зачастую не требуется авторизация в обоих направлениях. Несмотря на это, из соображений однородности и простоты записи SPD для ICMP задаются так же, как и для остальных протоколов. Отметим также, что для ICMP, Mobility Header и фрагментов, отличных от начального, в пакетах не содержатся номера портов. ICMP использует тип и код сообщения, а Mobility Header - тип заголовка. Поэтому записи SPD выражают правила контроля доступа для этих протоколов с использованием соответствующих полей вместо номеров портов. Для передаваемого в обход или отбрасываемого трафика поддерживаются раздельные записи для каждого направления, чтобы обеспечить возможность независимого контроля в обоих направлениях.

OPAQUE и ANY

Для каждого селектора в записи SPD в дополнение к литеральным значениям, определяющим соответствие, имеются два специальных значения ANY (любой) и OPAQUE. Значение ANY является шаблоном, которому соответствует любое значение указанного поля в пакете, а также пакеты, где это поле отсутствует или непонятно. OPAQUE показывает, что соответствующее поле селектора недоступно для проверки, поскольку его нет во фрагменте, не существует для данного протокола следующего уровня или предыдущее использование IPsec привело к шифровке значения этого поля. Значение ANY включает в себя и OPAQUE. Таким образом, OPAQUE нужно использовать лишь тогда, когда требуется отличать любые значения поля от случаев отсутствия или недоступности (например, в результате шифрования) поля.

Как получить значения для записи SAD

Для каждого селектора в записи SPD эта запись задаёт как получаются соответствующие значения для новой базы данных SA (SAD, см. параграф 4.4.2) из записей SPD и пакета. Задача состоит в том, чтобы обеспечить создание записи SAD и кэшированной записи SPD на основе значений указанного селектора из пакета или соответствующей записи SPD. Для исходящего трафика имеются кэшированные записи SPD-S и SPD-O. Для входящего трафика, не защищаемого IPsec, имеются кэшированные записи SPD-I и SAD, которая представляет кэш для входящего трафика, защищаемого IPsec (см. параграф 4.4.2). Если для записи задана обработка IPsec, может устанавливаться флаг PFP для одного или множества селекторов в записи SPD (локальный адрес IP, удалённый адрес IP, протокол следующего уровня и, в зависимости от протокола следующего уровня, локальный и удалённый порт, тип и код ICMP или тип Mobility Header). Установленный для данного селектора X флаг показывает, что для создаваемой связи SA следует брать значение X из пакета. При отсутствии флага SA следует брать значение для X из записи SPD.

Примечание: В случае отсутствия флага PFP значение селектора, согласуемое протоколом управления SA (например, IKEv2), может быть подмножеством значений в записи SPD, в зависимости от заданной партнёром политики SPD. Вопрос использования одного флага для всех селекторов (например, порт отправителя, тип/код ICMP или тип заголовка MH[2]) или отдельного флага для каждого селектора решается локально.

Приведённый ниже пример иллюстрирует использование флага PFP в контексте защитного шлюза или реализации BITS/BITW. Рассмотрим запись SPD, где разрешён диапазон удалённых адресов IPv4 от 192.0.2.1 до 192.0.2.10. Предположим, что исходящий пакет приходит с адресом получателя 192.0.2.3 и пока не существует SA для

[1]Populate from packet - заполнить из пакета.
[2]Mobility Header.

доставки этого пакета. Значение, используемое при создании SA для передачи этого пакета может быть одним из двух, показанных ниже в зависимости от того, что запись SPD для этого селектора задаёт в качестве источника значений для селектора:

Значение флага PPF для селектора удалённого адреса	***Пример значения селектора удалённого адреса для новой SAD***
a. PFP TRUE	192.0.2.3 (один хост)
b. PFP FALSE	192.0.2.1 - 192.0.2.10 (группа хостов)

Отметим, что если показанная выше запись SPD имеет значение ANY для удалённого (Remote) адреса, для значения селектора SAD будет выбрано ANY в случае (b), но сохранится показанное в примере значение для случая (a). Таким образом, флаг PFP можно использовать для запрета совместного использования SA даже среди пакетов, соответствующих одной записи SPD.

Интерфейс управления

Для каждой реализации IPsec **должен** поддерживаться интерфейс управления, обеспечивающий пользователю или системному администратору возможность управления SPD. Интерфейс должен позволять пользователю (или администратору) задавать режим обработки для каждого пакета, проходящего через границу IPsec[1]. Интерфейс управления для SPD **должен** позволять создание записей, совместимых с селекторами, определёнными в параграфе 4.4.1.1, а также **должен** поддерживать (полное) упорядочивание записей, видимых через этот интерфейс. Селекторы записей SPD аналогичны спискам контроля доступа ACL или пакетным фильтрам, которые повсеместно используются в МСЭ без учёта состояний (stateless firewall) или маршрутизаторах с фильтрацией пакетов и управляются аналогичным путём. На хостовых системах приложениям **может** быть разрешено создание записей SPD[2]. Однако системному администратору **должна** предоставляться возможность разрешать или запрещать пользователям или приложениям переписывать (принятые по умолчанию) системные правила. Форма интерфейса управления не задаётся данным документом и может различаться для хостов и защитных шлюзов, а также для хостов, использующих интерфейс сокетов, и реализаций BITS. Тем не менее данный документ задаёт стандартный набор элементов SPD, который **должны** поддерживать все реализации IPsec.

Декорреляция

Описываемая в этом документе модель обработки предполагает возможность декорреляции перекрывающихся записей SPD для обеспечения возможности кэширования, которое повышает эффективность обработки исходящего трафика в защитных шлюзах и реализациях BITS/BITW. Декорреляция [CoSa04] является лишь способом повышения производительности и упрощения описания обработки. Данный документ не требует от соответствующих спецификации реализаций обязательно использовать декорреляцию. Например, реализации в стеке протоколов хоста обычно используют косвенное кэширование (caching implicitly), поскольку они связывают SA с интерфейсами сокетов и в результате такого связывания в таких реализациях не требуется декоррелировать записи SPD.

Примечание: Если явно не указано иное, термин SPD относится к информации о политике как в упорядоченном, так и в декоррелированном (неупорядоченном) состоянии. Приложение B описывает алгоритм, который можно использовать для декоррелирования записей SPD. На практике можно использовать любой алгоритм, который даёт эквивалентный описанному в приложении результат. Отметим, что после декорреляции записей SPD все полученные в результате записи **должны** быть связаны воедино, чтобы все члены группы, полученной из одной (до декорреляции) записи SPD, могли быть одновременно помещены в кэш и SAD. Для примера предположим, что запись A (из упорядоченной базы SPD) после декорреляции даёт записи A1, A2 и A3. Когда приходящий пакет соответствует, например, записи A2 и вызывает создание SA, протокол управления SA (например, IKEv2) согласует A. Все три декоррелированные записи A1, A2, A3 помещаются в подходящий кэш SPD-S и связываются с SA. Смысл этого состоит в том, чтобы использование декоррелированных SPD не приводило к созданию большего числа SA, нежели было бы создано при использовании SPD без декорреляции.

При использовании декоррелированной SPD существуют три варианта информации, которую инициатор передаёт партнёру через протокол управления SA (например, IKE). Передавая полный набор декоррелированных, связанных записей, которые были выбраны из SPD, партнёр даёт наиболее полную информацию для того, чтобы можно было выбрать подходящую запись SPD на его стороне (особенно в тех случаях, когда партнёр также декоррелировал свою SPD). Однако, если связано большое число декоррелированных записей, это может вести к созданию больших пакетов для согласования SA и связанных с этим проблем фрагментации для протокола управления SA. В качестве альтернативы может быть сохранена и передана протоколу управления SA исходная запись из (коррелированной) базы SPD. Передача записи из коррелированной SPD сохраняет локальное использование декоррелированной SPD (невидимое для партнёров) и избавляет от возможных проблем с фрагментацией, хотя не обеспечивает столь точной информации отвечающей стороне для поиска соответствия в SPD.

В качестве промежуточного варианта может использоваться передача подмножества полного набора связанных, декоррелированных записей SPD. Этот вариант позволяет избежать проблем, связанных с фрагментацией, и обеспечивает больше информации, нежели передача исходной коррелированной записи. Основным недостатком этого метода является то, что он может вызывать создание впоследствии дополнительных связей SA, поскольку партнёру передаётся только часть связанных, декоррелированных записей. Реализация может выбрать любой из трёх рассмотренных выше методов.

Отвечающая сторона использует селекторы трафика, полученные через протокол управления SA для выбора подходящей записи в своей базе SPD. Цель поиска соответствия заключается в выборе записи SPD и создании связи SA, которая наиболее точно соответствует намерениям инициатора, чтобы трафик, передаваемый через созданную SA был воспринят обеими сторонами. Если отвечающая сторона использует декоррелированную SPD, ей **следует** SHOULD искать соответствие в декоррелированных записях SPD, поскольку это в общем случае будет приводить к созданию SA, которые наиболее точно соответствуют намерениям обеих сторон. Если отвечающая сторона использует коррелированную SPD, ей **следует** проводить поиск соответствия в коррелированных записях.

[1]При реализации IPsec непосредственно в стеке протоколов хоста с использованием интерфейса сокетов может не потребоваться обращений к SPD для каждого пакета, как отмечено в конце параграфа 4.4.1.1 и параграфа 5.
[2]Способы передачи сигналов о таких запросах реализации IPsec выходят за рамки настоящего стандарта.

Для IKEv2, использование декоррелированной SPD обеспечивает отвечающей стороне наилучшие возможности генерации наиболее подходящего отклика.

В любом случае при доступности декоррелированной SPD для заполнения кэша SPD-S используются декоррелированные записи. Если SPD не декоррелирована, кэширование не допускается и **должен** выполняться упорядоченный поиск в SPD для проверки того, что входящий трафик, поступающий в SA, соответствует политике контроля доступа, выраженной в SPD.

Обработка изменений в SPD во время работы системы

Если вносятся изменения в SPD во время работы системы, **следует** проверить воздействие этих изменений на существующие связи SA. Реализации **следует** проверить влияние изменений в SPD на существующие связи SA, а также **следует** предоставить пользователю/администратору механизм выбора предпринимаемых действий (например, удалить SA, на которые воздействуют внесённые изменения, позволить SA продолжать работу без изменений и т. п.).

4.4.1.1. Селекторы

Связи SA могут быть “крупнозернистыми” или “мелкозернистыми” в зависимости от селекторов, используемых при определении трафика для SA. Например, весь трафик между парой хостов может передаваться через одну связь SA с применением одного набора средств защиты. И напротив, трафик между парой хостов может передаваться с использованием множества SA, определяемых используемыми приложениями (задаются протоколом следующего уровня и связанными полями - например, номерами портов), с различными наборами средств защиты для разных SA. Подобно этому весь трафик между парой хостов может передаваться через одну SA или одна связь SA может выделяться для каждой пары обменивающихся данными хостов. Перечисленные ниже параметры селекторов **должны** поддерживаться всеми реализациями IPsec для облегчения контроля гранулярности SA. Отметим, что оба адреса - локальный и удалённый - должны быть IPv4 или IPv6, но не различных типов. Отметим также, что селекторы локального и удалённого портов (а также тип и код сообщений ICMP и тип Mobility Header) могут помечаться как OPAQUE для использования в тех ситуациях, когда эти поля становятся недоступными в результате фрагментации пакетов.

- **Remote IP Addresses** - удалённые адреса - (IPv4 или IPv6): Список диапазонов адресов IP (индивидуальных - unicast, широковещательных - broadcast (только IPv4)). Данная структура позволяет указать один адрес IP (вырожденный диапазон), список отдельных адресов (каждый адрес в виде вырожденного диапазона) или диапазон адресов (верхняя и нижняя граница, включительно), а также наиболее общую (most generic) форму списка диапазонов. Диапазоны адресов задаются для поддержки множества удалённых систем, использующих одну SA (например, после защитного шлюза).

- **Local IP Addresses** - локальные адреса - (IPv4 или IPv6): Список диапазонов адресов IP (индивидуальных - unicast, широковещательных - broadcast (только IPv4)). Данная структура позволяет указать один адрес IP (вырожденный диапазон), список отдельных адресов (каждый адрес в виде вырожденного диапазона) или диапазон адресов (верхняя и нижняя граница, включительно), а также наиболее общую (most generic) форму списка диапазонов. Диапазоны адресов задаются для поддержки множества удалённых систем, использующих одну SA (например, после защитного шлюза). Термин “локальный” обозначает, что эти адреса защищаются данной реализацией (или записью политики).

  Примечание: SPD не включает поддержку записей для групповых (multicast) адресов. Для поддержки групповых SA реализации следует использовать Group SPD (GSPD), как определено в [RFC3740]. Записи GSPD требуют иной структуры, т. е. Они не могут использовать симметричных отношений, связанных локальными или удалёнными адресами индивидуальных SA в multicast-контексте. В частности, исходящий трафик, направленный по групповому адресу в SA, не будет получен парной входной связью SA с групповым адресом в поле отправителя.

- **Next Layer Protocol -** протокол следующего уровня: Это значение берётся из поля IPv4 "Protocol" или IPv6 "Next Header". Это может быть номер конкретного протокола, значение ANY (все) или (только для IPv6) - OPAQUE. Значение Next Layer Protocol размещается после всех расширений заголовков. Для упрощения поиска Next Layer Protocol **следует** поддерживать механизм, который позволяет задать пропускаемые расширения заголовков IPv6. По умолчанию **следует** пропускать протоколы 0 (опции Hop-by-hop), 43 (Routing Header), 44 (Fragmentation Header) и 60 (Destination Options). Отметим, что этот список не включает значений 51 (AH) и 50 (ESP). С точки зрения поиска селекторов IPsec трактует AH и ESP как протоколы следующего уровня.

  Несколько дополнительных селекторов используются в зависимости от значение Next Layer Protocol:

  – Если протокол следующего уровня использует два порта (как это делают TCP, UDP, SCTP и др.), тогда существуют селекторы для локального и удалённого портов. Каждый из этих селекторов представляет собой список диапазонов значений. Отметим, что номера портов Local и Remote могут оказаться недоступными в случаях прихода фрагментов или при защите этих полей с помощью IPsec (шифрованные поля); таким образом, **должно** поддерживаться значение OPAQUE. Отметим, что в отличных от начального фрагментах номера портов недоступны. Если селектор портов задаёт значение, отличное от ANY или OPAQUE, ему не будут соответствовать никакие фрагменты, кроме начальных. Если SA требует для порта значение, отличное от ANY или OPAQUE, прибывающие фрагменты без номеров портов **должны** отбрасываться (см. главу 7, "Обработка фрагментов".)

  – Если протоколом следующего уровня является Mobility Header, существует селектор типа сообщения IPv6 Mobility Header (тип MH) [Mobip]. Это 8-битовое значение идентифицирует отдельное сообщение. Отметим, что тип MH может быть недоступен при получении фрагмента (см. главу 7). Для IKE тип MH помещается в 8 старших битов 16-битового селектора локального “порта”.

  – Если протоколом следующего уровня является ICMP, существует 16-битовый селектор типа и кода сообщения ICMP. Тип сообщения представляет собой одно 8-битовое значение, которое определяет тип сообщения ICMP или ANY. Код ICMP представляет собой одно 8-битовое значение, которое определяет специфический подтип для сообщения ICMP. Для IKE тип сообщения указывается в 8 старших битах 16-битового селектора, а код - в 8 младших битах этого селектора. Этот 16-битовый селектор может содержать один тип и диапазон кодов, один тип и любой (ANY) код, а также любой (ANY) тип с любым (ANY) кодом. Для правила с диапазоном типов от T-start до T-

end и кодов от C-start до C-end и пакета ICMP с типом t и кодом с реализация **должна** проверять выполнение условий

```
(T-start*256) + C-start <= (t*256) + c <= (T-end*256) + C-end
```

Отметим, что тип и код сообщения ICMP могут быть недоступны в случаях получения фрагментов (см. главу 7).

– **Name** - имя. Этот селектор не похож на перечисленные выше. Он не извлекается из пакета. Имя может использоваться как символьный идентификатор для локального или удалённого адреса IPsec. Именованные записи SPD используются двумя способами:

1. Именованные записи SPD используются отвечающей стороной (responder) для поддержки контроля доступа в тех случаях, когда для IP-адреса нет подходящего селектора Remote IP (например, для мобильного пользователя). Имя, используемое для сравнения с этим полем, передаётся в процессе согласования IKE в поле ID payload. В этом контексте адрес инициатора Source IP (внутренний заголовок IP в туннельном режиме) связан с адресом Remote IP в записи SAD, создаваемой при согласовании IKE. Этот адрес заменяется в SPD значением Remote IP, при выборе записи SPD означенным способом. Все реализации IPsec **должны** поддерживать такое использование имён.
2. Именованная запись SPD может использоваться инициатором для аутентификации пользователя, для которого будет создаваться IPsec SA (или чей трафик будет передаваться в обход). Для замены перечисленных ниже параметров используется IP-адрес инициатора (из внутреннего заголовка IP в туннельном режиме):

   – локальный адрес в кэшированной записи SPD;

   – локальный адрес в исходящей записи SAD;

   – удалённый адрес во входящей записи SAD.

   Поддержка такого использования является необязательной реализаций и не применима к некоторым реализациям. Отметим, что имя используется лишь локально - оно не передаётся протоколами обмена ключами. Отметим также, что в контексте инициатора (см. ниже)применимы имена, формируемые иначе, нежели в случае 1 (responder).

Запись SPD может содержать имя (или список имён.), а также значения локального и удалённого адресов IP.

Для случая 1 (responder) идентификаторы, используемые в именованных записях SPD относятся к одному из перечисленных типов:

a. полный адрес электронной почты пользователя - например, mozart@foo.example.com (соответствует ID_RFC822_ADDR в IKEv2)
b. полное доменное имя DNS - например, foo.example.com (соответствует ID_FQDN в IKEv2)
c. имя[1] X.500 - например, [WaKiHo97], CN = Stephen T. Kent, O = BBN Technologies, SP = MA, C = US (соответствует ID_DER_ASN1_DN в IKEv2 после декодирования)
d. строка байтов (соответствует Key_ID в IKEv2)

Для случая 2 (initiator) идентификаторы, используемые в именованных записях SPD представляют собой строки байтов. Ясно, что это могут быть идентификаторы пользователей Unix (UID), идентификаторы защиты Windows (security ID) или что-то в этом же роде, но могут также использоваться имена пользователей или учётных записей. В любом случае этот идентификатор имеет лишь локальное значение и не передаётся.

Контекст реализации IPsec определяет использование селекторов. Например, естественная реализация для хоста обычно использует интерфейс сокетов. При создании нового соединения может делаться запрос к SPD с привязкой SA к сокету. Таким образом трафик, передаваемый через сокет, не будет требовать дополнительного просмотра кэша SPD (SPD-O и SPD-S). Напротив, реализации BITS, BITW или защитных шлюзов должны выполнять просмотр для каждого пакета, выполняя поиск в кэше SPD-O/SPD-S на основе селекторов.

4.4.1.2. Структура записи SPD

В этом параграфе приводится описание структуры записей SPD. Приложение C содержит пример определения ASN.1 для записи SPD.

Описание SPD построено так, чтобы дать прямое отображение на данные IKE - это позволит согласовывать правила, требуемые SPD через IKE. К сожалению семантика IKEv2 [Kau05] была опубликована одновременно с этим документом, что не позволило обеспечить точное соответствие определений для SPD. В частности, IKEv2 не разрешает согласования для одной SA, которая связывает множество пар локальных и удалённых адресов и портов с одной SA. Вместо этого, при согласовании множества локальных и удалённых адресов и портов для SA, IKEv2 трактует их не как пары, а как (неупорядоченные) наборы локальных и удалённых значений, которые можно произвольно спаривать. Пока IKE не обеспечивает средства переноса семантики, выражаемой в SPD через наборы селекторов (как описано ниже), пользователям **недопустимо** включать множество наборов селекторов в одну запись SPD, если смысл контроля доступа не согласован с семантикой IKE «mix and match». Реализация **может** предупреждать пользователей для предотвращения проблем, если пользователь создаёт записи SPD со множеством наборов селекторов; синтаксис предупреждения показывает возможные конфликты с семантикой IKE.

Графический интерфейс (GUI) управления может предлагать пользователю другие формы записей (например, опцию использования адресных префиксов и диапазонов адресов, символьные имена[2] протоколов и портов и т. п.). Отметим, что опции Remote/Local (удалённый/локальный) применяются только к адресам IP и портам, но не типам/кодам ICMP или типам Mobility Header. Если зарезервированы символьные значения OPAQUE или ANY, они используются для данного типа селектора - только данное значение может появляться в списке для этого селектора и должно присутствовать в списке только один раз. Отметим, что для ANY и OPAQUE используются локальные синтаксические соглашения - IKEv2 согласует эти значения с использованием показанных ниже диапазонов.

[1] В оригинале - distinguished name. *Прим. перев.*
[2] Не следует путать символьные имена в интерфейсе управления с селектором SPD «Name».

```
ANY:     start = 0         end = <max>
OPAQUE:  start = <max>     end = 0
```

SPD представляет собой упорядоченный список элементов, каждый из которых содержит перечисленные ниже поля.

- **Name** - список идентификаторов. Этот квазиселектор является необязательным. Формы, которые **должны** поддерживаться, описаны выше в параграфе 4.4.1.1 (Name).
- **PFP flags** - флаги PPP (одно значение на селектор). Данный флаг (например, Next Layer Protocol - протокол следующего уровня) применяется к подходящим селекторам всех наборов селекторов (см. ниже), содержащихся в элементе SPD. При создании SA каждый флаг задаёт для соответствующего селектора трафика, создаётся селектор из соответствующего поля в пакете, вызвавшем создание SA, или из значения(й) в соответствующей записи SPD (см. параграф 4.4.1 «Как получить значения для записи SAD»). Использование одного флага для множества элементов (например, порта отправителя, типа/кода ICMP, типа MH) или раздельных флагов для каждого элемента определяется локально. Имеются флаги PFP для следующих элементов:
  - локальный адрес;
  - удалённый адрес;
  - протокол следующего уровня;
  - локальный порт, тип/код сообщения ICMP или тип заголовка Mobility (в зависимости от протокола следующего уровня);
  - удалённый порт, тип/код сообщения ICMP или тип заголовка Mobility (в зависимости от протокола следующего уровня).
- От 1 до N наборов селекторов, соответствующих «условию» применения определённых действий IPsec. Каждый набор селекторов содержит:
  - локальный адрес;
  - удалённый адрес;
  - протокол следующего уровня;
  - локальный порт, тип/код сообщения ICMP или тип заголовка Mobility (в зависимости от протокола следующего уровня);
  - удалённый порт, тип/код сообщения ICMP или тип заголовка Mobility (в зависимости от протокола следующего уровня).

Примечание. Селектор next protocol имеет индивидуальное значение (в отличие от локальных и удалённых адресов IP) в записи набора селекторов. Это согласуется с принятым в IKEv2 согласованием значений TS (селектор трафика) для SA. Это осмысленно ещё и потому, что может возникнуть необходимость связать различные поля портов с различными протоколами. Можно связать множество протоколов (и портов) в одной SA путём задания множества наборов селекторов для этой SA.

- **Processing info** - информация о требуемых действиях (PROTECT - защита, BYPASS - обход или DISCARD - отбрасывание). Это просто одно действие, относящееся ко всем наборам селекторов, а не отдельные действия для каждого набора. Если требуемой обработкой является защита (PROTECT) запись содержит перечисленную ниже информацию.
  - Режим IPsec - туннельный или транспортный.
  - (в туннельном режиме) локальный адрес туннеля - для стационарного (не мобильного) хоста - это просто адрес единственного интерфейса или (при наличии множества интерфейсов) специально указанный адрес. Для мобильный хостов задание локального адреса является внешним по отношению к IPsec.
  - (в туннельном режиме) удалённый адрес туннеля - для определения этого адреса не существует стандартных путей (см. параграф 4.5.3. Нахождение защитного шлюза).
  - расширенный порядковый номер (ESN) - использует ли данная SA расширенные номера?
  - проверка фрагментов с учётом состояния - выполняет ли данная SA такую проверку (см. раздел 7).
  - обход бита DF (T/F) - применимо к SA в туннельном режиме.
  - обход DSCP (T/F) или отображение на незащищённые значения DSCP (массив), если требуется ограничить обход значений DSCP - применимо к SA в туннельном режиме.
  - протокол IPsec protocol - AH или ESP.
  - Алгоритмы - используемые для AH, для ESP, для комбинированного режима в порядке убывания приоритета.

Набор сохраняемой информации, связанной с обслуживанием остающихся SA при изменении SPD задаётся локально.

4.4.1.3. Дополнительная информация о полях, связанных с протоколами следующего уровня

С полями в заголовке Next Layer Protocol часто связываются дополнительные селекторы. Конкретный заголовок Next Layer Protocol может иметь до 2 селекторов. Возможны ситуации, когда отсутствуют локальные и удалённые селекторы для полей, зависящих от Next Layer Protocol. Заголовок IPv6 Mobility имеет тип сообщения только Mobility Header. AH и ESP не имеют добавочных полей селекторов. Система может пожелать передать тип и код сообщений ICMP, которые

она не хочет получать. В приведённых ниже описаниях термин «port» служит для обозначения поля, зависящего от Next Layer Protocol.

A) Если Next Layer Protocol не имеет селекторов «port», для локального и удалённого селекторов «port» у соответствующей записи SPD указываются значения OPAQUE.

```
Локальный       next layer protocol = AH
                "port" selector     = OPAQUE
Удалённый       next layer protocol = AH
                "port" selector     = OPAQUE
```

B) Даже если Next Layer Protocol имеет единственный селектор (например, тип Mobility Header), селекторы локального и удалённого «портов» используются для индикации желания системы передавать и/или принимать трафик с заданными значениями «портов». Например, если разрешено передавать и принимать заголовки Mobility заданного типа через SA, соответствующая запись SPD будет иметь вид:

```
Локальный       next layer protocol = Mobility Header
                "port" selector     = тип сообщения Mobility Header
Удалённый       next layer protocol = Mobility Header
                "port" selector     = тип сообщения Mobility Header
```

Если указанный тип заголовков Mobility разрешён для передачи, но не принимается через SA, соответствующая запись SPD имеет вид:

```
Локальный       next layer protocol = Mobility Header
                "port" selector     = тип сообщения Mobility Header
Удалённый       next layer protocol = Mobility Header
                "port" selector     = OPAQUE
```

Если указанный тип заголовков Mobility разрешён для приёма, **но** не разрешён для передачи, соответствующая запись SPD будет иметь вид:

```
Локальный       next layer protocol = Mobility Header
                "port" selector     = OPAQUE
Удалённый       next layer protocol = Mobility Header
                "port" selector     = тип сообщения Mobility Header
```

C) Если система желает передавать трафик с определенным значением «порта», **но** не желает принимать трафик с таким значением, в соответствующей записи SPD системные селекторы трафика имеют вид:

```
Локальный       next layer protocol = ICMP
                "port" selector     = <указанный тип и код ICMP>
Удалённый       next layer protocol = ICMP
                "port" selector     = OPAQUE
```

D) Для индикации желания системы принимать трафик с определенным «портом», **но** не передавать такого трафика в соответствующей записи SPD системные селекторы трафика имеют вид:

```
Локальный       next layer protocol = ICMP
                "port" selector     = OPAQUE
Удалённый       next layer protocol = ICMP
                "port" selector     = <указанный тип и код ICMP>
```

Например, если защитный шлюз позволяет находящимся за ним системам трассировку ICMP, но не хочет открывать для внешних систем трассировку ICMP к расположенным за ним системам, в соответствующей записи SPD селекторы трафика защитного шлюза имеют вид:

```
Локальный       next layer protocol = 1 (ICMPv4)
                "port" selector     = 30 (traceroute)
Удалённый       next layer protocol = 1 (ICMPv4)
                "port" selector     = OPAQUE
```

4.4.2. База защищённых связей (SAD)

В каждой реализации IPsec существует номинальная база данных о защищённых связях (SAD), каждая запись которой определяет параметры, связанные с одной SA. Каждая SA имеет запись в SAD. Для обработки исходящего трафика каждая запись SAD указывается записями в части SPD-S кэша SPD. При обработке входящего трафика для SA с индивидуальной адресацией, SPI используется для поиска SA самостоятельно или в комбинации с типом протокола IPsec. Если реализация IPsec поддерживает групповую адресацию, для поиска SA используется SPI и адрес получателя или SPI в комбинации с адресами отправителя и получателя (в параграфе 4.1 подробно описаны алгоритмы, которые должны использоваться для отображения входящих дейтаграмм IPsec на SA). Описанные далее параметры связаны с каждой записью в SAD. Этим параметрам следует присутствовать всегда, если явно не указано иное (например, алгоритм аутентификации AH). Это описание не является MIB[1] и задаёт лишь минимальный набор данных, требуемых для поддержки SA в реализации IPsec.

Для каждого из селекторов, определённых в параграфе 4.4.1.1, запись SAD для входящей SA **должна** быть изначально заполнена значением или значениями, согласованными при создании SA (см. параграф 4.4.1 «Обработка изменений в SPD во время работы системы», где рассмотрено влияние изменений в SPD на остающиеся SA). Для получателя эти

[1]Management Information Base - база данных управления. *Прим. перев.*

значения используются при проверке соответствия полей входящих пакетов (после обработки IPsec) значениям селекторов, согласованным для SA. Таким образом, SAD действует, как кэш для проверки селекторов входящего трафика, поступающего в SA. Для получателя это является частью проверки того, что приходящий в SA пакет согласуется с политикой для данной SA (см. раздел 6, содержащий правила для сообщений ICMP). Эти поля могут иметь форму отдельных значений или диапазонов, а также ANY или OPAQUE, как описано в параграфе 4.4.1.1. Селекторы. Отметим также, что существуют ситуации, когда в SAD имеются записи для SA, которые не имеют соответствующих записей в SPD. Поскольку этот документ не задаётся обязательной селективной очистки SAD при изменении SPD, записи SAD могут оставаться в то время, как создавшие их записи SPD будут изменены или удалены. Также при ручном создании SA для них могут быть записи SAD, которые не соответствуют записям SPD.

Примечание. SAD может поддерживать SA с групповой адресацией, если это задано вручную. На исходящих SA с групповой адресацией структура совпадает с обычной SA. В качестве адреса отправителя указывается адрес передающего хоста, а в качестве адреса получателя - адрес группы. Для входящего трафика SA с групповой адресацией должны включать в качестве адресов отправителей адреса всех партнёров, которые уполномочены передавать в данную SA трафик в групповыми адресами. Значение SPI для групповой SA обеспечивается контроллером группы, а не получателем, как для обычных SA. Поскольку для записей SAD может требоваться включение множества индивидуальных IP-адресов отправителей, которые были частью записи SPD (для обычных SA), требуемое для входящих групповых SA свойство уже присутствует в реализации IPsec. Однако, поскольку SPD не имеет средств для аккомодации групповых записей, этот документ не задаёт способа автоматического создания записей SAD для входящих SA с групповой адресацией. Для аккомодации входящего трафика с групповой адресацией записи SAD могут создаваться лишь вручную.

Рекомендации для разработчиков. Этот документ не задаёт, как запись SPD-S ссылается на соответствующую запись SAD, поскольку это зависит от реализации. Однако известно, что некоторые реализации (основанные на опыте из RFC 2401) имеют проблемы в этой части. В частности, простого сохранения пары (IP-адрес заголовка удалённого туннеля, удалённый SPI) в кэше SPD недостаточно, поскольку такая пара не всегда позволяет однозначно идентифицировать одну запись SAD. Например, два хоста за одним устройством NAT могут выбрать одинаковое значение SPI. Аналогичная ситуация может возникать, когда хост получает адрес IP (например, от DHCP), который раньше использовался другим хостом и SA, связанные с тем хостом ещё не были удалены механизмом обнаружения «умерших» хостов. Это может приводить к тому, что пакеты будут передаваться через некорректную SA или, если управления ключами обеспечивает уникальность пары, отказу от создания корректных SA. Таким образом, реализациям следует поддерживать связь между кэшем SPD и SAD таким образом, чтобы не возникало упомянутых проблем.

4.4.2.1. Элементы данных в SAD

В SAD **должны** присутствовать следующие элементы данных:

- Список параметров защиты (SPI) - 32-битовое значение, выбираемое передающей стороной SA для уникальной идентификации SA. В записи SAD для исходящей SA значение SPI используется для создания заголовков пакетов AH и ESP. В записи SAD для входящей SA значение SPI используется для отображения трафика на соответствующую SA (см. параграф 4.1).

- Счётчик порядковых номеров - 64-битовое значение, используемое для генерации поля Sequence Number в заголовках пакетов AH и ESP. По умолчанию используются 64-битовые номера, но по согласованию могут использоваться и 32-битовые порядковые номера.

- Переполнение порядкового номера - флаг, показывающий нужно ли при переполнении счётчика порядковых номеров вносить запись в журнал аудита и прекращать дальнейшую передачу пакетов в SA или можно начинать отсчёт номеров заново. В журнальную запись о переполнении счётчика **следует** включать значение SPI, текущую дату и время, локальный и удалённый адрес, а также селекторы для соответствующей записи SAD.

- Окно предотвращения повторов - 64-битовый счётчик и битовое отображение (или эквивалент), используемое для детектирования повторного использования входящих пакетов AH или ESP[1].

- Алгоритм аутентификации, ключ и другие параметры AH (требуется только при включённой поддержке AH).

- Алгоритм шифрования, ключ, режим, IV и другие параметры ESP. При использовании комбинированного режима эти поля не будут применяться.

- Алгоритм защиты целостности, ключи и другие параметры ESP. Если защита целостности не выбрана, эти поля не будут применяться. При использовании комбинированного режима эти поля не будут применяться.

- Алгоритмы, ключи и другие параметры комбинированного режима ESP. Эти данные используются при выборе для ESP комбинированного алгоритма (шифрование и защита целостности). Если комбинированный алгоритм не используется, эти поля не будут применяться.

- Время жизни данной SA - интервал, по завершении которого данная SA должна быть заменена новой SA (с новым SPI) или прервана, с индикацией какое из двух означенных действий следует выполнять. Срок жизни может определяться временем или количеством байтов, а также обоими параметрами (прерывание по первому порогу). Соответствующие спецификации реализации **должны** поддерживать оба типа и возможность одновременного задания двух порогов. Если задан временной порог и IKE использует сертификаты X.509 для организации SA, время жизни SA должно быть ограничено периодом действия сертификатов, а также значениями NextIssueDate из списков CRL[2], использованных в обмене IKE для данной SA. В этом варианте обе стороны соединения (вызывающая и отвечающая) несут ответственность за ограничение времени жизни SA.

[1] Если предотвращение повторного использования пакетов отключено получателем для SA (например, в SA с задаваемыми вручную ключами), значение Anti-Replay Window игнорируется для этой SA. По умолчанию используются 64-битовые порядковые номера, но размер счётчика позволяет использовать и 32-битовые номера.

[2] Certificate Revocation List - список отзыва сертификатов.

Примечание. Детали обновления ключей при завершении срока жизни SA определяются локально. Ниже перечислено несколько подходящих вариантов.

(a) Если используется счётчик байтов, реализации **следует** подсчитывать число байтов, к которым применяется криптографический алгоритм IPsec. Для ESP это алгоритм шифрования (включая алгоритм Null), а для AH - алгоритм аутентификации. При подсчёте принимаются во внимание байты заполнения и т. п. Отметим, что реализации **должны** быть способны работать при потере синхронизации на концах SA (например, в результате потери пакетов или по причине использования разных механизмов по разные стороны SA).

(b) **Следует** поддерживать два типа ограничения времени жизни - мягкое, при завершении которого выдаётся предупреждение о необходимости инициировать замену SA, и жёсткое, при котором текущая SA завершается и уничтожается.

(c) Если пакет целиком не укладывается в срок жизни SA, этот пакет **следует** отбрасывать.

- Режим протокола IPsec - туннельный или транспортный. Показывает какой режим AH или ESP применяется к трафику данной SA.
- Флаг проверки фрагментов с учётом состояния. Показывает, применяется ли проверка фрагментов с учётом состояния для данной SA.
- Флаг обхода DF (T/F) - применяется к туннельным SA, когда внутренние и внешние заголовки относятся к IPv4.
- Значения DSCP - набор значений DSCP, разрешённых для пакетов через данную SA. Если значений не задано, фильтрации DSCP не происходит. Если задано одно или множество значений, они используются для выбора одной из нескольких SA, соответствующих селекторам трафика для исходящего пакета. Отметим, что фильтрация по этим значениям **не** применяется для входящих пакетов SA.
- Обход DSCP (T/F) или отображение на незащищённые значения DSCP (массив), если необходимо ограничить обход значений DSCP - применяется к туннельным SA. Это служит для отображения значений DSCP из внутренних заголовков в значения во внешних заголовках (например, для сокрытия канальной сигнализации).
- MTU для пути - любые переменные MTU и старения.
- Адреса отправителя и получателя из заголовка туннеля - оба адреса должны быть однотипными (IPv4 или IPv6). Версия определяет тип используемого заголовка. Используется только в туннельном режиме IPsec.

4.4.2.2. Соотношения между SPD, флагом PFP, пакетом и SAD

Приведённые ниже таблицы показывают для каждого селектора связи между значением SPD, флагом PFP, значением в триггерном пакете и результирующее значение в SAD. Отметим, что административный интерфейс для IPsec может использовать разные синтаксические опции для упрощения работы администратора по вводу правил. Например, хотя IKEv2 передаёт списки диапазонов, для пользователя может оказаться проще и удобней вводить адрес или префикс IP. Такой подход позволяет также снизить число ошибок.

Селектор	Запись SPD	PFP	Значение в триггерном пакете	Результирующая запись SAD
loc addr	список диапазонов	0	IP-адрес "S"	список диапазонов
	ANY	0	IP-адрес "S"	ANY
	список диапазонов	1	IP-адрес "S"	"S"
	ANY	1	IP-адрес "S"	"S"
rem addr	список диапазонов	0	IP-адрес "D"	список диапазонов
	ANY	0	IP-адрес "D"	ANY
	список диапазонов	1	IP-адрес "D"	"D"
	ANY	1	IP-адрес "D"	"D"
protocol	список протоколов[1]	0	протокол "P"	список протоколов[1]
	ANY[2]	0	протокол "P"	ANY
	OPAQUE[3]	0	протокол "P"	OPAQUE
	список протоколов[1]	0	нет	отбросить пакет
	ANY[2]	0	нет	ANY
	OPAQUE[3]	0	нет	OPAQUE
	список протоколов[4]	1	протокол "P"	"P"
	ANY[5]	1	протокол "P"	"P"
	OPAQUE[6]	1	протокол "P"	[7]
	список протоколов[4]	1	нет	отбросить пакет
	ANY[5]	1	нет	отбросить пакет
	OPAQUE[6]	1	нет	[7]

Если протокол относится к тем, что используют два порта, будут использоваться селекторы для локального и удалённого порта.

Селектор	Запись SPD	PFP	Значение в триггерном пакете	Результирующая запись SAD
loc port	список диапазонов	0	порт-источник "s"	список диапазонов
	ANY	0	порт-источник "s"	ANY
	OPAQUE	0	порт-источник "s"	OPAQUE
	список диапазонов	0	нет	отбросить пакет
	ANY	0	нет	ANY

[1] «Список протоколов» - это информация, а не способ представления этой информации в SPD, SAD или IKEv2.
[2] 0 (нуль) используется IKE в качестве значения ANY для протокола.
[3] Поле протокола на может принимать значение OPAQUE для IPv4. Эта запись применима только к IPv6.
[4] «Список протоколов» - это информация, а не способ представления этой информации в SPD, SAD или IKEv2.
[5] 0 (нуль) используется IKE в качестве значения ANY для протокола.
[6] Поле протокола на может принимать значение OPAQUE для IPv4. Эта запись применима только к IPv6.
[7] Использование PFP=1 с OPAQUE является ошибкой и его **следует** запрещать в реализации IPsec.

Селектор	Запись SPD	PFP	Значение в триггерном пакете	Результирующая запись SAD
	OPAQUE	0	нет	OPAQUE
	список диапазонов	1	порт-источник "s"	отбросить пакет
	ANY	1	порт-источник "s"	отбросить пакет
	OPAQUE	1	порт-источник "s"	[7]
	список диапазонов	1	нет	отбросить пакет
	ANY	1	нет	отбросить пакет
	OPAQUE	1	нет	[7]
rem port	список диапазонов	0	порт-получатель "d"	список диапазонов
	ANY	0	порт-получатель "d"	ANY
	OPAQUE	0	порт-получатель "d"	OPAQUE
	список диапазонов	0	нет	отбросить пакет
	ANY	0	нет	ANY
	OPAQUE	0	нет	OPAQUE
	список диапазонов	1	порт-получатель "d"	"d"
	ANY	1	порт-получатель "d"	"d"
	OPAQUE	1	порт-получатель "d"	[7]
	список диапазонов	1	нет	отбросить пакет
	ANY	1	нет	отбросить пакет
	OPAQUE	1	нет	[7]

Для протокола Mobility Header будет использоваться селектор для типа MH.

Селектор	Запись SPD	PFP	Значение в триггерном пакете	Результирующая запись SAD
mh type	список диапазонов	0	mh типа "T"	список диапазонов
	ANY	0	mh типа "T"	ANY
	OPAQUE	0	mh типа "T"	OPAQUE
	список диапазонов	0	нет	отбросить пакет
	ANY	0	нет	ANY
	OPAQUE	0	нет	OPAQUE
	список диапазонов	1	mh типа "T"	"T"
	ANY	1	mh типа "T"	"T"
	OPAQUE	1	mh типа "T"	[7]
	список диапазонов	1	нет	отбросить пакет
	ANY	1	нет	отбросить пакет
	OPAQUE	1	нет	[7]

Для протокола ICMP будет использоваться 16-битовый селектор типа и кода ICMP. Отметим, что тип и код связаны между собой, т. е., коды относятся к определённому типу. 16-битовый селектор может содержать один тип и диапазон кодов, один тип и любой код (ANY), любой (ANY) тип и любой (ANY) код.

Селектор	Запись SPD	PFP	Значение в триггерном пакете	Результирующая запись SAD
Тип и код ICMP	Один тип и диапазон кодов	0	тип "t" и код "c"	Один тип и диапазон кодов
	Один тип и код ANY	0	тип "t" и код "c"	Один тип и код ANY
	Код ANY и тип ANY	0	тип "t" и код "c"	Код ANY и тип ANY
	OPAQUE	0	тип "t" и код "c"	OPAQUE
	Один тип и диапазон кодов	0	нет	отбросить пакет
	Один тип и код ANY	0	нет	отбросить пакет
	Код ANY и тип ANY	0	нет	Код ANY и тип ANY
	OPAQUE	0	нет	OPAQUE
	Один тип и диапазон кодов	1	тип "t" и код "c"	"t" и "c"
	Один тип и код ANY	1	тип "t" и код "c"	"t" и "c"
	Код ANY и тип ANY	1	тип "t" и код "c"	"t" и "c"
	OPAQUE	1	тип "t" и код "c"	[1]
	Один тип и диапазон кодов	1	нет	отбросить пакет
	Один тип и код ANY	1	нет	отбросить пакет
	Код ANY и тип ANY	1	нет	отбросить пакет
	OPAQUE	1	нет	[1]

При использовании селектора name:

Селектор	Запись SPD	PFP	Значение в триггерном пакете	Результирующая запись SAD
name	Список пользоват. или сист. имён.	нет	нет	нет

4.4.3. База проверки полномочий партнёров (PAD)

База проверки полномочий партнёров (PAD) обеспечивает связь между SPD и протоколом управления SA (таким, как IKE). Она может реализовать несколько важных функций:

- аутентификация партнёров или групп партнёров, которые уполномочены взаимодействовать с данным объектом IPsec;
- указание протокола и метода, используемого для аутентификации каждого партнёра.;
- предоставление аутентификационных данных для каждого партнёра.;
- ограничение типов и значений идентификаторов, которые могут заявляться партнёром при создании дочерних SA, чтобы партнёр не мог заявлять для поиска в SPD аутентификационные данные, которые не разрешено представлять;
- информация о местоположении партнерского шлюза (например, адрес(а) IP или имена DNS), которая **может** быть включена для партнёров, находящихся «за защитным шлюзом».

[1]Использование PFP=1 с OPAQUE является ошибкой и его **следует** запрещать в реализации IPsec.

PAD обеспечивает эти функции для партнёра. IKE, когда тот действует в качестве инициатора или ответчика.

Для выполнения этих функций PAD содержит запись для каждого партнёра. или группы партнёров, с которыми объект IPsec будет взаимодействовать. Запись именует отдельного партнёра. (пользователя, конечную систему или защитный шлюз) или задаёт группу партнёров (используя правила соответствия идентификаторов, определённые ниже). Запись задаёт протокол аутентификации (например, IKEv1, IKEv2, KINK), используемый метод (например, сертификаты или разделяемые секреты), а также аутентификационные данные (например, разделяемый секрет или доверенную привязку, с помощью которой можно проверить сертификат партнёра.). Для аутентификации на основе сертификатов запись также может предоставлять информацию, помогающую проверить состояние отзыва сертификата для партнёра. (например, указатель на хранилище CRL, имя сервера OCSP[1], связанного с партнёром, или доверенную привязку для этого партнёра.).

Каждая запись также показывает, будет ли при создании дочерних SA использоваться элемент данных IKE ID в качестве символьного имени при просмотре SPD или для этого будет служить удалённый адрес IP из селекторов трафика.

Отметим, что информация PAD **может** использоваться для поддержки создания нескольких туннельных SA между парой партнёров (т. е., два туннеля для защиты тех же адресов/хостов, но с разными конечными точками туннелей).

4.4.3.1. Идентификаторы записей PAD и правила соответствия

PAD представляет собой упорядоченную базу данных, порядок которой определяется администратором (или пользователем в случае персональной конечной системы). Обычно один администратор отвечает за PAD и SPD, поскольку эти базы данных должны быть скоординированы. Требования по упорядочению для PAD основываются на таких же критериях, что для SPD.

Для записей в PAD поддерживается шесть типов идентификаторов, согласующихся с символьными именами и адресами IP, которые служат для идентификации записей SPD. Идентификатор для каждой записи может действовать для PAD, подобно индексу, т. е, его значение может служить для выбора записи. Все эти типы ID могут сопоставляться с типами элементов данных IKE ID. Типы идентификаторов включают:

- имя DNS (полное или частичное);
- DN[2] (полное имя или содержащее его субдерево);
- почтовый адрес RFC 822 (полный или частичный);
- адрес IPv4 (диапазон);
- адрес IPv6 (диапазон);
- Key ID (только точное соответствие).

Первые три типа могут поддерживать как полное, так и частичное (субдерево) соответствие. Имя DNS может быть FQDN[3] и соответствовать в точности одному имени (например, foo.example.com). Возможно и задание с помощью этого имени группы партнёров - например, строка «.example.com» будет соответствовать всем доменным именам домена example.com.

Аналогично, тип Distinguished Name может задавать полное имя для точного соответствия (например,CN = Stephen, O = BBN Technologies, SP = MA, C = US) или указывать группу партнёров путём задания субдерева (например, запись вида «C = US, SP = MA» может служить для выбора всех DN, содержащих эти два атрибута, как верхушку двух RDN[4].

Для почтовых адресов RFC 822 существует аналогичная возможность. Полный адрес типа foo@example.com соответствует только одному объекту, но субдерево типа @example.com будет соответствовать всем почтовым адресам домена (любой префикс слева от @).

Конкретный синтаксис соответствия части доменного имени, DN или почтового адреса RFC 822 определяется локальной реализацией. Однако **должно** поддерживаться по крайней мере соответствие субдереву, описанное выше. Соответствие подстрокам DN, имени DNS или адресе RFC 822 **возможно**, но не требуется.

Для адресов IPv4 и IPv6 **должен** поддерживаться такой же синтаксис адресных диапазонов, который используется для записей SPD. Это позволяет задавать отдельные адреса (вырожденный диапазон), префиксы (путём выбора диапазонов, соответствующих префиксам в стиле CIDR[5]) или произвольные диапазоны адресов.

Поле Key ID определено в IKE, как строка **октетов**. Для этого типа имён. **должно** поддерживаться только точное соответствие, поскольку этот тип идентификатора не имеет явной структуры. Для этого типа **могут** поддерживаться дополнительные функции соответствия.

4.4.3.2. Аутентификационные данные партнёра. IKE

После того, как запись найдена при упорядоченном поиске в PAD на основе соответствия поля ID, необходимо проверить предложенные аутентификационные данные (т. е., аутентифицировать предложенный ID). Для каждой записи PAD имеется индикация типа выполняемой аутентификации. Этот документ требует поддерживать два обязательных типа аутентификационных данных:

- сертификаты X.509;
- разделяемые (pre-shared) секреты.

---

[1]Online Certificate Status Protocol - протокол проверки состояния сертификатов.
[2]Distinguished Name.
[3]Full Qualified Domain Name - полное доменное имя. *Прим. перев.*
[4]Relative Distinguished Name.
[5]Classless Inter-Domain Routing - бесклассовая междоменная маршрутизация. Схема задания блоков адресов IP без использования классов адресов. *Прим. перев.*

Для аутентификации на основе сертификата X.509 запись PAD содержит доверенную привязку, посредством которой может быть проверен (напрямую или через путь сертификата) сертификат конечного элемента (EE[1]) для партнёра., которого требуется проверить. Определение доверенной привязки дано в RFC. Запись, используемая с аутентификацией на базе сертификата, **может** включать дополнительные данные для упрощения проверки состояния отзыва сертификата (например, список подходящих ответчиков OCSP или хранилищ CRL) и связанные аутентификационные данные. Для аутентификации на основе разделяемого ключа PAD содержит pre-shared-секрет, который будет использоваться IKE.

Этот документ не требует от элементов IKE ID, представляемых партнёром, синтаксической связи с конкретным полем конечного элемента сертификата, которые предназначен для аутентификации данного партнёра. Однако зачастую такое требование является целесообразным (например, когда одна запись представляет набор партнёров, каждый из которых может иметь свою запись SPD). Таким образом, реализация **должна** обеспечивать администратору возможность потребовать соответствия между представленным IKE ID и именем субъекта или его дополнительным именем (alt name) в сертификате. Первый вариант применим к IKE ID выражаемым через DN, а второй подходит для выражения через имена DNS, почтовые адреса RFC 822 и адреса IP. Поскольку значение KEY ID предназначено для аутентификации партнёров с использованием разделяемого секрета, не возникает требования соответствия между идентификаторами этого типа и полями сертификатов.

Подробное описание аутентификации партнёров в IKE с использованием сертификатов и разделяемых секретов приведено в документах IKEv1 [HarCar98] и IKEv2 [Kau05].

Этот документ не требует поддержки других методов аутентификации, хотя они **могут** быть развёрнуты.

4.4.3.3. Аутентификационные данные дочерних SA

После аутентификации партнёра. IKE можно создавать дочерние SA. Каждая запись PAD содержит данные для ограничения набора идентификаторов, которые могут представляться партнёром IKE для поиска соответствия в SPD. Каждая запись PAD показывает, какие IKE ID будут использоваться в качестве символических имён. для поиска в SPD или какие адреса IP из селекторов трафика будут использоваться.

Если запись показывает использование IKE ID, поле ID записи PAD определяет уполномоченный набор ID. Если запись показывает, что будут использоваться селекторы трафика дочерней SA, требуется дополнительный элемент данных в форме диапазона адресов IPv4 и/или IPv6 (партнёр может быть уполномочен на использование обоих типов адресов, следовательно **должны** указываться диапазоны для обоих типов).

4.4.3.4. Использование PAD

При начальном обмене IKE инициатор и ответчик должны представить свои аутентификационные данные в элементах IKE ID и передать элементы данных AUTH для проверки представленной информации. Может передаваться один или несколько элементов CERT для упрощения проверки представленных аутентификационных данных.

Когда объект IKE получает элемент данных IKE ID, он использует представленный ID для поиска записи в PAD с использованием описанных выше правил соответствия. Запись PAD задаёт метод аутентификации, используемый для партнёра. Это обеспечивает выбор правильного метода для каждого партнёра. и применение различных методов для разных партнёров. Запись также задаёт аутентификационные данные, которые будут использоваться при проверке представленной информации. Эти данные вместе с указанным методом применяются для того, чтобы аутентифицировать партнёра до создания какой-либо дочерней CHILD SA.

Дочерние SA создаются на базе обмена селекторами трафика в конце начального обмена IKE или в последующих обменах CREATE_CHILD_SA. Запись PAD для (уже аутентифицированного) партнёра. IKE используется для ограничения создания дочерних SA. В частности, запись PAD задаёт, способ поиска в SPD с использованием селектора трафика от партнёра. Здесь имеется два варианта - либо используется значение IKE ID, представленное партнёром, для поиска записи SPD по символьному имени, либо используется IP-адрес партнёра., представленный в селекторе трафика, для поиска в SPD по хранящейся там части поля с удаленным адресом IP. Требуется вносить некоторые ограничения на создание дочерних SA, чтобы обезопасить аутентифицированного партнёра. от обманных ID, связанных с другими легитимными партнёрами.

Отметим, что в результате проверки PAD до поиска записи SPD обеспечивается прочная защита инициатора от атак с использованием обманных пакетов (spoofing). Предположим в качестве примера, что IKE A получает исходящий пакет, направленный по IP-адресу X (хост, обслуживаемый защитным шлюзом). RFC 2401 [RFC2401] и данный документ не задают, как A определяет адрес партнёра. IKE, обслуживающего X. Однако любой партнёр, с которым контактирует A, как с возможным представителем X, должен быть зарегистрирован в PAD для того, чтобы позволить аутентификацию обмена IKE. Более того, когда аутентифицированный партнёр заявляет, что он представляет X в своём обмене селекторами трафика, будет запрашиваться PAD для проверки полномочий этого партнёра. в части представления X. Таким образом, PAD обеспечивает связывание диапазонов адресов (или подпространств имён.) с партнёрами для противодействия таким атакам.

4.5. Управление SA и ключами

Все реализации IPsec **должны** поддерживать ручное и автоматическое управление SA и криптографическими ключами. Протоколы IPsec (AH и ESP) в значительной степени независимы от методов управления SA, хотя эти методы оказывают влияние на некоторые защитные службы, предлагаемые протоколами. Например, необязательный сервис предотвращения повторного использования пакетов, доступный для AH и ESP, требует автоматического управления SA. Более того, гранулярность распространения ключей в IPsec определяет гранулярность обеспечиваемой аутентификации. В общем случае аутентификация источника данных в AH и ESP ограничивается сферой распространения секретов, используемый с механизмами защиты целостности (или с протоколом управления ключами, используемым для генерации таких секретов).

Далее описываются минимальные требования для обоих типов управления SA.

[1]The end entity.

4.5.1. Управление SA вручную

Простейшим способом управления является ручное управление, при котором человек вручную настраивает каждую систему, используя ключевой материал и данные управления SA, обеспечивающие защищённую связь с другими системами. Такое решение практично в небольших, статичных средах, но даже в таких случаях оно недостаточно хорошо масштабируется. Например, компания может создать виртуальную частную сеть (VPN[1]), используя IPsec на защитных шлюзах в нескольких сайтах. Если число сайтов невелико и все эти сайты находятся в одном административном домене, в таком контексте ручное управление может оказаться вполне приемлемым. В этом случае защитные шлюзы смогут селективно защищать трафик между некоторыми сайтами с использованием задаваемых вручную ключей, а остальной трафик останется незащищённым. Такое решение подойдёт для тех случаев, когда защита требуется лишь для части трафика. Подобные аргументы применимы и к развёртыванию IPsec в масштабе всей организации с небольшим числом хостов и/или шлюзов. Системы ручного управления зачастую настраивают статически с использованием симметричных ключей, хотя возможны и другие варианты.

4.5.2. Автоматизированное управление SA и ключами

Широкое распространение и использование IPsec требует стандарта Internet для масштабируемого, автоматизированного протокола управления SA. Такой протокол нужен для облегчения использования возможностей предотвращения повторов в AH и ESP, а также для создания SA по запросам (например, для ориентированных на сеансы и пользователей ключей). Отметим, что термин «смена ключей» (rekeying) SA на деле означает создание новой SA с новым SPI - этот процесс в общем случае предполагает использование протокола автоматизированного управления SA и ключами.

По умолчанию протоколом автоматизированного управления ключами в IPsec является IKEv2 [Kau05]. В этом документе предполагается доступность некоторых функций протокола управления ключами, которые не поддерживаются в IKEv1. **Могут** развёртываться и другие протоколы автоматизированного управления ключами.

При реализации протокола автоматизированного управления SA и ключами выходные данные этого протокола используются для генерации множества ключей для одной SA. SA создаются IKE. Если развёрнуты. защита целостности и конфиденциальности, требуется не менее четырёх ключей. В дополнение к этому некоторые криптографические алгоритмы (например, 3DES) также могут потребовать множества ключей.

Система управления ключами может обеспечивать раздельные битовые строки для каждого ключа или генерировать одну битовую строку, из которой выделяются все ключи. При использовании одной строки следует принять меры по обеспечению одинакового отображения битов строки на требуемые ключи по обе стороны SA. Чтобы гарантировать использование реализациями IPsec на разных сторонах SA использование одинаковых битов для одного ключа, независимо от того, какая часть системы делит строку битов на отдельные ключи, ключи шифрования **должны** браться первыми из левой части строки битов (старшие биты строки), а ключи для защиты целостности должны браться из оставшейся части. Число битов для каждого ключа определяется в RFC со спецификацией соответствующего криптографического алгоритма. При использовании множества ключей для шифрования и множества ключей для защиты целостности спецификация криптографического алгоритма должна задавать порядок выборки ключей из одной битовой строки, представленной криптографическому алгоритму.

4.5.3. Нахождение защитного шлюза

В этом параграфе рассматривается, как хост узнает о существовании интересующих его защитных шлюзов и как проверяет корректность найденного защитного шлюза. Детали хранения требуемой информации определяются локально, но база аутентификации партнёров (PAD), описанная в параграфе 4.4, является наиболее предпочтительным вариантом.

Далее системы, на которых используется IPsec, обозначаются S* (например, SH1 и SG2 ниже).

Рассмотрим ситуацию, когда удалённый хост (SH1) использует Internet для получения доступа к серверу или другой машине (H2) и имеется защитный шлюз (SG2), через который должен проходить трафик SH1 (например, межсетевой экран). Примером такой ситуации может служить мобильный хост, подключающийся через Internet к межсетевому экрану своей домашней сети (SG2). Такая ситуация порождает несколько вопросов:

1. Как SH1 узнает о существовании защитного шлюза SG2?
2. Как он аутентифицирует SG2 и как после аутентификации SG2 он сможет узнать о том, что SG2 уполномочен представлять H2?
3. Как SG2 аутентифицирует SH1 и проверяет, что SH1 имеет полномочия для доступа к H2?
4. Как SH1 может узнать о дополнительных шлюзах, которые обеспечивают альтернативный доступ к H2?

Для решения этих вопросов поддерживающий IPsec хост или шлюз **должен** иметь административный интерфейс, который позволяет пользователю/администратору настраивать адрес одного или множества защитных шлюзов для диапазонов адресов, требующих использования таких шлюзов. Сюда включается способность настроить информацию для поиска и аутентификации одного или множества защитных шлюзов и проверку полномочий этих шлюзов в плане представления целевого хоста (функции проверки полномочий реализуются в PAD). В этом документе не решается вопрос автоматизации поиска и верификации защитных шлюзов.

4.6. SA и групповая адресация

Ориентированная на получателя SA предполагает, что в случае индивидуального трафика система-адресат будет выбирать значение SPI. С учётом выбора значения SPI адресатом не возникает конфликтов между SA с ручной и автоматической (например, с помощью протокола управления ключами) настройкой или между SA из разных источников. Для группового трафика множество адресатов связывается с одной SA. Поэтому некоторым системам или лицам может потребоваться координация множества multicast-групп, чтобы выбрать SPI для каждой группы и после этого обмениваться групповыми данными IPsec со всеми членами группы с использованием механизмов, не определяемых здесь.

[1]Virtual private network.

Множеству отправителей в multicast-группу **следует** использовать одну SA (и, следовательно, один SPI) для всего трафика в данную группу, когда развернут алгоритм шифрования или защиты целостности с симметричными ключами. В таких обстоятельствах получатель знает лишь, что сообщение пришло от системы, владеющей ключом для данной группы. В этом случае получатель обычно не способен аутентифицировать систему, передающую групповой трафик. Спецификации более сложных моделей использования групповой адресации разрабатываются рабочей группой IETF Multicast Security.

5. Обработка трафика IP

Как отмечено в параграфе 4.4.1. База правил защиты (SPD), **необходимо** обращаться к SPD (или кэшированным данным) до начала обработки любого трафика, пересекающего границу защиты IPsec, включая управляющий трафик IPsec. Если в SPD не найдено правила, которому соответствует пакет (входящий или исходящий), пакет **должен** отбрасываться. Для упрощения обработки и ускорения поиска SA (для SG/BITS/BITW) в этом документе вводится понятие кэша SPD для всего исходящего трафика (SPD-O плюс SPD-S) и кэша для входящего трафика без защиты IPsec[1] (SPD-I). Номинально существует один кэш на SPD. Для целей данной спецификации предполагается, что каждая кэшированная запись будет отображать единственную SA. Отметим, однако, что возникают исключения, когда используется множество SA для передачи трафика с различными приоритетами (например, заданными разными значениями DSCP) но одинаковыми селекторами. Отметим также, что существуют ситуации, когда SAD может иметь записи для SA, у которых нет соответствующих записей в SPD. Поскольку этот документ не требует селективной очистки SAD при изменении SPD, записи SAD могут сохраняться, хотя создавшие их записи SPD изменены или удалены. Также при создании SA с заданием ключей вручную могут существовать записи SAD для SA, не имеющих записей в SPD.

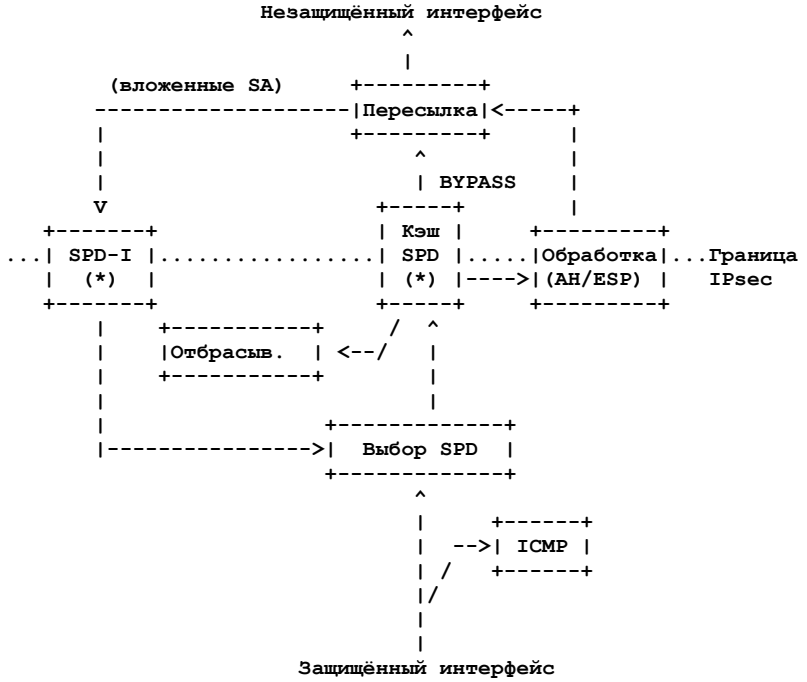
Поскольку записи SPD могут перекрываться, в общем случае кэширование таких записей не вполне безопасно. Простое кэширование может приводить к соответствию кэшированной записи, тогда как упорядоченный поиск в SPD может показывать соответствие другой записи. Если записи SPD сначала декоррелируются, результаты такой операции можно кэшировать без опаски. Каждая кэшированная запись будет показывать, что соответствующий ей трафик следует передать в обход или отбросить[2]. Если явно не указано иное, ниже все упоминания «SPD», «кэша SPD» или «кэша» относятся к декоррелированной SPD (SPD-I, SPD-O, SPD-S) или кэшу SPD, содержащему записи из декоррелированной SPD.

Примечание. В реализации хоста IPsec на основе сокетов, обращение к SPD будет происходить при каждом создании сокета для определения будет ли применяться (и какая) обработка IPsec к потоку трафика через этот сокет. Это обеспечивает механизм неявного кэширования и предшествующее обсуждение вопросов кэширования для таких реализаций можно игнорировать.

Примечание. Предполагается, что начальное состояние является коррелированным, поскольку это состояние зависит от того, как пользователи и администраторы привыкли управлять списками доступа и правилами фильтрации межсетевых экранов. После этого применяется механизм декорреляции, чтобы создать пригодные для кэширования записи SPD. Декорреляция не видна для интерфейса управления.

Для входящего трафика IPsec запись SAD, выбранная SPI, служит в качестве кэша для проверки селекторов прибывающих пакетов IPsec, после чего выполняется обработка AH или ESP.

5.1. Обработка исходящего трафика IP[3]



(*) На рисунке показан кэш SPD. При отсутствии этого кэша проверяется непосредственно SPD. Реализация не обязаны буферизовать пакет при отсутствии кэша.

Рисунок 2. Модель обработки исходящего трафика.

[1]Как было отмечено выше, SAD действует как кэш для проверки селекторов входящего трафика с защитой IPsec, поступающего в SA
[2]Исходная запись SPD может давать в результате множество SA (например, по причине наличия флага PFP).
[3]С защищённой стороны на незащищённую.

Сначала рассмотрим путь трафика, входящего в реализацию IPsec через защищённый интерфейс и выходящего через незащищённый.

Реализация IPsec **должна** выполнять следующие действия при обработке исходящих пакетов.

1. Когда пакет приходит с абонентского (защищённого) интерфейса, вызывается функция выбора SPD для получения SPD-ID, нужного при выборе подходящей SPD (если используется только одна SPD, этот этап становится пустым - no-op).

2. Заголовки пакета сравниваются с кэшем для SPD, заданной SPD-ID на этапе 1. Отметим, что этот кэш содержит записи из SPD-O и SPD-S.

3. a. При совпадении пакет обрабатывается, как задано соответствующей записью (т. е., BYPASS, DISCARD или PROTECT с использованием AH или ESP). Если обработка IPsec выполнена, имеется связь между кэш-записью SPD и соответствующей записью SAD (задаёт режим, криптографические алгоритмы, ключи, SPI, PMTU и пр.). Обработка IPsec определяется ранее для туннельного или транспортного режима и протокола AH или ESP, как задано в соответствующих RFC [Ken05b, Ken05a]. Отметим, что значение SA PMTU вкупе со значением флага проверки фрагментов с учётом состояния (а также бита DF в заголовке IP исходящего пакета) определяют возможность (необходимость) фрагментирования пакета до обработки IPsec или его отбрасывания с передачей сообщения ICMP PMTU.

   b. Если в кэше не найдено соответствия, осуществляется поиск в SPD (части SPD-S и SPD-O), заданной SPD-ID. Если запись SPD задаёт BYPASS или DISCARD, создаётся одна или несколько новых исходящих кэш-записей SPD, а для BYPASS создаётся одна или несколько входящих кэш-записей SPD. Множество записей может создаваться в результате того, что декоррелированная запись SPD может быть связана с другими такими же записями (побочный эффект процесса декорреляции). Если запись SPD задаёт PROTECT (т. е., создание новой SA), вызывается механизм управления ключами (например, IKEv2) для создания SA. При успешном создании SA создаётся новая исходящая (SPD-S) кэш-запись вместе с входящей и исходящей записью SAD. В противном случае пакет отбрасывается (пакет, вызвавший просмотр SPD, **может** быть отброшен реализацией или обработан в соответствии с вновь созданной кэш-записью). Поскольку SA создаются попарно, создаётся также запись SAD для соответствующей входящей SA, содержащая значения селекторов, определённые из записи SPD (и пакета, если установлен флаг PFP), использованной для создания входящей SA. Эта запись служит для проверки трафика, входящего через SA.

4. Пакет передаётся выходной функции пересылки (за пределы реализации IPsec) для выбора интерфейса, через который следует передать пакет. Эта функция может вернуть пакет через границу IPsec для дополнительной обработки IPsec (например, при поддержке вложенных SA). В таких случаях **должна** присутствовать дополнительная запись в базе SPD-I, которая разрешает такой обход, поскольку в противном случае пакет будет отброшен. При необходимости (т. е., при наличии множества SPD-I) завёрнутый назад трафик **может** быть помечен, как исходящий с внутреннего интерфейса. Это позволяет при необходимости использовать разные SPD-I для действительно внешнего трафика и «завёрнутого назад» трафика.

Примечание. За исключением транспортного режима IPv4 и IPv6, реализации SG, BITS или BITW **могут** фрагментировать пакеты до выполнения функций IPsec[1]. Устройству **следует** иметь конфигурационные параметры для запрета такого фрагментирования. Фрагменты обычным способом сравниваются с SPD. Таким образом фрагменты без номеров портов (сообщения ICMP без типа и кода или Mobility Header без типа) будут соответствовать лишь правилам, в которых селекторы для порта (типа и кода ICMP или типа MH) имеют значение OPAQUE или ANY (см. раздел 7).

Примечание. В части определения и реализации PMTU для SA системы IPsec **должны** следовать действиям, описанным в параграфе 8.2.

5.1.1. Обработка исходящих пакетов, которые должны быть отброшены

Если система IPsec получает исходящие пакеты, которые она считает нужным отбросить, ей **следует** поддерживать возможность генерации и передачи сообщения ICMP для индикации отправителю исходящего пакета информации об отбрасывании этого пакета. Причину отбрасывания **следует** записать в журнал аудита. В запись аудита **следует** включать причину отбрасывания, текущие значения даты и времени, а также значения селекторов из пакета.

a. Селекторы в пакете соответствуют записи SPD, требующей отбрасывания пакета.

   IPv4 Type = 3 (адресат недоступен) Code = 13 (связь запрещена административно)

   IPv6 Type = 1 (адресат недоступен) Code = 1 (связь с адресатом запрещена административно)

b1. Для системы IPsec удалённый партнёр доступен, но с ним не удаётся согласовать SA, требуемую запись SPD, которой соответствует пакет (например, удалённому партнёру административно запрещена связь с инициатором, инициатор не может аутентифицировать себя удалённому партнёру или SPD на удалённой стороне не имеет подходящей записи).

   IPv4 Type = 3 (адресат недоступен) Code = 13 (связь запрещена административно)

   IPv6 Type = 1 (адресат недоступен) Code = 1 (связь с адресатом запрещена административно)

b2. Система IPsec не способна организовать SA, требуемую запись SPD, которой соответствует пакет, поскольку не удалось связаться с партнёром IPsec на другой стороне.

   IPv4 Type = 3 (адресат недоступен) Code = 1 (хост недоступен)

   IPv6 Type = 1 (адресат недоступен) Code = 3 (адрес недоступен)

Отметим, что атакующий за защитным шлюзом может передавать пакеты с обманным адресом отправителя W.X.Y.Z объекту IPsec, заставляя того передавать сообщения ICMP по адресу W.X.Y.Z. Это открывает возможность организации атак на службы (DoS[2]) хостов, находящихся за защитным шлюзом. Для устранения такой возможности в

[1]Это применимо только к IPv4. Для пакетов IPv6 фрагментация разрешена только их исходному источнику.
[2]Denial of service - отказ в обслуживании.

защитные шлюзы **следует** включать средства управления, позволяющие администратору включать или отключать для реализации IPsec передачу сообщений ICMP в таких обстоятельствах и при включённой передаче сообщений ICMP ограничивать скорость такой передачи.

5.1.2. Создание заголовка для туннельного режима

В этом параграфе описывается обработка внутренних и внешних заголовков IP, расширений заголовков, а также опций туннелей AH и ESP для исходящего трафика. Обсуждается создание инкапсулирующего (внешнего) заголовка IP, обработка полей внутреннего заголовка и другие операции, которые следует применять к исходящему трафику в туннельном режиме. Обработка, описанная здесь, основана на модели RFC 2003, «Инкапсуляция IP в IP» [Per96]:

- Поля Source Address и Destination Address внешнего заголовка IP идентифицируют «конечные точки» туннеля (инкапсулятор и декапсулятор). Эти же поля внутреннего заголовка идентифицируют исходного отправителя и конечного получателя дейтаграмм (с точки зрения данного туннеля), соответственно (см. примечание 3 к таблице в параграфе 5.1.2.1 с комментариями по поводу инкапсуляции адреса отправителя).
- Внутренний заголовок IP не меняется за исключением поля TTL (или Hop Limit) и полей DS/ECN. В процессе доставки пакета к выходной точке туннеля внутренний заголовок сохраняется неизменным.
- Опции IP и расширения для внутреннего заголовка не меняются в процессе доставки инкапсулированной дейтаграммы через туннель.

Туннелирование IPsec имеет некоторые отличия от туннелирования IP-in-IP (RFC 2003 [Per96]):

- IPsec поддерживает средства, позволяющие администратору управлять скрытыми каналами (которые обычно не туннелируются) и обеспечить проверку получателем полученных пакетов в плане контроля доступа. Реализация IPsec **может** быть настраиваемой в плане обработки внешнего поля DS при передаче пакетов в туннельном режиме. Для исходящего трафика одна конфигурационная установка для внешнего поля DS будет действовать, как описано ниже, на обработку заголовков IPv4 и IPv6 для туннелей IPsec. Другая будет позволять отображение внешнего поля DS на фиксированное значение, которое **может** быть настраиваемым на уровне SA[1]. Эти конфигурационные опции позволяют локальному администратору решить, будет ли скрытый канал, обеспечиваемый копированием этих битов, перевешивать преимущества копирования.
- IPsec описывает обработку ECN и DS, а также обеспечивает возможность контроля за распространением изменений этих полей между защищёнными и открытыми доменами. В общем случае распространение из защищённой области в открытую представляет собой скрытый канал и для полосы этого канала поддерживаются средства управления. Распространение значений ECN в другом направлении контролируется так, что передаются только легитимные изменения ECN (показывающие насыщение между конечными точками туннеля). По умолчанию распространение DS из защищённой области в открытую запрещено. Однако, если отправитель и получатель не используют единое пространство кодов DS и получатель не может узнать способ отображения одного пространства на другое, распространение может быть разрешено. В частности, реализации IPsec **могут** настраиваться в части обработки внешнего поля DS для принятых пакетов (в туннельном режиме). Они могут быть настроены на отбрасывание внешнего значения DS (по умолчанию) **или** переписывание значения внутреннего поля DS в соответствии со значением внешнего. Выбор отбрасывания или изменения DS может быть задан на уровне SA. Эта опция позволяет локальному администратору самому выбрать между возникновением уязвимости при копировании поля DS и обеспечиваемыми таким копированием преимуществами. Описание каждого из этих вариантов дано в [RFC2983] вместе с описанием кондиционирования трафика до или после обработки IPsec (включая декапсуляцию).
- IPsec позволяет использовать разные версии протокола IP во внутреннем и внешнем заголовке.

Приведённые ниже таблицы описывают обработку полей заголовка и опций (термин «создаётся» означает, что значение поля во внешнем заголовке задаётся независимо от значения во внутреннем заголовке).

5.1.2.1. IPv4: создание заголовка для туннельного режима

	<-- Связь внешнего заголовка с внутренним -->	
Поля заголовка IPv4	**Внешний заголовок на инкапсуляторе**	**Внутренний заголовок на декапсуляторе**
Version	4 (1)	без изменения
Header length	создаётся	без изменения
DS	копируется из внутреннего заголовка (5)	без изменения
ECN	копируется из внутреннего заголовка	создаётся (6)
Total length	создаётся	без изменения
ID	создаётся	без изменения
Flags (DF, MF)	создаётся, DF (4)	без изменения
Fragment offset	создаётся	без изменения
TTL	создаётся (2)	уменьшается на 1 (2)
Protocol	AH, ESP	без изменения
Checksum	создаётся	создаётся (2)(6)
Src address	создаётся (3)	без изменения
Dest address	создаётся (3)	без изменения
Options	никогда не копируются	без изменения

Примечания:

(1) Версия IP в инкапсулирующем заголовке может отличаться от версии протокола во внутреннем заголовке.

(2) Значение TTL декрементируется инкапсулятором перед пересылкой пакета и декапсулятором - при пересылке[2] (контрольная сумма заголовка IPv4 меняется при изменении TTL).

[1]Это значение на практике может быть фиксированным для всего трафика, исходящего от устройства, и настройка на уровне SA позволяет сделать это.

[2]Декрементирование TTL является частью процесса пересылки. Для пакета, созданного декапсулятором, значение TTL не уменьшается, поскольку такая передача не является пересылкой. Это относится к BITS и естественным реализациям IPsec на хостах

(3) Локальные и удалённые адреса зависят от SA, которая служит для определения удалённого адреса, а тот, в свою очередь, определяет локальный адрес (сетевой интерфейс), используемый для пересылки пакета[1].

(4) Для поля DS копирование из внутреннего заголовка (IPv4), сброс или создание определяется настройкой.

(5) Если пакет будет непосредственно попадать в домен, где значение DSCP во внешнем заголовке неприемлемо, значение **должно** быть отображено на приемлемое для домена [NiBlBaBL98] (см. RFC 2475 [BBCDWW98]).

(6) Если поле ECN внутреннего заголовка имеет значение ECT(0)[2] или ECT(1) и внешнее поле ECN включает флаг CE (перегрузка), в поле ECN внутреннего заголовка устанавливается флаг CE; в остальных случаях значение внутреннего поля ECN не меняется (при изменении ECN меняется контрольная сумма заголовка IPv4).

Примечание. IPsec не копирует опции из внутреннего заголовка во внешний и не создаёт опций для внешнего заголовка. Однако после IPsec опции внешнего заголовка **могут** добавляться и изменяться.

5.1.2.2. IPv6: создание заголовка для туннельного режима

	<-- Связь внешнего заголовка с внутренним -->	
Поля заголовка IPv6	**Внешний заголовок на инкапсуляторе**	**Внутренний заголовок на декапсуляторе**
Version	6 (1)	без изменения
DS	копируется из внутреннего заголовка (5)	без изменения (9)
ECN	копируется из внутреннего заголовка	создаётся (6)
Flow label	копируется или настраивается (8)	без изменения
Payload length	создаётся	без изменения
Next header	AH, ESP, routing hdr	без изменения
Hop limit	создаётся (2)	уменьшается на 1 (2)
Src address	создаётся (3)	без изменения
Dest address	создаётся (3)	без изменения
Extension headers	никогда не копируется (7)	без изменения

Примечания:

(1) - (3), (5), (6) см. параграф 5.1.2.1.

(7) IPsec не копирует расширения заголовка из внутреннего пакета во внешний и не создаёт расширений во внешнем заголовке. Однако после IPsec **возможна** вставка и создание расширений для внешнего заголовка.

(8) См. [RaCoCaDe04]. Копирование допустимо только для конечных систем, но не для защитных шлюзов. Если шлюз копирует метки защиты из внутреннего заголовка во внешний, могут возникать конфликты.

(9) Реализация **может** выбрать способ передачи значения DS из внешнего заголовка во внутренний на уровне SA для принятых в туннельном режиме пакетов. Мотивацией для такого решения служит приспособление к ситуациям, когда пространство кодов DS на приёмной стороне отличается от пространства кодов передающей стороны и нет информации о способе трансляции из пространства отправителя. Копирование значения DS из внешнего заголовка во внутренний представляет опасность, поскольку у атакующего появляется возможность влияния на трафик за счёт изменения значений DSCP во внешнем заголовке. Следовательно, по умолчанию реализации IPsec **не** разрешают такое копирование.

5.2. Обработка входящего трафика IP[3]

Обработка входящего трафика несколько отличается от обработки исходящего, по причине использования SPI для отображения трафика IPsec на SA. Входной кэш SPD (SPD-I) применяется только для передаваемого в обход или отбрасываемого трафика. Если прибывающий с незащищённого интерфейса пакет выглядит, как фрагмент IPsec, сборка осуществляется до выполнения обработки IPsec. Смысл каждого кэша SPD заключается в том, что пакет, не соответствующий ни одной записи в кэше, относится к соответствующей SPD. Каждой SPD **следует** иметь номинальную, завершающую запись, которая «захватывает» все, что не соответствует другим записям и отбрасывает пакеты. Это обеспечивает отбрасывание всех входящих пакетов, которые не защищены IPsec и не соответствуют ни одной записи SPD-I.

До обработки AH или ESP все фрагменты IP, приходящие через незащищённый интерфейс собираются (средствами IP). Каждая входящая дейтаграмма IP, к которой будет применяться обработка IPsec, аутентифицируется наличием значений AH или ESP в поле IP Next Protocol (или AH/ESP в качестве протокола следующего уровня в контексте IPv6).

IPsec **необходимо** выполнить следующие действия:

1. Прибывающий пакет может быть помечен идентификатором интерфейса (физического или логического), через который пакет был принят, если это требуется для поддержки множества SPD и связанных с ними кэшей SPD-I (идентификатор интерфейса отображается на соответствующий SPD-ID).

2. Пакет проверяется и демультиплексируется в одну из двух категорий:

   – Если пакет представляется защищённым с помощью IPsec и адресован данному устройству, предпринимается попытка отобразить этот пакет на активную SA через SAD. Отметим, что устройство может иметь множество адресов IP, которые могут служить для поиска в SAD (например, в случае использования протоколов типа SCTP).

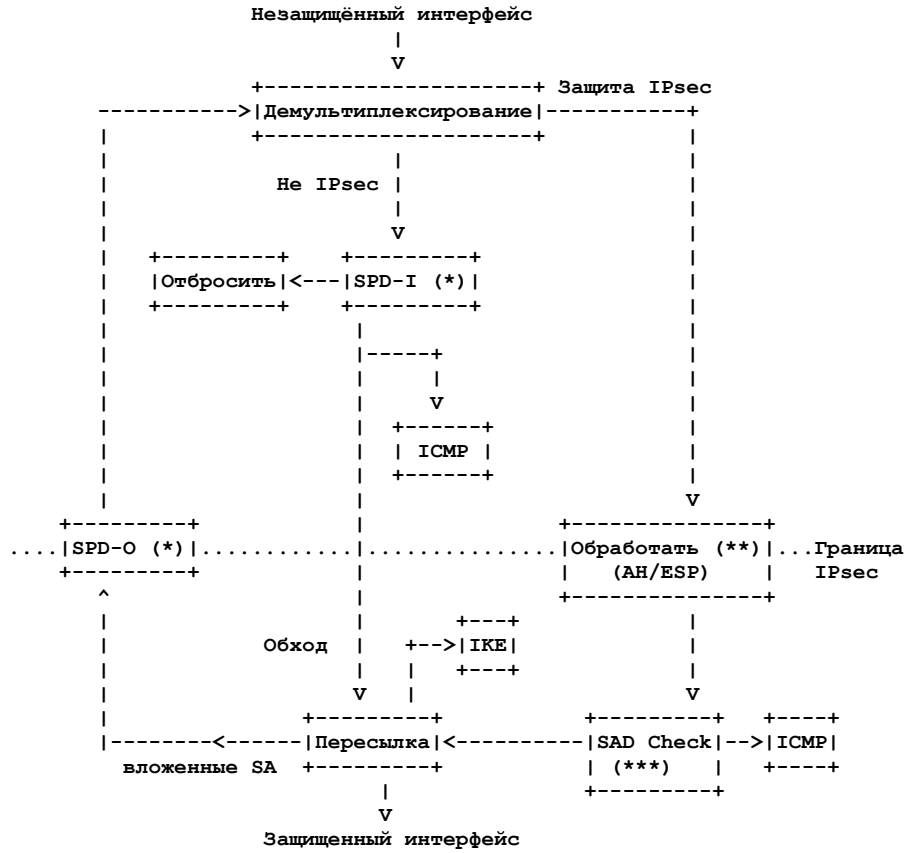
и маршрутизаторах. Однако модель обработки IPsec включает возможности внешней пересылки. Обработка TTL может использоваться для предотвращения маршрутных петель (например, в результате конфигурационных ошибок) вне контекста этой модели обработки

[1]Для группового трафика может потребоваться демультиплексирование адреса получателя или адресов получателя и отправителя. В этом случае важно обеспечить согласованность в течение всего срока жизни SA (гарантия совпадения адреса отправителя в инкапсулирующем заголовке с адресом, согласованным при создании SA). Из этого правила есть исключение - мобильные реализации IPsec будут менять адрес отправителя при перемещениях.

[2]ECN-Capable Transport - транспорт, поддерживающий ECN.

[3]С незащищённой стороны на защищённую.

– Трафик не адресованный данному устройству или адресованный устройству, но не относящийся к AH или ESP, направляется на просмотр SPD-I (это означает, что для трафик IKE в SPD **должна** присутствовать явная запись BYPASS). При наличии множества SPD для выбора SPD-I (и кэша) используется присвоенный на этапе 1 тег. Поиск в SPD-I определяет для пакета действие DISCARD (отбросить) или BYPASS (передать в обход IPsec).



(*) = На рисунке показаны кэшированные записи. Если кэш отсутствует, проверяется SPD. Требования буферизации пакета при отсутствии кэша не предъявляется к реализациям.

(**) = Эта обработка включает использование SPI и других параметров пакета для поиска SA в SAD, которая формирует кэш SPD для входящих пакетов (за исключением случаев, отмеченных в параграфах 4.4.2 и 5). См. п. 3a ниже.

(***) = Эта проверка SAD выполняется на этапе 4 (см. ниже).

Рисунок 3. Модель обработки входящего трафика.

3. a. Если пакет адресован устройству IPsec и в качестве протокола указан AH или ESP, выполняется поиск в SAD. Для индивидуального трафика используется только SPI (или SPI и протокол). Для группового трафика используется SPI в комбинации с адресом получателя или адресами отправителя и получателя, как указано в параграфе 4.1. В любом случае (групповой или индивидуальный трафик) при отсутствии соответствия пакеты отбрасываются с внесением записи в журнал аудита. В запись **следует** включать текущую дату и время, SPI, отправителя и получателя пакета, протокол IPsec и все прочие доступные из пакета селекторы. Если пакет найден в SAD, выполняется п. 4.

   b. Если пакет не адресован устройству или не относится к AH или ESP, просматривается соответствующий кэш SPD-I. Если соответствие найдено и пакет следует отбросить или передать в обход, выполняется нужная операция. Если в кэше не обнаружено соответствия, просматривается подходящая SPD-I и создаётся запись в кэше (не создаётся SA в ответ на получение пакета, требующего защиты IPsec; таким путём в кэше могут создаваться только записи BYPASS или DISCARD). Если соответствия не найдено, трафик отбрасывается с внесением записи в журнал аудита. В запись **следует** включать текущую дату и время, SPI, отправителя и получателя пакета, протокол IPsec и все прочие доступные из пакета селекторы.

   c. Предполагается, что обработка сообщений ICMP выполняется на незащищённой стороне границы IPsec. Незащищённые сообщения ICMP проверяются и к ним применяется локальная политика для определения дальнейших действий (принять или отбросить), а также действий при восприятии сообщения. Например, при получении сообщения ICMP unreachable реализация должна решить, что делать - отбросить сообщение или обработать его с учётом ограничений (см. раздел 6).

4. Выполняется обработка AH или ESP с использованием записи SAD, выбранной на этапе 3a. Затем проверяется соответствие пакета входным селекторам, определяемым записью SAD, чтобы убедиться в принадлежности пакета SA, через которую он был получен.

5. Если реализация IPsec получает пакет через SA и поля в заголовке пакета не согласуются с селекторами для SA, пакет **должен** отбрасываться с записью в журнал аудита. В запись **следует** включать текущую дату и время, SPI, отправителя и получателя пакета, протокол IPsec и все прочие доступные из пакета селекторы, а также значение селекторов для соответствующей записи SAD. Системе **следует** также поддерживать возможность генерации и передачи отправителю (партнёру IPsec) уведомления IKE INVALID_SELECTORS, показывающего, что принятый пакет был отброшен в результате несоответствия при проверке селекторов.

Для минимизации влияния DoS-атак и некорректно настроенных партнёров системам IPsec **следует** включать средства управления, позволяющие администратору настроить для реализации IPsec режим уведомлений IKE (передавать или не передавать) и ограничение частоты передачи таких уведомлений.

После того, как трафик передан в обход или обработан IPsec, он передаётся входной функции пересылки для его дальнейшей передачи. Эта функция может вызывать передачу пакета через границу IPsec (исходящий трафик) для дополнительной входной обработки IPsec (например, при использовании вложенных SA). В таких случаях, как для **всего** исходящего трафика, передаваемого в обход, **должно** проверяться соответствие пакета записи в SPD-O. В конечном итоге пакет следует переслать хосту-адресату или обработать пересылки дальше.

6. Обработка ICMP

В этом разделе описана обработка в IPsec трафика ICMP, который делится на две категории - сообщения об ошибках (например, type = destination unreachable) и прочие сообщения (например, type = echo). Данный раздел посвящён исключительно сообщениям об ошибках. Обработка остальных сообщений ICMP, которые не адресованы самой реализации IPsec, **должна** явно учитываться в плане использования записей SPD.

Обсуждение в этом разделе применимо как к ICMPv6, так и к ICMPv4. **Следует** также обеспечивать механизм, который позволит администратору протоколировать сообщения ICMP об ошибках (выбранные, все или никаких) для диагностики.

6.1. Обработка сообщений ICMP об ошибках, направленных реализации IPsec

6.1.1. Сообщения ICMP об ошибках на незащищённой стороне границы

Рисунок 3 в параграфе 5.2 показывает отдельный обработки ICMP на незащищённой стороне границы IPsec, предназначенный для обработки сообщений ICMP (ошибки и другие сообщения), адресованных устройству IPsec и не защищённых AH или ESP. Сообщения ICMP этого сорта являются неаутентифицированными и их обработка может приводить к отказу или снижению производительности сервиса. По этой причине в общем случае желательно игнорировать такие сообщения. Однако хосты и защитные шлюзы получают много сообщений ICMP из неаутентифицированных источников (например, маршрутизаторов в открытой сети Internet. Игнорирование этих сообщений сообщений PMTU или перенаправлений). Таким образом, нужна мотивация для восприятия и выполняемых действий в случаях получения неаутентифицированных сообщений ICMP.

Для учёта всех особенностей, связанных с обработкой сообщений ICMP соответствующая спецификации реализация IPsec должна позволять локальному администратору возможность настройки восприятия или отбрасывания неаутентифицированного трафика ICMP. Эти средства **должны** обеспечивать гранулярность по типу ICMP и **могут** обеспечивать гранулярность по типу и коду. В дополнение к этому реализации **следует** поддерживать механизмы и параметры для работы с таким трафиком. Например, может обеспечиваться возможность поддержки минимального PMTU для трафика (по адресатам), чтобы предотвратить установку слишком малых значение PMTU в результате обработки неаутентифицированных сообщений ICMP.

Если сообщение ICMP PMTU проходит указанную выше проверку и система настроена на восприятие такого сообщения, возникают две возможности. Если реализация использует фрагментацию на шифрованной стороне границы, принятая информация PMTU передаётся модулю пересылки (не относится к реализации IPsec), который использует её при фрагментации исходящих пакетов. Если реализация настроена на фрагментацию нешифрованного трафика, информация PMTU передаётся на защищённую (нешифрованную) сторону и обрабатывается, как описано в параграфе 8.2.

6.1.2. Сообщения ICMP об ошибках, принимаемые на защищённой стороне

Эти сообщения ICMP не аутентифицированы, но они приходят с защищённой стороны границы IPsec. Таким образом, эти сообщения в общем случае рассматриваются, как более «доверенные» по сравнению с описанными выше сообщениями с незащищённой стороны. Основная проблема безопасности для таких сообщений связана с тем, что скомпрометированный хост или маршрутизатор может генерировать не соответствующие действительности сообщения ICMP об ошибках, которые могут снижать производительность сервиса для других устройств, находящихся «за шлюзом», и даже приводить к нарушению конфиденциальности. Например, если ложные сообщения ICMP redirect будут восприниматься защитным шлюзом, это может приводить к изменению таблицы пересылки на защищённой стороне, в результате чего трафик будет попадать неприемлемому получателю «за шлюзом». Таким образом, реализации **должны** обеспечивать средства контроля, позволяющие локальному администратору ограничить обработку сообщений ICMP об ошибках, полученных с защищённой стороны границы и направленных реализации IPsec. Эти средства контроля имеют тот же тип, который описан для незащищённой стороны в параграфе 6.1.1.

6.2. Обработка защищённых транзитных сообщений ICMP об ошибках

Когда сообщение ICMP об ошибке передаётся через SA устройству, находящемуся за реализацией IPsec, требуется проверять заголовок и содержимое сообщения ICMP с точки зрения контроля доступа. Если одно из таких сообщений пересылается хосту за защитным шлюзом, реализация принимающего хоста будет принимать решение на основе содержимого сообщения (т. е., заголовка пакета, вызвавшего передачу сообщения об ошибке). Таким образом, реализация IPsec **должна** обеспечивать возможность настройки на проверку соответствия этого заголовка SA, через которую был получен пакет (это означает, что заголовок с обращёнными адресами о номерами портов отправителя и получателя должен соответствовать селекторам трафика для SA). Если такую проверку не проводить, тогда любой, с кем принимающая система IPsec (A) имеет активную SA, сможет передать сообщение ICMP Destination Unreachable, указывающее на недоступность хоста/сети, с которым A взаимодействует, что создаёт возможность организации очень эффективных DoS-атак на соединения. Обычный получатель IPsec, обрабатывающий трафик, недостаточно защищён от таких атак. Однако упомянутая выше проверка требуется не в каждом контексте, поэтому необходимо предоставить локальному администратору возможность **отказа** от такой проверки.

Для обеспечения обеих возможностей принимается описанное далее соглашение. Если администратор разрешит передачу сообщений ICMP об ошибках через SA без проверки содержимого, создаётся запись SPD, которая явно позволяет передавать такой трафик. Если администратор решит проверять содержимое сообщений ICMP об ошибках

на предмет соответствия SA, записи SPD для восприятия такого трафика на основе заголовка пакета ICMP не создаётся. Это соглашение служит мотивацией для описанной далее обработки.

Отправители и получатели IPsec **должны** поддерживать описанную ниже обработку сообщений ICMP об ошибках, передаваемых и принимаемых через SA.

Если существует SA, которая принимает исходящее сообщении ICMP об ошибке, сообщение отображается на эту SA и при получении проверяются только заголовки IP и ICMP, как для других типов трафика. Если не существует SA, соответствующей селекторам трафика, связанным с сообщением ICMP об ошибке, просматривается SPD для определения возможности создания такой SA. При положительном результате создаётся SA и сообщение ICMP передаётся через неё. При получении к такому сообщению применяется обычная проверка селекторов. Такая обработка ничем не отличается от обработки обычного трафика и не включает каких-либо специальных действий применительно к обработке сообщений ICMP об ошибках.

Если нет SA, которая будет использоваться для передачи исходящих сообщений ICMP, о которых идёт речь, и нет записи SPD, которая позволяет передавать исходящие сообщения ICMP об ошибках, реализация IPsec **должна** отобразить сообщение на SA, которая будет передавать трафик, связанных с пакетом, вызвавшим сообщение ICMP об ошибке. Это требует от реализации IPsec детектировать исходящие сообщения ICMP об ошибках, которые отображаются на несуществующую SA или запись SPD и специальным образом обрабатывать их в плане поиска и создания SA. Реализация просматривает заголовок пакета для определения информации о пакете, вызвавшем ошибку (из поля данных сообщения ICMP), меняет местами адреса отправителя и получателя, выделяет поле протокола и меняет местами номера портов отправителя и получателя (если они доступны). Полученная информация служит для определения подходящей активной исходящей SA и передачи сообщения об ошибке через эту SA. Если SA не существует, новая SA не создаётся и в журнал аудита вносится запись об этом факте.

Если реализация IPsec получает входящее сообщение ICMP об ошибке на SA, f заголовки IP и ICMP в этом сообщении не соответствуют селекторам трафика для SA, получатель **должен** обрабатывать полученное сообщение особо. В частности, получатель **должен** выделить заголовок вызвавшего ошибку пакета из сообщения ICMP и выполнить обращение полей, как описано выше, для определения соответствия пакета SA, через которую было получено сообщение ICMP. При несоответствии реализации IPsec **недопустимо** пересылать сообщение ICMP об ошибке указанному в нем адресату. Информация о таких событиях заносится в журнал аудита.

7. Обработка фрагментов на защищённой стороне границы IPsec

В предыдущих разделах документа описаны механизмы для (a) фрагментации исходящих пакетов после обработки IPsec и сборки фрагментов на приёмной стороне до обработки IPsec и (b) механизмы обработки входящих фрагментов, полученных с незащищённой стороны границы IPsec. В этом разделе рассматривается, как реализации следует выполнять обработку исходящих незашифрованных фрагментов на защищённой стороне границы IPsec (см. Приложение D: Обоснование обработки фрагментов) В частности, здесь рассматривается:

- отображение исходящих фрагментов, не являющихся первыми на корректную SA (или поиск записи SPD);
- проверка полномочности передачи фрагмента, отличного от первого, через SA, в которой он был получен;
- отображение исходящих и входящих фрагментов, не являющихся первыми, на корректную запись SPD-O/SPD-I или подходящую запись в кэше для трафика BYPASS/DISCARD.

Примечание. В параграфе 4.1 SA транспортного режима определены, как не передающие фрагменты (IPv4 или IPv6). Отметим также, что в параграфе 4.4.1, были определены два специальных случая селекторов - ANY и OPAQUE, причём ANY включает в себя OPAQUE. Для селекторов, отличных от OPAQUE и ANY используется термин «нетривиальный».

Примечание. Термин «фрагмент, отличный от первого» используется для индикации фрагментов, не содержащих значения всех селекторов, которые могут потребоваться для контроля доступа. Как отмечено в параграфе 4.4.1, в зависимости от значения Next Layer Protocol, кроме номеров порта в таких фрагментах может отсутствовать тип/код ICMP или тип Mobility Header. Для IPv6 даже первый фрагмент может **не** включать значений Next Layer Protocol или Port (тип/код ICMP или тип Mobility Header) в зависимости от числа и типа имеющихся расширений заголовка. Если отличный от первого фрагмент содержит Port (тип и код ICMP или тип Mobility Header), но не включает Next Layer Protocol, тогда при отсутствии записи SPD для подходящих адресов Local/Remote с ANY в полях Next Layer Protocol и Port (тип и код ICMP или тип Mobility Header) фрагмент не будет включать полного набора информации, требуемой для контроля доступа.

Для решения этих вопросов определены три модели:

- SA туннельного режима, передающие отличные от первых фрагменты (параграф 7.1.);
- отдельные SA туннельного режима для фрагментов, отличных от первых (параграф 7.2.);
- контроль фрагментов с учётом состояний (параграф 7.3.).

7.1. SA туннельного режима, передающие любые фрагменты

Все реализации **должны** поддерживать SA туннельного режима, которые настраиваются на передачу трафика вне зависимости от номера порта (типа/кода ICMP или типа Mobility Header type). Если SA будет передавать трафик для заданных протоколов, набор селекторов для SA **должен** задавать поля портов (типа/кода ICMP или типа Mobility Header), как ANY. Определённые таким способом SA будут передавать весь трафик, включая первые и последующие фрагменты для указанных адресов Local/Remote и заданного протокола (протоколов) следующего уровня. Если SA будет передавать трафик независимо от протокола (т. е, Next Layer = ANY), значения портов не определены и **должны** быть указаны, как ANY (как отмечено в параграфе 4.4.1, значение ANY включает OPAQUE и все конкретные номера).

7.2. Отдельные туннельные SA для фрагментов, отличных от первых

Реализация **может** поддерживать SA туннельного режима, через которые будут передаваться только нефрагментированные пакеты и первые фрагменты. Для задания поля порта (типа/кода ICMP или типа Mobility Header) селекторов таких SA будет использоваться значение OPAQUE. Получатели **должны** проверять минимальное смещение в (отличных от первого) фрагментах IPv4 для защиты от атак с перекрывающимися фрагментами, когда используются SA такого типа. Поскольку такие проверки не могут осуществляться для (отличных от первого) фрагментов IPv6, пользователям и администраторам следует принимать во внимание опасность, связанную с передачей таких фрагментов и реализации **могут** отказываться от поддержки таких SA для трафика IPv6. SA этого типа также будут передавать все отличные от первых фрагменты, которые соответствуют заданной паре адресов Local/Remote и значению протокола (т. е., фрагменты, передаваемые в таких SA, относятся к пакетам, которые не будучи фрагментированными, могли бы передаваться через другие SA с иной защитой). Следовательно, пользователям и администраторам нужно защищать такой трафик с использованием ESP (с защитой целостности) и «самыми строгими» алгоритмами шифрования и защиты целостности для данной пары партнёров (определение «самого строгого» алгоритма требует упорядочивания доступных алгоритмов со стороны инициатора данной SA).

Значения селекторов порта (типа/кода ICMP или типа Mobility Header) будут использоваться для определения SA, передающих нефрагментированные пакеты и первые фрагменты. Такая модель может использоваться, если пользователь или администратор хочет создать одну или более SA туннельного режима между одной парой адресов Local/Remote и эти связи разделяются по полям портов (типов/кодов ICMP или типов Mobility Header). Эти SA **должны** иметь нетривиальные значения селекторов протокола, иначе **должен** использоваться описанный выше вариант #1.

Примечание. В общем случае для описанной здесь модели требуется только одна SA между двумя реализациями для передачи фрагментов, отличных от первого. Однако, если выбрано использование множества SA между парой реализаций для дифференциации QoS, может возникнуть желание создать множество SA для передачи «фрагментов без номера порта», каждая из которых поддерживает класс QoS. Поскольку поддержка QoS через разные SA задаётся на локальном уровне и не определяется данным документом, выбор использования множества SA для передачи фрагментов, отличных от первого, также решается на локальном уровне.

7.3. Проверка фрагментов с учётом состояния

Реализация **может** поддерживать ту или иную форму проверки фрагментов с учётом состояния для SA туннельного режима с нетривиальным (отличным от ANY и OPAQUE) значением поля порта (типа/кода ICMP или типа MH). Реализации, которые будут передавать отличные от первых фрагменты в SA туннельного режима, использующие нетривиальные селекторы порта (типа/кода ICMP или типа MH), **должны** уведомлять партнёра. с помощью элемента данных IKE NOTIFY NON_FIRST_FRAGMENTS_ALSO.

Партнёр **должен** отвергать такие предложения, если он не будет принимать отличные от начального фрагменты в таком контексте. Если реализация не согласовала передачу фрагментов, отличных от первого, для такой SA, ей **недопустимо** передавать эти фрагменты через SA. Этот стандарт не задаёт для партнёров способа обработки фрагментов (например, сборки или иные операции на стороне отправителя или получателя). Однако получатель **должен** отбрасывать отличные от первого фрагменты, которые приходят через SA с нетривиальным селектором порта (типа/кода ICMP или типа MH), если приём таких фрагментов не был согласован. Получатель также **должен** отбрасывать отличные от первого фрагменты, которые не соответствуют политике защиты для полного пакета. Записи о таких событиях заносятся в журнал аудита. Отметим, что в сетевых конфигурациях, где фрагменты пакетов могут передаваться и приниматься через разные защитные шлюзы или реализации BITW, стратегии отслеживания состояния фрагментации могут давать отказы.

7.4. Операции BYPASS/DISCARD для трафика

Все реализации **должны** поддерживать отбрасывание фрагментов с использованием нормальных механизмов классификации SPD. Все реализации **должны** поддерживать проверку фрагментов с учётом состояния, чтобы принимать передаваемый в обход (BYPASS) трафик, для которого задан нетривиальный диапазон портов. Дело в том, что передаваемый в обход трафик представляет собой нешифрованные фрагменты, отличные от первого и прибывающие в реализацию IPsec, могут подрывать защиту, обеспечиваемую для трафика IPsec, направленного тому же получателю. Рассмотрим в качестве примера реализацию IPsec с записью SPD, которая обеспечивает защиту трафика между парой адресов (отправитель-получатель) для заданного протокола и порта получателя (например, трафик TCP в порт 23 - Telnet). Предположим, что реализация также разрешает передавать в обход трафик между той же парой хостов для того же протокола, но с другим номером порта (например, порт 119 - NNTP). Атакующий может передать отличный от первого фрагмент (с обманным адресом отправителя), который при передаче в обход может перекрываться с трафиком IPsec с того же адреса и, таким образом, нарушать целостность трафика IPsec. Требование проверки с учётом состояния для отличных от первого фрагментов с нетривиальными значениями портов, направляемых в обход, позволяет предотвратить такие атаки. Как отмечено выше, в сетях, где фрагменты могут передаваться и приниматься через разные защитные шлюзы или реализации BITW, могут возникать проблемы с проверкой фрагментов с учётом состояния.

8. Обработка Path MTU и DF

Операции AH и ESP, применяемые к исходящим пакетам, увеличивают размер пакетов и могут в результате приводить к тому, что размер пакета превысит значение PMTU для SA, через которую пакет будет передаваться. Реализация IPsec может также получить незащищённое сообщение ICMP PMTU, реакция на которое будет воздействовать на обработку исходящего трафика. В этом разделе описаны операции, которые реализация IPsec должна выполнять в упомянутых ситуациях.

8.1. Бит DF

Все реализации IPsec **должны** поддерживать опцию копирования флага DF из заголовка исходящего пакета в заголовок создаваемого пакета туннельного режима при передаче трафика через туннельную SA. Это означает **необходимость** настройки для реализации трактовки флага DF (установить, сбросить, копировать из внутреннего заголовка) применительно к каждой SA (в том случае, когда внешний и внутренний заголовок относятся к IPv4).

8.2. Определение PMTU

В этом параграфе рассматривается обработка IPsec для незащищённых сообщений Path MTU Discovery. Обозначение ICMP PMTU[1] используется здесь для сообщений ICMP в:

IPv4 (RFC 792 [Pos81b]):

- Type = 3 (Адресат недоступен)
- Code = 4 (Нужна фрагментация, но установлен флаг DF)
- Next-Hop MTU в младших 16 второго слова заголовка ICMP (указаны в RFC 792, как неиспользуемые) с нулевым значением старших 16 битов.

IPv6 (RFC 2463 [CD98]):

- Type = 2 (Пакет слишком велик)
- Code = 0 (Нужна фрагментация)
- Next-Hop MTU в 32-битовом поле MTU сообщения ICMP6.

8.2.1. Распространение PMTU

Когда реализация IPsec получает неаутентифицированное сообщение PMTU, будучи настроенной на обработку (вместо игнорирования) таких сообщений, она отображает это сообщение на SA, которой оно соответствует. Отображение осуществляется с использованием информации из заголовка в поле данных сообщения PMTU по процедуре, описанной в параграфе 5.2. Определяемое таким сообщением значение PMTU используется для обновления поля SAD PMTU с учётом размера заголовка AH или ESP, который будет использован, всех данных криптографической синхронизации, а также издержек, связанных с дополнительным заголовком IP для туннельных SA.

В естественной реализации хоста возможно поддерживать данные PMTU с такой же гранулярностью, как для незащищённых коммуникаций, т. е., без потери функциональности. Сигнализация для данных PMTU является внутренним делом хоста. Для остальных реализаций IPsec данные PMTU должны распространяться с использованием синтезированного сообщения ICMP PMTU. В таких случаях реализации IPsec **следует** дождаться отображения исходящего трафика на запись SAD. При поступлении такого трафика с превосходящим PMTU размером пакетов должна выполняться следующая обработка:

Случай 1 - исходный (нешифрованный) пакет относится к IPv4 и имеет флаг DF. Реализации **следует** отбросить пакет и передать сообщение PMTU ICMP.

Случай 2 - исходный (нешифрованный) пакет относится к IPv4 и не имеет флага DF. Реализации **следует** фрагментировать пакет (до шифрования или после него в зависимости от настроек) и переслать фрагменты. Сообщение PMTU ICMP передавать **не следует**.

Случай 3 - исходный (нешифрованный) пакет относится к IPv6. Реализации **следует** отбросить пакет и передать сообщение PMTU ICMP.

8.2.2. Старение PMTU

Во всех реализациях IPsec значение PMTU, связанное с SA **должно «**стареть» с использованием того или иного механизма для своевременного обновления PMTU, особенно с целью проверки того, что найденное значение PMTU не меньше требуемого текущими условиями в сети. Данное значение PMTU сохраняется достаточно долго, чтобы доставить пакет от источника SA до партнёра. и передать сообщение ICMP, если текущее значение PMTU слишком велико.

Реализациям следует использовать модель, описанную с документе Path MTU Discovery (RFC 1191 [MD90], параграф 6.3), которая предлагает периодически сбрасывать PMTU для значения MTU канала данных на первом интервале, а потом обновлять его при необходимости с помощью обычного процесса PMTU Discovery. Период **следует** делать настраиваемым.

9. Проведение проверок

Реализации IPsec не обязаны поддерживать аудит. В большинстве случаев гранулярность аудита определяется на локальном уровне. Однако в этом документе отмечены некоторые события и для каждого из таких событий указан минимальный набор информации, которую следует включать в журнал аудита. Для каждого из таких событий в журнал также **может** включаться дополнительная информация. **Возможно** включение записей в журнал аудита в связи с событиями, которые не указаны явно в этом документе. Получатель не обязан передавать все сообщения предполагаемому отправителю в ответ на заносимые в журнал аудита события, поскольку такое поведение открывало бы возможность для организации атак на отказ служб.

10. Соответствие требованиям

Все реализации IPv4 IPsec **должны** удовлетворять всем требованиям, приведённым в данном документе. Все реализации IPv6 IPsec **должны** удовлетворять всем требованиям, приведённым в данном документе.

11. Вопросы безопасности

Основной темой этого документа является защита - следовательно, вопросы безопасности являются неотъемлемой частью данной спецификации.

IPsec вносит строгие ограничения на обход данных в заголовках IP для обоих направления прохождения через границу IPsec, в частности, при использовании SA туннельного режима. Некоторые ограничения являются абсолютными, иные

[1]Path MTU - MTU для пути.

находятся в ведении локального администратора (зачастую, на уровне SA). Для исходящего трафика ограничения направлены на сужение полосы скрытых каналов. Для входящего трафика целью ограничений является предотвращение враждебного воздействия на другие потоки данных (на защищённой стороне IPsec) со стороны тех злоумышленников, которые могут перехватывать поток данных (на незащищённой стороне IPsec). Иллюстрацией может служить приведённое в разделе 5 обсуждение обработки значений DSCP для SA туннельного режима.

Если реализация IPsec настроена на пропускание сообщений ICMP об ошибках через SA по значениям заголовков ICMP без проверки данных из заголовков, содержащихся в поле данных ICMP, может возникнуть серьёзная уязвимость. Рассмотрим сценарий с несколькими сайтами (A, B, C), соединёнными между собой туннелями ESP ((A-B, A-C, B-C). Предположим также, что селекторы трафика для каждого туннеля содержат значение ANY в полях протокола и порта, а адреса IP для отправителя и получателя содержат диапазоны, охватывающие блоки адресов, находящиеся за защитными шлюзами каждого из сайтов. Это позволяет хосту сайта B передавать любому хосту сайта A сообщения ICMP Destination Unreachable, которые говорят о недоступности всех хостов сети на сайте C. В результате возникает возможность организации очень эффективной DoS-атаки, которая могла бы быть предотвращена, если бы в сообщениях ICMP не только проверялись заголовки, но и выполнялись бы другие проверки, обеспечиваемые IPsec (если SPD подобающим образом настроена, как описано в параграфе 6.2).

12. Взаимодействие с IANA

Агентство IANA выделило значение (3) для реестра asn1-modules и идентификатор объекта 1.3.6.1.5.5.8.3.1[1] для модуля SPD (см. Приложение C, «ASN.1 для записи SPD»).

13. Отличия от RFC 2401

Этот документ, посвящённый архитектуре защиты, существенно отличается от RFC 2401 [RFC2401] в деталях и организации самого документа, но основы архитектуры остались неизменными.

- Модель обработки была пересмотрена для реализации новых сценариев IPsec, повышения производительности и упрощения реализации. Изменения включают разделение пересылки (маршрутизации) и выбора SPD, раздельное изменение SPD, добавление исходящего кэша SPD и входящего кэша SPD для передаваемого в обход и отбрасываемого трафика. Добавлена также база данных аутентификации партнёров (PAD). Это обеспечило связь между протоколом управления SA (таким, как IKE) и SPD.
- Снято требование поддержки вложенных SA или «связок SA». Соответствующая функциональность может быть достигнута за счёт SPD и таблицы пересылки. Пример конфигурации приведён в Приложении E.
- Переопределены записи SPD для повышения уровня гибкости. Каждая запись SPD сейчас включает от 1 до N наборов селекторов, каждый из которых содержит один протокол, а список диапазонов может включать адреса Local IP, Remote IP и другие поля (если они есть), связанные с протоколом следующего уровня (локальный порт, удалённый порт, тип и код сообщения ICMP, тип Mobility Header). Отдельные значения представляются тривиальным (вырожденным) диапазоном, а значение ANY представляется диапазоном, который включает все значения селектора. Пример описания ASN.1 включён в Приложение C.
- TOS (IPv4) и Traffic Class (IPv6) были заменены на DSCP и ECN. Обновлён раздел, посвящённый туннелям, в плане обработки битов DSCP и ECN.
- Для SA туннельного режима реализациям SG, BITS и BITW сейчас разрешено фрагментировать пакеты до обработки IPsec. Это относится только к IPv4, а пакеты IPv6 разрешается фрагментировать только их источнику.
- Кода требуется защита между двумя промежуточными точками пути или между промежуточной и конечной точками, можно использовать транспортный режим между защитными шлюзами или между хостом и защитным шлюзом.
- В этом документе уточнено, что для всего трафика, проходящего через границу IPsec (включая трафик управления IPsec), должна запрашиваться SPD или связанные с ней кэшированные записи.
- В этом документе описана обработка ситуаций, когда на защитном шлюзе с множеством абонентов требуется организация раздельных контекстов IPsec.
- Добавлено определение резервных SPI.
- Расширено объяснение необходимости проверки **всех** пакетов IP - IPsec включает минимальную функциональность межсетевого экрана для контроля доступа на уровне IP.
- В разделе, посвящённом туннелям, разъяснена обработка поля опций IP и расширенных заголовков IPv6 при создании внешнего заголовка.
- Обновлено отображение SA для входящего трафика с целью обеспечения согласованности с изменениями, внесёнными в AH и ESP для поддержки индивидуальных и групповых SA.
- Добавлено руководство, касающееся работы со скрытыми каналами, создаваемыми в туннельном режиме при копировании значения DSCP во внешний заголовок.
- Отменено требование поддержки AH для IPv4 и IPv6 одновременно.
- Обновлена обработка PMTU. Приложение, посвящённое обработке PMTU/DF/Fragmentation удалено.
- Добавлены три модели обработки нешифрованных фрагментов на защищённой стороне границы IPsec. В Приложении D приведено обоснование работы с фрагментами.
- Добавлено описание процесса создания значений селекторов для SA из записей SPD, полей пакета и т. п.

[1] В оригинале ошибочно указан идентификатор 1.3.6.1.5.8.3.1. *Прим. перев.*

- Добавлена таблица, описывающая связи между значениями селекторов в записи SPD, флагом PFP и получаемыми в результате значениями селекторов соответствующей записи SAD.
- Добавлено Приложение B, описывающее декорреляцию.
- Добавлен текст, описывающий обработку исходящих пакетов, которые должны быть отброшены.
- Добавлен текст, описывающий обработку **отбрасываемых** входящих пакетов (т. е., пакетов, не соответствующих SA, через которую они были приняты).
- Добавлен заголовок IPv6 Mobility Header в качестве возможного значения Next Layer Protocol. Тип IPv6 Mobility Header добавлен в качестве селектора.
- Тип и код сообщения ICMP добавлены в качестве селектора.
- В целях упрощения удалён селектор «data sensitivity level».
- Обновлён текст, описывающий обработку сообщений ICMP об ошибках. Приложение «Categorization of ICMP Messages» было удалено.
- Обновлён и прояснён текст для имён. селекторов.
- Приведены дополнительные разъяснения для Next Layer Protocol (протокол следующего уровня) и добавлен принятый по умолчанию список протоколов, пропускаемых при поиске протокола следующего уровня.
- Обновлён текст, в котором сказано, что данный документ предполагает использование протокола IKEv2 или протокола управления SA, обеспечивающего сравнимые возможности.
- Добавлен текст, разъясняющий алгоритм отображения входящих дейтаграмм IPsec на SA в присутствии групповых SA.
- Удалёно приложение Sequence Space Window Code Example.
- Применительно к адресам IP и портам в правилах политики используются термины «локальный» и «удалённый» (взамен отправителя и получателя). Локальный относится к объекту, защищённому реализацией IPsec (т. е., отправителю исходящих пакетов или получателю входящих), а удалённый - к его партнёру. Термины «отправитель» и «получатель» по-прежнему используются применительно к полям заголовков.

14. Благодарности

Авторы отмечают значительный вклад Ran Atkinson в работы на начальном этапе IPsec и подготовку первых вариантов стандарта IPsec - RFC 1825-1827, а также существенный вклад Charlie Lynn в подготовку второй версии стандарта IPsec - RFC 2401, 2402, 2406 и текущей версии (в части IPv6). Авторы также благодарят членов рабочих групп IPsec и MSEC, внёсших важный вклад в разработку спецификации протокола.

Приложение A: Глоссарий

В этом разделе даны определения ключевых терминов, используемых в этом документе. Дополнительные определения и базовая информация содержатся также в ряде дополнительных документов, посвящённых этой технологии (например, [Shi00], [VK83], [HA94]). В этот глоссарий включены базовые термины, связанные с вопросами защиты, и термины IPsec.

Access Control - контроль доступа

Услуга защиты, предотвращающая несанкционированное использование ресурсов, включая использование ресурсов сверх имеющихся полномочий. В контексте IPsec к контролируемым ресурсам относятся:

- вычислительные ресурсы и данные на хостах;
- защищаемые шлюзами сети и полоса сетевых соединений.

Anti-replay - предотвращение повторного использования пакетов

См. Integrity ниже.

Authentication - проверка подлинности

Неформально используется для обозначения комбинации двух номинально разных услуг защиты - проверки подлинности источника данных и защиты целостности соединений (см. приведённые ниже определения этих терминов).

Availability - доступность

В контексте защиты - решение проблем, вызываемых атаками на сети, связанными со снижением производительности или нарушением работы служб. Например, в контексте IPsec доступность обеспечивается механизмами предотвращения повторного использования пакетов в протоколах AH и ESP.

Confidentiality - конфиденциальность

Услуга защиты, обеспечивающая предотвращение раскрытия информации. Основной проблемой конфиденциальности в большинстве случаев является несанкционированное раскрытие данных прикладного уровня, однако раскрытие коммуникационных параметров также может нарушать конфиденциальность при определённых обстоятельствах. Конфиденциальность потока трафика представляет собой услугу по сокрытию адресов отправителей и получателей, размера сообщений, интенсивности обмена информацией. В контексте IPsec использование ESP в туннельном режиме (особенно на защитных шлюзах) может обеспечивать некоторый уровень конфиденциальности потока трафика (см. также Traffic Analysis).

Data Origin Authentication - аутентификация источника данных

Услуга защиты, обеспечивающая проверку подлинности заявленного источника данных. Эта услуга обычно объединяется с защитой целостности (без привязки к соединению).

Encryption - шифрование

Механизм защиты, используемый для преобразования данных из понятной формы (нешифрованной) в совершенно непонятную (шифрованную) для обеспечения конфиденциальности информации. Обратное преобразование называется расшифровкой или дешифровкой. Зачастую термин «шифрование» используется для обозначения обоих процессов вместе.

Integrity - целостность

Услуга защиты, обеспечивающая детектирование внесённых в данные изменений. Защита целостности может обеспечиваться в различных формах в зависимости от потребностей приложений. IPsec поддерживает две формы защиты целостности - не связанную с соединениями (connectionless) и частичную защиту порядка (partial sequence integrity). Защита целостности, не связанная с соединениями, представляет собой услугу по обнаружению изменения отдельных дейтаграмм IP без связи с порядком их следования в потоке трафика. Частичная защита порядка, предлагаемая в IPsec, называется также предотвращением повторного использования пакетов, и обеспечивает детектирование дубликатов дейтаграмм IP (в пределах ограниченного окна). Этим она отличается от защиты целостности соединений, которая обеспечивает более жёсткий контроль упорядоченности трафика (например, может детектировать потерю или нарушение порядка доставки сообщений). Хотя услуги аутентификации и защиты целостности часто рассматривают отдельно, на практике они обычно тесно связаны и почти всегда предоставляются в тандеме.

Protected vs. Unprotected - защищённый и незащищённый

«Защищённой» называют систему или интерфейс, находящиеся внутри защитной границы IPsec, а «незащищённой» - систему или интерфейс, находящиеся за пределами защитной границы. IPsec обеспечивает границу, через которую проходит трафик. На границе имеется асимметрия, которая отражена в модели обработки. Исходящие данные, если они не передаются в обход и не отбрасываются, будут защищены с использованием AH или ESP и добавлением соответствующих заголовков. Входящие данные, если они не передаются в обход и не отбрасываются, будут обрабатываться путём удаления заголовков AH или ESP. В этом документе входящим считается трафик, который приходит реализации IPsec от «незащищённого» интерфейса. Исходящий трафик поступает с «защищённого» интерфейса или генерируется самой реализацией на «защищённой» стороне границы и направляется в сторону «незащищённого» интерфейса. Реализация IPsec может поддерживать более одного интерфейса с каждой стороны границы. Защищённый интерфейс может быть внутренним (например, для реализации IPsec на хосте). Защищённый интерфейс может быть связан с интерфейсом уровня сокета, обеспечиваемым операционной системой (OS).

Security Association (SA) - защищённая связь

Симплексное (одностороннее) логическое соединение, создаваемое в целях защиты. Для всего трафика, проходящего через SA обеспечивается одинаковая защитная обработка. В IPsec защищённая связь (SA) представляет собой абстракцию уровня Internet, реализуемую за счёт использования AH или ESP. Данные о состоянии, связанные с SA представляются в базе данных SA (SAD).

Security Gateway (SG) - защитный шлюз

Промежуточная система, выполняющая функции коммуникационного интерфейса между двумя сетями. Хосты (сети), расположенные с внешней стороны защитного шлюза, называют незащищёнными (они в любом случае менее защищены, чем те, которые находятся «за шлюзом»), а сети и хосты с внутренней стороны шлюза называют защищёнными. Внутренние подсети и хосты, обслуживаемые защитным шлюзом, будут доверенными в силу использования общего, локального администрирования защиты. В контексте IPsec защитный шлюз является точкой, в которой реализуются AH и/или ESP для обслуживания внутренних хостов с целью предоставления им услуг защиты при обмене информацией с внешними хостами, которые также используют IPsec (непосредственно или через другой защитный шлюз).

Security Parameters Index (SPI) - список параметров защиты

Произвольное 32-битовое значение, используемое получателем для аутентификации SA, с которой следует связать входящий поток. Для индивидуальных SA значение SPI можно использовать для указания SA само по себе или в комбинации с типом протокола IPsec. Для аутентификации групповых SA дополнительно используются адреса IP. Значения SPI передаются в протоколах AH и ESP для того, чтобы принимающая система могла выбрать SA, в которой будут обрабатываться принимаемые пакеты. SPI имеет только локальную значимость, как определяется создателем SA (обычно, получатель пакета, содержащего SPI), поэтому в общем случае SPI представляется неформатированной строкой битов. Однако создатель SA может выбрать интерпретацию битов SPI для упрощения локальной обработки.

Traffic Analysis - анализ трафика

Анализ потока сетевого трафика с целью обнаружения информации, которая может быть полезна противнику. Примерами такой информации могут служить интенсивность обмена данными, аутентификация сторон, размеры пакетов, идентификаторы потоков [Sch94].

Приложение B: Декорреляция

Это приложение основано на работах по кэшированию правил, выполненных Luis Sanchez, Matt Condell и John Zao в рамках рабочей группы IP Security Policy.

Две записи SPD коррелируют между собой, если имеется непустое пересечение значений соответствующих селекторов в каждой записи. Кэширование коррелирующих записей SPD может приводить к некорректному выполнению правил. Решением этой проблемы, сохраняющим возможность кэширования, является удаление

неоднозначностей путём декорреляции записей. Т. е., записи SPD должны быть переписаны таким образом, чтобы для каждой пары записей существовал селектор, для которого пересечение значений в обеих записях было бы пустым. После выполнения декорреляции не возникает необходимости задавать отношения порядка между записями, поскольку при поиске будет соответствовать только одна из записей. В следующем параграфе декорреляция описана более детально и представлен алгоритм, который можно использовать для реализации.

B.1. Алгоритм декорреляции

Базовый алгоритм декорреляции принимает каждую запись в коррелирующих SPD и делит её на множество записей, используя древовидную структуру. Узлы дерева являются селекторами, которые могут перекрываться между правилами. На каждом узле алгоритм создаёт ветвь для каждого значения селектора. Создаётся также одна ветвь для дополнения к объединению значений всех селекторов. Правила формируются путём прохождения по дереву от корня до каждого листа. Правила на листьях сравниваются с набором ранее декоррелированных правил. Каждое правило на каждом листе или полностью перекрывается набором уже декоррелированных правил и отбрасывается, или никак не коррелирует с набором ранее декоррелированных правил и добавляется к нему.

Базовый алгоритм не гарантирует создания оптимального набора декоррелированных записей. Т. е., записи могут представлять собой набор, который меньше необходимого, хотя они по-прежнему будут содержать всю требуемую информацию о правилах. Некоторые расширения базового алгоритма, описанные ниже, позволяют преодолеть этот недостаток и повышают производительность алгоритма.

C набор упорядоченных, коррелирующих записей (коррелирующая SPD).

Ci i-ая запись в C.

U набор декоррелированных записей, созданный из C.

Ui i-ая запись в U.

Sik k-ый выбор для правила Ci.

Ai действие для правила Ci.

Правило (запись SPD) P можно выразить, как последовательность значений селекторов и действие(BYPASS - обход, DISCARD - отбрасывание, PROTECT - защита):

```
Ci = Si1 x Si2 x ... x Sik -> Ai
```

1) Поместим C1 в набор U, как U1

   Для каждого правила Cj (j > 1) в наборе C

2) Если Cj декоррелирована с каждой записью в U, Cj добавляется в U.

3) Если Cj коррелирует с одной или множеством записей в U, создаётся дерево с корнем у Cj, которое делит Cj на множество декоррелированных записей. Алгоритм начинает работу с корневого узла, где ещё не были выбраны селекторы.

   A) Выбирается селектор Sjn в Cj, который ещё не был выбран при прохождении от корня дерева к данному узлу. Если ни одного селектора ещё не было использовано, продолжается работа со следующей незавершённой ветвью, пока все ветви не будут завершены. При завершении дерева переход к п. D.

      T представляет собой набор записей в U, коррелирующих с записью для данного узла.

      Запись у этого узла представляет собой запись, формируемую значениями селекторов каждой из ветвей между корнем и данным узлом. Любые значения селекторов, которые ещё не представлены ветвями, предполагаются соответствующими значению селектора в Cj, поскольку значения в Cj представляют максимальное значение для каждого селектора.

   B) К дереву добавляется ветвь для каждого значения селектора Sjn, которое появляется в любой из записей в T (если значение представляет собой надмножество значения Sjn в Cj, используется значение в Cj, поскольку это значение представляет универсальный набор). Добавляется также ветвь для дополнения к объединению значений селектора Sjn в T. При создании дополнения следует помнить, что универсальный набор является значением Sjn в Cj. Для пустых наборов ветви не создаются.

   C) П. A и B повторяются до завершения дерева.

   D) Запись для каждого листа сейчас представляет запись, являющуюся подмножеством Cj. Записи у листьев полностью делят Cj так, что каждая запись полностью перекрывается записью в U или декоррелирована со всеми записями U.

   Все декоррелированные записи у листьев дерева добавляются к U.

4) Переход к следующей Cj и п. 2.

5) При завершении обработки всех записей C набор U будет представлять собой декоррелированную версию C.

Этот алгоритм можно оптимизировать и некоторые из вариантов оптимизации представлены ниже.

Можно оптимизировать число ветвлений за счёт аккуратного выбора селекторов, используемых для следующей ветви. Например, если селектор Sjn был выбран так, что все значения для этого селектора в T совпадают или являются надмножеством значений Sjn в Cj, нужно будет создавать только одну ветвь (поскольку дополнение будет пустым).

Ветви дерева не требуется обрабатывать с использованием полного алгоритма декорреляции. Например, если узел представляет запись, которая декоррелирована со всеми записями в U, нет причин продолжать декорреляцию этой ветви. Точно так же, если ветвь полностью перекрывается записью в U, эту ветвь больше не нужно декоррелировать.

Дополнительная оптимизация обеспечивается проверкой перекрытия ветви одной из **коррелированных** записей в наборе C, которые уже были декоррелированы. Т. е., если ветвь является частью декоррелированной Cj, проверяется перекрытие этой ветви записью Cm (m < j). Такая проверка корректна, поскольку все записи Cm уже выражены в U.

Вместе с проверкой того, что запись уже была декоррелирована на этапе 2, проверяется перекрытие Cj любой записью в U. Если такое перекрытие присутствует, запись пропускается, поскольку она не имеет отношения к делу. Запись x перекрывается другой записью y, если селектор в x совпадает с соответствующим селектором в y или является его подмножеством.

Приложение C: ASN.1 для записи SPD

Это приложение включено в качестве дополнительного способа описания записей SPD, определённых в параграфе 4.4.1. Здесь используется синтаксис ASN.1, который был успешно скомпилирован. Синтаксис служит исключительно для иллюстрации и его не требуется воплощать в реализациях для обеспечения совместимости. Описание SPD в параграфе 4.4.1 является нормативным.

```
SPDModule
    {iso(1) org (3) dod (6) internet (1) security (5) mechanisms (5)
     ipsec (8) asn1-modules (3) spd-module (1) }

       DEFINITIONS IMPLICIT TAGS ::=

       BEGIN

       IMPORTS
           RDNSequence FROM PKIX1Explicit88
             { iso(1) identified-organization(3)
               dod(6) internet(1) security(5) mechanisms(5) pkix(7)
               id-mod(0) id-pkix1-explicit(18) } ;

       -- SPD представляет собой список правил в порядке убывания предпочтений
       SPD ::= SEQUENCE OF SPDEntry

       SPDEntry ::= CHOICE {
           iPsecEntry       IPsecEntry,                -- PROTECT (защита трафика)
           bypassOrDiscard  [0] BypassOrDiscardEntry } -- DISCARD/BYPASS (обход/отбрасывание)

       IPsecEntry ::= SEQUENCE {       -- Каждая запись включает
           name         NameSets OPTIONAL,
           pFPs         PacketFlags,    -- Заполняется в соответствии с флагами в пакете
                          -- Применяется ко всем соответствующим селекторам трафика
                          -- из SelectorLists
           condition    SelectorLists,  -- Правило "condition" - условие
           processing   Processing      -- Правило "action" - действие
           }

       BypassOrDiscardEntry ::= SEQUENCE {
           bypass       BOOLEAN,        -- TRUE BYPASS (обход), FALSE DISCARD (отбрасывание)
           condition    InOutBound }

       InOutBound ::= CHOICE {
           outbound     [0] SelectorLists,
           inbound      [1] SelectorLists,
           bothways     [2] BothWays }

       BothWays ::= SEQUENCE {
           inbound      SelectorLists,
           outbound     SelectorLists }

       NameSets ::= SEQUENCE {
           passed       SET OF Names-R,  -- Соответствует IKE ID
           local        SET OF Names-I } -- Используется инициатором IKE

       Names-R ::= CHOICE {                    -- IKEv2 IDs
           dName        RDNSequence,           -- ID_DER_ASN1_DN
           fqdn         FQDN,                  -- ID_FQDN
           rfc822       [0] RFC822Name,        -- ID_RFC822_ADDR
           keyID        OCTET STRING }         -- KEY_ID

       Names-I ::= OCTET STRING       -- Используется инициатором IKE

       FQDN ::= IA5String

       RFC822Name ::= IA5String

       PacketFlags ::= BIT STRING {
                  -- при установленном флаге берётся значение селектора из пакета,
                  -- создающего SA, иначе используется значение из записи SPD
           localAddr  (0),
           remoteAddr (1),
           protocol   (2),
           localPort  (3),
           remotePort (4)  }

       SelectorLists ::= SET OF SelectorList
```

```
SelectorList ::= SEQUENCE {
    localAddr   AddrList,
    remoteAddr  AddrList,
    protocol    ProtocolChoice }

Processing ::= SEQUENCE {
    extSeqNum   BOOLEAN, -- TRUE 64-битовый счётчик, FALSE - 32-битовый
    seqOverflow BOOLEAN, -- TRUE смена ключа, FALSE прерывание с записью в журнал
    fragCheck   BOOLEAN, -- TRUE проверка фрагментов с учётом состояния,
                         -- FALSE без проверки фрагментов с учётом состояния
    lifetime    SALifetime,
    spi         ManualSPI,
    algorithms  ProcessingAlgs,
    tunnel      TunnelOptions OPTIONAL } -- при отсутствии использ. транспортный режим

SALifetime ::= SEQUENCE {
    seconds   [0] INTEGER OPTIONAL,
    bytes     [1] INTEGER OPTIONAL }

ManualSPI ::= SEQUENCE {
    spi     INTEGER,
    keys    KeyIDs }

KeyIDs ::= SEQUENCE OF OCTET STRING

ProcessingAlgs ::= CHOICE {
    ah          [0] IntegrityAlgs,  -- AH
    esp         [1] ESPAlgs}        -- ESP

ESPAlgs ::= CHOICE {
    integrity       [0] IntegrityAlgs,       -- только защита целостности
    confidentiality [1] ConfidentialityAlgs, -- только защита конфиденциальности
    both            [2] IntegrityConfidentialityAlgs,
    combined        [3] CombinedModeAlgs }

IntegrityConfidentialityAlgs ::= SEQUENCE {
    integrity       IntegrityAlgs,
    confidentiality ConfidentialityAlgs }

-- Алгоритмы защиты целостности в порядке снижения предпочтений
IntegrityAlgs ::= SEQUENCE OF IntegrityAlg

-- Алгоритмы защиты конфиденциальности в порядке снижения предпочтений
ConfidentialityAlgs ::= SEQUENCE OF ConfidentialityAlg

--  Алгоритмы защиты целостности
IntegrityAlg ::= SEQUENCE {
    algorithm   IntegrityAlgType,
    parameters  ANY -- Определяются алгоритмом -- OPTIONAL }

IntegrityAlgType ::= INTEGER {
    none               (0),
    auth-HMAC-MD5-96   (1),
    auth-HMAC-SHA1-96  (2),
    auth-DES-MAC       (3),
    auth-KPDK-MD5      (4),
    auth-AES-XCBC-96   (5)
--  tbd (6..65535)
    }

-- Алгоритмы защиты конфиденциальности
ConfidentialityAlg ::= SEQUENCE {
    algorithm   ConfidentialityAlgType,
    parameters  ANY -- Определяются алгоритмом -- OPTIONAL }

ConfidentialityAlgType ::= INTEGER {
    encr-DES-IV64    (1),
    encr-DES         (2),
    encr-3DES        (3),
    encr-RC5         (4),
    encr-IDEA        (5),
    encr-CAST        (6),
    encr-BLOWFISH    (7),
    encr-3IDEA       (8),
    encr-DES-IV32    (9),
    encr-RC4        (10),
    encr-NULL       (11),
    encr-AES-CBC    (12),
    encr-AES-CTR    (13)
--  tbd (14..65535)
    }

CombinedModeAlgs ::= SEQUENCE OF CombinedModeAlg

CombinedModeAlg ::= SEQUENCE {
```

```
    algorithm   CombinedModeType,
    parameters  ANY -- Определяются алгоритмом} -- определены в других документах
                                                -- для режимов AES.

CombinedModeType ::= INTEGER {
    comb-AES-CCM    (1),
    comb-AES-GCM    (2)
--  будут определены впоследствии (3..65535)
    }

TunnelOptions ::= SEQUENCE {
    dscp        DSCP,
    ecn         BOOLEAN,    -- TRUE копировать CE во внутренний заголовок
    df          DF,
    addresses   TunnelAddresses }

TunnelAddresses ::= CHOICE {
    ipv4        IPv4Pair,
    ipv6        [0] IPv6Pair }

IPv4Pair ::= SEQUENCE {
    local       OCTET STRING (SIZE(4)),
    remote      OCTET STRING (SIZE(4)) }

IPv6Pair ::= SEQUENCE {
    local       OCTET STRING (SIZE(16)),
    remote      OCTET STRING (SIZE(16)) }

DSCP ::= SEQUENCE {
    copy      BOOLEAN, -- TRUE копировать из внутреннего заголовка
                       -- FALSE не  копировать
    mapping   OCTET STRING OPTIONAL} -- указывает на таблицу, если копирования нет

DF ::= INTEGER {
    clear   (0),
    set     (1),
    copy    (2) }

ProtocolChoice::= CHOICE {
    anyProt  AnyProtocol,              -- для протокола ANY (любой)
    noNext   [0] NoNextLayerProtocol,  -- нет элементов следующего уровня
    oneNext  [1] OneNextLayerProtocol, -- имеется 1 элемент следующего уровня
    twoNext  [2] TwoNextLayerProtocol, -- имеется 2 элемента следующего уровня
    fragment FragmentNoNext }          -- нет информации следующего уровня

AnyProtocol ::= SEQUENCE {
    id          INTEGER (0),    -- протокол ANY (любой)
    nextLayer   AnyNextLayers }

AnyNextLayers ::= SEQUENCE {      -- с любым из
    first       AnyNextLayer,     -- ANY - селектор следующего уровня
    second      AnyNextLayer }    -- ANY - селектор следующего уровня

NoNextLayerProtocol ::= INTEGER (2..254)

FragmentNoNext ::= INTEGER (44)   -- Идентификатор фрагмента

OneNextLayerProtocol ::= SEQUENCE {
    id          INTEGER (1..254),   -- ICMP, MH, ICMPv6
    nextLayer   NextLayerChoice }   -- ICMP Type*256+Code
                                    -- MH   Type*256

TwoNextLayerProtocol ::= SEQUENCE {
    id          INTEGER (2..254),   -- Протокол
    local       NextLayerChoice,    -- Локальный и
    remote      NextLayerChoice }   -- удалённый порт

NextLayerChoice ::= CHOICE {
    any         AnyNextLayer,
    opaque      [0] OpaqueNextLayer,
    range       [1] NextLayerRange }

-- представление ANY в поле следующего уровня
AnyNextLayer ::= SEQUENCE {
    start       INTEGER (0),
    end         INTEGER (65535) }

-- представление OPAQUE в поле следующего уровня
-- соответствует соглашениям IKE
OpaqueNextLayer ::= SEQUENCE {
    start       INTEGER (65535),
    end         INTEGER (0) }

-- диапазон для поля следующего уровня
NextLayerRange ::= SEQUENCE {
    start       INTEGER (0..65535),
```

```
        end         INTEGER (0..65535) }

    -- список адресов IP
    AddrList ::= SEQUENCE {
        v4List      IPv4List OPTIONAL,
        v6List      [0] IPv6List OPTIONAL }

    -- представление адреса IPv4
    IPv4List ::= SEQUENCE OF IPv4Range

    IPv4Range ::= SEQUENCE {    -- закрыто, но не вполне корректно ...
        ipv4Start   OCTET STRING (SIZE (4)),
        ipv4End     OCTET STRING (SIZE (4)) }

    -- представление адреса IPv6
    IPv6List ::= SEQUENCE OF IPv6Range

    IPv6Range ::= SEQUENCE {    -- закрыто, но не вполне корректно ...
        ipv6Start   OCTET STRING (SIZE (16)),
        ipv6End     OCTET STRING (SIZE (16)) }

    END
```

Приложение D: Обоснование обработки фрагментов

Имеется три вопроса, связанных с обработкой (нешифрованных) фрагментов в IPsec и требующих решения:

- отображение отличных от первого исходящих фрагментов на нужную SA (или привязка к нужной записи SPD);
- проверка полномочности принятого фрагмента, отличного от первого, по отношению к SA, через которую он был получен;
- отображений отличных от первого входящих и исходящих фрагментов на нужную запись в SPD/кэше для трафика BYPASS/DISCARD.

Первый и третий вопросы обусловлены необходимостью наличия алгоритма детерминированного отображения трафика на SA (и записи в SPD/кэше). Все три вопроса достаточно важны, поскольку нужна уверенность в том, что отличные от первых фрагменты, проходящие через границу IPsec, не будут приводить к нарушению политики контроля доступа на приёмной или передающей стороне.

D.1. Транспортный режим и фрагменты

Сначала отметим, что SA транспортного режима по определению не могут передавать фрагменты. Это перенесено из RFC 2401, в соответствии с которым SA транспортного режима всегда завершаются на конечных точках[1]. Данное требование имеет фундаментальный характер, поскольку (в наихудшем случае) фрагмент IPv4, обработанный IPsec, может быть снова фрагментирован (в зашифрованном виде) на пути к адресату. Процедура сборки фрагментов IP на приёмной стороне IPsec не сможет отличить фрагменты, созданные до обработки IPsec от фрагментов обработанных IPsec дейтаграмм.

В IPv6 фрагментировать пакеты может только отправитель. Как и для IPv4, реализации IPsec разрешено фрагментировать пакеты туннельного режима после обработки IPsec, поскольку именно реализация IPsec является отправителем с точки зрения (внешнего) заголовка пакета. Однако, в отличие от IPv4, в этом случае можно передавать нешифрованные фрагменты через транспортные SA, поскольку заголовок фрагмента IPv6 размещается после заголовка AH или ESP, что позволяет избавиться от конфликтов при сборке на приёмной стороне. В частности, получатель не будет пытаться собирать фрагменты до выполнения обработки IPsec. Для упрощения данная спецификация запрещает передавать фрагменты через транспортные SA (в том числе и) для трафика IPv6.

Когда транспортные SA используются только конечными системами, проблем с передачей фрагментов не возникает, поскольку предполагается, что конечная система может быть настроена на отказ от фрагментирования IPsec. Для естественной реализации на хосте и представляется оправданным и, как уже было отмечено, RFC 2401 предупреждает, что реализации BITS могут собирать фрагменты до просмотра SA (они в таких случаях будут применять AH или ESP и могут заново фрагментировать пакет после обработки IPsec). Поскольку предполагается, что реализации BITS способны иметь доступ ко всему трафику, исходящему с хоста, даже при наличии множества интерфейсов, данное требование представляется разумным.

В данной спецификации использование транспортного режима приемлемо в тех случаях, когда реализация IPsec не является конечным получателем (например, между двумя SG). В принципе, это создаёт новую возможность для отображения исходящих нешифрованных фрагментов на транспортные SA для обработки IPsec. Однако в тех, достаточно редких, случаях, когда транспортные SA пригодны для такого использования, представляется понятным, что запрет на передачу фрагментов, видимых IPsec (т. е. Тех пакетов, где во внешнем заголовке указано отличное от нуля смещение фрагмента). Например, наложенной сети IP пакеты будут передаваться через транспортные SA в туннелях IP-in-IP и, таким образом, внутренний заголовок должен учитывать фрагментация до обработки IPsec. При передаче через транспортные SA IPsec не будет проверять внутренние заголовки IP для такого трафика и, таким образом, пакет не будет рассматриваться в качестве фрагмента.

D.2. Туннельный режим и фрагменты

Для туннельных SA исходящие фрагменты могут в любой момент поступать для обработки в реализацию IPsec. Необходимость обработки фрагментированных исходящих пакетов может вызывать проблемы, поскольку фрагменты, отличные от первого, не будут содержать полей портов, связанных с протоколом следующего уровня (таким, как TCP, UDP, SCTP). Таким образом, в зависимости от конфигурации SPD для данной реализации IPsec, нешифрованные фрагменты могут вызывать или не вызывать проблемы.

[1] Не на защитных шлюзах, в отличие от туннельных SA. *Прим. перев.*

Например, если SPD требует, чтобы для всего трафика между парой диапазонов адресов обеспечивалась защита IPsec (для этого диапазона адресов не применяются записи SPD BYPASS или DISCARD), передавать отличные от первого фрагменты достаточно просто для через SA, определённых для этого диапазона адресов, поскольку запись SPD предназначена для передачи **всего** трафика между диапазонами адресов. Однако при наличии множества записей SPD, которым может соответствовать фрагмент, указывающих на разные подмножества поля портов (в отличие от ANY), невозможно однозначно отобразить отличные от первого входящие фрагменты на корректную запись. Если разрешена передача фрагментов через транспортные SA для IPv6, описанная проблема будет наблюдаться и в этом контексте.

Эта проблема в значительной мере обусловила определение в RFC 2401 OPAQUE в качестве значения для поля портов. Другой причиной ввода значения OPAQUE послужило тот факт, что поля портов могут быть недоступны до применения IPsec. Например, если хост использует IPsec для своего трафика при поступлении этого трафика на SG поля портов будут зашифрованы. Описанный в RFC 2401 алгоритма нахождения поля next layer protocol также является мотивом использования OPAQUE для восприятия в таких ситуациях зашифрованного поля протокола следующего уровня. Тем не менее, основным назначением OPAQUE является использование этого значения в качестве селектора, соответствующего пакетам, не содержащим поля портов (отличные от первого фрагменты), и пакетам, в которых поля портов уже зашифрованы (в результате использования IPsec). В RFC 2401 содержались неоднозначности в плане использования OPAQUE и ANY (отмечалось, что в некоторых случаях ANY может служить альтернативой OPAQUE).

Мы усилили возможности дополнительного контроля доступа определив оба значения ANY и OPAQUE. Значение OPAQUE можно определить, как соответствующее только тем полям, которые не доступны. ANY определим, как дополнение к OPAQUE, которое соответствует все доступным значениям полей портов. Следовательно, упрощается процедура нахождения поля следующего протокола, поскольку ESP и AH трактуются, как протоколы следующего уровня. В результате значимость зашифрованного поля протокола следующего уровня снижается и не нужно заботиться о шифрованных полях портов. В соответствии со сказанным, значение OPAQUE будет применимо только к фрагментам, не являющимся первыми.

Поскольку мы принимаем приведённые выше определения ANY и OPAQUE, нужно разъяснить использование этих значений в тех случаях, когда указанный протокол не использует поля портов, а в качестве селектора протокола указано значение ANY. Соответственно, если указанное значение протокола используется в качестве селектора и этот протокол не имеет поля портов, селекторы поля порта игнорируются и в качестве значения полей портов **должно** использоваться ANY. В этом контексте значения типа и кода ICMP указываются совместно в одном поле порта (для согласования IKEv2), как и значение типа IPv6 Mobility Header. Если селектор протокола имеет значение ANY, его следует трактовать, как эквивалент задания протокола, для которого не определено поле порта (селекторы портов в этом случае игнорируются и **должны** иметь значение ANY).

D.3. Проблема фрагментов, не являющихся начальными

Для реализаций SG обычной картиной является получение фрагментов от конечных систем, расположенных за SG. Реализации BITW могут сталкиваться с фрагментами от расположенных за ними хостов или шлюзов (как было отмечено выше, реализации хостов и BITS могут не сталкиваться с описанными ниже проблемами). В наихудшем случае фрагменты пакета могут приходить на разные BITW или SG, что не позволяет воспользоваться сборкой фрагментов на таких устройствах. Поэтому в RFC 2401 были описаны общие требования по обработке фрагментов в туннельном режиме для всех реализаций. Однако RFC 2401 не обеспечивает решения для всех случаев. Использование значения OPAQUE в качестве селектора для полей портов (уровень требований **следует** в RFC 2401) разрешается для SA, передающих фрагменты, отличные от первых.

При использовании определённых в RFC 2401 возможностей для SA между двумя реализациями IPsec (SG или BITW), указывающими значение OPAQUE в полях портов все отличные от первых фрагменты, соответствующие по адресам отправителя/получателя (S/D) и протоколам, будут отображаться на такие SA. Начальные фрагменты **не** будут отображаться на эти SA, если мы примем строгое определение OPAQUE. Однако в RFC 2401 не дано детального руководства для таких случаев и, следовательно, может показаться не очевидным, что использование этих возможностей позволяет создавать SA только для фрагментов, не являющихся первыми.

При обсуждении модели SA «только для фрагментов» было отмечено наличие трудно уловимых проблем, которые не были рассмотрены в RFC 2401, - этих проблем удаётся избежать. Например, SA такого типа должны настраиваться так, чтобы обеспечивалось «высшее качество» услуг защиты для любого трафика между указанными адресами S/D (для заданного протокола). Это требуется для того, чтобы для трафика, проходящего через SA «только для фрагментов», не снижался уровень защиты по сравнению с пакетами, для которых не требуется фрагментирования. Возможная проблема заключается в том, что не удастся идентифицировать «высшее качество» услуг защиты, которые могут обеспечиваться между двумя реализациями IPsec, поскольку выбор протоколов, опций и алгоритмов защиты осуществляется из неупорядоченного линейно множества (можно сказать, что BYPASS < AH < ESP с контролем целостности, но ситуация усложнится, если используется множество алгоритмов шифрования или контроля целостности ESP). Поэтому упорядочение таких параметров защиты может иметь лишь локальную значимость.

Однако такая консервативная стратегия может приводить к снижению производительности. Если большая часть трафика, проходящего через реализацию IPsec для данной пары адресов S/D и заданного протокола будет передаваться в обход (bypass), SA «только для фрагментов» для данной паря адресов может вызывать существенный рост объёма трафика, требующего криптографической обработки. Если реализация криптографических механизмов не способна обрабатывать такой объем трафика, могут возникнуть проблемы[1].

Другим предметом для беспокойства являются то, что отличные от первого фрагменты, передаваемые через выделенную SA, могут использоваться для организации атак (перекрытие фрагментов при сборке), когда они будут объединяться с явно допустимыми первыми фрагментами[2]. Этот вопрос легко решается в IPv4 путём проверки величины смещения фрагмента с целью убедиться, что смещение отличных от первого фрагментов не было настолько мало, чтобы в такие фрагменты попадали поля с номерами портов, которые должны находиться в первом фрагменте.

[1]Реализации IPsec, работающие со скоростью среды (или близкой), не подвержены влиянию выбора такой конфигурации SA. Снижение производительности может сильно зависеть от возможностей конкретной реализации.
[2]Этот тип атак предполагает создание поддельных и не оказывает влияния на нормальную фрагментацию.

Поскольку, что минимальный размер MTU для IPv4 составляет 576 байтов, а размер заголовка IP не может превышать 60 байтов, номера портов в любом случае должны оказаться в первом фрагменте. Если мы потребуем, чтобы все отличные от первого фрагменты имели смещение не менее 128, это позволит предотвратить атаки данного типа. Если задача состоит лишь в защите от атак на процесс сборки, проверку достаточно обеспечить лишь на приёмной стороне.

В IPv6 также используется смещение фрагментов, значение которого передаётся в расширенном заголовке фрагментов. Однако расширенные заголовки IPv6 имеют переменный размер и здесь не существует аналогичного значения предельного размера заголовка, которое можно использовать для проверки отличных от первого фрагментов с целью отбрасывания пакетов, связанных с упомянутыми выше атаками. Получателю для обеспечения эффективной защиты требуется поддерживать информацию о состоянии, аналогичную информации о состоянии сборки фрагментов. По этой причине предотвращение атак с использованием обманных номеров портов за счёт проверки смещения фрагментов, отличных от первых, в реализации IPsec (или межсетевом экране), возможно только для IPv4.

Другим поводом для беспокойства является то, что для некоторых топологий и конфигураций SPD эта модель может приводить к неожиданностям в плане контроля доступа. Дело в том, что при создании SA для передачи **всех** (ALL) отличных от начальных фрагментов такие SA могут передавать некий трафик, который в противном случае приходил бы по иному пути (например, через межсетевой экран-посредник[1]) в незашифрованном виде. Но эта проблема возникает только в тех случаях, когда альтернативный путь пропускает отличные от первого фрагменты без промежуточной сборки, что является заведомо неправильным для межсетевых экранов-посредников. Тем не менее, это может вызывать проблемы в некоторых топологиях при определённых условиях для правил SPD и межсетевого экранирования, поэтому администраторам нужно принимать во внимание такую возможность.

Менее серьёзная проблема связана с тем, что отличные от начальных фрагменты, передаваемые через предназначенные только для них SA могут открывать возможность для DoS-атак за счёт использования некорректных первых фрагментов. Это может использоваться для организации атак на хосты, расположенные за устройствами SG или BITW. Однако дополнительный риск от атак этого типа, которые могут быть направлен лишь на хосты, расположенные за устройствами SG или BITW, представляется достаточно малым.

Если мы интерпретируем значение селектора ANY, как включающее в себя значение OPAQUE, одна связь SA со значениями ANY для обоих полей портов будет способна принимать весь трафик, соответствующих селекторам адресов S/D и протокола, - это будет альтернативой использованию значения OPAQUE. Однако использование ANY препятствует созданию множества разных SA между одинаковыми реализациями IPsec для одного набора адресов и протокола. Поэтому описанные варианты не совсем эквивалентны.

В общем случае проблема обслуживания фрагментов возникает лишь в тех ситуациях, где определено множество SA для одного набора селекторов адресов S/D и протокола, но с разными значениями селекторов портов.

D.4. Обход/отбрасывание трафика

Рассмотрим также вопрос обработки отличных от начального фрагментов для записей BYPASS/DISCARD, независимо от обработки SA. Эта задача решается в значительной степени локально по двум причинам:

1) нет возможности координации записей SPD для такого трафика между разными реализациями IPsec, поскольку IKE не используется;

2) многие из таких элементов относятся к трафику, который **не** связан с узлами, использующими IPsec, - следовательно, нет партнёра. IPsec, с которым можно организовать координацию.

Однако этот документ должен обеспечить руководство, в котором в соответствии с провозглашёнными целями, должны быть описаны функции контроля доступа для всего трафика на границах IPsec. Поэтому в документе сказано, что реализации **должны** поддерживать сборку фрагментов трафика BYPASS/DISCARD, когда задано значение поля порта. Реализация также **должна** предоставлять пользователю или администратору возможность принять или отвергнуть такой трафик с использованием соглашений SPD, описанных в разделе 4.4.1. Дело в том, что передача в обход (BYPASS) нешифрованных фрагментов, отличных от начального, которые приходят реализации IPsec, может снижать уровень защиты для трафика IPsec, поступающего по тому же адресу. В качестве примера рассмотрим реализацию IPsec записью SPD, которая обеспечивает защиту IPsec для всего трафика между заданной парой отправитель-получатель с указанным протоколом и номером порта (например, TCP через порт 23 - Telnet). Предположим, что реализация также разрешает обход (BYPASS) для трафика той же пары отправитель-получатель и протокола, но с другим номером порта (например, 119 - NNTP). Атакующий может передать отличный от начального фрагмент (с подставным адресом отправителя), который, будучи переданным в обход, сможет перекрыться с защищённым IPsec трафиком от корректного отправителя с тем же адресом и это приведёт к нарушению целостности защищённого IPsec трафика. Требование проверки передаваемого в обход (BYPASS) трафика с учётом состояния для отличных от начального фрагментов позволяет предотвратить атаки этого типа.

D.5. Просто запретить использование портов?

Предлагалось решение рассмотренной выше проблемы путём запрета использования селекторов порта в туннельном режиме. Однако обсуждение этого вопроса показало, что такой запрет будет слишком жёсткой мерой, поскольку для реализаций в OS и BITS описанной проблемы не возникает. Более того, некоторые члены рабочей группы описали сценарии, в которых использование туннельных SA со значимыми номерами порта в заголовках отличных от начального фрагментов вполне допустимо. Таким образом, задача состоит в определении стратегии решения этой проблемы в контексте BITW и SG. Отметим также, что записи BYPASS/DISCARD в SPD, для которых допускается использование портов, вызывают одинаковые проблемы как в транспортном, так и в туннельном режиме.

Существуют предложения сохранить для межсетевых экранов, находящихся за SG или BITW, возможность контроля доступа на уровне портов для фрагментов. Однако это такой подход представляется неуместным, поскольку в IPsec (например, для данных IKE) имеются ситуации, когда межсетевые экраны отбрасывают все фрагменты. Если многие межсетевые экраны совсем не будут пропускать фрагменты, почему мы должны поступать по-иному в данном случае? Поэтому приведённый здесь анализ отвергает предложение запрета на использование селекторов портов в туннельных SA.

[1] В оригинале «proxy firewall». *Прим. перев.*

D.6. Другие решения

Существует предложение по сборке фрагментов на передающей стороне IPsec, что позволило бы полностью избавиться от проблемы. Это решение остаётся просто невидимым для приёмной стороны и, таким образом, может быть полностью реализовано локально.

Более изощренным вариантом является организация и поддержка минимальной информации о состоянии для каждого начального фрагмента, которая позволила бы сопоставить отличные от начального фрагменты с корректной SA или записью SPD/кэша. Этот вариант предполагает расширение современной (и предшествующей) модели обработки. Реализация IPsec будет перехватывать все фрагменты, считывать поля IP-адресов отправителя и получателя, протокола, идентификатора пакета и номеров портов, а потом использовать эти данные для отображения отличных от начального фрагментов на SA, которые задают поля портов. При реализации этой модели получатель должен будет поддерживать эквивалентную схему, поскольку он тоже должен будет убедиться, что полученные фрагменты согласуются со значениями селекторов SA. Отличный от начального фрагмент, полученный раньше начального, может быть кэширован или отброшен.

Недостатком обоих рассмотренных вариантов является то, что они работают не во всех случаях. Когда устройство BITW или SG работают в топологии, позволяющей обработку некоторых фрагментов пакета на других SG или BITW, не будет гарантии поступления всех фрагментов на одно устройство IPsec. Могут также возникать проблемы при обработке. Если отправитель кэширует отличные от начального фрагменты, пока не прибудет соответствующий начальный фрагмент, могут возникать проблемы с буферизацией (особенно при высоких скоростях). Если отличные от начального фрагменты будут отбрасываться вместо кэширования, не будет даже гарантий, прохождения, поскольку при повторной передаче в пакетах будут другие значения идентификаторов, не соответствующие изначальным. В любом случае потребуются специальные процедуры для решения вопроса о возможности удаления информации о состоянии фрагментов, что приведёт к усложнению системы. Тем не менее это решение вполне для ряда достаточно распространённых топологий.

Рабочая группа отказалась от прежнего соглашения в части создания SA для передачи отличных от начального фрагментов, которое неявно поддерживалось в модели RFC 2401 за счёт использования значения OPAQUE в поле порта, но не было указано в RFC 2401 явно. (Отвергнутое) соглашение предлагает каждый фрагмент, отличный от начального, трактовать, как протокол 44 (идентификатор протокола в заголовке IPv6) на передающей и приёмной стороне. Это позволяет унифицировать обработку фрагментов IPv4 и IPv6, но не даёт полного решения проблемы и даже не решает вопроса обработки фрагментов для отбрасываемого и передаваемого в обход (BYPASS/DISCARD) трафика. С учётом проблемы атак перекрывающимися фрагментами в IPv6 эта стратегия не представляется эффективной.

D.7. Согласованность

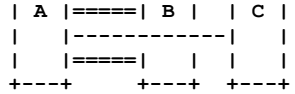
Ранее рабочая группа согласилась с разрешением реализациям IPsec для BITS, BITW или SG выполнять фрагментацию до обработки IPsec. Если такая фрагментация выполняется после поиска SA на стороне отправителя, проблемы отображения на корректную SA не возникает. Но для получателя сохраняется необходимость проверки соответствия отличных от начального фрагментов связи SA, через которую они были получены. Поскольку начальный фрагмент может быть потерян в пути, у получателя могут возникать все перечисленные выше проблемы. Таким образом, если мы будем поддерживать принятое ранее решение, нужно будет сказать, как вести себя получателю отличных от начального фрагментов.

D.8. Заключение

Не существует простого, однотипного решения для всех случаев. Разные варианты по-разному работают в различных контекстах. Этот документ предлагает 3 варианта - один **обязательный** и два **возможных**. В будущем, если сообщество наберет достаточно опыта использования возможных вариантов, их статус может быть повышен до следует или должно, а также могут быть предложены другие решения.

Приложение E: Пример поддержки вложенных SA через записи SPD и таблицы пересылки

В этом приложении даётся пример настройки SPD и таблиц пересылки для поддержки вложенной пары SA в соответствии с новой моделью обработки. Для простоты в этом примере предполагается лишь одна SPD-I.



Задачей является поддержка транспортной SA от A к C, передаваемой через туннельную SA от A к B. В качестве A может использоваться, например, переносный компьютер, подключенный к сети Internet, B может быть межсетевым экраном, защищающим корпоративную сеть, а C - сервером в корпоративной сети, которому требуется сквозная проверка подлинности трафика от A.

SPD содержит записи в форме, показанное в таблице.

Правило	Локальн.	Удалён.	Протокол следующего уровня	Действие
1	C	A	ESP	BYPASS
2	A	C	ICMP,ESP	PROTECT(ESP, туннель, целостность и конфиденциальность)
3	A	C	ANY	PROTECT(ESP, транспорт, только целостность)
4	A	B	ICMP,IKE	BYPASS

На незащищённой стороне A таблица пересылки организована так, что исходящие пакеты, адресованные C возвращаются на защищённую сторону. Таблица пересылки на защищённой стороне A организована так, что входящие пакеты ESP возвращаются на незащищённую сторону. Таблицы пересылки A показаны ниже.

Таблица пересылки на незащищённой стороне.

Правило	Локальн.	Удалён.	Протокол следующего уровня	Действие
1	A	C	ANY	Петля с возвратом на защищённую сторону
2	A	B	ANY	Пересылка B

Таблица пересылки на защищённой стороне.

Правило	Локальн.	Удалён.	Протокол следующего уровня	Действие
1	A	C	ESP	Петля с возвратом на незащищённую сторону

Пакет TCP идущий от A к C будет соответствовать правилу 3 в SPD и к нему будет применяться ESP транспортного режима. Таблица пересылки на незащищённой стороне будет возвращать пакет назад. Пакет сравнивается с SPD-I (см. Рисунок 2), констатируется соответствие правилу 1 и пакет передаётся в обход (BYPASS). Пакет трактуется, как исходящий и сравнивается с SPD в третий раз. Сейчас он будет соответствовать правиле 2 и к нему будет применяться ESP туннельного режима. В этом случае таблица пересылки уже не будет возвращать пакет, поскольку его получателем указан B, следовательно, пакет отправится в сеть.

Входящий пакет TCP от C к A будет «завернут» в два заголовка ESP - внутренний (ESP туннельного режима) будет указывать в качестве отправителя B, а внешний (ESP транспортного режима) - C. По прибытии на A пакет будет отображён на SA на основе значения SPI, после чего внешний заголовок будет удалён, а пакет - расшифрован и проверен на целостность. Далее пакет будет проверен на соответствие селекторам SAD для этой SA, которые будут указывать C в качестве отправителя пакета, A - в качестве получателя (правила 2 в SPD). Таблица пересылки на защищённой стороне будет возвращать пакет на незащищённую сторону на основе адресов и протокола следующего уровня (ESP), показывающих вложенность. Пакет сравнивается с SPD-O (см. Рисунок 3) и обнаруживается соответствие правилу 1 в SPD, поэтому пакет передаётся в обход (BYPASS). Пакет отображается на SA на основе значения SPI, выполняется проверка целостности и сравнение с селекторами SAD полученными из правила 3 в SPD. После этого функция пересылки передаст пакет на следующий уровень, поскольку он не является пакетом ESP.

Литература

Нормативные документы

[BBCDWW98] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Service", RFC 2475[1], December 1998.

[Bra97] Bradner, S., "Key words for use in RFCs to Indicate Requirement Level", BCP 14, RFC 2119, March 1997.

[CD98] Conta, A. and S. Deering, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 2463, December 1998.

[DH98] Deering, S., and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[Eas05] 3rd Eastlake, D., "Cryptographic Algorithm Implementation Requirements For Encapsulating Security Payload (ESP) and Authentication Header (AH)", RFC 4305, December 2005.

[HarCar98] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[Kau05] Kaufman, C., Ed., "The Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005.

[Ken05a] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

[Ken05b] Kent, S., "IP Authentication Header", RFC 4302, December 2005.

[MD90] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, November 1990.

[Mobip] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support in IPv6", RFC 3775, June 2004.

[Pos81a] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[Pos81b] Postel, J., "Internet Control Message Protocol", RFC 792, September 1981.

[Sch05] Schiller, J., "Cryptographic Algorithms for use in the Internet Key Exchange Version 2 (IKEv2)", RFC 4307, December 2005.

[WaKiHo97] Wahl, M., Kille, S., and T. Howes, "Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names", RFC 2253, December 1997.

Дополнительная литература

[CoSa04] Condell, M., and L. Sanchez, "On the Deterministic Enforcement of Un-ordered Security Policies", BBN Technical Memo 1346, March 2004.

[FaLiHaMeTr00] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

[Gro02] Grossman, D., "New Terminology and Clarifications for Diffserv", RFC 3260, April 2002.

[HC03] Holbrook, H. and B. Cain, "Source Specific Multicast for IP", Work in Progress, November 3, 2002.

[HA94] Haller, N. and R. Atkinson, "On Internet Authentication", RFC 1704, October 1994.

[NiBlBaBL98] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[Per96] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

[RaFlBl01] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, September 2001.

[1]Этот документ частично изменён RFC 3260. *Прим. перев.*

[RFC2401] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[RFC2983] Black, D., "Differentiated Services and Tunnels", RFC 2983, October 2000.

[RFC3547] Baugher, M., Weis, B., Hardjono, T., and H. Harney, "The Group Domain of Interpretation", RFC 3547, July 2003.

[RFC3740] Hardjono, T. and B. Weis, "The Multicast Group Security Architecture", RFC 3740, March 2004.

[RaCoCaDe04] Rajahalme, J., Conta, A., Carpenter, B., and S. Deering, "IPv6 Flow Label Specification", RFC 3697, March 2004.

[Sch94] Schneier, B., Applied Cryptography, Section 8.6, John Wiley & Sons, New York, NY, 1994.

[Shi00] Shirey, R., "Internet Security Glossary", RFC 2828, May 2000.

[SMPT01] Shacham, A., Monsour, B., Pereira, R., and M. Thomas, "IP Payload Compression Protocol (IPComp)", RFC 3173, September 2001.

[ToEgWa04] Touch, J., Eggert, L., and Y. Wang, "Use of IPsec Transport Mode for Dynamic Routing", RFC 3884, September 2004.

[VK83] V.L. Voydock & S.T. Kent, "Security Mechanisms in High-level Networks", ACM Computing Surveys, Vol. 15, No. 2, June 1983.

Адреса авторов

Stephen Kent
BBN Technologies
10 Moulton Street
Cambridge, MA 02138
USA
Phone: +1 (617) 873-3988
EMail: kent@bbn.com

Karen Seo
BBN Technologies
10 Moulton Street
Cambridge, MA 02138
USA
Phone: +1 (617) 873-3152
EMail: kseo@bbn.com

Перевод на русский язык
Николай Малых
nmalykh@protokols.ru

Полное заявление авторских прав

Интеллектуальная собственность

Подтверждение

Финансирование функций RFC Editor обеспечено Internet Society.