Internet Engineering Task Force (IETF)
Request for Comments: 9063
Obsoletes: 4423
Category: Informational
ISSN: 2070-1721

R. Moskowitz, Ed.
HTT Consulting
M. Komu
Ericsson

July 2021

Host Identity Protocol Architecture

Архитектура протокола отождествления хостов

Аннотация

В этом документе описано пространство имён отождествления хостов (Host Identity), которое обеспечивает криптографическое пространство имён для приложений, протокол отождествления хоста (Host Identity Protocol или HIP), размещаемый между сетевым[1] и транспортным уровнем, который поддерживает мобильность конечных хостов, многодомность и работу через NAT. В документе рассмотрены основы используемых в настоящее время пространств имён с их преимуществами и недостатками, а также их дополнение пространством HI. Определены роли пространства имён отождествления хостов в протоколах.

Этот документ отменяет RFC 4423 и решает проблемы, отмеченные IESG, в частности, вопрос гибкости шифрования. В разделе 11. Вопросы безопасности рассмотрены меры предотвращения лавинных атак (flooding), применение идентификаторов в списках контроля доступа, слабые типы идентификаторов и доверие при первом применении. Документ включает в себя уроки, извлечённые из реализации RFC 7401, и идёт дальше в разъяснении работы HIP как защищённого сигнального канала.

Статус документа

Документ относится к категории информационных и не задаёт стандартов Internet.

Документ является результатом работы IETF[2] и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG[3]. Дополнительную информацию о стандартах Internet можно найти в разделе 2 в RFC 7841.

Информацию о текущем статусе документа, ошибках и способах обратной связи можно найти по ссылке https://www.rfc-editor.org/info/rfc9063.

Оглавление

[1] В оригинале применяется термин internetworking - межсетевой.
[2] Internet Engineering Task Force - комиссия по решению инженерных задач Internet.
[3] Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

1. Введение

В Internet имеется два важных пространства имён - адреса IP и доменные имена DNS. Эти пространства включают набор функций и абстракций, обеспечивающих современное состояние Internet. Пространства также имеют ряд слабостей. По сути, эти пространства являются основой сети и мы пытаемся все делать в них. Семантическая перегрузка и функциональные расширения существенно усложнили эти пространства имён.

Предлагаемое пространство имён отождествления хостов также является глобальными занимает место между пространствами IP и DNS. Концептуально это пространство имён. относится к вычислительной платформе и такая платформа может иметь несколько идентификаторов (поскольку платформа может по разному идентифицировать себя разным партнёрам). Пространство имён. отождествления хостов состоит из идентификаторов хостов (Host Identifier или HI). Для каждого отождествления применяется в точности один идентификатор HI (хотя могут возникать переходные периоды, например, в момент замены ключей, когда могут быть активны несколько идентификаторов). Хотя далее в тексте обсуждаются некриптографические идентификаторы, архитектура фокусируется на криптографических идентификаторах хостов (HI). В частности, HI является открытым ключом асимметричной пары. Каждое отождествление хоста однозначно указывает один хост, т. е. два хоста не могут иметь совпадающие Host Identity. Если идентификаторы HI совпадают у двух или более вычислительных платформ, эти платформы являются экземплярами распределенного хоста. Идентификатор HI может быть публичным (например, доступным через DNS) или непубликуемым. В клиентских системах будут применяться оба типа HI.

Имеется незначительное но важное различие между отождествлением (Host Identity) и идентификатором (HI). Отождествление указывает абстрактную сущность, которая идентифицируется, а идентификатор - конкретную последовательность битов, используемую в процессе идентификации.

Хотя HI могут применяться во многих системах проверки подлинности (аутентификации), таких как IKEv2 [RFC7296], представленная архитектура задаёт новый протокол, названный протоколом отождествления хоста (Host Identity Protocol или HIP), и криптографический обмен, названный базовым обменом HIP (см. 6. Плоскость управления. HIP обеспечивает ограниченные варианты доверия между системами, повышает уровни мобильности, многодомности и динамической смены адресов IP, помогая в трансляции и изменении протоколов, а также снижая возможности организации некоторых DoS[1]-атак.

При использовании HIP фактический трафик данных между двумя хостами HIP обычно (но не обязательно) защищается с помощью ESP[2] [RFC7402]. Отождествления хостов применяются для создания защищённых связей ESP (Security Association или SA) и аутентификации хостов. При использовании ESP фактические данные пакетов IP не отличаются от передаваемых в обычных пакетах IP с защитой ESP.

С момента публикации [RFC4423] было получено много информации о HIP [RFC6538]. Этот документ расширяет отождествление хоста за пределы первоначального применения для обеспечения связности IP и защиты, с целью предоставления общей защищённой сигнализации между хостами на уровне любого протокола. Сигналы могут организовать защищённую связь между хостами или просто передавать информацию внутри канала.

[1]Denial-of-service - отказ в обслуживании.
[2]Encapsulating Security Payload - инкапсуляция данных защиты.

2. Терминология

2.1. Общие с другими документами термины

Таблица 1.

Термин	*Определение*
Public key Открытый ключ	Открытый ключ асимметричной криптографической пары ключей. Применяется в качестве доступного идентификатора для криптографической проверки подлинности. Открытость в данном случае трактуется в диапазоне от «известно партнёру» до «известно всем».
Private key Секретный ключ	Секретный ключ асимметричной криптографической пары ключей. Предполагается, что этот ключ известен лишь стороне, идентифицируемой соответствующим открытым ключом. Применяется идентифицированной стороной для подтверждения своей подлинности партнёрам.
Public key pair Пара с открытым ключом	Асимметричная пара криптографических ключей, содержащая открытый и секретный ключ. Например, это может быть пара ключей Rivest-Shamir-Adleman (RSA), Digital Signature Algorithm (DSA), Elliptic Curve DSA (ECDSA).
Endpoint Конечная точка	Взаимодействующая сущность (элемент). В силу исторических причин в этом документе в качестве (близкого) синонима используется термин «вычислительная платформа».

2.2. Термины, относящиеся к документам HIP

Следует отметить, что многие термины в этом документе являются тавтологическими, самоопределяющими или содержащими циклические ссылки на другие термины. Это связано с лаконичностью определений. Более подробные объяснения приведены в тексте документа и базовой спецификации [RFC7401].

Таблица 2.

Термин	*Определение*
Computing platform Вычислительная платформа	Элемент, способный к взаимодействию и расчётам, например, компьютер. См. определение Endpoint выше.
HIP base exchange Базовый обмен HIP	Криптографический протокол (см. также раздел 6).
HIP packet Пакет HIP	Пакет IP, содержащий сообщение HIP.
Host Identity Отождествление хоста	Абстрактная концепция, связанная с вычислительной платформе. См. Host Identifier ниже.
Host Identifier Идентификатор хоста	Открытый ключ, служащий именем для отождествления хоста (Host Identity).
Host Identity namespace Пространство имён отождествления хостов	Пространство имён, содержащее все возможные HI.
Host Identity Protocol Протокол отождествления хоста	Протокол, используемый для передачи и аутентификации HI и другой информации.
Host Identity Hash Хэш отождествления хоста	Криптографический хэш, применяемый для создания HIT из HI.
Host Identity Tag Тег отождествления хоста	128-битовый блок данных, создаваемый применением криптографического хэширования к HI, с битами идентификации метода хэширования.
Local Scope Identifier Локальный идентификатор	32-битовый блок данных, обозначающий отождествление хоста.
Public Host Identifier and Identity Публичный идентификатор и отождествление хоста	Опубликованный или общедоступный идентификатор HI, служащий публичным именем для отождествления хоста, и соответствующее отождествление.
Unpublished Host Identifier and Identity Неопубликованный идентификатор и отождествление хоста	Идентификатор HI, не размещаемый в общедоступном репозитории, и соответствующее отождествление хоста. Неопубликованные отождествления хостов обычно будут иметь короткий срок действия с частой заменой и возможно однократным применением.
Rendezvous Mechanism Механизм встречи (рандеву)	Механизм, используемый для нахождения мобильных хостов по их HIT.

3. Основы

Тремя основными компонентами Internet являются вычислительные платформы (конечные системы), инфраструктура транспортировки пакетов (межсетевое взаимодействие) и службы (приложения). Internet служит для предоставления услуг людям и роботизированным системам («кремниевым людям», если угодно). Все упомянутые компоненты нужно именовать для обеспечения между ними расширяемого взаимодействия. В этом документе рассматривается именование вычислительных платформ и элементов доставки пакетов.

Для этих элементов (платформ) в Internet применяется два основных пространства имён - IP-адреса и доменные имена. Система доменных имён обеспечивает иерархическое выделение имён для некоторых вычислительных платформ и служб. Каждый уровень иерархии получает полномочия от вышележащего уровня и в доменных именах нет анонимности. Примерами доменных имён являются адреса Email, HTTP, SIP.

Пространство адресов IP перегружено их применением для интерфейсов (L3) и конечных точек (связанная с конечной точкой часть L3 и уровень L4). В части именования интерфейсов адреса IP иногда называют «локаторами» (locator) и они служат конечными точками в топологии маршрутизации.

IP-адреса являются числами, назначаемыми сетевым интерфейсам, и обычно применяются лишь при подключении интерфейса к сети. Изначально адреса IP служили для долгосрочной идентификации. Сегодня огромное число интерфейсов используют временные и/или неуникальные адреса IP, т. е. интерфейс получает адрес IP при подключении к сети.

В современной сети Internet транспортные уровни неразрывно связаны с адресами IP и не могут развиваться отдельно. На разработку IPng существенно повлиял отказ от создания соответствующего транспорта TCPng.

Используемым пространствам имён присущи три крупных недостатка. Во-первых, организация начального контакта и поддержка потока данных между двумя хостами могут осложняться при использовании временных и частных адресов. Во-вторых, не обеспечивается согласованной и доверенной защиты конфиденциальности. В-третьих, не обеспечивается проверка подлинности систем и дейтаграмм. Все эти недостатки обусловлены используемым сегодня именованием вычислительных платформ.

3.1. Пространство имён для вычислительных платформ

Независимое пространство имён для вычислительных платформ может применяться для сквозных (end-to-end) операций независимо от развития сетевого уровня и между разными сетевыми уровнями. Это позволит быстро менять адреса на сетевом уровне при перемещении, переносе или переадресации.

Если пространство имён вычислительных платформ организовать на основе криптографии с открытым ключом, его можно будет применять также для услуг аутентификации. При локальном создании пространств имён без регистрации они могут обеспечивать анонимность.

Ниже приведены характеристики пространства имён (для вычислительных платформ) и имён в них.

- Пространство имён следует применять на уровне «ядра» или стека IP. Стек IP размещается между приложениями и инфраструктурой доставки пакетов.
- Пространству следует полностью отделять (меж)сетевой уровень от вышележащих уровней. Именам следует заменять собой все вхождения адресов IP внутри приложений (как в блоке управления транспортом Transport Control Block или TCB). Замена может быть выполнена незаметно для унаследованных приложений с помощью локальных идентификаторов (Local Scope Identifier или LSI) и HIT, совместимых с адресами IPv4 и IPv6 [RFC5338]. Однако понимающие HIP приложения потребуется несколько изменить, например, в расширениях API для HIP [RFC6317].
- Для введения пространства имён не следует требовать какой-либо административной инфраструктуры. Развёртывание должно выполняться снизу вверх попарно.
- Для имён следует использовать представление фиксированного размера для простоты включения в заголовки дейтаграмм и имеющиеся программные интерфейсы (например, TCB).
- Пространству имён следует быть доступным при использовании в протоколах. Это связано в основном с размером пакетов, а также вычислениями.
- По возможности следует избегать конфликтов имён. Можно использовать математический «парадокс дней рождения» для оценки вероятности конфликта в данной популяции и хэш-пространстве. В общем случае для случайного хэш-пространства размером n битов конфликт предполагается после примерно $1,2*sqrt(2^n)$ хэш-значений. Для 64 это составит около 4 миллиардов. Размер 64 бита может оказаться слишком малым для крупных популяций, например вероятность конфликта составит 1% для популяции 640M. Для 100 битов (или более) возникновение конфликта ожидается после расчёта 2^{50} (1 квадриллион) значений. При используемом в настоящее время размере 96 битов [RFC7343] можно рассчитать без конфликтов 2^{48} (281 триллион) значений.
- Именам следует иметь локализованную абстракцию, чтобы их можно было применять в имеющихся протоколах и API.
- Должна обеспечиваться возможность локального создания имён. Когда имена не публикуются, это может обеспечивать анонимность за счёт сложности предсказания.
- Пространству имён следует поддерживать услуги проверки подлинности.
- Именам следует быть долгоживущими, но с возможностью замены в любой момент. Это влияет на списки управления доступом - короткий срок действия повышает трудоёмкость поддержки списков или потребует в пространстве имён инфраструктуры для централизованного управления списками доступа.

В этом документе пространство имён, соответствующее приведённым выше характеристикам, называется пространством отождествления хостов - Host Identity. Использование идентификаторов HI требует своего протокольного уровня (Host Identity Protocol) между (меж)сетевым и транспортным уровнем. Имена создаются на основе криптографии с открытым ключом для предоставления услуг аутентификации. При корректной реализации можно обеспечить соответствие всем требованиям, приведённым выше.

4. Пространство имён отождествления хостов

Имя в пространстве Host Identity - идентификатор хоста (HI) представляет статистически уникальное значение для указания любой системы со стеком IP. Это отождествление обычно связано со стеком IP, но не ограничивается такой связью. Система может иметь множество идентификаторов, некоторые могут быть общеизвестными (well known), а другие - анонимными. Система может самостоятельно отождествлять себя или использовать сторонний аутентификатор, например DNSSEC [RFC4033], Pretty Good Privacy (PGP) или X.509 для «нотариального заверения» своего отождествления в другом пространстве имён.

Теоретически, любое имя, статистически уникальное в глобальном масштабе, может служить в качестве HI. В архитектуре HIP в качестве такого имени выбран открытый ключ пары асимметричных ключей, поскольку им можно управлять самостоятельно и такой ключ сложно подделать. Как указано в спецификации протокола HIP [RFC7401], HI на основе открытых ключей позволяет аутентифицировать пакеты HIP и защитить их от MitM[1]-атак. Поскольку аутентифицированные дейтаграммы обязательны для обеспечения основной защиты HIP от DoS-атак, обмен Diffie-Hellman в базовом обмене HIP использует аутентификацию. Таким образом, на практике поддерживаются лишь HI на базе открытых ключей и аутентифицированные сообщения HIP.

[1]Man-in-the-middle - перехват и изменение пакетов в пути с участием человека.

В этом документе упоминаются некриптографические формы HI и HIP, но следует предпочитать криптографические варианты, поскольку они более защищены. В прошлом исследовались варианты применения некриптографических HI для радио-меток (Radio Frequency IDentification или RFID) в обмене HIP, адаптированном для работы с такими задачами ([urien-rfid], [urien-rfid-draft]).

4.1. Идентификаторы хостов

Отождествление хостов добавляет в протоколы Internet два основных свойства. Первое заключается в развязывании (меж)сетевого и транспортного уровня (5. Новая архитектура стека), которое обеспечивает этим уровням возможность независимого развития, а также может обеспечить сквозные услуги через множество областей межсетевого взаимодействия. Вторым свойством является аутентификация хостов. Поскольку HI является открытым ключом, он может применяться для проверки подлинности в таких протоколах защиты как ESP.

Отождествление в HIP основано на парах из секретного и открытого ключа и представляется открытым ключом. Таким образом, имя, представляющее отождествление хоста в пространстве Host Identity, т. е. HI, является открытым ключом. В некотором смысле отождествление определяется владением секретным ключом. Если секретным ключом владеет несколько узлов, отождествление может считаться распределенным.

Архитектурно в качестве HI можно использовать любое другое соглашение от именовании в Internet, однако некриптографичекие имена следует применять лишь в средах с высоким уровнем доверия и/или малыми рисками. Это могут быть места, где не требуется аутентификация (нет риска подмены хоста) или не нужна защита ESP. Однако для соединённых между собой сетей, охватывающих несколько операционных доменов и сред, где существует риск подмены хостов, использование некриптографических HI вносит существенный риск. Поэтому в текущих документах HIP не задано использование HI, не основанных на открытых ключах. Например, служба Back to My Mac [RFC6281] от Apple очень близка по функциональности к HIP, но основана на некриптографических идентификаторах.

Реальные HI не применяются напрямую на транспортном или сетевом уровне. Соответствующие HI (открытые ключи) могут храниться в DNS или иных каталогах, как указано в этом документе, и могут передаваться в базовом обмене HIP. Другие протоколы применяют тег отождествления хоста (Host Identity Tag или HIT) для представления Host Identity. Другое представления отождествлений хостов - локальный идентификатор (Local Scope Identifier или LSI) также можно применять в протоколах и API.

4.2. Хэш отождествления хоста (HIH)

Хэш отождествления хоста (Host Identity Hash или HIH) - это криптографических алгоритм, служащий для создания HIT из HI, а также применяемый в HIP для простоты и единообразия. Два хоста в обмене HIP могут применять разные алгоритмы.

Требуется несколько HIH внутри HIP для решения вопросов создания перемещающихся целей и разрешения возможных конфликтов хэш-значений. Это существенно усложняет протокол HIP и ослабляет возможность атак на понижение в HIP [RFC7401].

4.3. Тег отождествления хоста (HIT)

Тег отождествления хоста (Host Identity Tag или HIT) - это 128-битовое представление Host Identity. Благодаря размеру, тег подходит для использования в имеющихся API сокетов вместо адреса IPv6 (например, sin6_addr в структуре sockaddr_in6) без внесения изменений в приложения. Тег создаётся из HIH, префикса IPv6 [RFC7343] и идентификатора хэш-функции. Имеется два преимущества использования в протоколах HIT вместо HI. Первым является фиксированный размер, что упрощает кодирование протоколов и более эффективное управление размером пакетов. Во-вторых, тег представляет отождествление протоколу в согласованном формате независимо от используемых криптоалгоритмов.

По сути, HIT представляет собой хэш открытого ключа, поэтому не его создание влияют два алгоритма, используемые для открытого ключа HI и HIH. Эти алгоритмы кодируются в битовом представлении HIT. Поскольку две взаимодействующих стороны могут поддерживать разные алгоритмы, в [RFC7401] определён минимальный набор для надёжного взаимодействия. Для расширения возможностей взаимодействия отвечающая сторона (Responder) может сохранять свои ключи в записях DNS и инициатор может связать HIT адресата с соответствующими HIT источника по совпадению HIH.

В пакетах HIP идентификаторы HIT указывают отправителя и получателя пакетов, поэтому значениям HIT следуют быть уникальными во всем пространстве IP, где они применяются. В очень редких случаях, когда одно значение HIT сопоставляется с несколькими Host Identity, идентификаторы HI (открытые ключи) будут обеспечивать различие. Если для данного узла имеется несколько открытых ключей, HIT служит подсказкой для выбора нужного ключа.

Хотя случайные конфликты, когда одно значение HIT сопоставляется с несколькими Host Identity, могут возникать редко, атакующий может с помощью перебора или использования слабости алгоритма, найти второй хэш Host Identity с тем же HIT. Этот тип атак называют атаками на прообраз (preimage attack) и устойчивость к нахождению второго идентификатора HI (открытого ключа), который хэшируется в то же значение HIT называется устойчивостью ко второму прообразу. Такая устойчивость в HIP основана на силе алгоритма хэширования и выходном размере хэш-функции. Для HIPv2 [RFC7401] устойчивость составляет 96 битов (меньше 128-битового размера адреса IPv6 по причине наличия префикса ORCHID[1] [RFC7343]). Такая устойчивость сочтена достаточной на момент разработки HIP, но её может не хватить для модели угроз предполагаемого развёртывания. Одним из возможных решений может быть расширение использования HIT в развёртывании за счёт идентификаторов HI (и механизмов защищённой привязки HI к HIT) так, что HI становится окончательным основанием. Можно также усложнить атаки с перебором за счёт увеличения сложности расчёта HI, например, с помощью защищенного обнаружения криптографически созданных адресов соседей (Secure Neighbor Discovery Cryptographically Generated Addres) [RFC3972], хотя спецификации HIP до HIPv2 не предоставляют такого механизма. Если при развёртывании не применяются идентификаторы ORCHID (например, в некоторых типах наложенных сетей), можно использовать полный размер 128 битовых адресов IPv6 для HIT.

[1]Overlay Routable Cryptographic Hash Identifier - идентификатор наложенного маршрутизируемого криптографического хэша.

4.4. Идентификатор с локальной областью действия (LSI)

LSI - это 32-битовое локализованное представление Host Identity, которое, благодаря его размеру, можно применять в имеющихся API сокетов вместо адресов IPv4 (например, sin_addr в структуре sockaddr_in), не изменяя приложений. Назначение LSI состоит в облегчении использования HI в имеющихся API для приложений IPv4. Значения LSI не передаются в линию и при передаче приложением данных с использованием пары LSI уровень HIP (или обработчик сокетов) транслирует LSI в соответствующие HIT (и обратно в случае приёма данных). Кроме упрощения связности на основе HIP для традиционных приложений IPv4, идентификаторы LSI полезны в 2 других сценариях [RFC6538].

В первом случае два приложения, поддерживающих лишь IPv4, размещены на двух разных хостах, соединённых через сеть с поддержкой только IPv6. Связность на основе HIP позволяет этим приложениям взаимодействовать, несмотря на различие семейств протоколов приложений и базовой сети. Это обусловлено тем, что уровень HIP транслирует LSI от вышележащих уровней в маршрутизируемые локаторы (адреса) IPv6 перед отправкой пакетов в линию.

Второй случай похож на описанный, но одно из приложений поддерживает лишь IPv6. Здесь имеется два препятствия взаимодействию приложений - разные семейства адресов, используемые двумя приложениями, и невозможность приложения IPv4 обмениваться данными с сетью IPv6. Связность на основе HIP решает эту проблему, транслируя локатор (адрес) входящего пакета в LSI или HIT.

Фактически LSI расширяет возможности взаимодействия IPv6 на сетевом уровне, как описано для первого случая, и на прикладном, как описано для второго. Механизм расширения возможностей взаимодействия не следует применять для отказа от перехода на IPv6, авторы твердо верят в принятие IPv6 и призывают разработчиков переводить имеющиеся приложения с поддержкой лишь IPv4 на использование IPv6. Однако некоторые фирменные приложения с закрытым кодом, поддерживающие лишь IPv4, могут не увидеть света IPv6 и здесь механизм LSI поможет продлить их жизнь даже в сетях, поддерживающих лишь IPv6.

Основным недостатком LSI является локальная значимость идентификаторов. Приложения могут нарушать принципы уровней и передавать между собой LSI в протоколах прикладного уровня. Поскольку LSI действительны лишь в контексте локального хоста, они могут представлять совершенно иной хост при передаче идентификатора другому хосту. Однако следует подчеркнуть, что контекст LSI фактически представляет собой NAT на хосте и не создаёт проблем больше, чем NAT в промежуточных устройствам для IPv4. Иными словами, приложениям, нарушающим принципы уровней, уже препятствуют устройства NAT, развёрнутые повсеместно.

4.5. Сохранение HI в каталогах

Общедоступные HI следует сохранять в DNS, а неопубликованные не следует сохранять нигде, кроме самих взаимодействующих хостов. Для хранения (общедоступных) HI вместе с поддерживаемыми HIH имеется новый тип записи о ресурсах (Resource Record или RR), определённый в расширении HIP DNS [RFC8005].

В дополнение к сохранению HI в DNS их можно хранить в других каталогах, например, LDAP (Lightweight Directory Access Protocol) или PKI (Public Key Infrastructure) [RFC8002]. Кроме того, были успешно использованы [RFC6538] распределенные таблицы хэшей (Distributed Hash Table или DHT) [RFC6537]. Такая практика позволяет применять идентификаторы не только для распознавания хостов.

Некоторые типы приложений могут кэшировать и использовать HI напрямую, а другие опосредованно находить идентификаторы по символьным именам хостов, таким как полные доменные имена (Fully Qualified Domain Name или FQDN), путём просмотра каталогов. Хотя HI могут действовать существенно дольше связанных с ними маршрутизируемых адресов IP, каталоги могут оказаться лучшим подходом для управления сроком действия HI. Например, каталог на основе LDAP или DHT можно применять для опубликованных локально идентификаторов, а DNS лучше подходит для общедоступных.

5. Новая архитектура стека

Одним из способов охарактеризовать Host Identity является сравнение предложенной архитектуры на основе HI с используемой в настоящее время. Как отмечено в отчёте IRTF Name Space Research Group Report [nsrg-report] и, например, документе «Endpoints and Endpoint Names» [chiappa-endpoints], адреса IP в настоящее время играют двойную роль - «локаторов» и идентификаторов конечных точек, т. е. каждый адрес IP указывает топологическое местоположение в Internet, действуя как вектор направления маршрутизации или «локатор», и в то же время именует физический сетевой интерфейс, размещённый в данный момент в точке подключения, выступая как имя конечной точки.

В архитектуре HIP имена конечных точек и «локаторы» отделены одно от другого. Идентификаторы HI указывают конечные точки. Важно понимать, что имена конечных точек, основанные на HI несколько отличаются от имён интерфейсов и доступ к Host Identity может одновременно обеспечиваться через несколько интерфейсов. Различия в привязках логических объектов показаны на рисунке 1, где слева показана текущая архитектура TCP/IP, а справа - архитектура на основе HIP

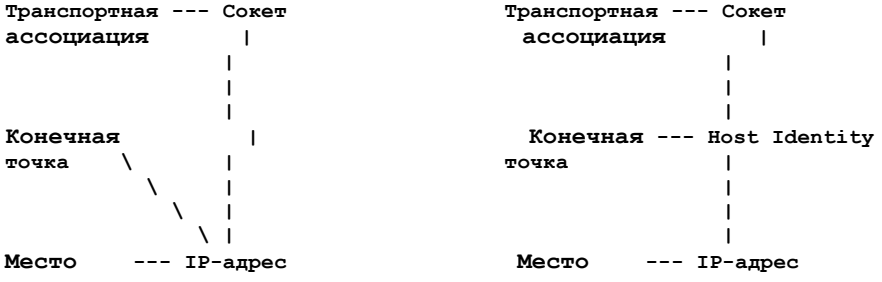

Рисунок 1.

Архитектурно HIP обеспечивает разную привязку протоколов транспортного уровня, т. е. ассоциации транспортного уровня (например, соединения TCP и ассоциации UDP) связаны не с адресами IP. А с идентификаторами HI. На

практике отождествление хостов раскрывается через LSI и HIT для унаследованных приложений и транспортного уровня, чтобы повысить уровень совместимости с имеющимися сетевыми API и стеками протоколов.

Уровень HIP логически размещён как L3,5 между транспортным и сетевым уровнем сетевого стека и действует как «прокладка», использующая LSI или HIT, но не затрагивающая другие данные. Уровень HIP выполняет преобразование двух форм идентификаторов HIP, приходящих от транспортного уровня, в маршрутизируемые адреса IPv4 или IPv6 для сетевого уровня и обратно.

5.1. Множественное отождествление

Хост может иметь множество отождествлений как на клиентской, так и на серверной стороне. Это вызывает некоторые дополнительные проблемы, рассматриваемые в этом параграфе.

Из соображений безопасности идея дублирования Host Identity на нескольких хостах может быть неудачной, поскольку компрометация одного хоста портит отождествление других. Управления машинами с одинаковыми Host Identity тоже может вызывать сложности, поэтому каждому хосту рекомендуется применять уникальное отождествление.

На стороне сервера для распределения нагрузки лучше применять DNS, нежели общее отождествление Host Identity. Можно настроить одну запись FQDN указывающую разные Host Identity. Каждую запись FQDN можно связать с соответствующими «локаторами» или общим «локатором», если серверы используют общий сервер HIP rendezvous (6.3. Механизм встречи) или ретранслятор HIP (6.4. Механизм ретрансляции).

Вместо дублирования отождествлений в HIP реализован специальный (opportunistic) режим, в котором инициатор (Initiator) не учитывает идентификатор отвечающего (Responder) при инициировании обмена ключами и узнает его по завершении обмена. Этот компромисс имеет слабые гарантии защиты, но позволяет избежать публикации HI [komu-leap]. Поскольку многие общедоступные серверы уже используют DNS в качестве каталога, такой подход может оказаться более подходящим, например, для соединений «точка-точка». Следует также отметить, что этот режим нужен на практике при использовании в качестве «локаторов» anycast-адресов IP. Этот режим может применяться вместе с серверами HIP rendezvous или ретрансляторами HIP [komu-diss]. В таких случаях инициатор передаёт сообщение I1 с шаблонным HIT адресата «локатору» сервера HIP rendezvous или ретранслятора. Когда такой сервер обслуживает множество зарегистрированных Responder, он может выбирать HIT адресата, выступая балансировщиком на основе HIP. Однако этот подход остаётся экспериментальным и требует дополнительных исследований.

На стороне клиента хост может иметь несколько Host Identity, например, из соображений приватности или при работе пользователя хоста с разными административными доменами в качестве дополнительной меры защиты. Если на пути между клиентом и сервером имеется понимающее HIP промежуточное устройство, например, межсетевой экран (МСЭ) на основе HIP, пользователю или базовой системе следует аккуратно выбирать нужное отождествление, чтобы предотвратить ненужные препятствия со стороны МСЭ на основе HIP [komu-diss].

Сервер также может иметь несколько Host Identity, например, web-сервер может обслуживать несколько административных доменов. Обычно различия реализуются на основе имени DNS, но для этого может служить и Host Identity. Однако более веской причиной использования нескольких отождествлений являются понимающие HIP МСЭ, которые не способны видеть трафик HTTP внутри шифрованных туннелей IPsec. В этом случае для каждой службы можно настроить своё отождествление, позволяя МСЭ разделять службы одного web-сервера [lindqvist-enterprise].

6. Плоскость управления

HIP разделяет плоскости данных и управления. Два конечных хоста инициализируют плоскость управления с помощью обмена ключами, называемого базовым обменом. В этой процедуре могут помогать инфраструктурные посредники HIP, называемые серверами встреч (rendezvous) или ретрансляторами. При смене адреса IP конечные точки сохраняют связь с плоскостью управления с помощью расширений для мобильности и множественной адресации (multihoming). По завершении работы конечные хосты удаляют плоскость управления и связанные с ней состояния.

6.1. Базовый обмен

Базовым обменом называется процедура обмена ключами, где Initiator и Responder проверяют подлинность друг друга, используя свои открытые ключи. Обычно инициатором является клиентский хост, а отвечающим (Responder) - сервер. Роли используются конечным автоматом реализации HIP и отбрасываются по завершении.

Обмен включает 4 сообщения и создание симметричных ключей для защиты плоскости управления с помощью хэшированных кодов аутентификации сообщений (Hash-based Message Authentication Code или HMAC). Эти ключи могут также служить для защиты плоскости данных, где обычно применяется IPsec ESP [RFC7402], хотя HIP поддерживает и другие протоколы. Плоскости данных и управления удаляются с помощью процедуры закрытия, включающей 2 сообщения.

Кроме того, базовый обмен включает вычислительную задачу (puzzle) [RFC7401], которую должен решить инициатор. Отвечающий выбирает уровень сложности этой задачи, что позволяет ему задерживать запросы новых инициаторов в соответствии с локальной политикой, например, при высокой нагрузке. Задача-головоломка может обеспечивать некоторую защиту от DoS-атак, поскольку механизм позволяет отвечающему не создавать состояния (stateless) до завершений базового обмена [aura-dos]. Головоломки HIP изучались для установившихся атак DDoS [beal-dos] на множестве моделей злоумышленников с изменением сложности задачи (puzzle) [tritilanunt-dos] и эфемерными отождествлениями хостов [komu-mitigation].

6.2. Мобильные и многоадресные хосты

HIP отделяет транспортный уровень от (меж)сетевого и связывает транспортные ассоциации с отождествлением хоста (через HIT или LSI). После начального обмена ключами уровень HIP поддерживает связность на транспортном уровне и потоки данных с использованием расширений для мобильности [RFC8046] и множественных адресов [RFC8047]. Таким образом, HIP может обеспечить некоторый уровень мобильности и поддержки множественных адресов без больших затрат на инфраструктуру. Поддержка мобильности в HIP включает смену адресов IP (любым способом) у любой из сторон. Система считается мобильной, если её адрес IP может меняться динамически по любой причине, такой как переназначение префикса PPP, DHCP, IPv6 или изменение трансляции NAT. Многодомной (многоадресной)

считается система, имеющая одновременно несколько глобально маршрутизируемых адресов IP. HIP связывает адреса IP, если несколько адресов соответствует одному отождествлению хоста. Если один из адресов становится не применимым или появляется более предпочтительный адрес, имеющиеся транспортные ассоциации легко переносятся на другой адрес.

Когда мобильный узел перемещается в процессе обмена данными, смена адреса выполняется достаточно просто - узел передаёт пакет HIP UPDATE для информирования партнёра о новом адресе, а партнёр проверяет доступность мобильного узла по этому адресу. Это позволяет партнёру избежать лавинных атак, описанных в параграфе 11.2.

6.3. Механизм встречи

Организация контакта с перемещающимся мобильным узлом несколько сложней. Для начала обмена HIP узлу-инициатору нужно узнать, как связаться с мобильным узлом. Например, мобильный узел может реализовать Dynamic DNS [RFC2136] для обновления сведений о своей доступности в DNS. Чтобы избежать зависимости от DNS, в HIP имеется своё решение - механизм встречи HIP rendezvous, определённый в [RFC8004].

С помощью расширений HIP rendezvous мобильный узел постоянно обновляет инфраструктуру встреч, используя текущий адрес(а) IP. Мобильные узлы доверяют механизму встречи HIP для должной поддержки сопоставления их HIT с адресом IP. Механизм встречи особенно полезен в случаях, когда предполагается возможность изменения адресов одновременно обоими партнёрами. В этом случае пакеты HIP UPDATE будут «пересекаться» в сети и не попадут к партнёру.

6.4. Механизм ретрансляции

Механизм ретрансляции HIP [RFC9028] является альтернативой HIP rendezvous и более подходит для сетей IPv4 с NAT, поскольку способен пересылать все коммуникации управления и данных для гарантированного прохождения NAT.

6.5. Завершение плоскости управления

Плоскость управления между парой хостов удаляется с использованием защищённого обмена двумя сообщениями, описанного в базовой спецификации [RFC7401]. При удалении плоскости следует удалять и связанные с ней состояния (т. е., ассоциации между хостами).

7. Плоскость данных

Формат инкапсуляции для плоскости данных, служащей для переноса трафика прикладного уровня, может согласовываться динамически в процессе обмена ключами. Например, расширения HICCUPS [RFC6078] определяют один из способов доставки дейтаграмм прикладного уровня непосредственно через плоскость управления HIP с защитой на основе асимметричных ключей. Защищённый транспорт в реальном масштабе времени (Secure Real-time Transport Protocol или SRTP) также рассматривался в качестве протокола инкапсуляции данных [hip-srtp]. Однако более широкое распространение получил метод инкапсуляции защищённых данных (Encapsulated Security Payload или ESP) [RFC7402], использующий симметричные ключи, выведенные в процессе обмена ключами. Защищённые связи ESP (Security Association или SA) обеспечивают защиту конфиденциальности и целостности, причём первую можно отключить в процессе обмена ключами. В будущем могут быть определены иные способы доставки данных прикладного уровня.

ESP SA организуются и завершаются между инициатором и отвечающим хостом. Обычно хосты создают не менее 2 SA, по одной в каждом направлении (от инициатора к отвечающему и обратно). При смене IP-адреса любого из хостов можно использовать расширение HIP для повторного согласования соответствующих SA.

В линии разница при использовании идентификаторов между плоскостями управления и данных HIP состоит в том, что теги HIT включаются во все пакеты управления, но не включаются в пакеты данных при использовании ESP. Вместо этого применяются индексы параметров защиты ESP (Security Parameter Index или SPI), которые действуют как сжатые HIT. Любым промежуточным устройствам с поддержкой HIP (например, HIP МСЭ), заинтересованным в трафике данных на основе ESP, нужно отслеживать идентификаторы плоскостей данных и управления, чтобы связать их между собой.

Поскольку HIP не согласует срок действия SA, этот срок определяется локальными правилами. Реализация HIP должна поддерживать ограничение срока действия лишь на основе достижения максимального порядкового номера (защита от повторов) и тайм-аута SA, возникающего при отсутствии пакетов, принимаемых через SA. Реализации могут поддерживать сроки действия для разных преобразований ESP и других протоколов плоскости данных.

8. HIP и NAT

Передача пакетов между разными областями адресации IP требует изменения адресов в заголовках пакетов IP. Это может происходить, например, при передаче пакета между Internet и приватным пространством адресов или между сетями IPv4 и IPv6. Трансляция адресов обычно обеспечивается устройствами NAT (Network Address Translation) [RFC3022] или NAT-PT (NAT Protocol Translation) [RFC2766].

В среде с идентификацией хостов по адресам IP указание взаимодействующих узлов осложняется при использовании NAT, поскольку пространства приватных адресов перекрываются. Иными словами, два хоста невозможно отличить друг от друга на основе лишь адресов IP. В HIP конечные точки транспортного уровня (приложения) связаны с уникальными отождествлениями хостов вместо перекрывающихся приватных адресов. Это позволяет различать две конечные точки при их размещении в разных приватных областях адресации. В результате адреса IP применяются лишь для маршрутизации и могут свободно меняться устройствами NAT при прохождении пакетов между парой поддерживающих HIP хостов через области с разной приватной адресацией.

Расширения для прохождения NAT в протоколе HIP [RFC9028] могут служить для организации сквозной связности через устройства NAT. Для поддержки базовой совместимости с унаследованными устройствами NAT расширения инкапсулируют плоскости управления и данных HIP в протокол UDP. Расширения определяют механизмы для пересылки двух плоскостей через промежуточный хост, называемый ретранслятором HIP (relay) и процедуры для

организации прямой сквозной связности через NAT. Поскольку это «естественный» режим прохождения через NAT для HIP, можно использовать и другие механизмы работы через NAT, например, Teredo [RFC4380] (см. [varjonen-split]).

Помимо традиционных NAT была разработана и реализована система NAT с поддержкой протокола HIP [ylitalo-spinat]. Для потоков на основе HIP поддерживающий HIP транслятор NAT или NAT-PT отслеживает сопоставления HIT и соответствующих ESP SPI с адресами IP. Система NAT узнает сопоставления HIT и SPI с адресами IP. Множество HIT (и SPI) может отображаться на один адрес IP в системе NAT, что упрощает соединения на интерфейсах NAT с недостаточным числом адресов. NAT может получать большую часть сведений из самих пакетов HIP, однако может потребоваться некоторая настройка конфигурации NAT.

8.1. HIP и контрольная сумма вышележащего уровня

Хост не может узнать, применялись ли адреса из заголовка IP при расчёте контрольной суммы TCP, т. е. невозможно рассчитать корректную контрольную сумму TCP, используя фактические адреса IP из псевдозаголовка, поскольку адрес в принятом пакете может отличаться от указанного передающим хостом. Кроме того, невозможно пересчитать контрольную сумму вышележащего уровня в системах NAT/NAT-PT, поскольку трафик защищён ESP. Поэтому контрольные суммы TCP и UDP рассчитываются с использованием HIT вместо адресов IP в псевдозаголовке. Используется лишь формат псевдозаголовков IPv6. Это обеспечивает трансляцию протоколов IPv4 и IPv6.

9. Групповая адресация

Был опубликован ряд исследований групповой адресации на основе HIP, включая [shields-hip], [zhu-hip], [amir-hip], [kovacshazi-host], [zhu-secure]. В частности, так называемые фильтры Блума, позволяющие сжимать несколько меток в небольшие структуры данных, могут быть многообещающим шагом вперёд [sarela-bloom]. Однако разные схемы не были приняты рабочей группой HIP (и исследовательской группой HIP в IRTF), поэтому детали здесь не приведены.

10. Правила HIP

Каждый хост должен поддерживать множество переменных, влияющих на обмен HIP. Всем реализациям HIP следует поддерживать по меньшей мере 2 HI, один из которых публикуется в DNS или иной службе каталогов, а второй не публикуется и служит для анонимного использования (предполагается его частая смена для предотвращения сопоставлений и отслеживания). Хотя неопубликованные HI редко применяются в качестве Responder HI, их часто используют инициаторы. Как указано в [RFC7401], «все реализации HIP **должны** поддерживать одновременно более одного HI, и по меньшей мере один идентификатор **следует** резервировать для анонимного использования» и «**рекомендуется** поддерживать более двух HI». Это ставит перед системами и пользователями вопрос выбора HI, раскрываемого при организации новой сессии.

Специальный (Opportunistic) режим, где инициатор начинает обмен HIP, не зная Responder HI, является компромиссом в части безопасности. Этот режим позволяет инициатору узнать отождествление отвечающего в процессе взаимодействия, а не из внешнего каталога, но это делает его уязвимым для MitM-атак. Этот режим можно применять для регистрации основанных на HIP служб [RFC8003] (т. е. использующих HIP для внутренних целей) или на уровне приложений [komu-leap]. Соображения безопасности, особенно во втором случае, требуют вовлечения пользователя в решение вопроса о восприятии отождествления отвечающего, подобно приглашению в протоколе SSH (Secure Shell) при первом подключении к серверу [pham-leap]. На практике этом может быть реализовано в МСЭ на конечных хостах в случае унаследованных приложений [karvonen-usable] или в естественных API для HIP [RFC6317] в приложениях с поддержкой HIP.

В [RFC7401] сказано: «Инициаторы **могут** использовать разные HI для разных отвечающих, чтобы обеспечить базовую приватность. Применение таких приватных HI повторно для одного отвечающего и срок действия HI определяется локальными правилами и зависит от требований приватности инициатора». Согласно [RFC7401]: «Responder, отвечающий лишь выбранным инициаторам, требует наличия списка управления доступом (Access Control List или ACL), представляющего хосты, от которых он воспринимает базовый обмен HIP, предпочтительный формат транспорта и локальные сроки действия. **Следует** поддерживать шаблонные записи в таких ACL, а также для отвечающих, которые предоставляют общедоступные или анонимные услуги.

11. Вопросы безопасности

В этом разделе рассматриваются некоторые вопросы и решения, связанные с безопасностью архитектуры HIP.

11.1. MitM-атаки

Протокол HIP использует парадигму отождествления хоста для защищённой аутентификации хостов и обеспечения быстрого обмена ключами для ESP. HIP также пытается ограничить раскрытие хоста для различных DoS и MitM-атак. При этом сам протокол HIP подвержен атакам DoS и MitM, которые могут нанести большой ущерб работе хоста.

DoS-атаки с истощением ресурсов используют «дороговизну» установки состояния протокола на отвечающей стороне по сравнению с «дешевизной» у инициатора. HIP позволяет отвечающему повысить стоимость запуска состояния на стороне инициатора и пытается снизить расходы отвечающего. Это делается за счёт запуска обмена Diffie-Hellman отвечающим вместо инициатора, что ведёт к базовому обмену HIP из 4 пакетов. Первый пакет, отправленный отвечающим, может быть создан заранее для дополнительного снижения затрат. Этот пакет также включает вычислительную задачу (puzzle), которая может служить для дополнительной задержки инициатора, например, при перегрузке отвечающего. Детали процесса приведены в спецификации базового обмена [RFC7401].

Защититься от MitM-атак сложно без сторонней аутентификации. Опытный атакующий может легко обработать все части базового обмена HIP, но протокол HIP опосредованно обеспечивает дополнительную защиту от MitM-атак. Если значение Responder HI получено из подписанной зоны DNS или иным защищённым способом, инициатор может применить это для аутентификации подписанных пакетов HIP. Если Initiator HI хранится в защищённой зоне DNS, отвечающий может найти идентификатор и проверить подписанные пакеты HIP. Однако инициатор может применять неопубликованный HI, осознанно принимая риск MitM-атаки. Responder может отказаться воспринимать обмен HIP с инициаторами, использующими неизвестный идентификатор HI.

Другие типы MitM-атак на HIP могут быть организованы с использованием сообщений ICMP, сообщающих о проблемах. В качестве общей рекомендации можно указать рассмотрение сообщений ICMP как ненадёжных «советов» и реагирование на них лишь после некоторой паузы (тайм-аут). Точные сценарии атак и меры противодействия описаны более подробно в спецификации базового обмена [RFC7401].

В MitM-атаке может быть предпринята попытка использования старых сообщений I1 или R1 с более слабыми криптографическими алгоритмами, как указано в параграфе 4.1.4 [RFC7401]. Базовый обмен был усилен для борьбы с такими атаками путём перезапуска при обнаружении атаки. В худшем случае это приведёт лишь к бесконечному повтору базового обмена или его прерыванию после некоторого числа попыток. Недостатком этого является 6-этапный базовый обмен, который может показаться неудачным решением. Однако такое возможно лишь при атаке, которая может быть обработана (чтобы повторять её не было смысла), поэтому предполагается, что последующие сообщения не представляют угрозы безопасности. Поскольку MitM-атаки не позволяют снизить версию, их можно считать лишь помехой. Таким образом, базовый обмен будет включать обычно лишь 4 пакета даже при готовности реализации к защите от понижения версии.

В протоколе HIP защищённые связи SA для ESP индексируются по SPI, адрес отправителя всегда игнорируется, а адрес получателя также можно проигнорировать. Поэтому при включённом HIP работа ESP не зависит от адресов IP. Это может показаться упрощением для организации атак, но ESP с защитой от повторного использования (replay) уже обеспечивает высокий уровень безопасности и удаление адреса IP из проверки не увеличивает раскрытие ESP для DoS-атак.

11.2. Защита от лавинных атак

Хотя идея информирования о смене адреса путём простой отправки пакетов с новым адресом источника кажется привлекательной, она недостаточно безопасна. Даже если HIP ни в чем не полагается на адрес отправителя (после завершения базового обмена), представляется необходимой проверка доступности мобильного узла по новому адресу до отправки туда большего объёма данных.

Восприятие новых адресов вслепую (без проверки) открывает возможность для организации лавинных DoS-атак против третьей стороны [RFC4225]. В распределенной лавинной атаке злоумышленник может создать соединения HIP с большим объёмом трафика и множеством хостов (неопубликованные HI) а затем объявить всем этим хостам о своём переходе на другой адрес IP. Если партнерские хосты будут просто воспринимать такой перенос, жертва с указанным адресом может получить лавину пакетов. Для предотвращения таких атак расширения HIP mobility включают процедуру проверки маршрута по обратному пути, где доступность узла проверяется независимо для каждого адреса до отправки по этому адресу большего объёма трафика.

До завершения проверки адреса для передачи данных между хостами можно воспользоваться основанном на кредите подходе «Host Mobility with the Host Identity Protocol» [RFC8046]. При использовании HIP между доверяющими один другому хостами можно отказаться от проверки адресов, однако такое решение следует принимать лишь в случаях, когда узлы заведомо заслуживают доверия и способны защитить себя от вредоносных программ.

11.3. Применение HIT в ACL

На конечных точках теги HIT могут применяться в списках контроля доступа на основе IP для прикладного или сетевого уровня. На промежуточных устройствах понимающие HIP МСЭ [lindqvist-enterprise] могут использовать HIT или открытые ключи для входного или выходного контроля на уровне сети или отдельных хостов даже в присутствии мобильных устройств, поскольку теги HIT и открытые ключи не связаны с топологией. Как отмечено в разделе 7, после организации сессии HIP значение SPI в пакете ESP может служить индексом, указывающим HIT. На практике МСЭ могут проверять пакеты HIP для изучения привязок между HIT, SPI и адресами IP. Можно даже явно контролировать использование ESP, динамически открывая ESP лишь для конкретных SPI и адресов IP. Подписи в пакетах HIP позволяют соответствующему МСЭ гарантировать, что обмен HIP действительно происходит между двумя известными хостами. Это может повысить уровень безопасности МСЭ.

Возможным недостатком HIT в ACL является их «плоская» природа, не позволяющая агрегировать теги, что может приводить к большому размеру таблиц поиска в МСЭ с поддержкой HIP. Способом оптимизации может служить применение фильтров Блума (Bloom) для группировки HIT [sarela-bloom]. Однако следует отметить, что правила для отдельных HIT, а не групп позволяют легко исключить хосты с некорректным поведением, не затрагивая других.

Имеется заметный негативный опыт работы с распределенными ACL, содержащими материал, относящийся к открытым ключам, например, для SSH. Если владельцу ключа нужно отозвать его по той или иной причине, задача поиска всех мест хранения ключа в ACL может оказаться непосильной. Если причина отзыва связана с кражей секретного ключа, это может привести к серьёзным проблемам.

Хост может отслеживать всех своих партнёров, которые могут применять его HIT в ACL, регистрируя все удалённые HIT достаточно регистрировать отвечающие хосты). На основе этой информации хост может уведомить другие хосты о смене HIT. Были попытки разработать защищённый метод выдачи уведомлений об отзыве HIT [zhang-revocation].

Некоторые промежуточные устройства с поддержкой HIP, такие как МСЭ [lindqvist-enterprise] или NAT [ylitalo-spinat], могут пассивно просматривать трафик в пути. Такие устройства по своей природе прозрачны и не могут получать уведомлений о переходе хоста в другую сеть. В результате от таких устройств требуется поддержка состояния и тайм-аутов для слишком долгого простоя плоскостей управления и данных между парой конечных хостов HIP. Соответственно, два конечных хоста могут периодически передавать пакеты сохранения (keepalive), такие как UPDATE или сообщения ICMP в туннеле ESP, для поддержки статуса в промежуточном устройстве.

Одним из общих ограничений для сквозного шифрования является неспособность промежуточных устройств участвовать в защите потоков данных. Хотя эта проблема может влиять и на другие протоколы, Heer и др. [heer-end-host] проанализировали её в контексте HIP. В частности, при использовании ESP в качестве протокола плоскости данных для HIP связь между плоскостями данных и управления слаба и может использоваться при некоторых допущениях. Предположим, что злоумышленник получил доступ к целевой сети, защищённой МСЭ с поддержкой HIP, но хочет обойти HIP МСЭ. Для этого атакующий пассивно наблюдает базовый обмен между двумя хостами HIP, а затем воспроизводит его. Таким образом злоумышленнику удаётся преодолеть МСЭ и он может использовать туннель ESP для доставки своих данных. Эта возможность обусловлена тем, что МСЭ не может проверить действительность

туннеля ESP. Для решения проблемы промежуточные устройства с поддержкой HIP могут участвовать во взаимодействии с плоскостью управления путём добавления в трафик управления одноразовых параметров (nonce), которые конечные хосты должны подписывать для обеспечения свежести трафика управления [heer-midauth]. Как вариант можно использовать расширения для доставки трафика плоскости данных напрямую через плоскость управления [RFC6078].

11.4. Варианты для HI

В определении идентификатора хоста сказано, что HI не обязательно является открытым ключом. Это означает, что HI может быть любым значением, например, FQDN. Этот документ не описывает поддержку некриптографических HI, но примеры таких вариантов протокола имеются ([urien-rfid], [urien-rfid-draft]). Некриптографические HI по-прежнему будут предоставлять услуги HIT или LSI для прохождения через NAT. Можно переносить теги HIT в пакетах HIP без защиты приватности и конфиденциальности. Такие схемы могут быть пригодны для устройств с ограниченными ресурсами, таких как небольшие датчики с батарейным питанием, но здесь подобные устройства не рассматриваются.

Если желательно использовать HIP в ситуации с низким уровнем защиты, где расчёт открытых ключей считается избыточным, HIP можно применять с очень короткими ключами Diffie-Hellman и Host Identity. Это делает участвующие хосты уязвимыми для MitM-атак и захвата соединений, однако не вызывает опасности лавинных атак, поскольку механизм проверки адресов полагается на систему маршрутизации, а не криптостойкость.

11.5. Доверие при первом применении

В [RFC7435] выделено 4 принципа разработки для принятия на веру (Leap of Faith) или доверия при первом использовании (Trust On First Use или TOFU) для протоколов, применимые также к opportunistic HIP.

1. Сосуществование с явной политикой.
2. Приоритизация коммуникаций.
3. Максимальная защита партнёров.
4. Отсутствие ложного представления о безопасности.

Согласно первому принципу TOFU: «Защита с использованием обстоятельств (Opportunistic security) никогда не заменит и не вытеснит явные правила.» некоторые данные приложений могут быть слишком конфиденциальными (sensitive), поэтому соответствующая политика может требовать проверки подлинности (т. е. открытого ключа или сертификата) вместо защиты по обстоятельствам без аутентификации. На практике это реализовано в HIP в соответствии с [RFC6538].

OpenHIP позволяет инициатору применять режим Opportunistic лишь с явно заданным IP-адресом отвечающего, когда Responder HIT неизвестен. У отвечающего реализация OpenHIP позволяет включить режим Opportunistic для любого инициатора (доверие к любому инициатору).

Разработчики HIP for Linux (HIPL) экспериментировали с более детализированными правилами на уровне приложения. Реализация HIPL использует так называемую ловушку LD_PRELOAD на уровне приложения, которая позволяет динамически подключаемой библиотеке перехватывать связанные с сокетом вызовы без пересборки соответствующих двоичных файлов приложения. Библиотека выступает промежуточным уровнем между приложениям и транспортом, транслируя не связанные с HIP вызовы сокета из приложения в вызовы на основе HIP. Хотя такая библиотека вносит определённые сложности, описанные в [komu-leap], она достигает цели применения режима Opportunistic на уровне детализации отдельных приложений.

Второй принцип TOFU по сути провозглашает приоритет коммуникаций над безопасностью. Поэтому в общем случае режим Opportunistic следует разрешать даже при отсутствии аутентификации и возможности отката к нешифрованной связи (если правила позволяют) вместо блокировки. На практике это можно реализовать в три этапа. Сначала инициатор HIP может выполнить поиск Responder HI в каталоге (например, DNS). При нахождении инициатором HI он может применить идентификатор для аутентификации и пропустить следующие шаги. При отказе в поиске HI инициатор может попробовать режим Opportunistic с отвечающим. На третьем шаге инициатор может отказаться от коммуникаций на базе HIP после отказа режима Opportunistic, если политика разрешает это. Эта трехступенчатая модель была реализована и описана более подробно в [komu-leap].

Третий принцип TOFU предлагает максимизировать защиту для использования по меньшей мере защиты по обстоятельствам (opportunistic security). Описанная выше трехэтапная модель предпочитает по возможности применять аутентификацию, например, через записи DNS (а при доступности - через DNSSEC) и возвращается в режим Opportunistic при недоступности свидетельств по отдельному каналу (out-of-band credentials). В крайнем случае может происходить отказ от коммуникаций на основе HIP, если правила разрешают это. Поскольку в третьем принципе явно упоминается совершенная защита (perfect forward secrecy или PFS), следует упомянуть её поддержку в HIP.

Четвёртый принцип TOFU гласит, что пользователей и неинтерактивные приложения следует должным образом информировать о применяемом уровне защиты. На практике не поддерживающие HIP приложения будут предполагать отсутствие дополнительной защиты, поэтому ложное представление, по крайней мере у неинтерактивных приложений, возникать не должно. В случае интерактивных desktop-приложений в ранних экспериментах с HIP [karvonen-usable] [RFC6538] использовались приглашения системного уровня для выбора пользователем базовой защиты на основе HIP. Обычно в этих экспериментах пользователи понимали, когда применяется защита на основе HIP. Однако пользователи не замечали разницы между Opportunistic HIP без аутентификации и HIP с аутентификацией но без режима Opportunistic. Причина заключалась в том, что режим Opportunistic HIP (пониженный уровень защиты) не был чётко указан в системном приложении. Это стало уроком в части улучшения пользовательского интерфейса.

В случае понимающих HIP приложений можно использовать естественные API сокетов для HIP, описанные в [RFC6317], для разработки зависящей от приложения логики вместо базового системного приглашения. Здесь приложение само может спросить пользователя или иным способом управлять ситуацией. В этом случае неинтерактивные приложения также могут должным образом осознать уровень защиты, поскольку разработчик может явно задать использование HIP с аутентификацией, Opportunistic HIP или обмен открытыми данными (plain-text).

Следует упомянуть несколько дополнительных пунктов, отмеченных в [RFC7435]. Для активных атак в HIP имеется встроенная защита против понижения версии шифра, как описано в [RFC7401]. Кроме того, могут применяться заранее установленные сертификаты для ослабления атак в случае режима Opportunistic, как отмечено в [RFC6538].

Обнаружение возможностей партнёра также упоминается в контексте TOFU и для этого может применяться трехэтапная модель, отмеченная выше. Хост может выполнить первый этап аутентификации, т. е. обнаружить открытый ключ, например, через DNS. Если хост не находит ключей, он может в качестве второго шага проверить режим Opportunistic. При возникновении тайм-аута хост может перейти к третьему шагу, возвращаясь к взаимодействию без HIP, если политика разрешает это. Последний этап основан на неявном тайм-ауте вместо явного (негативного) подтверждения как в случае DNS, поэтому пользователь может сделать вывод об отказе преждевременно. Для ускорения фазы обнаружения путём явной проверки, поддерживает ли партнёр режим Opportunistic HIP, исследователи предложили расширения для TCP [RFC6538] [komu-leap]. Инициатор передаёт партнёру одновременно пакет (opportunistic) I1 и соответствующую дейтаграмму TCP SYN со специальной опцией TCP. Если партнёр поддерживает HIP, он отбросит пакет SYN и ответит пакетом R1. Если же партнёр не понимает HIP, он отбросит пакет HIP (и неизвестную опцию TCP) и передаст в ответ TCP SYN-ACK. Преимуществом предложенной схемы является быстрый отказ от попытки взаимодействия на основе HIP за один период кругового обхода. Недостаток заключается в привязке к TCP (опции IP также рассматривались, но они плохо работают через МСЭ и NAT). Такой подход не работает против активных атак, но режим Opportunistic в любом случае не предназначен для этого.

Следует отметить, что, несмотря на некоторые преимущества режима Opportunistic, связанные с постепенным развёртыванием, он уступает HIP с аутентификацией [komu-diss], поскольку тот поддерживает постоянные идентификаторы (хост указывается одним HI независимо от перемещений). Opportunistic HIP решает эту задачу лишь частично - после первого контакта между хостами HIP может поддерживать соединение с помощью расширений для мобильности, но возникают проблемы, когда хосты разрывают ассоциацию HIP и пытаются связаться снова. Хост может сменить своё местоположение и нет гарантии привязки адреса IP к хосту, поскольку один адрес может временно выделяться разным хостам (например, сервером DHCP), области адресации могут перекрываться (см. Приложение A.1) или из-за попытки организации атаки.

12. Взаимодействие с IANA

Этот документ не требует действий IANA.

13. Отличия от RFC 4423

Отличия от RFC 4423 [RFC4423] являются в основном редакторскими правками, включая разъяснения сложно описанных тем и исключение не относящихся к архитектуре деталей, уже описанных в других документах. Добавлен ряд пропущенных. Включено рассмотрение недостатков HIP, а также безопасности 802.15.4 и MAC, вариантов HIP для IoT, вопросов развёртывания и описание базового обмена.

14. Литература

14.1. Нормативные документы

[RFC5482] Eggert, L. and F. Gont, "TCP User Timeout Option", RFC 5482, DOI 10.17487/RFC5482, March 2009, <https://www.rfc-editor.org/info/rfc5482>.

[RFC6079] Camarillo, G., Nikander, P., Hautakorpi, J., Keranen, A., and A. Johnston, "HIP BONE: Host Identity Protocol (HIP) Based Overlay Networking Environment (BONE)", RFC 6079, DOI 10.17487/RFC6079, January 2011, <https://www.rfc-editor.org/info/rfc6079>.

[RFC7086] Keranen, A., Camarillo, G., and J. Maenpaa, "Host Identity Protocol-Based Overlay Networking Environment (HIP BONE) Instance Specification for REsource LOcation And Discovery (RELOAD)", RFC 7086, DOI 10.17487/RFC7086, January 2014, <https://www.rfc-editor.org/info/rfc7086>.

[RFC7343] Laganier, J. and F. Dupont, "An IPv6 Prefix for Overlay Routable Cryptographic Hash Identifiers Version 2 (ORCHIDv2)", RFC 7343, DOI 10.17487/RFC7343, September 2014, <https://www.rfc-editor.org/info/rfc7343>.

[RFC7401] Moskowitz, R., Ed., Heer, T., Jokela, P., and T. Henderson, "Host Identity Protocol Version 2 (HIPv2)", RFC 7401, DOI 10.17487/RFC7401, April 2015, <https://www.rfc-editor.org/info/rfc7401>.

[RFC7402] Jokela, P., Moskowitz, R., and J. Melen, "Using the Encapsulating Security Payload (ESP) Transport Format with the Host Identity Protocol (HIP)", RFC 7402, DOI 10.17487/RFC7402, April 2015, <https://www.rfc-editor.org/info/rfc7402>.

[RFC8002] Heer, T. and S. Varjonen, "Host Identity Protocol Certificates", RFC 8002, DOI 10.17487/RFC8002, October 2016, <https://www.rfc-editor.org/info/rfc8002>.

[RFC8003] Laganier, J. and L. Eggert, "Host Identity Protocol (HIP) Registration Extension", RFC 8003, DOI 10.17487/RFC8003, October 2016, <https://www.rfc-editor.org/info/rfc8003>.

[RFC8004] Laganier, J. and L. Eggert, "Host Identity Protocol (HIP) Rendezvous Extension", RFC 8004, DOI 10.17487/RFC8004, October 2016, <https://www.rfc-editor.org/info/rfc8004>.

[RFC8005] Laganier, J., "Host Identity Protocol (HIP) Domain Name System (DNS) Extension", RFC 8005, DOI 10.17487/RFC8005, October 2016, <https://www.rfc-editor.org/info/rfc8005>.

[RFC8046] Henderson, T., Ed., Vogt, C., and J. Arkko, "Host Mobility with the Host Identity Protocol", RFC 8046, DOI 10.17487/RFC8046, February 2017, <https://www.rfc-editor.org/info/rfc8046>.

[RFC8047] Henderson, T., Ed., Vogt, C., and J. Arkko, "Host Multihoming with the Host Identity Protocol", RFC 8047, DOI 10.17487/RFC8047, February 2017, <https://www.rfc-editor.org/info/rfc8047>.

[RFC9028] Keränen, A., Melén, J., and M. Komu, Ed., "Native NAT Traversal Mode for the Host Identity Protocol", RFC 9028, DOI 10.17487/RFC9028, July 2021, <https://www.rfc-editor.org/info/rfc9028>.

14.2. Дополнительная литература

[amir-hip] Amir, K., Forsgren, H., Grahn, K., Karvi, T., and G. Pulkkis, "Security and Trust of Public Key Cryptography for HIP and HIP Multicast", International Journal of Dependable and Trustworthy Information Systems (IJDTIS), Vol. 2, Issue 3, pp. 17-35, DOI 10.4018/jdtis.2011070102, 2013, <https://doi.org/10.4018/jdtis.2011070102>.

[aura-dos] Aura, T., Nikander, P., and J. Leiwo, "DOS-Resistant Authentication with Client Puzzles", 8th International Workshop on Security Protocols, Security Protocols 2000, Lecture Notes in Computer Science, Vol. 2133, pp. 170-177, Springer, DOI 10.1007/3-540-44810-1_22, September 2001, <https://doi.org/10.1007/3-540-44810-1_22>.

[beal-dos] Beal, J. and T. Shepard, "Deamplification of DoS Attacks via Puzzles", October 2004.

[camarillo-p2psip] Camarillo, G., Mäenpää, J., Keränen, A., and V. Anderson, "Reducing delays related to NAT traversal in P2PSIP session establishments", IEEE Consumer Communications and Networking Conference (CCNC), pp. 549-553, DOI 10.1109/CCNC.2011.5766540, 2011, <https://doi.org/10.1109/CCNC.2011.5766540>.

[chiappa-endpoints] Chiappa, J., "Endpoints and Endpoint Names: A Proposed Enhancement to the Internet Architecture", 1999, <http://mercury.lcs.mit.edu/~jnc/tech/endpoints.txt>.

[heer-end-host] Heer, T., Hummen, R., Komu, M., Gotz, S., and K. Wehrle, "End-Host Authentication and Authorization for Middleboxes Based on a Cryptographic Namespace", 2009 IEEE International Conference on Communications, DOI 10.1109/ICC.2009.5198984, 2009, <https://doi.org/10.1109/ICC.2009.5198984>.

[heer-midauth] Heer, T., Ed., Hummen, R., Wehrle, K., and M. Komu, "End-Host Authentication for HIP Middleboxes", Work in Progress, Internet-Draft, draft-heer-hip-middle-auth-04, 31 October 2011, <https://datatracker.ietf.org/doc/html/draft-heer-hip-middle-auth-04>.

[henderson-vpls] Henderson, T. R., Venema, S. C., and D. Mattes, "HIP-based Virtual Private LAN Service (HIPLS)", Work in Progress, Internet-Draft, draft-henderson-hip-vpls-11, 3 August 2016, <https://datatracker.ietf.org/doc/html/draft-henderson-hip-vpls-11>.

[hip-dex] Moskowitz, R., Ed., Hummen, R., and M. Komu, "HIP Diet EXchange (DEX)", Work in Progress, Internet-Draft, draft-ietf-hip-dex-24, 19 January 2021, <https://datatracker.ietf.org/doc/html/draft-ietf-hip-dex-24>.

[hip-lte] Liyanage, M., Kumar, P., Ylianttila, M., and A. Gurtov, "Novel secure VPN architectures for LTE backhaul networks", Security and Communication Networks, Vol. 9, pp. 1198-1215, DOI 10.1002/sec.1411, January 2016, <https://doi.org/10.1002/sec.1411>.

[hip-srtp] Tschofenig, H., Shanmugam, M., and F. Muenz, "Using SRTP transport format with HIP", Work in Progress, Internet-Draft, draft-tschofenig-hiprg-hip-srtp-02, 25 October 2006, <https://datatracker.ietf.org/doc/html/draft-tschofenig-hiprg-hip-srtp-02>.

[hummen] Hummen, R., Hiller, J., Henze, M., and K. Wehrle, "Slimfit - A HIP DEX compression layer for the IP-based Internet of Things", 2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), pp. 259-266, DOI 10.1109/WiMOB.2013.6673370, October 2013, <https://doi.org/10.1109/WiMOB.2013.6673370>.

[IEEE.802.15.4] IEEE, "IEEE Standard for Low-Rate Wireless Networks", IEEE Standard 802.15.4, DOI 10.1109/IEEESTD.2020.9144691, July 2020, <https://ieeexplore.ieee.org/document/9144691>.

[IEEE.802.15.9] IEEE, "IEEE Draft Recommended Practice for Transport of Key Management Protocol (KMP) Datagrams", IEEE P802.15.9/D04, May 2015.

[karvonen-usable] Karvonen, K., Komu, M., and A. Gurtov, "Usable security management with host identity protocol", 2009 IEEE/ACS International Conference on Computer Systems and Applications, pp. 279-286, DOI 10.1109/AICCSA.2009.5069337, 2009, <https://doi.org/10.1109/AICCSA.2009.5069337>.

[komu-cloud] Komu, M., Sethi, M., Mallavarapu, R., Oirola, H., Khan, R., and S. Tarkoma, "Secure Networking for Virtual Machines in the Cloud", 2012 IEEE International Conference on Cluster Computing Workshops, pp. 88-96, DOI 10.1109/ClusterW.2012.29, 2012, <https://doi.org/10.1109/ClusterW.2012.29>.

[komu-diss] Komu, M., "A Consolidated Namespace for Network Applications, Developers, Administrators and Users", Dissertation, Aalto University, Espoo, Finland, ISBN 978-952-60-4904-5 (printed), ISBN 978-952-60-4905-2 (electronic), December 2012.

[komu-leap] Komu, M. and J. Lindqvist, "Leap-of-Faith Security is Enough for IP Mobility", 2009 6th IEEE Consumer Communications and Networking Conference, Las Vegas, NV, USA, pp. 1-5, DOI 10.1109/CCNC.2009.4784729, January 2009, <https://doi.org/10.1109/CCNC.2009.4784729>.

[komu-mitigation] Komu, M., Tarkoma, S., and A. Lukyanenko, "Mitigation of Unsolicited Traffic Across Domains with Host Identities and Puzzles", 15th Nordic Conference on Secure IT Systems, NordSec 2010, Lecture Notes in Computer Science, Vol. 7127, pp. 33-48, Springer, ISBN 978-3-642-27936-2, DOI 10.1007/978-3-642-27937-9_3, October 2010, <https://doi.org/10.1007/978-3-642-27937-9_3>.

[kovacshazi-host] Kovacshazi, Z. and R. Vida, "Host Identity Specific Multicast", International Conference on Networking and Services (ICNS '07), Athens, Greece, pp. 1-1, DOI 10.1109/ICNS.2007.66, 2007, <https://doi.org/10.1109/ICNS.2007.66>.

[levae-barriers] Levä, T., Komu, M., and S. Luukkainen, "Adoption barriers of network layer protocols: the case of host identity protocol", Computer Networks, Vol. 57, Issue 10, pp. 2218-2232, ISSN 1389-1286, DOI 10.1016/j.comnet.2012.11.024, March 2013, <https://doi.org/10.1016/j.comnet.2012.11.024>.

[lindqvist-enterprise] Lindqvist, J., Vehmersalo, E., Komu, M., and J. Manner, "Enterprise Network Packet Filtering for Mobile Cryptographic Identities", International Journal of Handheld Computing Research (IJHCR), Vol. 1, Issue 1, pp. 79-94, DOI 10.4018/jhcr.2010090905, 2010, <https://doi.org/10.4018/jhcr.2010090905>.

[Nik2001] Nikander, P., "Denial-of-Service, Address Ownership, and Early Authentication in the IPv6 World", 9th International Workshop on Security Protocols, Security Protocols 2001, Lecture Notes in Computer Science, Vol. 2467, pp. 12-21, Springer, DOI 10.1007/3-540-45807-7_3, 2002, <https://doi.org/10.1007/3-540-45807-7_3>.

[nsrg-report] Lear, E. and R. Droms, "What's In A Name: Thoughts from the NSRG", Work in Progress, Internet-Draft, draft-irtf-nsrg-report-10, 22 September 2003, <https://datatracker.ietf.org/doc/html/draft-irtf-nsrg-report-10>.

[paine-hip] Paine, R. H., "Beyond HIP: The End to Hacking As We Know It", BookSurge Publishing, ISBN-10 1439256047, ISBN-13 978-1439256046, 2009.

[pham-leap] Pham, V. and T. Aura, "Security Analysis of Leap-of-Faith Protocols", 7th International ICST Conference, Security and Privacy for Communication Networks, SecureComm 2011, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Vol. 96, DOI 10.1007/978-3-642-31909-9_19, 2012, <https://doi.org/10.1007/978-3-642-31909-9_19>.

[ranjbar-synaptic] Ranjbar, A., Komu, M., Salmela, P., and T. Aura, "SynAPTIC: Secure and Persistent Connectivity for Containers", 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), Madrid, 2017, pp. 262-267, DOI 10.1109/CCGRID.2017.62, 2017, <https://doi.org/10.1109/CCGRID.2017.62>.

[RFC2136] Vixie, P., Ed., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, DOI 10.17487/RFC2136, April 1997, <https://www.rfc-editor.org/info/rfc2136>.

[RFC2766] Tsirtsis, G. and P. Srisuresh, "Network Address Translation - Protocol Translation (NAT-PT)", RFC 2766, DOI 10.17487/RFC2766, February 2000, <https://www.rfc-editor.org/info/rfc2766>.

[RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, DOI 10.17487/RFC3022, January 2001, <https://www.rfc-editor.org/info/rfc3022>.

[RFC3102] Borella, M., Lo, J., Grabelsky, D., and G. Montenegro, "Realm Specific IP: Framework", RFC 3102, DOI 10.17487/RFC3102, October 2001, <https://www.rfc-editor.org/info/rfc3102>.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, Ed., "Extensible Authentication Protocol (EAP)", RFC 3748, DOI 10.17487/RFC3748, June 2004, <https://www.rfc-editor.org/info/rfc3748>.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, DOI 10.17487/RFC3972, March 2005, <https://www.rfc-editor.org/info/rfc3972>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <https://www.rfc-editor.org/info/rfc4033>.

[RFC4225] Nikander, P., Arkko, J., Aura, T., Montenegro, G., and E. Nordmark, "Mobile IP Version 6 Route Optimization Security Design Background", RFC 4225, DOI 10.17487/RFC4225, December 2005, <https://www.rfc-editor.org/info/rfc4225>.

[RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, DOI 10.17487/RFC4380, February 2006, <https://www.rfc-editor.org/info/rfc4380>.

[RFC4423] Moskowitz, R. and P. Nikander, "Host Identity Protocol (HIP) Architecture", RFC 4423, DOI 10.17487/RFC4423, May 2006, <https://www.rfc-editor.org/info/rfc4423>.

[RFC5218] Thaler, D. and B. Aboba, "What Makes for a Successful Protocol?", RFC 5218, DOI 10.17487/RFC5218, July 2008, <https://www.rfc-editor.org/info/rfc5218>.

[RFC5338] Henderson, T., Nikander, P., and M. Komu, "Using the Host Identity Protocol with Legacy Applications", RFC 5338, DOI 10.17487/RFC5338, September 2008, <https://www.rfc-editor.org/info/rfc5338>.

[RFC5887] Carpenter, B., Atkinson, R., and H. Flinck, "Renumbering Still Needs Work", RFC 5887, DOI 10.17487/RFC5887, May 2010, <https://www.rfc-editor.org/info/rfc5887>.

[RFC6078] Camarillo, G. and J. Melen, "Host Identity Protocol (HIP) Immediate Carriage and Conveyance of Upper-Layer Protocol Signaling (HICCUPS)", RFC 6078, DOI 10.17487/RFC6078, January 2011, <https://www.rfc-editor.org/info/rfc6078>.

[RFC6250] Thaler, D., "Evolution of the IP Model", RFC 6250, DOI 10.17487/RFC6250, May 2011, <https://www.rfc-editor.org/info/rfc6250>.

[RFC6281] Cheshire, S., Zhu, Z., Wakikawa, R., and L. Zhang, "Understanding Apple's Back to My Mac (BTMM) Service", RFC 6281, DOI 10.17487/RFC6281, June 2011, <https://www.rfc-editor.org/info/rfc6281>.

[RFC6317] Komu, M. and T. Henderson, "Basic Socket Interface Extensions for the Host Identity Protocol (HIP)", RFC 6317, DOI 10.17487/RFC6317, July 2011, <https://www.rfc-editor.org/info/rfc6317>.

[RFC6537] Ahrenholz, J., "Host Identity Protocol Distributed Hash Table Interface", RFC 6537, DOI 10.17487/RFC6537, February 2012, <https://www.rfc-editor.org/info/rfc6537>.

[RFC6538] Henderson, T. and A. Gurtov, "The Host Identity Protocol (HIP) Experiment Report", RFC 6538, DOI 10.17487/RFC6538, March 2012, <https://www.rfc-editor.org/info/rfc6538>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC7435] Dukhovni, V., "Opportunistic Security: Some Protection Most of the Time", RFC 7435, DOI 10.17487/RFC7435, December 2014, <https://www.rfc-editor.org/info/rfc7435>.

[sarela-bloom] Särelä, M., Esteve Rothenberg, C., Zahemszky, A., Nikander, P., and J. Ott, "BloomCasting: Security in Bloom Filter Based Multicast", Information Security Technology for Applications, NordSec 2010, Lecture Notes in Computer Science, Vol. 7127, pages 1-16, Springer, DOI 10.1007/978-3-642-27937-9_1, 2012, <https://doi.org/10.1007/978-3-642-27937-9_1>.

[schuetz-intermittent] Schütz, S., Eggert, L., Schmid, S., and M. Brunner, "Protocol enhancements for intermittently connected hosts", ACM SIGCOMM Computer Communication Review, Vol. 35, Issue 3, pp. 5-18, DOI 10.1145/1070873.1070875, July 2005, <https://doi.org/10.1145/1070873.1070875>.

[shields-hip] Shields, C. and J. J. Garcia-Luna-Aceves, "The HIP protocol for hierarchical multicast routing", Proceedings of the seventeenth annual ACM symposium on Principles of distributed computing, pp. 257-266, ISBN 0-89791-977-7, DOI 10.1145/277697.277744, 1998, <https://doi.org/10.1145/277697.277744>.

[tempered-networks] Tempered Networks, "Identity-Defined Network (IDN) Architecture: Unified, Secure Networking Made Simple", White Paper, 2016.

[tritilanunt-dos] Tritilanunt, S., Boyd, C., Foo, E., and J.M.G. Nieto, "Examining the DoS Resistance of HIP", On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops, Lecture Notes in Computer Science, Vol. 4277, pp. 616-625, Springer, DOI 10.1007/11915034_85, 2006, <https://doi.org/10.1007/11915034_85>.

[urien-rfid] Urien, P., Chabanne, H., Pepin, C., Orga, S., Bouet, M., de Cunha, D.O., Guyot, V., Pujolle, G., Paradinas, P., Gressier, E., and J.-F. Susini, "HIP-based RFID Networking Architecture", 2007 IFIP International Conference on Wireless and Optical Communications Networks, pp. 1-5, DOI 10.1109/WOCN.2007.4284140, 2007, <https://doi.org/10.1109/WOCN.2007.4284140>.

[urien-rfid-draft] Urien, P., Lee, G. M., and G. Pujolle, "HIP support for RFIDs", Work in Progress, Internet-Draft, draft-irtf-hiprg-rfid-07, 23 April 2013, <https://datatracker.ietf.org/doc/html/draft-irtf-hiprg-rfid-07>.

[varjonen-split] Varjonen, S., Komu, M., and A. Gurtov, "Secure and Efficient IPv4/IPv6 Handovers Using Host-Based Identifier-Location Split", Journal of Communications Software and Systems, Vol. 6, Issue 1, ISSN 18456421, DOI 10.24138/jcomss.v6i1.193, 2010, <https://doi.org/10.24138/jcomss.v6i1.193>.

[xin-hip-lib] Xin, G., "Host Identity Protocol Version 2.5", Master's Thesis, Aalto University, Espoo, Finland, June 2012.

[ylitalo-diss] Ylitalo, J., "Secure Mobility at Multiple Granularity Levels over Heterogeneous Datacom Networks", Dissertation, Helsinki University of Technology, Espoo, Finland, ISBN 978-951-22-9531-9, 2008.

[ylitalo-spinat] Ylitalo, J., Salmela, P., and H. Tschofenig, "SPINAT: Integrating IPsec into Overlay Routing", First International Conference on Security and Privacy for Emerging Areas in Communication Networks, SECURECOMM'05, Athens, Greece, pp. 315-326, ISBN 0-7695-2369-2, DOI 10.1109/SECURECOMM.2005.53, 2005, <https://doi.org/10.1109/SECURECOMM.2005.53>.

[zhang-revocation] Zhang, D., Kuptsov, D., and S. Shen, "Host Identifier Revocation in HIP", Work in Progress, Internet-Draft, draft-irtf-hiprg-revocation-05, 9 March 2012, <https://datatracker.ietf.org/doc/html/draft-irtf-hiprg-revocation-05>.

[zhu-hip] Zhu, X., Ding, Z., and X. Wang, "A Multicast Routing Algorithm Applied to HIP-Multicast Model", 2011 International Conference on Network Computing and Information Security, Guilin, China, pp. 169-174, DOI 10.1109/NCIS.2011.42, 2011, <https://doi.org/10.1109/NCIS.2011.42>.

[zhu-secure] Zhu, X. and J. W. Atwood, "A Secure Multicast Model for Peer-to-Peer and Access Networks Using the Host Identity Protocol", 2007 4th IEEE Consumer Communications and Networking Conference, Las Vegas, NV, USA, pages 1098-1102, DOI 10.1109/CCNC.2007.221, 2007, <https://doi.org/10.1109/CCNC.2007.221>.

Приложение A. Соображения по устройству

A.1. Преимущества HIP

Изначально протокол сетевого уровня (IP) имел 4 «классических» инварианта:

1. неизменность - переданный адрес совпадал с принятым;
2. отсутствие мобильности - адрес не менялся в процессе взаимодействия;
3. обратимость - адрес возврата всегда определялся перестановкой адресов отправителя и получателя;
4. всеведение - каждый хост знал, какой адрес можно использовать для передачи пакетов партнёру.

Четвёртый инвариант можно вывести из 1 и 3, но он упомянут явно по причинам, рассмотренным ниже.

В современном «постклассическом» мире предпринимаются попытки исключить п. 2 (для мобильности и применения нескольких адресов), а также произошёл отказ от пп. 1 и 4. Попыткой сохранить п. 4 без п. 1, был документ Realm Specific IP [RFC3102], а IPv6 пытается восстановить п. 1.

Значимые имена DNS имеют немногие клиентские системы в Internet. Т. е. при наличии у клиентской системы имени FQDN это имя обычно относится к устройству NAT или серверу доступа (dial-up), а не указывает реально

подключающуюся систему. FQDN (и их расширения в форме почтовых адресов) относятся к уровню приложений (чаще именуются службы, а не отдельные системы). Поэтому многие системы в Internet не зарегистрированы в DNS - у них просто нет служб, интересных другим хостам Internet.

Имена DNS являются ссылками на адреса IP и это лишь демонстрирует связь между сетевым и прикладным уровнем. DNS, как единственная и распределенная база данных Internet, является также хранилищем для других пространств имён, отчасти благодаря DNSSEC и записям ключей для приложений. Хотя каждое пространство имён можно растянуть (IP с помощью v6, DNS с помощью записей KEY), ни одно из них не может адекватно обеспечить аутентификацию хостов или послужить разделом между прикладным и сетевым уровнем.

Пространство имён HI заполняет важный пробел между пространствами IP и DNS. Интересным в HI является возможность отказа хоста от всего, кроме п. 3 для сетевого уровня. Иными словами, пока адреса отправителя и получателя в протоколе сетевого уровня обратимы, HIP заботится об идентификации хоста, а обратимость позволяет локальному хосту получать пакеты от удалённого. Смена адресов в результате прохождения через NAT (изменяемость) или перемещения хоста (мобильность и отсутствие всеведения) может обрабатываться уровнем HIP.

За исключением высокопроизводительных расчётных приложений, API сокетов являются наиболее общим вариантом разработки сетевых приложений, которые используют API напрямую или опосредованно через те или иные библиотеки или модели. Однако API сокетов основаны на допущении о статических адресах IP, а DNS со сроками действия записей придумали позже в процессе развития Internet. Поэтому API сокетов не работают со сроками действия адресов [RFC6250]. Сегодня большая часть пользовательского оборудования стала мобильной, а его адреса эфемерными, но API сокетов по-прежнему создают иллюзию постоянства адресов IP для неосторожных разработчиков. Протокол HIP может служить для укрепления этой иллюзии, поскольку HIP обеспечивает постоянные суррогатные адреса в форме LSI и HIT.

Постоянные идентификаторы, предоставляемые HIP, полезны во многих случаях (см. [ylitalo-diss] [komu-diss]).

- При перемещении мобильного хоста между WLAN со сменой адреса использующее идентификаторы приложение не зависит от изменений топологии, а нижележащий уровень HIP восстанавливает связность (передача обслуживания по горизонтали).
- При переходе мобильного устройства из сети WLAN в сотовую сеть использующему идентификаторы приложению не нужно знать о смене технологии (передача обслуживания по вертикали).
- При размещении хостов в разных сетях с приватными адресами приложение может однозначно распознавать хосты по их идентификаторам. Иными словами, HIP улучшает прозрачность Internet для уровня приложений [komu-diss].
- При смене адресов сайта для служб в результате раздела или слияния компаний, а также при смене Internet-провайдера может полностью измениться сетевой префикс организации, что может создавать проблемы для трафика из-за жёсткого задания адресов в файлах конфигурации служб или кэширования адресов IP на стороне клиентов [RFC5887]. С учётом возможных человеческих ошибок использование независимых от местоположения идентификаторов, предоставляемых HIP, может смягчить проблему смены адресов.
- Может быть повышена гибкость взаимодействия с IPv6, как отмечено в параграфе 4.4. Приложения на основе IPv6 могут взаимодействовать с помощью HIT с приложениями IPv4, использующими идентификаторы LSI. Кроме того, семейство адресов в приложении становится независимым от типа базовой сети (IPv4 или IPv6).
- Можно применять HIT (или LSI) в списках доступа на основе IP как более защищённую замену адресов IPv6. Помимо защиты, контроль доступа на основе HIT обеспечивает два других преимущества. Во-первых, применение HIT может вдвое сократить размер списков, поскольку не будут нужны отдельные правила для IPv4 [komu-diss]. Во-вторых, правила конфигурации на основе HIT в промежуточных устройствах с поддержкой HIP остаются статическими и не зависят от смены топологии, что упрощает администрирование, особенно для сред с мобильными устройствами. Например, преимущества списков управления доступом на основе HIT применимы в HIP МСЭ, но могут использоваться также непосредственно на конечных хостах [RFC6538].

Хотя некоторые из этих преимуществ могут быть и были реализованы в отдельных приложениях, обеспечение такой базовой функциональности на более низком уровне полезно, поскольку снижает объем работы при создании приложений и число ошибок в сетевых программах (поскольку уровень тестируется с разными приложениями). Это также позволяет разработчикам сосредоточиться на приложении, а не вникать в тонкости работы мобильных сетей, тем самым облегчая разделение задач.

HIP можно объединять с разными протоколами, но сложность получаемых в результате программ может существенно возрасти, а взаимодействие между разными (возможно многоуровневыми) протоколами может негативно повлиять на задержку и пропускную способность. Следует также отметить, что практически нет помех в реализации архитектуры HIP в форме, например, библиотеки прикладного уровня, которая уже фактически была реализована в [xin-hip-lib]. Однако при переносе HIP на уровень приложений могут не поддерживаться унаследованные приложения.

A.2. Недостатки HIP

В информатике многие проблемы можно решить с помощью дополнительного уровня опосредованности. Однако косвенные обращения всегда связаны с определёнными расходами и «бесплатных обедов» не бывает. Издержки для случая HIP перечислены ниже.

- В общем случае дополнительный уровень и пространство имён всегда требуют начальных усилий в плане разработки, развёртывания и обслуживания. Может потребоваться также некоторое обучение для разработчиков и администраторов. Сообщество HIP в IETF потратило годы на эксперименты, исследования, тестирование, документирование и реализацию HIP для снижения расходов на внедрение.
- HIP требует управления идентификаторами HI и централизованного подхода к масштабному управлению конечными точками с поддержкой HIP. Прежние ACL на основе адресов IP сейчас стали списками на основе доверенных HIT и сопоставление HIT с IP, а также правила доступа нужно администрировать. Конечные точки с поддержкой HIP должны также быть способны работать автономно для обеспечения мобильности и

доступности (конечная точка должна сохранять работоспособность в отсутствие постоянного управляющего соединения). Пользователи, которым нужна повышенная защищенность и мобильность на основе HI вместо ACL по адресам IP, должны затем поддерживать этот дополнительный «уровень отождествления». Как показано в Приложении A.3.5, эти задачи уже решены в настройке инфраструктуры распространения политики и управления отображениями и отношениями доверия между конечными точками HIP.

- HIP разделяет роли IP адресов как идентификаторов и «локаторов», поэтому требуется механизм сопоставления, чтобы связать их между собой. Неспособность сопоставить HIT с соответствующим «локатором» может приводить к отказам связности, поскольку идентификаторы HIT являются «плоскими» по своей природе и не поддерживают поиска через иерархическую систему DNS. Плоское пространство HIT обусловлено компромиссом в части защиты. Увеличение размера хэша в HIT снижает вероятность (злонамеренных) конфликтов (совпадений).
- С точки зрения производительности обработка плоскостей данных и управления в HIP вносит некоторые издержки в части пропускной способности и задержки, как описано ниже.

Из-за недостатков развёртывания для контроля доступа к различным службам и устройствам в современной сети Internet обычно применяются МСЭ. Поскольку HIP вносит дополнительное пространство имён, предполагается что для него также будет применяться фильтрация на предмет нежелательных подключений. Хотя это может быть достигнуто с помощью имеющихся средств непосредственно на конечных хостах, фильтры промежуточных устройств потребуется менять с внесением правок в программы имеющихся МСЭ или дополнительных промежуточных устройств [RFC6538].

Обмен ключами вносит дополнительную задержку *2 периода кругового обхода) в организацию транспортного соединения между парой конечных хостов. В TCP дополнительная задержка возникает, если реализация базового сетевого стека отбрасывает пакет SYN в процессе обмена ключами. Такие же издержки могут добавляться процедурами передачи обслуживания в HIP. Однако последующие сеансы TCP с той же ассоциацией HIP не будут добавлять издержек (в течение срока действия ключа). Издержки при обмене ключами и передаче обслуживания можно минимизировать за счёт кэширования пакетов TCP, а передачу обслуживания можно дополнительно оптимизировать с помощью расширений для пользовательского тайм-аута TCP [RFC5482], как подробно описал Schütz с соавторами [schuetz-intermittent].

Наиболее загружающие CPU операции связаны с использованием асимметричных ключей и вывода ключей методом Diffie-Hellman в плоскости управления, но это происходит при обмене ключами, их обслуживании (передача обслуживания и обновление ключевого материала) и завершении ассоциаций HIP. Плоскость данных обычно реализуется с помощью ESP, поскольку это снижает издержки за счёт применения симметричных ключей. Однако и с ESP связаны дополнительные издержки в части задержек (обработка) и пропускной способности (туннелирование), как отмечено в [ylitalo-diss] для оценки производительности.

A.3. Вопросы развёртывания и адаптации

В этом разделе рассмотрены некоторые соображения по развёртыванию и адаптации HIP с технической точки зрения.

A.3.1. Анализ развёртывания

Протокол HIP был адаптирован и развернут в промышленной сети управления производственного предприятия, где строгая идентификация HIP на сетевом уровне поддерживала безопасное сосуществование множества незащищённых сетевых устройств разных производителей [paine-hip]. Протокол HIP был также включён в продукцию защиты для поддержки L2 VPN [henderson-vpls] с целью поддержки зон безопасности в сети диспетчерского контроля и сбора данных (supervisory control and data acquisition или SCADA). Однако HIP не получил «большого успеха» [RFC5218] в Internet, как отмечено Levä и др. [levae-barriers]. Здесь кратко освещены некоторые выводы, основанные на общении с 19 специалистами из сферы промышленности и академических кругов.

С точки зрения маркетинга потребность в HIP была невелика и предпочтение отдавалось другим технологиям. Другая выявленная причина заключалась в сохранении некоторых технических заблуждений, связанных с ранними спецификациями HIP. Два обнаруженных заблуждения состояли в том, что HIP не поддерживает работу через NAT и требует реализации в ядре OS. Оба этих утверждения не соответствуют действительности - HIP включает расширения для работы через NAT [RFC9028], а изменения ядра можно избежать в современных ОС перенаправляя пакеты для обработки в пользовательское пространство.

В анализе Levä и др. приведены инфраструктурные требования для HIP. В минимальном варианте на машинах клиента и сервера должны работать программы HIP. Однако для предотвращения настройки вручную для HIP обычно создаются записи в DNS. Например, популярных DNS-сервер Bind9 не требует каких-либо изменений для размещения записей, связанных с HIP, поскольку он поддерживает двоичный формат в файлах конфигурации [RFC6538]. Серверы HIP rendezvous и МСЭ не являются обязательными. Не требуется вносить изменения в сетевые адреса, NAT, граничные маршрутизаторы и сети ядра.

Анализ также проясняет требования к компонентам хоста, состоящие из трёх частей. Во-первых, требуется плоскость управления HIP, обычно реализуемая в виде демона в пользовательском пространстве. Во-вторых, нужна плоскость данных и большинство реализаций HIP использует режим туннеля со сквозной привязкой (Bound End-to-End Tunnel или BEET) для ESP, доступный в Linux, начиная с ядра 2.6.27, и включенный также в нескольких реализациях в форме программы в пользовательском пространстве. В-третьих, системы HIP обычно обеспечивают DNS-прокси для локального хоста, транслирующий записи HIP DNS в LSI или HIT и передающий соответствующие «локаторы» демону HIP в пользовательском пространстве. Хотя третья часть не является обязательной, она очень полезна для предотвращения настройки вручную. Дополнительное описание этих модулей приведено в отчёте [RFC6538].

На основе обсуждения со специалистами в работе Levä и др. предложены дальнейшие направления для упрощения развёртывания HIP. Перенос ряда спецификаций HIP в IETF Standards Track уже произошёл, но авторы предлагают дополнительные меры, в частности, реализацию HIP в виде библиотеки прикладного уровня [xin-hip-lib] или иной промежуточной программы (middleware). С другой стороны, более осторожные меры включают сосредоточение на частных развёртываниях, контролируемых одной заинтересованной стороной. В качестве более конкретного примера такого сценария HIP может применяться одним сервис-провайдером для организации защищённых соединений между его серверами [komu-cloud].

A.3.2. HIP в сетях 802.15.4

Стандарты IEEE 802 определяют защиту на уровне MAC и многие из них используют расширяемый протокол аутентификации (Extensible Authentication Protocol или EAP) [RFC3748] в качестве системы управления ключами (Key Management System или KMS), но некоторые стандарты, такие как IEEE 802.15.4 [IEEE.802.15.4], оставляют систему KMS и её транспорт вне «области действия». HIP хорошо подходит для таких сред в качестве KMS.

- HIP не зависит от адресации IP и может транспортироваться напрямую по любому сетевому протоколу.
- Первичные ключи в протоколах 802 обычно являются парными и групповые ключи доставляются из группового контроллера с помощью парных ключей.
- Одноранговая KMS может лучше обслуживать специализированные (Ad-hoc) сети 802, чем модель «клиент-сервер» в EAP.
- Память в некоторых устройствах может быть сильно ограничена, а общая система KMS для защиты MAC и IP даёт существенную экономию кода.

A.3.3. HIP и IoT

HIP требует на устройстве наличия вычислительных ресурсов для криптографической обработки. Протокол может работать в телефонах и небольших устройствах system-on-chip (таких как Raspberry Pi, Intel Edison), но мелкие датчики со слабыми батареями остаются проблематичными. Были разработаны расширения HIP для переноса на мелкие устройства обычно со снижением уровня защиты. Например, были предложены некриптографические идентификаторы для RFID. Подход Slimfit [hummen] предлагает уровень сжатия для HIP, делающий протокол более подходящим для сетей с ограничениями. Этот подход применён в облегчённой версии HIP (Diet HIP) для мелких датчиков.

Обмен HIP Diet EXchange (DEX) [hip-dex] нацелен на снижение издержек, связанных с криптографическими примитивами, за чет отказа от подписей открытых ключей и хэш-функций. При этом сохраняется цель обеспечить свойства защиты, аналогичные базовому обмену (Base Exchange или BEX). Обмен DEX разработан прежде всего для устройств и датчиков с ограниченной памятью и вычислительными ресурсами. Предполагается, что он будет применяться с подходящим протоколом защиты данных вышележащего протокола, таким как ESP. Кроме того, DEX может служить механизмом ввода ключей для примитивов защиты на уровне MAC, например, для сетей IEEE 802.15.9 [IEEE.802.15.9]. Основные отличия между BEX от DEX указаны ниже.

1. Минимальный набор криптографических примитивов для снижения протокольных издержек:
   - статические пары ключей Elliptic Curve Diffie-Hellman (ECDH) для аутентификации и шифрования сеансового ключа;
   - AES-CTR для симметричного шифрования и AES-CMAC для функции MACing;
   - простая функций свёртки (fold) для генерации HIT.
2. Отказ совершенной защиты (PFS) с отменой эфемерного согласования ключей Diffie-Hellman.
3. Отказ от цифровых подписей с исключением хэш-функции. Использование выведенного из ECDH ключа в HIP_MAC для подтверждения владения секретным ключом.
4. Выводимый с помощью Diffie-Hellman ключ применяется **лишь** для защиты пакетов HIP. Для создания сеансовых ключей применяется отдельный обмен секретами в пакетах HIP.
5. Необязательная стратегия повтора передачи адаптирована для потенциально длительного выполнения криптографических на устройствах с ограниченными вычислительными ресурсами.

A.3.4. Инфраструктурные приложения

В отчёте об экспериментах с HIP [RFC6538] указан ряд реализаций клиентских и серверных приложений, опробованных с HIP. На основе этого отчёта ниже рассматриваются и дополняются некоторые возможные способы применения HIP в имеющейся инфраструктуре (маршрутизаторы, шлюзы, прокси).

HIP успешно применялся в пересылающих web-прокси (прокси у клиента) между хостом клиента (web-браузер) и пересылающим прокси (сервер Apache), завершающим туннель HIP/ESP. Пересылающий web-прокси транслировал основанный на HIP трафик от клиента в обычный (не HIP) трафик с направлении web-сервера в Internet. Поддерживающий HIP клиент мог взаимодействовать с не понимающими HIP серверами. Таким способом клиент мог использовать поддержку мобильности в HIP при использовании фиксированного IP-адреса на web-прокси, например, для доступа к услугам, которые разрешены лишь для адресов IP из диапазона прокси.

В случае с обратным web-прокси (на стороне сервера), который также был исследован [komu-cloud], не поддерживающий HIP клиент обращался к понимающему HIP сервису web через промежуточный балансировщик нагрузки (HAProxy). Балансировщик транслировал обычный (не HIP) трафик от клиента в трафик на основе HIP для web-сервиса (серверы front-end и back-end). Балансировщик и web-сервис размещались в ЦОД. Одним из ключевых преимуществ шифрования web-трафика с помощью HIP в этом случае была поддержка частного и публичного (гибридного) облака, где балансировщик и серверы front-end и back-end размещаются в разных ЦОД и трафик нужно защищать при передаче через потенциально незащищённые сети между границами частного и публичного облака.

Хотя HIP можно применять для защиты доступа к промежуточным устройствам (например, доступа к коммутаторам по протоколу telnet), он использовался также для защиты соединений в инфраструктуре промежуточных устройств. Например, в более ранним исследовании [komu-mitigation] HIP применялся между почтовыми серверами (Simple Mail Transport Protocol или SMTP) для применения вычислительной головоломки (puzzle) HIP в качестве механизма снижения спама. Достаточно очевидной проблемой в этом случае было отсутствие адаптации HIP на серверах SMTP.

Чтобы избежать проблем развёртывания в имеющейся инфраструктуре, HIP можно использовать в контексте новых протоколов с минимальным развёртыванием. HIP был изучен в контексте некого протокола peer-to-peer SIP [camarillo-p2psip], результатом чего стал набор связанных RFC [RFC6078], [RFC6079], [RFC7086]. Основная идея исследования состояла в предотвращении избыточных трудоёмких процедур ICE путём группировки разных соединений (т. е. SIP и

медиапотоки) вместе с использованием низкоуровневого HIP, выполняющего процедуру прохождения NAT лишь один раз на хост. Интересным аспектом было применение инфраструктуры P2P-SIP в качестве серверов встречи для плоскости управления HIP вместо использования традиционных служб HIP rendezvous [RFC8004].

Исследователи предложили применять HIP в сотовых сетях как решение для мобильности, множества адресов и защиты. В [hip-lte] приведён анализ защиты и моделирование применения HIP в транспортных сетях LTE.

HIP изучали в части защиты внутриоблачных соединений сначала с виртуальными машинами [komu-cloud], затем между контейнерами Linux [ranjbar-synaptic]. В обоих случаях HIP предоставлял решение для работы через NAT, которое подходило как внутри облака, так и между разными облаками. В частности, для первого случая HIP обеспечивал постоянную связность с виртуальной машиной при её переносе в другое место, а во втором контроллер программно-определяемой сети (Software-Defined Networking или SDN) служил сервером встреч для поддерживающих HIP контейнеров, обеспечивая надёжную защиту от повторного использования путём добавления middlebox nonce [heer-end-host] для базового обмена HIP и сообщений UPDATE.

A.3.5. Поддержка отождествлений в коммерческих системах

Tempered Networks выпускает продукцию на основе HIP, называя свою платформу сетью на основе отождествлений (Identity-Defined Networking или IDN) [tempered-networks] из за ориентированной на идентификацию архитектуры HIP. Задача заключалась в упрощении и бесперебойном развёртывании служб с поддержкой HIP в рабочих средах для обеспечения прозрачной аутентификации и проверки полномочий устройств, сокрытия, сегментации и сквозного взаимодействия в сети. Цель состоит в устранении большого числа циклических зависимостей, эксплойтов и сложности традиционных сетей на основе адресов, которые препятствуют мобильности и проверяемому контролю доступа к устройствам. Продукция Tempered Networks представляет HIP в нескольких типах устройств.

Коммутаторы и шлюзы HIP

Физические и виртуальные устройства, служащие шлюзами HIP и точками применения правил для приложений без поддержки HIP и расположенных за ними устройств. Для подключения, маскировки и защиты устройств без поддержки HIP не нужно менять IP или инфраструктуру. В настоящее время шлюзы HIP поддерживаются на устройствах x86 и ARM, а также в облаках ESXi, Hyper-V, KVM, AWS, Azure, Google.

Трансляторы и серверы встречи HIP

Физические и виртуальные устройства, служащие маршрутизаторами на основе отождествлений для проверки полномочий и соединения конечных точек HIP без шифрования сессий HIP. Ретранслятор HIP можно развернуть как автономное устройство или в кластере для горизонтальной расширяемости. Все конечные точки с поддержкой HIP, соединяемые и защищаемые ими устройства могут иметь приватные адреса. Платформы предотвращают конфликты IP, поддерживают туннели через NAT (включая NAT операторского класса) и не требуют менять базовую инфраструктуру. Единственным требованием является наличие у конечной точки HIP исходящего доступа в Internet а у ретранслятора HIP - публичного адреса.

Клиенты и серверы с поддержкой HIP

программы, устанавливаемые в сетевой стек хоста и обеспечивающие выполнение правил на хосте. Клиенты HIP поддерживают раздельное туннелирование. Клиенты и серверы HIP могут взаимодействовать с МСЭ на локальном хосте, а сервер можно заблокировать для прослушивания лишь применяемого для HIP порта, что делает его невидимым для неуполномоченных устройств. В настоящее время поддерживаются платформы Windows, OS X, iOS, Android, Ubuntu, CentOS и другие дистрибутивы Linux.

Менеджеры оркестровки правил

Физические и виртуальные устройства для задания и распространения правил сети и защиты (сопоставления HI и IP, наложенные сети, «белые» списки и т. п.) в конечные точки с поддержкой HIP. Оркестровка не требует сохранения в конечных точках HIP и наоборот, что позволяет создавать автономные сети и обеспечивать защиту.

A.4. Ответы на вопросы NSRG

Исследовательская группа IRTF Name Space задала в своём оценочном отчёте ряд вопросов [nsrg-report], ответы на которые представлены ниже.

1. Как имя стека улучшит общую функциональность Internet?

   HIP отделяет (меж)сетевой уровень от транспортного, позволяя им развиваться независимо. Разделение упрощает мобильность и многоадресность конечных хостов, а также позволяет работать через сети IPv4 и IPv6. HI упрощает смену адресов и миграцию процессов, а также реализацию кластеризованных серверов. Кроме того, криптографическая природа идентификаторов обеспечивает базу для решения вопросов безопасности, связанных с мобильностью и многоадресностью конечных хостов.

2. Как выглядит имя стека?

   HI является криптографическим открытым ключом, однако вместо непосредственного использования ключа многие протоколы применяют хэш открытого ключа с фиксированным размером.

3. Каков срок действия идентификатора?

   HIP поддерживает стабильные и временные HI. Стабильные HI обычно имеют длительный срок действия (годы), а для временных срок определяют соединения и приложения вышележащего уровня (от секунд до лет).

4. Где HI размещаются в стеке?

   Идентификаторы HI находятся между транспортным и (меж)сетевым уровнем.

5. Как HI используются конечными точками?

   Идентификаторы HI могут применяться приложениями напрямую или опосредовано (в форме HIT или LSI) при обращении к сетевым службам. Кроме того, HI как открытые ключи используются во встроенном протоколе согласования ключей, называемом базовым обменом HIP, для взаимной аутентификации хостов.

6. Какая административная инфраструктура требуется для поддержки?

В некоторых средах возможно применение HIP без какой-либо инфраструктуры, однако для полного использования преимуществ HIP идентификаторы HI должны храниться в DNS или PKI, а также нужен «механизм встречи (rendezvous) [RFC8005].

7. Не делает ли дополнительный уровень ненужным список адресов в SCTP?

   Да, список не нужен.

8. Какие дополнительные преимущества обеспечивает новая схема именования в части безопасности?

   HIP снижает зависимость от адресов IP, упрощая решение проблемы владения адресами [Nik2001]. На практике HIP обеспечивает защиту для мобильности и многоадресности конечных хостов. Кроме того, идентификаторы HI являются открытыми ключами и поверх HIP может применяться стандартная инфраструктура открытых ключей.

9. Каким может быть механизм распознавания и какие характеристики требуются от него?

   Для большинства случаев достаточно модели с преобразованием имён DNS сразу в HI и адреса IP. Однако, если требуется преобразование HI в адреса IP или их обратное преобразование в имена DNS, нужна инфраструктура плоского распознавания, которая может быть реализована на основе распределенных хэш-таблиц, но это потребует новых разработок и развёртывания.

Благодарности

Люди, участвовавшие в ранних этапах разработки HIP, перечислены в разделе благодарностей спецификации HIP. На финальных этапах подготовки этого документа, когда редактором стал Pekka Nikander, бесценны были комментарии ранних разработчиков и других людей, включая Jari Arkko, Jeff Ahrenholz, Tom Henderson, Petri Jokela, Miika Komu, Mika Kousa, Andrew McGregor, Jan Melen, Tim Shepard, Jukka Ylitalo, Sasu Tarkoma, Jorma Wall. Спасибо также Lars Eggert, Spencer Dawkins, Dave Crocker, Erik Giesa за полезные комментарии.

Авторы выражают особую благодарность Tom Henderson, взявшему на себя задачи редактирования документа в ответ на комментарии IESG, когда оба автора были заняты другими делами. Без его настойчивости документ не достиг бы уровня RFC 4423.

Основная работа по обновлению и продвижению HIP в рамках процесса IETF была проделана несколькими командами разработчиков HIP. Авторы признательны компании Boeing, Helsinki Institute for Information Technology (HIIT), NomadicLab из Ericsson и трём университетам - RWTH Aachen, Aalto, University of Helsinki за их усилия. Без коллективной работы протокол HIP засох бы на лозе IETF как прекрасная концепция.

Спасибо также Suvi Koskinen за помощь в корректуре и джунглях ссылок.

Адреса авторов

Robert Moskowitz (editor)
HTT Consulting
Oak Park, Michigan
United States of America
Email: rgm@labs.htt-consult.com

Miika Komu
Ericsson
Hirsalantie 11
FI-02420 Jorvas
Finland
Email: miika.komu@ericsson.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru