Internet Engineering Task Force (IETF)
Request for Comments: 5857
Category: Standards Track
ISSN: 2070-1721

E. Ertekin
C. Christou
R. Jasani
Booz Allen Hamilton
T. Kivinen
AuthenTec, Inc.
C. Bormann
Universitaet Bremen TZI

May 2010

Расширения IKEv2 для поддержки компрессии ROHC с IPsec

IKEv2 Extensions to Support Robust Header Compression over IPsec

Аннотация

Для интеграции ROHC[1] с IPSec требуется сигнальный механизм для передачи параметров канала ROHC между конечными точками. Механизм IKE[2] подходит для решения этой задачи. В этом документе предложены расширения IKEv2, позволяющие передавать сигнализацию ROHC и канальные параметры для защищённых связей IPsec.

Статус документа

Этот документ является проектом стандарта Internet (Internet Standards Track)..

Документ подготовлен IETF[3] и содержит согласованный взгляд сообщества IETF. Документ обсуждался публично и одобрен для публикации IESG[4]. Дополнительная информация о стандартах Internet приведена в разделе 2 RFC 5741.

Информацию о текущем состоянии данного документа, обнаруженных ошибках и способах обратной связи можно найти по ссылке http://www.rfc-editor.org/info/rfc5795.

Оглавление

1. Введение

Увеличение размера заголовков в результате использования IPsec [IPSEC] может приводить к неэффективному использованию полосы каналов. Объединение модели ROHC [ROHC] с IPsec обеспечивает эффективный способ передачи защищённого трафика IP.

ROHCoIPsec [ROHCOIPSEC] требует инициализации конфигурационных параметров на компрессоре и декомпрессоре. В современных спецификациях для поэтапного (hop-by-hop) сжатия ROHC эти параметры согласуются с

[1]Robust Header Compression - отказоустойчивое сжатие заголовков.
[2]Internet Key Exchange.
[3]Internet Engineering Task Force - комиссия по решению инженерных задач Internet.
[4]Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

использованием протоколов канального уровня типа PPP[1] (т. е. ROHC через PPP [ROHC-PPP]). Поскольку для динамического обмена параметрами между партнёрами IPsec можно использовать протоколы обмена ключами (например, IKEv2 [IKEV2]), в этом документе определяются расширения IKEv2 для передачи параметров ROHC в ROHCoIPsec.

2. Уровни требований

Ключевые слова **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не нужно** (SHALL NOT), **следует** (SHOULD), **не следует** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе должны интерпретироваться в соответствии с RFC 2119 [BRA97].

3. Инициализация канала ROHC для ROHCoIPsec

В последующих параграфах определены расширения IKEv2, которые позволят инициатору и ответчику обмениваться параметрами, требуемыми при организации канала ROHC для сеансов ROHCoIPsec.

3.1. Сообщение ROHC_SUPPORTED Notify

Параметры канала ROHC **должны** передаваться раздельно для каждой поддерживающей ROHC связи IPsec SA. В частности, должен быть включён новый тип сообщений Notify в обмены IKE_AUTH и CREATE_CHILD_SA при каждой организации IPsec SA с поддержкой ROHC или смене ключей в существующих связях.

Данные Notify, передаваемые инициатором, **должны** включать параметры канала для сессии ROHC. Эти параметры показывают возможности декомпрессора ROHC на стороне инициатора. При получении запроса от инициатора ответчик будет игнорировать эти данные (если он не поддерживает ROHC или предлагаемые параметры) или отвечать на них данными Notify, содержащими его параметры для канала ROHC.

Отметим, что для переноса параметров ROHC используется только одно поле Notify. При получении множества Notify с параметрами ROHC все такие данные Notify, кроме первых, **должны** быть отброшены. Если инициатор не получил данных Notify с параметрами канала ROHC у ответчика, включать ROHC для Child SA **недопустимо**.

Новое значение Notify Message Type, обозначаемое ROHC_SUPPORTED, показывает, что данные Notify передают параметры канала ROHC (параграф 3.1.2[2]).

Формат данных Notify (определён в 4306 [IKEV2]) показан на рисунке 1.

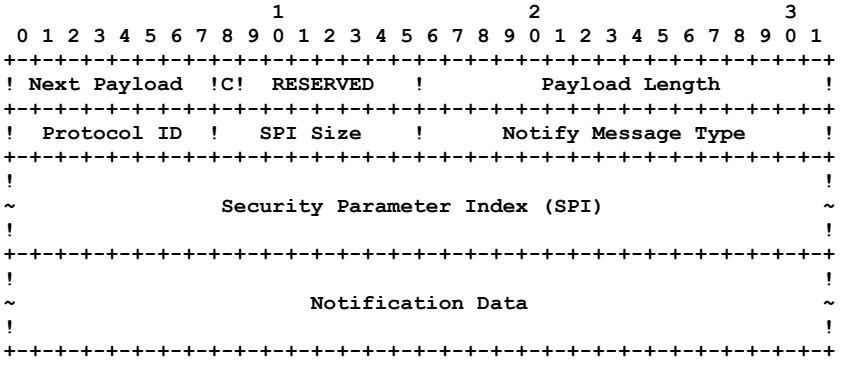

Рисунок 1. Формат Notify Payload.

Поля Notify Payload описаны ниже.

Next Payload (1 октет)
Идентификатор типа следующих данных в сообщении. Дополнительная информация в RFC 4306 [IKEV2].

Critical (1 бит)
Поскольку все реализации IKEv2 поддерживают данные Notify, это поле **должно** иметь значение 0.

Payload Length (2 октета)
В соответствии с определением RFC 4306 [IKEV2] это поле показывает размер данных (payload) вместе с их базовым заголовком.

Protocol ID (1 октет)
Поскольку эти сообщения используются в процессе создания Child SA, данное поле **должно** иметь значение 0.

SPI Size (1 октет)
Это поле **должно** иметь значение 0, поскольку применимых SPI нет (параметры ROHC задаются при создании SA, когда SPI не определены).

Notify Message Type (2 октета)
В это поле **должно** помещаться значение ROHC_SUPPORTED.

Security Parameter Index (SPI)
Поскольку поле SPI Size = 0, передавать данное поле **недопустимо**.

Notification Data (переменный размер)
В этом поле **должно** содержаться не менее трёх атрибутов ROHC (см. параграф 3.1.1).

3.1.1. Атрибуты ROHC

Сообщение ROHC_SUPPORTED Notify используется для передачи канальных параметров между компрессором и декомпрессором ROHCoIPsec. Сообщение содержит список атрибутов ROHC, включающих параметры, требуемые для сеанса ROHCoIPsec.

Формат сигнальных атрибутов ROHC похож на формат атрибутов преобразования, описанных в параграфе 3.3.5 документа RFC 4306 [IKEV2]. Формат атрибута ROHC показан на рисунке 2.

[1] Point-to-Point Protocol - протокол «точка-точка».
[2] В оригинале ошибочно приведена ссылка на раздел 4. См. http://www.rfc-editor.org/errata_search.php?rfc=5857. *Прим. перев.*

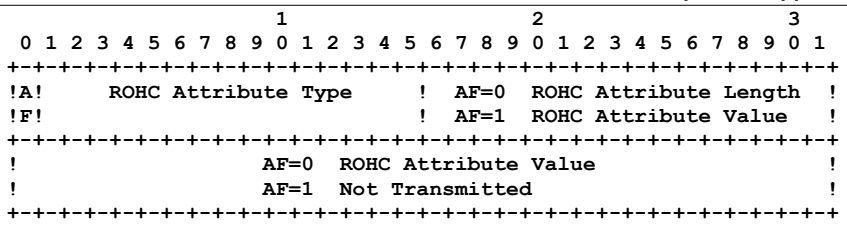

Рисунок 2. Формат атрибута ROHC.

- **Attribute Format (AF)** (1 бит) - если AF=0, атрибут ROHC представляется в формате TLV[1]. При AF=1 атрибут указывается в формате TV (тип/значение).
- **ROHC Attribute Type** (15 битов) - уникальный идентификатор для каждого типа атрибутов ROHC (см. параграф 3.1.2).
- **ROHC Attribute Length** (2 октета) - размер Attribute Value в октетах. Если AF=1, значение атрибута ROHC занимает 2 октета и поле ROHC Attribute Length отсутствует.
- **ROHC Attribute Value** (переменный размер) - значение атрибута ROHC, связанное с ROHC Attribute Type. Если AF=0, размер этого поля определяется значением поля ROHC Attribute Length. При AF=1 размер поля ROHC Attribute Value составляет 2 октета.

3.1.2. Типы атрибутов ROHC

В этом параграфе описаны типы атрибутов ROHC: MAX_CID, ROHC_PROFILE, ROHC_INTEG, ROHC_ICV_LEN и MRRU. Связанные с каждым из типов значения рассмотрены в разделе 5[2].

MAX_CID (Максимальное значение идентификатора контекста, AF = 1)

Атрибут MAX_CID является обязательным и **должен** передаваться в единственном экземпляре. Поле MAX_CID указывает максимальное значение идентификатора контекста, поддерживаемое декомпрессором ROHCoIPsec. Значение атрибута имеет размер 2 октета и **должно** находиться в диапазоне от 0 до 16383, включительно. Поскольку идентификаторы CID могут принимать значения от 0 до MAX_CID, число контекстов, которые могут использоваться реально, составляет MAX_CID+1. Значение MAX_CID = 0 говорит о единственном контексте. Получатель атрибута MAX_CID **должен** использовать для компрессии только контексты, номера атрибутов для которых не превышают MAX_CID.

Отметим, что параметр MAX_CID является односторонним уведомлением (т. е.,отправитель атрибута показывает свои возможности другой стороне), поэтому в каждом направление может использоваться своё значение MAX_CID.

ROHC_PROFILE (профиль ROHC, AF = 1)

Атрибут ROHC_PROFILE является обязательным. Каждый атрибут ROHC_PROFILE имеет фиксированный размер в 4 октета, а значением атрибута является 2-октетный идентификатор профиля ROHC [ROHCPROF]. В сообщение ROHC_SUPPORTED Notify **должен** включаться по крайней мере один атрибут ROHC_PROFILE. При передаче множества атрибутов ROHC_PROFILE они могут размещаться в произвольном порядке. Получатель атрибутов ROHC_PROFILE **должен** использовать для компрессии только предложенные этими атрибутами профили.

Некоторые профили общего назначения определены в RFC 3095 [ROHCV1] и RFC 5225 [ROHCV2]. Следует отметить, что передавать в атрибутах две версии одного профиля **недопустимо**. Например, если декомпрессор ROHCoIPsec поддерживает ROHCv1 UDP (0x0002) и ROHCv2 UDP (0x0102), **недопустимо** указывать оба профиля. Это ограничение обусловлено тем, что пакеты, сжатые с использованием ROHC, содержат только 8 младших битов идентификатора профиля, а они совпадают для профилей ROHCv1 и ROHCv2, поэтому декомпрессор не сможет определить версию профиля, использованного для сжатия пакета.

Отметим, что атрибут ROHC_PROFILE является односторонним уведомлением и в каждом направлении могут анонсироваться разные наборы ROHC_PROFILE.

ROHC_INTEG (Алгоритм контроля целостности для проверки заголовков после декомпрессии, AF = 1)

Атрибут ROHC_INTEG является обязательным и в каждом сообщении ROHC_SUPPORTED Notify **должно** указываться не менее одного атрибута ROHC_INTEG. Значение атрибута содержит идентификатор алгоритма контроля целостности, используемого для обеспечения целостности декомпрессированных пакетов (т. е., гарантии совпадения заголовка пакета после декомпрессии с заголовком исходного пакета до сжатия).

Алгоритмы аутентификации, которые **должны** поддерживаться, заданы в таблице «Алгоритмы аутентификации» параграфа 3.1.1 (Алгоритмы шифрования и аутентификации ESP) RFC 4835 [CRYPTO-ALG] (или его замены).

Алгоритм контроля целостности представляется 2-октетным значением, соответствующим значению из реестра параметров IKEv2 [IKEV2-PARA], раздел Transform Type 3 - Integrity Algorithm Transform IDs[3]. При получении атрибутов ROHC_INTEG ответчик **должен** выбрать единственный из предложенных алгоритмов и передать свой выбор в сообщении ROHC_SUPPORTED Notify, адресованном инициатору. Выбранный алгоритм контроля целостности **должен** использоваться для обоих направлений. Если ответчик не принимает ни одного из предложенных алгоритмов, включать ROHC для данной SA **недопустимо**.

Использование алгоритма контроля целостности включает ряд перечисленных ниже аспектов.

1. Ключи (по одному для каждого направления) для алгоритма контроля целостности создаются из IKEv2 KEYMAT (см. [IKEV2], параграф 2.17). При создании ключей ROHC рассматривается, как протокол IPsec. При смене ключей для поддерживающей ROHC связи CHILD_SA меняются и ключи, связанные с этим алгоритмом контроля целостности.
2. Инициатор ROHCoIPsec **может** передать в атрибуте ROHC_INTEG нулевое значение (0x0000). Это соответствует значению NONE из реестра IKEv2 Integrity Algorithm Transform IDs. Ответчик ROHCoIPsec **может** выбрать это значение, отвечая инициатору атрибутом ROHC_INTEG = 0x0000. В этом случае для обоих направлений алгоритм контроля целостности не применяется.

[1]Type/Length/Value - тип - размер - значение.
[2]В оригинале ошибочно приведена ссылка на раздел 4. См. http://www.rfc-editor.org/errata_search.php?rfc=5857. *Прим. перев.*
[3]Тип преобразования 3 - Идентификаторы преобразования для алгоритмов контроля целостности.

3. ROHC_INTEG является параметром, согласуемым обеими сторонами. Иными словами, инициатор показывает поддерживаемые им алгоритмы, а ответчик выбирает одно из предложенных значений ROHC_INTEG и передаёт свой выбор инициатору.

ROHC_ICV_LEN (Размер алгоритма контроля целостности, AF = 1)

Атрибут ROHC_ICV_LEN относится к числу необязательных. В сообщения ROHC_SUPPORTED Notify **может** включаться множество атрибутов ROHC_ICV_LEN. Атрибут задает число октетов значения ICV[1], которые отправитель сообщения ожидает получить во входящих пакетах ROHC. Значение ICV согласованного алгоритма ROHC_INTEG **должно** сокращаться до ROHC_ICV_LEN байтов, путём отбрасывания лишних октетов в конце. Инициатор и ответчик анонсируют свои значения размера ICV. Получатель атрибута ROHC_ICV_LEN **должен** уменьшить размер ICV до величины, указанной в сообщении. Если ROHC_ICV_LEN = 0, передача ICV **недопустима**. Если атрибут ROHC_ICV_LEN не был задан совсем или его значение превышает размер ICV для выбранного алгоритма, **должно** использоваться полное значение ICV соответствующего алгоритма ROHC_INTEG . Отметим, что атрибут ROHC_ICV_LEN служит односторонним уведомлением и в каждом направлении может анонсироваться своё значение ROHC_ICV_LEN.

MRRU (Максимальный восстанавливаемый блок для приёма, AF = 1)

Атрибут MRRU относится к числу необязательных. В сообщения ROHC_SUPPORTED Notify **может** включаться не более одного атрибута MRRU. Размер атрибута составляет 2 октета. Атрибут задает размер (в октетах) максимального блока, который декомпрессор ROHCoIPsec ожидает для восстановления из сегментов ROHC (см. параграф 5.2.5 [ROHCV1]). Этот размер включает поля контрольной суммы (CRC) и ROHC ICV. Если MRRU = 0 или значение MRRU не задано, сегменты заголовков **недопустимо** передавать в канал ROHCoIPsec.

Отметим, что атрибут MRRU служит односторонним уведомлением и в каждом направлении может анонсироваться своё значение MRRU.

При получении атрибута ROHC неизвестного типа, такой атрибут **должен** отбрасываться без уведомления.

3.2. Неявно устанавливаемые канальные параметры ROHC

Перечисленные ниже параметры каналов ROHC **недопустимо** передавать в сигнализации.

- **LARGE_CIDS**. Это значение неявно определяется значением MAX_CID (т. е., при MAX_CID <= 15, принимается LARGE_CIDS = 0).
- **FEEDBACK_FOR**. При создании пары SA (по одной для каждого направления) параметр FEEDBACK_FOR для канала ROHC **должен** неявно задаваться для другой SA в паре (т. е., SA в обратном направлении).

4. Вопросы безопасности

Возможность согласования размера ROHC ICV может создавать уязвимость для протокола ROHCoIPsec. В частности, возможность задать малый размер может приводить к ситуациям, когда в защищённый домен будут пересылаться ошибочные пакеты. Более детально эта проблема рассмотрена в параграфе 6.1.4 документа [ROHCOIPSEC] и в разделе 5 [IPSEC-ROHC].

Проблему можно смягчить за счёт использования больших ICV, но это приведёт к добавочным издержкам и снизит эффективность, обеспечиваемую ROHCoIPsec.

5. Согласование с IANA

В этом документе определено новое сообщение Notify (Status Type). Следовательно, агентство IANA выделило одно значение из реестра «IKEv2 Notify Message Types» для индикации поддержки ROHC (ROHC_SUPPORTED).

В дополнение к этому агентство IANA создало новый субреестр ROHC Attribute Types в рамках реестра Internet Key Exchange Version 2 (IKEv2) Parameters [IKEV2-PARA]. В реестре ROHC Attribute Types этот документ выделяет значения, перечисленные в таблице.

Значение в реестре	***Тип атрибута ROHC***	***Формат***	***Документ***
0	Резерв		[RFC5857]
1	Максимальный идентификатор контекста MAX_CID	TV	[RFC5857]
2	Профиль ROHC (ROHC_PROFILE)	TV	[RFC5857]
3	Алгоритм контроля целостности ROHC (ROHC_INTEG)	TV	[RFC5857]
4	Размер ROHC ICV в байтах (ROHC_ICV_LEN)	TV	[RFC5857]
5	Максимальный восстанавливаемый блок на приёме (MRRU)	TV	[RFC5857]
6-16383	Не распределены	TV	[RFC5857]
16384-32767	Для приватного использования	TV	[RFC5857]

В соответствии с [IANA-CONSIDERATIONS] выделение IANA новых значений для типов атрибутов ROHC должно осуществляться после рецензии эксперта (Expert Review).

Для регистрационных запросов ответственный руководитель направления IESG (IESG Area Director) будет назначать эксперта (Designated Expert), который будет направлять запрос в списки рассылки ROHC и IPsec (или заменяющие их списки, указанные руководителем направления) для получения комментариев и рецензий. После этого назначенный эксперт будет принимать или отвергать запрос на регистрацию, направляя своё решение в оба списка рассылки и информируя IANA. Отказ от регистрации должен быть мотивирован.

6. Благодарности

Авторы благодарят Sean O'Keeffe, James Kohler, и Linda Noone из Министерства Обороны[2], а также Rich Espy из OPnet за их вклад и поддержку при разработке этого документа.

Авторы также благодарны Yoav Nir и Robert A Stangarone Jr., которые выступили рецензентами данной спецификации.

[1] Integrity Check Value - значение для проверки целостности.
[2] США. *Прим. перев.*

Кроме того, авторы рады поблагодарить перечисленных ниже людей за их рецензии и комментарии к документу:

- Magnus Westerlund
- Stephen Kent
- Lars-Erik Jonsson
- Pasi Eronen
- Jonah Pezeshki
- Carl Knutsson
- Joseph Touch
- David Black
- Glen Zorn

В заключение авторы выражают свою признательность Tom Conkle, Michele Casey и Etzel Brower.

7. Литература

7.1. Нормативные документы

[IPSEC] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[ROHC] Sandlund, K., Pelletier, G., and L-E. Jonsson, "The RObust Header Compression (ROHC) Framework", RFC 5795, March 2010.

[IKEV2] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005.

[BRA97] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[ROHCV1] Bormann, C., Burmeister, C., Degermark, M., Fukushima, H., Hannu, H., Jonsson, L-E., Hakenberg, R., Koren, T., Le, K., Liu, Z., Martensson, A., Miyazaki, A., Svanbro, K., Wiebke, T., Yoshimura, T., and H. Zheng, "Robust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed", RFC 3095, July 2001.

[ROHCV2] Pelletier, G. and K. Sandlund, "RObust Header Compression Version 2 (ROHCv2): Profiles for RTP, UDP, IP, ESP and UDP-Lite", RFC 5225, April 2008.

[IPSEC-ROHC] Ertekin, E., Christou, C., and C. Bormann, "IPsec Extensions to Support Robust Header Compression over IPsec", RFC 5858, May 2010.

[IANA-CONSIDERATIONS] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

7.2. Дополнительная литература

[ROHCOIPSEC] Ertekin, E., Jasani, R., Christou, C., and C. Bormann, "Integration of Header Compression over IPsec Security Associations", RFC 5856, May 2010.

[ROHC-PPP] Bormann, C., "Robust Header Compression (ROHC) over PPP", RFC 3241, April 2002.

[ROHCPROF] IANA, "RObust Header Compression (ROHC) Profile Identifiers", <http://www.iana.org>.

[CRYPTO-ALG] Manral, V., "Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)", RFC 4835, April 2007.

[IKEV2-PARA] IANA, "Internet Key Exchange Version 2 (Kev2) Parameters", <http://www.iana.org>.

Адреса авторов

Emre Ertekin
Booz Allen Hamilton
5220 Pacific Concourse Drive, Suite 200
Los Angeles, CA 90045
US
EMail: ertekin_emre@bah.com

Chris Christou
Booz Allen Hamilton
13200 Woodland Park Dr.
Herndon, VA 20171
US
EMail: christou_chris@bah.com

Rohan Jasani
Booz Allen Hamilton
13200 Woodland Park Dr.
Herndon, VA 20171
US
EMail: ro@breakcheck.com

Tero Kivinen
AuthenTec, Inc.
Fredrikinkatu 47
HELSINKI
FI
EMail: kivinen@iki.fi

Carsten Bormann
Universitaet Bremen TZI
Postfach 330440
Bremen D-28334
Germany
EMail: cabo@tzi.org

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru