

Network Working Group  
Request for Comments: 3173  
Obsoletes: 2393  
Category: Standards Track

A. Shacham  
Juniper  
B. Monsour  
Consultant  
R. Pereira  
Cisco  
M. Thomas  
Consultant  
September 2001

## Протокол компрессии данных IP

IP Payload Compression Protocol (IPComp)

### Статус документа

Этот документ содержит спецификацию предложенного стандарта для сообщества Internet и служит запросом к обсуждению в целях совершенствования протокола. Текущее состояние стандартизации можно выяснить из актуальной версии документа Internet Official Protocol Standards (STD 1). Документ может распространяться свободно.

### Авторские права

Copyright (C) The Internet Society (2001). All Rights Reserved.

### Аннотация

Документ содержит описание протокола, предназначенного для неразрушающей компрессии дейтаграмм IP в среде Internet.

### 1. Введение

Протокол сжатия данных (payload) IP (IPComp) предназначен для снижения размера дейтаграмм IP. Этот протокол будет повышать общую производительность связи между парой обменивающихся данными устройств (хосты, шлюзы), которые будем для простоты называть узлами (node), за счёт сжатия дейтаграмм, обеспечиваемого вычислительными ресурсами узлов (основного процессора - CPU или специального сопроцессора компрессии), при передаче данных по низкоскоростным или загруженным каналам.

Компрессия данных IP особенно полезна для зашифрованных дейтаграмм IP. Шифрование дейтаграмм делает распределение кодов случайным, поэтому компрессия на нижележащих уровнях (например, PPP Compression Control Protocol [RFC1962]) становится неэффективной. При совместном использовании компрессия должна выполняться до шифрования.

Этот документ определяет протокол компрессии данных IP (IPComp), структуру пакетов IPComp, ассоциации IPComp (IPCA) и несколько методов согласования IPCA.

Использование алгоритмов компрессии для протокола IPComp должно быть рассмотрено в других документах, поскольку этот вопрос выходит за пределы данной спецификации.

### Спецификация 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 [RFC2119].

### 2. Процесс сжатия

Процесс компрессии дейтаграмм IP состоит из 2 фаз – сжатие исходящих дейтаграмм IP (compression - компрессия) и декомпрессия входящих дейтаграмм (decompression). Процесс компрессии **должен** осуществляться без потерь (lossless), чтобы дейтаграммы IP после декомпрессии оставались тождественными исходным дейтаграммам.

Процессы сжатия и декомпрессии выполняются независимо для каждой дейтаграммы IP, вне связи с компрессией других дейтаграмм (stateless compression), поскольку доставка дейтаграмм IP может происходить с нарушением их порядка, а некоторые дейтаграммы могут просто оказаться недоставленными. Каждая сжатая дейтаграмма IP инкапсулирует одну порцию пользовательских данных (single IP payload).

На приёмной стороне **должна** обеспечиваться обработка сжатых и несжатых дейтаграмм IP для того, в соответствии с политикой pop-expansion, описанной в параграфе 2.2.

Компрессия исходящих дейтаграмм IP **должна** выполняться до того, как начнётся какая-либо обработка, связанная с безопасностью IP (например, шифрование или аутентификация) и до фрагментации дейтаграмм IP. Кроме того, в IPv6 [RFC2460] компрессия исходящих дейтаграмм должна выполняться перед добавлением заголовка Hop-by-Hop Options или Routing Header, поскольку оба эти заголовка в обоих заголовках содержится информация, которая может проверяться и обрабатываться каждым узлом на пути доставки пакета, и, следовательно, **должна** передаваться в исходном виде.

Подобно этому декомпрессия принятых дейтаграмм IP **должна** происходить после сборки фрагментов и выполнения операций, связанных с безопасностью, типа аутентификации и шифрования.

## 2.1. Сжатые данные

Сжатие применяется к одному массиву октетов, который является непрерывным в дейтаграмме IP. Этот массив всегда завершается на последнем октете данных в пакете IP. Отметим, что непрерывный массив октетов дейтаграммы может быть фрагментирован в физической памяти узла.

В IPv4 [RFC0791] компрессия применяется к пользовательским данным (payload) дейтаграммы IP, начиная с первого октета после заголовка IP и продолжается до последнего октета дейтаграммы. Ни одна часть заголовка IP или опций IP не сжимается. Отметим, что в случае инкапсулированного заголовка IP (например, туннельная инкапсуляция в Ipsec) начало данных в дейтаграмме располагается сразу же после внешнего (outer) заголовка IP. В результате вложенный (инкапсулированный) заголовок IP рассматривается как часть данных и сжимается.

В контексте IPv6 протокол IPComp рассматривается как данные на всем пути (end-to-end) и **не требуется** поэтапный (hop-by-hop) анализ и преобразование заголовков. В результате компрессия используется, начиная с первого поля IP Header Option, которое не содержит информации, проверяемой и обрабатываемой узлами на пути доставки пакета (если такое поле имеется в заголовке) и продолжается до данных ULP в дейтаграмме IP.

Размер данных после использования алгоритма сжатия **должен** составлять целое число октетов.

Как показано в параграфе 3, заголовок IPComp вставляется непосредственно перед сжатыми данными. Исходный заголовок IP изменяется для учёта нового размера дейтаграммы и индикации протокола IPComp. Исходное содержимое Next Header (IPv6) или поля протокола (IPv4) сохраняется в заголовке IPComp.

Декомпрессия применяется к одному непрерывному массиву октетов дейтаграммы IP. Массив начинается сразу же после заголовка IPComp и заканчивается последним байтом данных в дейтаграмме IP. При успешном завершении декомпрессии заголовок IP изменяется (восстанавливается) с учётом изменения размера дейтаграммы и поля next header (или протокол), сохранённого в заголовке IPComp. Заголовок IPComp удаляется из дейтаграммы IP и данные после декомпрессии размещаются вслед за оригинальным заголовком IP.

## 2.2. Правило Non-Expansion

Если общий размер сжатых данных и заголовка IPComp (см. Параграф 3), не меньше размера исходных данных, дейтаграмма IP **должна** передаваться в исходном виде (без компрессии). При передаче дейтаграммы IP без компрессии заголовок IPComp не добавляется в дейтаграмму. Это правило избавляет от затраты времени на бессмысленную декомпрессию и предотвращает избыточную фрагментацию, когда размер дейтаграммы после декомпрессии превысит MTU.

Очевидно, что небольшие дейтаграммы IP могут увеличиваться в размере после компрессии. Следовательно, перед компрессией дейтаграмм их размер сравнивается с пороговым значением и дейтаграммы с размером меньше порогового не сжимаются. Значение порогового уровня зависит от реализации.

Дейтаграммы IP со сжатыми данными нет смысла подвергать дополнительной компрессии. Предыдущая компрессия данных может быть результатом внешних процессов, выполненных протоколами более высоких уровней или внешними системами сжатия. Во избежание ненужного расхода ресурсов следует использовать адаптивные алгоритмы компрессии. Например, если сжатие  $i$  последовательных дейтаграмм IP с помощью IPComp даёт отрицательный результат, следующие дейтаграммы (скажем,  $k$ ) передаются без попыток сжатия. Если сжатие последующих  $j$  дейтаграмм также не даёт результата, без попыток компрессии передаётся большее число дейтаграмм ( $k + n$ ). После успешного сжатия дейтаграммы восстанавливается обычная работа IPComp. Выбор адаптивного алгоритма и пороговых значений зависит от реализации.

В процессе обработки сжимаемых данных алгоритм **может** проводить периодические проверки эффективности сжатия подобно требованиям [V42BIS]. Природа проверки будет зависеть от используемого алгоритма компрессии. После того, как проверка показала несжимаемость данных, алгоритму следует прекратить дальнейшую обработку данных и дейтаграмма должна передаваться без компрессии.

## 3. Структура заголовка сжатых дейтаграмм IP

Сжатые дейтаграммы IP инкапсулируются путём изменения стандартного заголовка IP и включения после него заголовка IPComp, непосредственно за которым располагаются сжатые данные. В этом разделе определяется процедура изменения заголовков IPv4 и IPv6, а также структура заголовка IPComp.

### 3.1. Изменение заголовка IPv4

Перед отправкой сжатой дейтаграммы изменяются следующие поля заголовка IPv4:

#### **Total Length**

Общая длина инкапсулированной дейтаграммы IP с учётом стандартного заголовка IP, заголовка IPComp и сжатых данных.

#### **Protocol**

В поле протокола помещается значение 108, выделенное для дейтаграмм IPComp [RFC1700].

#### **Header Checksum**

Контрольная сумма заголовка IP [RFC0791].

Остальные поля заголовка IPv4 (включая поле опций) не изменяются.

### 3.2. Изменение заголовка IPv6

Перед отправкой сжатой дейтаграммы изменяются следующие поля заголовка IPv6:

#### **Payload Length**

Размер сжатых данных IP.

#### **Next Header**

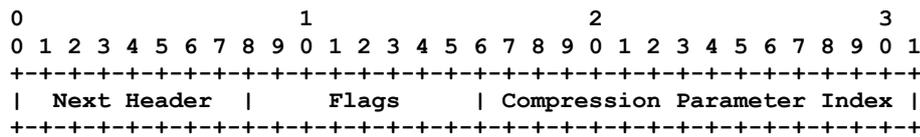
В поле Next Header помещается значение 108 – идентификатор дейтаграммы IPComp [RFC1700].

Все остальные поля заголовка IPv6 (включая все несжатые поля) сохраняются неизменными.

Заголовок IPComp помещается в пакет IPv6 с использованием таких же правил, как для заголовка IPv6 Fragment Header. Однако для пакетов Ipv6, содержащих одновременно заголовки IPv6 Fragment Header и IPComp, заголовок IPv6 Fragment Header **должен** предшествовать IPComp. Отметим, что между IPv6 Fragment Header и заголовком IPComp могут размещаться другие заголовки IPv6.

### 3.3. Структура заголовка IPComp

4-октетный заголовок IPComp имеет следующую структуру:



#### Next Header

8-битовый селектор, сохраняющий значение поля Protocol (IPv4) или IPv6 Next Header из оригинального заголовка IP.

#### Flags

8-битовое поле флагов зарезервировано для использования в будущем и **должно** иметь нулевое значение. На приёмной стороне это поле **должно** игнорироваться.

#### Compression Parameter Index (CPI)

16-битовый индекс, определяющий тип компрессии. Значения 0-63 обозначают хорошо известные алгоритмы компрессии, не требующие дополнительной информации и устанавливаемые вручную. Сами значения индексов совпадают с идентификаторами IPCOMP Transform, определёнными в [SECDOI]. Для выяснения распределённых значений и получения инструкций по выделению новых индексов следует обращаться к работе [SECDOI]. Значения 64-255 зарезервированы для использования в будущем. Значения 256-61439 согласуются между парой узлов при создании IPComp Association<sup>1</sup>, как описано в разделе 4. Значения 61440-65535 выделены для частного использования по согласованию сторон. Оба узла могут выбирать значения CPI независимо и не существует каких-либо соотношений между выбранными каждой стороной CPI. Заголовок IPComp в исходящих пакетах должен использовать значение CPI, выбранное принимающей стороной. CPI вместе с IP-адресом получателя обеспечивает уникальную идентификацию параметров компрессии для дейтаграмм.

## 4. Согласование IPComp Association (IPCA)

Для использования протокола IPComp два узла **должны** сначала создать ассоциацию IPComp Association (IPCA) между собой. IPCA включает всю информацию, требуемую для работы IPComp, включая значения CPI, режим работы, используемые алгоритмы компрессии и все требуемые для выбранных алгоритмов параметры.

Правила организации IPComp могут задаваться на уровне пары узлов (тогда IPComp используется для всего трафика между узлами) или на уровне сеансов, когда компрессия используется только для выбранных сессий между парой узлов.

Пара узлов может согласовать использование IPComp в одном или обоих направлениях и допускается использование разных алгоритмов компрессии для каждого направления. Узлы, однако, **должны** согласовать между собой алгоритм компрессии для каждого из направлений, в котором они будут использовать IPCA – по умолчанию никакой алгоритм компрессии не определён.

Никакой из алгоритмов компрессии не является обязательным для реализации IPComp.

Ассоциации IPCA создаются путём динамического согласования параметров или вручную. При динамическом согласовании следует использовать протокол обмена ключами Internet Key Exchange [IKE] с IPsec. Динамическое согласование **может** выполняться на основе разных протоколов.

### 4.1. Использование IKE

Для IPComp в контексте безопасности IP протокол IKE обеспечивает требуемые механизмы и рекомендации по созданию IPCA. Используя IKE, протокол IPComp может согласовывать ассоциации как автономный или совместно с другими протоколами IPsec.

Ассоциации IPComp согласуются инициатором с использованием Proposal Payload (предложенные данные) и включением Transform Payload. Proposal Payload задаёт протокол компрессии данных IP (Payload Compression Protocol) в поле идентификатора протокола, а каждый элемент Transform Payload содержит конкретный алгоритм, предлагаемый другой стороне.

Значение CPI передаётся в поле SPI с соответствующим значением поля размера SPI. Значение CPI **следует** передавать как 16-битовое целое, устанавливая в поле размера SPI значение 2. **Возможна** также передача CPI в форме 32-битового значения с установкой размера SPI = 4. В этом случае 16-битовое значение CPI **должно** помещаться в два младших октета поля SPI, а старшие октеты **должны** содержать нулевое значение и приёмная сторона должна игнорировать эти октеты. Принимающий узел **должен** уметь обрабатывать обе формы предложения CPI.

В домене интерпретации IP Security (Internet IP Security Domain of Interpretation или DOI) протокол IPComp должен согласовываться как Protocol ID PROTO\_IPCOMP. Алгоритм компрессии согласуется как один из определённых транспортных идентификаторов IPCOMP (Transform Identifier).

Предложения IPComp могут содержать следующие атрибуты:

#### Encapsulation Mode

Чтобы предложить нестандартный режим инкапсуляции (например, Tunnel Mode), предложение IPComp **должно** включать атрибут Encapsulation Mode. Если этот атрибут не задан, используется принятая по умолчанию инкапсуляция Transport Mode.

#### Lifetime

<sup>1</sup>Отметим, что при согласовании одного из хорошо известных алгоритмов узлы **могут** выбрать CPI из предопределённого диапазона 0-63.

Предложения IPComp используют атрибуты Life Duration (время жизни) и Life Type (жизненный тип) для определения времени жизни IPCA.

Когда согласование IPComp является частью Protection Suite, все логически связанные предложения должны быть согласованы. Однако в предложения IPComp **не следует** включать атрибуты, неприменимые к IPComp. **Недопустимо** отвергать предложения IPComp из-за того, что они не включают атрибутов других протоколов Protection Suite, не имеющих отношения к IPComp. Когда предложение IPComp включает такие атрибуты, они **должны** игнорироваться при создании ассоциации IPCA и, следовательно, не учитываются в работе IPComp.

Замечания для разработчиков

1. Узел может избавиться от необходимости вычислений для определения алгоритма компрессии из SPI, используя один из хорошо известных алгоритмов – это может сократить время декомпрессии. Для решения этой задачи узел может согласовать SPI, значение которого совпадает с одним из предопределённых идентификаторов Transform для данного алгоритма компрессии. В частности, узел **может** предложить SPI из предопределённого диапазона, передавая Proposal Payload с **единственным** значением Transform Payload, которое совпадает с SPI. Когда предлагается более одного значения Transform Payload, узел **может** предложить SPI из предопределённого диапазона, используя множество предложений IPComp, каждое из которых **должно** включать единственное значение Transform Payload. Иными словами, если Proposal Payload содержит более одного значения Transform Payload, значения SPI должны находиться в согласованном диапазоне. Принимающий узел **должен** быть способен обрабатывать каждую из предложенных форм.
2. Ассоциации IPCA перестают быть уникальными при организации двух или более сеансов IPComp между парой узлов и использовании одинаковых SPI из числа хорошо известных по крайней мере для двух сессий. Отсутствие уникальности IPCA порождает проблемы при поддержке специфических для каждой ассоциации IPCA атрибутов – согласованных (например, время жизни) или внутренних (например, счётчики адаптивного алгоритма для обработки предварительно сжатого трафика). Для обеспечения уникальности всех IPCA данной пары узлов при наличии двух и более ассоциаций IPCA, использующих одинаковый алгоритм компрессии, значения SPI следует выбирать из согласованного диапазона. Однако в тех случаях, когда уникальность IPCA не требуется (например, при использовании IPCA без атрибутов), **можно** использовать хорошо известные SPI. Отметим, что ассоциация IPCA является уникальной, когда между парой узлов существует единственный сеанс, использующий данное хорошо известное значение SPI.

## 4.2. Использование протоколов, отличных от IKE

Динамическое согласование **может** выполняться не только на основе протокола IKE, но этот вопрос выходит за пределы данной спецификации.

## 4.3. Ручная настройка конфигурации

Можно создавать ассоциации IPCA (IPComp Associations) между парами узлов путём настройки конфигурации вручную. Для этого варианта выделен ограниченный диапазон индексов SPI, представляющих алгоритмы компрессии.

## 5. Вопросы безопасности

При использовании IPComp в контексте IPsec, предполагается отсутствие влияния протокола компрессии на функционирование средств обеспечения безопасности IPsec – использование компрессии не должно снижать или изменять природу нижележащей архитектуры обеспечения безопасности или используемых для её реализации методов шифрования.

При использовании IPComp без IPsec, компрессия данных IP может снижать уровень безопасности в Internet, подобно IP-инкапсуляции [RFC2003]. Например, IPComp может затруднять работу граничных маршрутизаторов по фильтрации дейтаграмм на основе полей заголовков. В частности, исходное значение поля Protocol из заголовка IP перемещается в заголовок IPComp, а все заголовки транспортного уровня внутри дейтаграммы (такие, как номера портов) просто оказываются в сжатой части дейтаграммы и напрямую недоступны. Фильтрующий граничный маршрутизатор сможет выполнять фильтрацию только в тех случаях, когда он принимает участие в ассоциации IPCA, используемой для компрессии. Для использования компрессии в средах. Где требуется фильтрация (или, по крайней мере, учёт) всех пакетов, для принимающих узлов должен обеспечиваться механизм безопасного обмена IPCA с граничными маршрутизаторами. Это (в более редких случаях) может быть применимо к IPCA, используемым для исходящих дейтаграмм.

## 6. Взаимодействие с IANA

Этот документ не требует каких либо действий со стороны IANA. Используемые в данной спецификации хорошо известные номера уже определены в других документах [SECD01].

## 7. Отличия от RFC 2393

В этом параграфе перечислены изменения, внесённые в данный документ по сравнению с RFC 2393, о которых должны быть предупреждены разработчики, использующие RFC 2393. Все изменения связаны с уточнением процедуры организации IPCA (IPComp Association) с использованием протокола IKE [IKE] в контексте IPsec.

- 1) Уточнены процедуры согласования при автономном использовании IPComp и в случаях совместной работы с другими протоколами Protection Suite.
- 2) Определена передача SPI в поле SPI предложений IKE – **следует** использовать двухоктетные поля, но **могут** применяться и 4-октетные. Определено размещение 16-битовых значений SPI в 4-октетном поле. Указано, что получатель **должен** обрабатывать поля обоих размеров.
- 3) Добавлено использование по умолчанию режима инкапсуляции (Encapsulation Mode) Transport. Добавлено требование, в соответствии с которым предложения IPComp **должны** включать атрибут Encapsulation Mode, если они предлагают использование инкапсуляции, не совпадающей с принятой по умолчанию (например, Tunnel Mode).

- 4) В список поддерживаемых атрибутов добавлено время жизни – Lifetime (вместе с Transport Mode).
- 5) Описана обработка атрибутов преобразований в Protection Suite, которые не применимы к IPComp – такие атрибуты **не следует** включать в предложения IPComp и они **должны** игнорироваться при установке IPComp и в процессе работы IPComp. Для реализаций IPComp **недопустимо** отвергать предложения IPComp. Не включающие атрибуты других преобразований.
- 6) Добавлены примечания для разработчиков по вопросам согласования и использования CPI из предопределённого диапазона (хорошо известные значения).

## 8. Литература

[RFC0791] Postel, J., Editor, "Internet Protocol", STD 5, [RFC 791](#), September 1981.

[RFC1700] Reynolds, J. and J. Postel, "Assigned Numbers", STD 2, [RFC 1700](#), October 1994. См. также [www.iana.org/numbers.html](http://www.iana.org/numbers.html)

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), December 1998.

[RFC1962] Rand, D., "The PPP Compression Control Protocol (CCP)", RFC 1962, June 1996.

[RFC2003] Perkins, C., "IP Encapsulation within IP", [RFC 2003](#), October 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, [RFC 2119](#), March 1997.

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

[SECD0I] Piper, D., "The Internet IP Security Domain of Interpretation for ISAKMP", [RFC 2407](#), November 1998.

[V42BIS] CCITT, "Data Compression Procedures for Data Circuit Terminating Equipment (DCE) Using Error Correction Procedures", Recommendation V.42 bis, January 1990.

## Адреса авторов

### Abraham Shacham

Juniper Networks, Inc.  
1194 North Mathilda Avenue  
Sunnyvale, California 94089  
United States of America  
E-Mail: [shacham@shacham.net](mailto:shacham@shacham.net)

### Roy Pereira

Cisco Systems, Inc.  
55 Metcalfe Street  
Ottawa, Ontario K1P 6L5  
Canada  
E-Mail: [royp@cisco.com](mailto:royp@cisco.com)

### Bob Monsour

18 Stout Road  
Princeton, New Jersey 08540  
United States of America  
E-Mail: [bob@bobmonsour.com](mailto:bob@bobmonsour.com)

### Matt Thomas

3am Software Foundry  
8053 Park Villa Circle  
Cupertino, California 95014  
United States of America  
E-Mail: [matt@3am-software.com](mailto:matt@3am-software.com)

## Перевод на русский язык

Николай Малых

[nmalykh@protokols.ru](mailto:nmalykh@protokols.ru)

## Комментарии

Комментарии следует направлять по адресу [ippcp@external.cisco.com](mailto:ippcp@external.cisco.com) (список рассылки) или авторам.

## Полное заявление авторских прав

Copyright (C) The Internet Society (2001). Все права защищены.

Этот документ и его переводы могут копироваться и предоставляться другим лицам, а производные работы, комментирующие или иначе разъясняющие документ или помогающие в его реализации, могут подготавливаться, копироваться, публиковаться и распространяться целиком или частично без каких-либо ограничений при условии сохранения указанного выше уведомления об авторских правах и этого параграфа в копии или производной работе. Однако сам документ не может быть изменён каким-либо способом, таким как удаление уведомления об авторских правах или ссылок на Internet Society или иные организации Internet, за исключением случаев, когда это необходимо для разработки стандартов Internet (в этом случае нужно следовать процедурам для авторских прав, заданных процессом Internet Standards), а также при переводе документа на другие языки.

Предоставленные выше ограниченные права являются бессрочными и не могут быть отозваны Internet Society или правопреемниками.

Этот документ и содержащаяся в нем информация представлены "как есть" и автор, организация, которую он/она представляет или которая выступает спонсором (если таковой имеется), Internet Society и IETF отказываются от каких-либо гарантий (явных или подразумеваемых), включая (но не ограничиваясь) любые гарантии того, что использование представленной здесь информации не будет нарушать чьих-либо прав, и любые предполагаемые гарантии коммерческого использования или применимости для тех или иных задач.

## Подтверждение

Финансирование функций RFC Editor обеспечивается Internet Society.