Internet Engineering Task Force (IETF)
Request for Comments: 8900
BCP: 230
Category: Best Current Practice
ISSN: 2070-1721

R. Bonica
Juniper Networks
F. Baker
Unaffiliated
G. Huston
APNIC
R. Hinden
Check Point Software
O. Troan
Cisco
F. Gont
SI6 Networks
September 2020

IP Fragmentation Considered Fragile

Проблемы, связанные с фрагментацией IP

Аннотация

В этом документе описывается фрагментация IP и её влияние на нестабильность Internet-коммуникаций. Документ также предлагает альтернативу фрагментации IP и даёт рекомендации для разработчиков и сетевых операторов.

Статус документа

Документ относится к категории Internet Best Current Practice.

Документ является результатом работы IETF[1] и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG[2]. 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/rfc8900.

Оглавление

[1]Internet Engineering Task Force - комиссия по решению инженерных задач Internet.
[2]Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

1. Введение

Опыт эксплуатации [Kent] [Huston] [RFC7872] показывает, что фрагментация IP ведёт к уязвимости коммуникаций Internet. Этот документ описывает фрагментацию IP и разъясняет её «хрупкость», а также предлагает альтернативу фрагментации и рекомендации для разработчиков и сетевых операторов.

Хотя в документе идентифицированы проблемы, связанные с фрагментацией IP, он не предлагает отказа от неё. Унаследованные протоколы, зависящие от фрагментации IP, следует обновить для исключения зависимости. Однако некоторые приложения и среды (см. раздел 5) требуют использовать фрагментацию IP. В таких случаях протокол будет продолжать её использование, но разработчикам следует осознавать, что фрагментированные пакеты могут приводить к «чёрным дырам». При разработке следует предусматривать соответствующие меры предосторожности.

Вместо отказа от фрагментации IP этот документ рекомендует протоколам вышележащего уровня решать эту проблему, снижая зависимость от фрагментации, насколько это возможно.

1.1. Уровни требований

Ключевые слова **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не следует** (SHALL NOT), **следует** (SHOULD), **не нужно** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **не рекомендуется** (NOT RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе интерпретируются в соответствии с BCP 14 [RFC2119] [RFC8174] тогда и только тогда, когда они выделены шрифтом, как показано здесь.

2. Фрагментация IP

2.1. Каналы, пути, MTU, PMTU

Путь Internet соединяет узлы источника и получателя. Путь может включать каналы и маршрутизаторы. Если путь состоит из нескольких каналов они соединяются маршрутизаторами в цепочку. Пути Internet являются динамическими. Предположим, что путь от одного узла к другому содержит набор каналов и маршрутизаторов. При отказе одного из них путь будет изменён и пройдёт через другой набор каналов и маршрутизаторов.

Для каждого канала число октетов, которые можно передать в одном пакете IP, ограничено. Этот параметр называют максимальным передаваемым блоком (Maximum Transmission Unit или MTU). В IPv4 [RFC0791] от каждого канала требуется поддержка MTU не менее 68 октетов[1]. В IPv6 [RFC8200] от каждого канала требуется поддержка MTU не менее 1280 октетов. Эти значения называются минимальными MTU для каналов IPv4 и IPv6.

Некоторые каналы и способы использования вносят дополнительные издержки переменного размера. Для простого случая туннелей этот документ следует другим документам, а в других случаях, таких как MPLS, в этом документе считается, что MTU на канале учитывает возможность таких издержек.

Точно также размер пакетов IP ограничен на каждом пути Internet. Это ограничение называют MTU для пути (Path MTU или PMTU). Для любого пути PMTU совпадает с наименьшим значением MTU образующих путь каналов. Пути в Internet имеют динамическую природу и значения PMTU также являются динамическими.

По описанным выше причинам узел-источник оценивает PMTU между собой и узлами-получателями. Оценка PMTU может быть чрезвычайно сдержанной, где:

- для каждого пути IPv4 принимается минимальное значение IPv4 MTU;
- для каждого пути IPv6 принимается минимальное значение IPv6 MTU.

Эта консервативная оценка в любом случае даёт значение не выше фактического PMTU и вероятно будут существенно ниже его, что может негативно влиять на производительность протокола вышележащего уровня.

С помощью процедур определения PMTU (Path MTU Discovery или PMTUD) [RFC1191] [RFC8201] узел-источник может более точно оценить PMTU между собой и получателем. В PMTUD узел-источник выполняет начальную оценку PMTU как значение MTU на первом канале пути к получателю. Это значение может быть больше фактического PMTU.

[1] В IPv4 каждый хост должен быть способен собрать пакет, размером не меньше 576 октетов. Однако для IPv4 минимальное значение MTU на канале не равно 576. В параграфе 3.2 RFC 791 [RFC0791] явно указано минимальное значение IPv4 MTU в 68 октетов.

Получив начальную оценку PMTU, узел-источник передаёт получателю нефрагментируемые[1] пакеты IP. Если какой-то из этих пакетов окажется больше фактического PMTU, маршрутизатор не сможет его переслать, поэтому он отбросит пакет и передаст сообщение ICMP [RFC0792] [RFC4443] Packet Too Big (PTB - пакет слишком велик) узлу-источнику[2]. Сообщение ICMP PTB указывает значение MTU на канале, в который не удалось передать пакет. Узел-источник использует эти сведения для уточнения оценки PMTU.

PMTUD даёт текущую оценку PMTU между источником и получателем. Поскольку PMTU является динамическим параметром, оценка PMTU может оказаться выше фактического PMTU. Для обнаружения роста PMTU механизм PMTUD время от времени сбрасывает PMTU к начальному значению и повторяет описанную выше процедуру.

В идеальном случае PMTUD работает, как описано выше, но в некоторых случаях PMTUD может отказывать.

- PMTUD полагается на способность сети доставлять сообщения ICMP PTB узлу-источнику и при неспособности сети доставить сообщения ICMP PTB возникает отказ PMTUD.
- Механизм PMTUD уязвим для атак, поскольку сообщения ICMP легко подделать [RFC5927] и получатель не проверяет их подлинность. Такие атаки ведут PMTUD к заниженной оценке PMTU.

2.2. Процедуры фрагментации

Когда протокол вышележащего уровня представляет данные базовому модулю IP и размер получаемого пакета IP превышает PMTU, пакет делится на фрагменты, каждый из которых включает заголовок IP и часть исходного пакета.

Процедуры фрагментации IPv4 описаны в [RFC0791]. Пакет IPv4 с DF = 1 может быть фрагментирован узлом-источником[3], но маршрутизаторы на пути не могут фрагментировать его. Пакет IPv4 с DF = 0 может фрагментировать узел-источник или маршрутизатор на пути. При фрагментировании пакета IPv4 все опции IP (из заголовка IPv4) указываются в первом фрагменте, а в последующие включаются лишь опции с установленным (1) флагом копирования.

В [RFC8200] (особенно параграф 4.5) описаны процедуры фрагментации IPv6. Пакет IPv6 может фрагментировать лишь узел-источник. При фрагментировании пакета IPv6 в первом фрагменте присутствуют все заголовки расширения, но в последующих присутствуют лишь заголовки фрагмента, включающие:

- заголовок IPv6;
- Hop-by-Hop Options (при наличии);
- Destination Options (при наличии и размещении до заголовка Routing);
- Routing (при наличии);
- Fragment.

В IPv4 заголовок вышележащего уровня обычно присутствует в первом фрагменте из-за размера включённых заголовков. В IPv6 заголовок вышележащего уровня должен присутствовать в первом фрагменте.

2.3. Зависимость вышележащего уровня от фрагментации IP

Протоколы вышележащего уровня могут работать в нескольких режимах:

- не полагаться на фрагментацию IP;
- полагаться на фрагментацию лишь узлом-источником;
- полагаться на фрагментацию любым узлом.

Протоколы вышележащего уровня на основе IPv4 могут работать во всех перечисленных режимах, протоколы на основе IPv6 - лишь в первом и втором. Протоколам, работающим в первых двух режимах, требуется оценка PMTU, для чего они могут:

- задать PMTU равным минимальному MTU канала IPv4 или IPv6;
- использовать оценку, выполненную PMTUD;
- выполнить PMTUD самостоятельно;
- выполнить процедуры PMTUD уровня пакетизации (Packetization Layer PMTUD или PLPMTUD) [RFC4821] [RFC8899].

В соответствии с процедурами PLPMTUD протокол вышележащего уровня поддерживает текущую оценку PMTU путём отправки зондов разного размера вышележащему уровню своего партнёра и обработки подтверждений от того. Эта стратегия отличается от PMTUD тем, что полагается на подтверждения принятых сообщений, а не на ICMP PTB для отброшенных сообщений. Поэтому PLPMTUD не зависит от возможности сети доставлять сообщения ICMP PTB.

3. Связанные с фрагментацией проблемы

В этом разделе разъясняется, как фрагментация IP приводит к уязвимости коммуникаций Internet.

---

[1]Нефрагментируемый пакет может быть фрагментирован источником, однако другие узлы не могут фрагментировать его. В IPv4 нефрагментируемыми являются пакеты с установленным флагом DF (Don't Fragment), а в IPv6 все пакеты являются нефрагментируемыми.

[2]Сообщения ICMP PTB существуют в 2 вариантах. В ICMPv4 [RFC0792] сообщением ICMP PTB служит сообщение Destination Unreachable с Code = 4 (нужна фрагментация, но установлен флаг DF). В [RFC1191] это сообщение было дополнено для указания MTU на канале, через который не удалось передать пакет. В ICMPv6 [RFC4443] роль ICMP PTB играет сообщение Packet Too Big Message с Code = 0, которое тоже включает MTU канала, где возникла проблема.

[3]Строго говоря, это не следует считать фрагментацией, поскольку пакета, как такового ещё нет. Это скорее часть процесса пакетизации. Прим. перев.

3.1. Виртуальная сборка

Виртуальной сборкой называют процедуру, в которой устройство концептуально собирает пакет, пересылает его фрагменты и отбрасывает собранную копию. В системах A+P (Address plus Port) [RFC6346] и NAT операторского класса (Carrier Grade NAT или CGN) [RFC6888] виртуальная сборка требуется для корректной трансляции адресов во фрагментах. Такая сборка может быть полезна для решения проблем, указанных в параграфах 3.2, 3.3, 3.4, 3.5.

Виртуальная сборка требует больших вычислительных ресурсов и хранения состояний в течение неопределённого времени, поэтому она подвержена ошибкам и атакам (параграф 3.7).

3.2. Маршрутизация на основе правил

Фрагментация IP вызывает проблемы в маршрутизаторах, реализующих пересылку на основе правил. При получении маршрутизатором пакета он определяет следующий интервал пересылки (next hop) на маршруте к адресату и пересылает пакет туда. Для определения next hop маршрутизатор обращается к локальной структуре данных, называемой базой данных о пересылке (Forwarding Information Base или FIB).

Обычно FIB содержит записи, сопоставляющие префикс получателей с next hop и организованные по префиксам. Маршрутизация на основе правил позволяет держать в одной базе FIB записи, организованные по префиксам и правилам маршрутизации. Записи FIB на основе правил сопоставляют несколько полей из заголовка IP или транспортного уровня с next hop.

Таблица 1. Routing FIB на основе правил.

Запись	***Тип***	***Префикс адресата***	***След. заголовок/порт адресата***	***Next Hop***
1	По префиксам	2001:db8::1/128	Any/Any	2001:db8:2::2
2	По правилам	2001:db8::1/128	TCP/80	2001:db8:3::3

Предположим, что маршрутизатор поддерживает базу FIB, показанную в таблице 1. Первая запись FIB сопоставляет префикс 2001:db8::1/128 со следующим узлом пересылки 2001:db8:2::2. Вторая запись FIB связана с правилом и сопоставляет тот же префикс 2001:db8::1/128 и порт получателя TCP/80 с другим next hop (2001:db8:3::3). Вторая запись более конкретна (специфична) по сравнению с первой. Когда маршрутизатор получает первый фрагмент пакета, направленного в порт TCP 80 по адресу 2001:db8::1, он обращается к FIB. Этому запросу соответствуют обе записи и маршрутизатор выберет вторую, как более конкретную, и перешлёт пакет по адресу 2001:db8:3::3. При получении второго фрагмента маршрутизатор снова обратится к FIB и в этом случае запросу будет соответствовать лишь первая запись, поскольку во втором фрагменте порт TCP 80 не указан. В результате маршрутизатор выберет первую запись и перешлёт пакет по адресу 2001:db8:2::2.

Маршрутизацию на основе правил (policy-based) называют также маршрутизацией на основе фильтров (filter-based).

3.3. Трансляция сетевых адресов (NAT)

Фрагментация IP вызывает проблемы для устройств трансляции сетевых адресов (Network Address Translation или NAT). Когда устройство NAT обнаруживает новый исходящий поток, оно отображает его порт-источник и адрес IP на другой порт и IP-адрес. Создав такое отображение, устройство NAT транслирует:

- IP-адрес отправителя и порт-источник для каждого исходящего пакета;
- IP-адрес и порт получателя для каждого входящего пакета.

Двумя основными стратегиями NAT являются A+P [RFC6346] и CGN [RFC6888]. В обоих случаях устройство NAT должно выполнять виртуальную сборку фрагментов для трансляции и корректной пересылки каждого фрагмента.

3.4. Межсетевые экраны без поддержки состояний

Как подробно рассматривается в параграфе 3.7, фрагментация IP вызывает проблемы для межсетевых экранов (МСЭ) без поддержки состояния, которые используют правила, включающие порты TCP и UDP. Поскольку сведения о портах имеются лишь в первом фрагменте, для МСЭ возможны два варианта:

- воспринимать все последующие фрагменты, которые могут быть частью атаки;
- блокировать все последующие фрагменты, которые могут содержать легитимный трафик.

Ни один из этих вариантов не представляется привлекательным.

3.5. ECMP, LAG и балансировщики без учёта состояния

Фрагментация IP вызывает проблемы для балансировщиков нагрузки на основе ECMP (Equal-Cost Multipath - множество равноценных путей), LAG (Link Aggregate Group - группировка каналов) и других технологий без учёта состояния. Для направления пакета или фрагмента в канал промежуточный узел применяет хэш-алгоритм (распределение нагрузки). Ниже перечислены наиболее распространённые алгоритмы хэширования.

Если пакет или фрагмент содержит заголовок транспортного уровня, алгоритм принимает на входе квинтет (5-tuple):

- IP-адрес отправителя;
- IP-адрес получателя;
- IPv4 Protocol или IPv6 Next Header;
- порт отправителя на транспортном уровне;
- порт получателя на транспортном уровне.

Если в пакете или фрагменте нет заголовка транспортного уровня, алгоритм получает на входе лишь триплет (3-tuple):

- IP-адрес отправителя;

- IP-адрес получателя;
- IPv4 Protocol или IPv6 Next Header.

В результате нефрагментированные пакеты потока могут быть привязаны к одному каналу, а фрагментированные могут быть разделены между этим каналом и другим. В результате распределение нагрузки может стать неоптимальным.

В [RFC6438] предложено частичное решение этой проблемы для IPv6.

> На промежуточных маршрутизаторах с распределением нагрузки на основе алгоритма хэширования для определения исходящего канала ECMP или LAG в направлении следующего узла, **должен** учитываться по меньшей мере триплет {dest addr, source addr, flow label} и **могут** использоваться оставшиеся компоненты 5-tuple.

Если алгоритм включает лишь триплет {dest addr, source addr, flow label}, он направит все фрагменты пакета в один канал (см. [RFC6437] и [RFC7098]).

Для предотвращения описанной выше проблемы, реализациям **следует** выполнять рекомендации параграфа 6.4 в этом документе.

3.6. Ошибки сборки IPv4 при высоких скоростях

Фрагментация IPv4 недостаточно отказоустойчива в современной сети Internet. При высоких скоростях передачи 16-битовое поле IP identification недостаточно велико для предотвращения дубликатов, что ведёт к частым ошибкам при сборке фрагментов, а контрольные суммы TCP и UDP не смогут предотвратить доставку собранных некорректно дейтаграмм протоколам вышележащих уровней. В [RFC4963] описаны некоторые легко воспроизводимые эксперименты, демонстрирующие проблему, и рассмотрено практическое влияние этих наблюдений.

Эти проблемы сборки не возникают часто в IPv6, поскольку там применяется 32-битовое поле идентификации.

3.7. Уязвимости безопасности

Исследователи безопасности описали несколько атак с использованием фрагментации IP, включая:

- атаки с перекрытием фрагментов [RFC1858] [RFC3128] [RFC5722];
- атаки на исчерпание ресурсов;
- атаки на основе предсказуемости идентификации фрагментов [RFC7739];
- обход систем обнаружения вторжений (Network Intrusion Detection System или NIDS) [Ptacek1998].

В атаках с перекрывающимися фрагментами злоумышленник создаёт серию фрагментов пакетов, где первый фрагмент содержит заголовки IP и транспортного уровня, а также часть данных транспортного уровня (payload). Этот фрагмент соответствует локальной политике безопасности и может пройти через МСЭ без учёта состояния. Второй фрагмент с отличным от 0 смещением перекрывается с первым и тоже проходит через МСЭ. При сборке фрагментов заголовок транспортного уровня из первого фрагмента переписывается данными из второго фрагмента и полученный пакет уже не соответствует локальной политике безопасности. При попытке передачи через МСЭ без фрагментации пакет был бы отвергнут. МСЭ без учёта состояния не могут защитить от атак с перекрывающимися фрагментами. Однако целевой узел может обеспечить свою защиту путём реализации процедур, описанных в RFC 1858, RFC 3128, RFC 8200. Эти процедуры сборки обнаруживают наложение фрагментов и отбрасывают пакет.

Алгоритм сборки фрагментов является процедурой с учётом состояния в протоколе, не учитывающем состояние. Поэтому сборку фрагментов можно использовать для истощения ресурсов собирающего хоста. Атакующий может создать серию фрагментированных пакетов, пропуская в каждом пакете один фрагмент, что делает сборку невозможной. Таким образом можно вызвать истощение ресурсов на целевом хосте, что может препятствовать сборке пакетов из других потоков. Этот тип атак можно смягчить путём очистки буферов сборки при необходимости ценой возможной потери легитимных фрагментов.

Каждый фрагмент IP содержит поле Identification, которое целевой узел использует для сборки пакета из фрагментов. Некоторые реализации устанавливают в поле Identification предсказуемое значение, что упрощает атакующему создание вредоносных фрагментов IP, которые могут препятствовать сборке легитимных фрагментов.

Системы NIDS предназначены для обнаружения вредоносных действий путём анализа сетевого трафика. Возможная неопределённость результата сборки фрагментов может позволить атакующему обойти такие системы. Многие из систем NIDS пытаются учесть некоторые из таких методов обхода (например, путём расчёта всех возможных вариантов сборки фрагментов ценой роста издержек на обработку).

3.8. Чёрные дыры PMTU в результате потери ICMP

Как отмечено в параграфе 2.3, протоколы вышележащего уровня можно настроить на использование PMTUD. Поскольку PMTUD полагается на доставку сообщений ICMP PTB, эти протоколы также будут зависеть от доставки. В соответствии с [RFC4890] фильтрация сообщений ICMPv6 PTB недопустима. Однако доставка ICMP PTB не гарантируется из-за временных и сохраняющихся потерь.

Временная потеря сообщений ICMP PTB может вызвать временные «чёрные дыры» PMTU, которые исчезают при восстановлении доставки сообщений ICMP PTB и восстановлении совместимости между отправителем и получателем. В параграфе 3.8.1 этого документа описаны условия, при которых возникают временные потери сообщений ICMP PTB.

Постоянные потери ICMP PTB вызывают сохраняющиеся чёрные дыры и в параграфах 3.8.2 - 3.8.4 описаны условия, ведущие к постоянной потере сообщений ICMP PTB.

Описанная здесь проблема относится к PMTUD и не возникает, когда протоколы вышележащего уровня получают оценку PMTU от PLPMTUD или иного источника.

3.8.1. Временные потери

Причинами временных потерь сообщений ICMP PTB могут быть:

- перегрузки в сети;
- повреждение пакетов;
- временные петли в маршрутизации;
- ограничение скорости отправки ICMP.

Влияние ограничения скорости может быть серьёзным, поскольку RFC 4443 рекомендует строго ограничивать частоту передачи трафика ICMPv6.

3.8.2. Некорректная реализация правил безопасности

Некорректная реализация правил безопасности может вести к постоянной потере сообщений ICMP PTB. Предположим, например, что абонентский маршрутизатор (Customer Premises Equipment или CPE) реализует на уровне зоны правила:

- разрешать любой трафик из зоны наружу;
- не разрешать трафик в зону извне, если он не является частью существующего потока (т. е. был вызван исходящим пакетом).

При получении корректной реализацией такой политики сообщения ICMP PTB проверяются данные ICMP PTB для сопоставления с исходным пакетом (т. е. пакетом, вызвавшим ICMP PTB) на предмет принадлежности к имеющемуся потоку. Если исходный пакет относился к существующему потоку, реализация пропустит сообщение ICMP PTB извне внутрь зоны. В противном случае сообщение ICMP PTB будет отброшено.

Некорректная реализация этой политики будет отбрасывать сообщения ICMP PTB, поскольку адрес отправителя в них не связан с имеющимся потоком.

Ситуация с некорректной реализацией политики часто встречается на абонентских маршрутизаторах CPE.

3.8.3. Постоянные потери, вызванные Anycast

Использование Anycast может вызвать постоянную потерю сообщений ICMP PTB. Например, клиент DNS может отправлять запрос по anycast-адресу. Сеть маршрутизирует запрос DNS ближайшему экземпляру адреса (т. е. серверу DNS). Сервер генерирует отклик и отправляет его клиенту DNS. Хотя отклик не превышает оценку PMTU на сервере DNS, он может превысить фактическое значение PMTU. Маршрутизатор будет отбрасывать пакет и передавать ICMP PTB его отправителю (по anycast-адресу). Сеть маршрутизирует ICMP PTB экземпляру anycast, ближайшему к этому маршрутизатору. Этот экземпляр anycast может не быть сервером DNS, который отправил исходный отклик на запрос клиента. Это может оказаться другой сервер DNS с тем же anycast-адресом. Сервер DNS, отправивший исходный пакет, может просто не получить сообщение ICMP PTB и не обновить своё значение PMTU.

3.8.4. Постоянные потери, вызванные асимметричной маршрутизацией

Асимметричная маршрутизация может вызывать постоянные потери сообщений ICMP PTB. Например, узел-источник передаёт пакет получателю. Все промежуточные узлы поддерживают маршрут к получателю, но могут не поддерживать маршрута к отправителю. В этом случае, промежуточный узел, столкнувшийся с проблемой MTU не сможет передать отправителю сообщение ICMP PTB.

3.9. «Чёрные дыры» из-за фильтрации или потерь

В RFC 7872 исследователи выбрали пути Internet для проверки передачи по ним пакетов с заголовками расширения IPv6. Выбранные пути завершались на популярных сайтах Internet (например, серверы web, DNS, электронной почты). Исследование показало, что по меньшей мере 28% выбранных путей не поддерживали передачу пакетов с заголовками расширения IPv6 Fragment. В большинстве случаев фрагменты отбрасывались в автономной системе получателя, иногда отбрасывание происходило в транзитных автономных системах.

В другом исследовании [Huston] выводы были подтверждены и показано, что 37% выбранных конечных точек, использовавших распознаватели DNS с поддержкой IPv6, не смогли получить фрагментированные отклики IPv6.

Причины отбрасывания пакетов сетевыми операторами определить сложно, но они могут включать:

- неспособность оборудования обрабатывать фрагментированные пакеты;
- отказ изменения от установленных производителем настроек;
- непреднамеренные ошибки в конфигурации;
- преднамеренная настройка (например, отбрасывание фрагментов IPv6 для предотвращения проблем, рассмотренных в параграфах 3.2 - 3.8).

4. Альтернативы фрагментации IP

4.1. Решения на транспортном уровне

Протокол управления транспортом (Transport Control Protocol или TCP) [RFC0793]) может работать в режиме, не требующем фрагментации IP. Приложения представляют протоколу TCP поток данных, которые TCP делит на сегменты размером не более TCP MSS (Maximum Segment Size - максимальный размер сегмента). Каждый сегмент инкапсулируется с заголовком TCP и представляется базовому модулю IP, который добавляет в начало заголовок IP и пересылает полученный пакет. Если значение TCP MSS достаточно мало, модуль IP не будет создавать пакетов, размер которых превосходит фактическое значение PMTU и фрагментация IP не потребуется. TCP предоставляет несколько механизмов управления MSS:

- настройка вручную;
- PMTUD;
- PLPMTUD.

Ручная настройка возможна всегда. При установке достаточно малого значения MSS уровень IP никогда не создаст пакетов, размер которых превосходит минимальное значение MTU для канала. Однако ручная настройка препятствует TCP использовать преимущества больших MTU.

Протоколы вышележащего уровня могут реализовать PMTUD для обнаружения и использования больших значений Path MTU. Однако, как отмечено в параграфе 2.1, PMTUD полагается на доставку сообщений ICMP PTB, поэтому PMTUD может обеспечить оценку PMTU лишь в средах, где невысок риск потери ICMP PTB (например, известно, что сообщения не фильтруются).

PLPMTUD не использует сообщений ICMP PTB, работая на основе сообщений-зондов, передаваемых в сегментах TCP, для определения возможности использования PMTU на пути через сеть. В PLPMTUD зондирование отделено от контроля перегрузок, поэтому потеря пробного сегмента TCP не вызывает сокращения окна контроля перегрузки. В [RFC4821] определены процедуры PLPMTUD для протокола TCP.

Хотя TCP никогда осознанно не вынуждает базовый модуль IP выдавать пакеты размером больше оценённого значения PMTU, он может заставить модуль IP выдавать пакеты размером больше фактического PMTU. Например, при изменении маршрутизации, вызывающем снижение PMTU, TCP не узнает об этом до прибытия сообщения ICMP PTB. При передаче слишком большого пакета он отбрасывается, оценка PMTU обновляется и сегмент делится на меньшие с представлением каждого базовому модулю IP.

Протоколы DCCP (Datagram Congestion Control Protocol) [RFC4340] и SCTP (Stream Control Transmission Protocol) [RFC4960] также могут работать в режиме, не требующем фрагментации IP. Оба протокола воспринимают данные от приложения и делят их на сегменты, не превосходящие максимальный размер. DCCP поддерживает ручную настройку, PMTUD и PLPMTUD для установки максимального размера. Для дейтаграмм можно реализовать PLPMTUD для оценки PMTU через [RFC8899]. Это предполагает выполнение процедур PLPMTUD с UDP, опциями UDP, SCTP, QUIC и другими протоколами доставки дейтаграмм.

В настоящее время протокол UDP (User Datagram Protocol) [RFC0768] не имеет своего механизма фрагментации и применяет фрагментацию IP. Однако в [UDP-OPTIONS] предложен механизм фрагментации для протокола UDP.

4.2. Решения на прикладном уровне

В [RFC8085] отмечено, что фрагментация IP снижает надёжность коммуникаций Internet. Отмечено также отсутствие в UDP своего механизма фрагментации и работа протокола на основе фрагментации IP. Поэтому в [RFC8085] внесено специальное предложение для приложений на основе транспорта UDP.

> Приложению **не следует** передавать дейтаграммы UDP, которые приводят в пакетам IP размером больше MTU на пути к получателю. Поэтому приложению **следует** использовать данные о MTU на пути от уровня IP или самостоятельно реализовать определение MTU на пути (Path MTU Discovery или PMTUD) [RFC1191] [RFC1981] [RFC4821] для определения возможности передачи сообщений без фрагментации.

В RFC 8085 также сказано:

> Приложениям, не использующим PMTU или PLPMTUD, **следует** избегать отправки дейтаграмм UDP, которые будут приводить к передаче пакетов IP размером больше MTU. Поскольку реальное значение MTU для пути не известно, таким приложениям **следует** передавать сообщения, которые короче принятого по умолчанию эффективного значения MTU для передачи (EMTU_S в [RFC1122]). Для IPv4 EMTU_S будет меньшим из значение 576 байтов и MTU первого этапа пересылки [RFC1122], для IPv6 EMTU_S будет 1280 байтов [RFC2460]. Эффективное значение PMTU для подключённого напрямую адресата (без маршрутизаторов на пути) будет определяться заданным для интерфейса значением MTU, которое может быть меньше разрешённого на канале максимума. Передача дейтаграмм UDP минимального размера не эффективна для путей, поддерживающих большие PMTU, и это служит другой причиной реализации определения PMTU.

В RFC 8085 предполагается, что в IPv4 значение EMTU_S = 576 достаточно мало и поддерживается большинством современных путей Internet, несмотря на то, что минимальное значение IPv4 MTU составляет 68.

Приведённые рекомендации применимы к любым приложениям, работающим непосредственно по протоколу IP.

5. Приложения, полагающиеся на фрагментацию IPv6

Ряд приложений полагается на фрагментацию IPv6:

- DNS [RFC1035];
- OSPFv2 [RFC2328];
- OSPFv3 [RFC5340];
- инкапсуляция «пакет в пакете».

Каждое из этих приложений в той или иной степени полагается на фрагментацию IPv6. В некоторых случаях такая поддержка необходима и от неё нельзя отказаться без существенного изменения протокола. В других случаях зависимость случайна и приложения уже предпринимают шаги от избавлению от неё. Приведённый список не является полным и могут быть другие протоколы, полагающиеся на фрагментацию IP. Однако здесь эти протоколы не рассматриваются.

5.1. Служба доменных имён (DNS)

DNS использует протокол UDP для эффективности и, следовательно, полагается на фрагментацию IP для больших откликов, как это разрешено опциями механизмов расширения (Extension Mechanisms for DNS или EDNS0) в запросе. Можно смягчить проблему потери пакетов из-за фрагментации, применяя для запросов меньшие размеры буферов EDNS0 UDP или ограничивая на серверах DNS максимальный размер откликов UDP неким самоустанавливаемым размером пакета, который будет меньше предпочитаемого размера буфера EDNS0 UDP. В обоих случаях большие отклики усекаются в DNS с рекомендацией клиенту повторить запрос по протоколу TCP для получения полного отклика. Однако ограниченная поддержка DNS по протоколу TCP, особенно в случае IPv6, не позволяет эффективно реализовать этот подход [Damas].

Больших откликов DNS обычно можно избежать жёсткой отсечкой раздела Additional в откликах DNS. Одним из случаев, когда такая отсечка неэффективна, является использование DNSSEC, где большие размеры ключей увеличивают размер откликов на определённые запросы DNS. В DNS нет эффективного решения этой проблемы, кроме сокращения размера криптографических ключей и принятия в DNSSEC административных мер для сокращения размера откликов DNS.

5.2. Протокол OSPF

Реализации OSPF могут создавать сообщения достаточно большого размера, который потребует фрагментации. Однако для оптимизации производительности многие реализации OSPF ограничивают максимальный размер сообщений до величин, не вызывающих фрагментации.

5.3. Инкапсуляция пакета в пакет

Этот документ подтверждает, что в некоторых случаях пакеты должны фрагментироваться в туннелях IP-in-IP, но не даёт дополнительных рекомендаций, относящихся к таким туннелям.

В этом документе инкапсуляцией пакета в пакет (packet-in-packet) считается IP-in-IP [RFC2003], GRE (Generic Routing Encapsulation) [RFC2784], GRE-in-UDP [RFC8086], Generic Packet Tunneling in IPv6 [RFC2473]. В [RFC4459] описаны проблемы фрагментации, связанные с этими методами инкапсуляции. Стратегия фрагментации для GRE, описанная в [RFC7588], была развёрнута для всех перечисленных методов инкапсуляции. Эта стратегия не полагается на фрагментацию IP за исключением одного крайнего случая (см. параграф 3.3.2.2 в [RFC7588] и параграф 7.1 в [RFC2473].) Этот случай описан в параграфе 3.3 [RFC7676].

Дополнительное рассмотрение вопросов, связанных с туннелями приведено в [TUNNELS].

5.4. Повышение производительности приложений UDP

Некоторые приложения UDP пытаются повысить производительность на основе использования фрагментации IP. В таких приложения используются размер дейтаграмм UDP, превышающий Path MTU, чтобы передать больший объем данных между приложением и ядром за один системный вызов.

Например, протокол LTP (Licklider Transmission Protocol) [RFC5326], применяемый в настоящее время на МКС (International Space Station или ISS), использует дейтаграммы UDP размером больше Path MTU для обеспечения приемлемой производительности даже при возникновении фрагментации IP. В более общем смысле SNMP и видео-приложения могут передавать кванты данных уровня приложения в зависимости сетевого уровня для фрагментации и последующей сборки при возникновении необходимости.

6. Рекомендации

6.1. Для разработчиков приложений и протоколов

Разработчикам **не следует** создавать новые протоколы и приложения, полагающиеся на фрагментацию IP. При развёртывании нового протокола или приложения в среде, не полностью поддерживающей фрагментацию IP, **следует** обеспечивать корректность работы для принятой по умолчанию или специально подготовленной конфигурации.

Хотя в некоторых контролируемых средах фрагментация IP работает надёжно, это проблема развёртывания, о которой невозможно знать заранее при разработке приложения или протокола. Не рекомендуется создавать новые протоколы и приложения, полагающиеся на фрагментацию IP, поскольку они не обеспечивают достаточно надёжную работу в Internet.

Унаследованные протоколы, зависящие от фрагментации IP, следует обновить для исключения этой зависимости. Однако в некоторых случаях может не быть реальной замены фрагментации IP (например, для туннельного режима IPSEC, инкапсуляции IP-in-IP). Приложения и протоколы могут не знать и не контролировать использование фрагментации на нижележащих уровнях и путях через сеть. В таких случаях протокол будет по-прежнему полагаться на фрагментацию IP, но применять её следует лишь в средах, где известно о поддержке фрагментации.

Протоколы могут избежать фрагментации IP за счёт применения достаточно малых значений MTU (например, минимальное значение для протокола), запрета фрагментации IP и обеспечения адаптации транспортным протоколом размера сегментов в соответствии с MTU. Другие протоколы могут реализовать достаточно надёжный механизм определения PMTU (например, PLPMTUD).

Приложениям UDP **следует** соблюдать рекомендации, приведённые в параграфе 3.2 [RFC8085].

6.2. Для системных разработчиков

В программные библиотеки **следует** включать предоставления механизма PLPMTUD для каждого поддерживаемого протокола.

6.3. Для разработчиков промежуточных устройств

Промежуточным устройствам, которые являются системами, «прозрачно» выполняющими функции политики передачи трафика, но не участвующими в маршрутизации, следует обрабатывать фрагменты IP в соответствии с [RFC0791] и [RFC8200]. Во многих случаях промежуточные устройства для этого должны поддерживать состояния.

Вопросы цены и производительности часто побуждают сетевых операторов развёртывать промежуточные устройства, не хранящие состояний. Такие устройства могут неоптимально обрабатывать фрагменты IP, не выполняя требований RFC 791 или RFC 8200, и могут даже полностью отбрасывать фрагменты IP. Такое поведение **не рекомендуется**. Если промежуточное устройство реализует нестандартное поведение для фрагментов IP, это поведение **должно** быть чётко документировано.

6.4. Для разработчиков и операторов ECMP, LAG и балансировщиков

В принятой по умолчанию конфигурации устройствам IPv6, реализующим ECMP с маршрутизацией OSPF [RFC2328] и других протоколов, LAG [RFC7424] или иную технологию распределения нагрузки для случая IPv6 Flow Label с нулевым значением, следует принимать на входе алгоритма хэширования лишь поля:

- IP Source Address;
- IP Destination Address;
- Flow Label.

Операторам **следует** использовать при развёртывании принятую по умолчанию конфигурацию устройств.

Эти рекомендации похожи на представленные в [RFC6438] и [RFC7098] и отличаются лишь выбором принятой по умолчанию конфигурации.

6.5. Для сетевых операторов

Операторы **должны** обеспечивать надлежащую работу PMTUD в своей сети, включая гарантию генерации сообщений PTB при отбрасывании пакетов, размер которых превышает MTU на выходном интерфейсе. Однако реализации **могут** ограничивать скорость генерации сообщений ICMP в соответствии с [RFC1812] и [RFC4443].

В соответствии с RFC 4890 сетевым операторам **недопустимо** фильтровать сообщения ICMPv6 PTB, если нет уверенности в их подделке или иных нарушениях. Как отмечено в параграфе 3.8, фильтрация пакетов ICMPv6 PTB ведёт к отказам PMTUD. Многие протоколы вышележащих уровней полагаются на PMTUD.

В соответствии с RFC 8200 сетевым операторам **недопустимо** развёртывать каналы IPv6, где MTU меньше 1280 октетов.

Сетевым операторам **не следует** фильтровать фрагменты IP, если известно, что они исходят от сервера DNS или направлены такому серверу. Это связано с важностью серверов имён для работы Internet.

7. Взаимодействие с IANA

Этот документ не предполагает действий IANA.

8. Вопросы безопасности

Этот документ смягчает некоторые проблемы безопасности, связанные с фрагментацией IP, за счёт отказа от её использования. Документ не создаёт новых уязвимостей, поскольку не предлагает новые замены фрагментации IP, рекомендуя лишь известные ранее и хорошо понятные варианты.

9. Литература

9.1. Нормативные документы

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, DOI 10.17487/RFC0768, August 1980, <https://www.rfc-editor.org/info/rfc768>.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <https://www.rfc-editor.org/info/rfc791>.

[RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, DOI 10.17487/RFC0792, September 1981, <https://www.rfc-editor.org/info/rfc792>.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, DOI 10.17487/RFC0793, September 1981, <https://www.rfc-editor.org/info/rfc793>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, DOI 10.17487/RFC1191, November 1990, <https://www.rfc-editor.org/info/rfc1191>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/info/rfc4443>.

[RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, DOI 10.17487/RFC4821, March 2007, <https://www.rfc-editor.org/info/rfc4821>.

[RFC6437] Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme, "IPv6 Flow Label Specification", RFC 6437, DOI 10.17487/RFC6437, November 2011, <https://www.rfc-editor.org/info/rfc6437>.

[RFC6438] Carpenter, B. and S. Amante, "Using the IPv6 Flow Label for Equal Cost Multipath Routing and Link Aggregation in Tunnels", RFC 6438, DOI 10.17487/RFC6438, November 2011, <https://www.rfc-editor.org/info/rfc6438>.

[RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage Guidelines", BCP 145, RFC 8085, DOI 10.17487/RFC8085, March 2017, <https://www.rfc-editor.org/info/rfc8085>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8201] McCann, J., Deering, S., Mogul, J., and R. Hinden, Ed., "Path MTU Discovery for IP version 6", STD 87, RFC 8201, DOI 10.17487/RFC8201, July 2017, <https://www.rfc-editor.org/info/rfc8201>.

[RFC8899] Fairhurst, G., Jones, T., Tüxen, M., Rüngeler, I., and T. Völker, "Packetization Layer Path MTU Discovery for Datagram Transports", RFC 8899, DOI 10.17487/RFC8899, September 2020, <https://www.rfc-editor.org/info/rfc8899>.

9.2. Дополнительная литература

[Damas] Damas, J. and G. Huston, "Measuring ATR", April 2018, <http://www.potaroo.net/ispcol/2018-04/atr.html>.

[Huston] Huston, G., "IPv6, Large UDP Packets and the DNS", August 2017, <http://www.potaroo.net/ispcol/2017-08/xtn-hdrs.html>.

[Kent] Kent, C. and J. Mogul, "Fragmentation Considered Harmful", SIGCOMM '87: Proceedings of the ACM workshop on Frontiers in computer communications technology, DOI 10.1145/55482.55524, August 1987, <http://www.hpl.hp.com/techreports/Compaq-DEC/WRL-87-3.pdf>.

[Ptacek1998] Ptacek, T. H. and T. N. Newsham, "Insertion, Evasion and Denial of Service: Eluding Network Intrusion Detection", 1998, <http://www.aciri.org/vern/Ptacek-Newsham-Evasion-98.ps>.

[RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, DOI 10.17487/RFC1122, October 1989, <https://www.rfc-editor.org/info/rfc1122>.

[RFC1812] Baker, F., Ed., "Requirements for IP Version 4 Routers", RFC 1812, DOI 10.17487/RFC1812, June 1995, <https://www.rfc-editor.org/info/rfc1812>.

[RFC1858] Ziemba, G., Reed, D., and P. Traina, "Security Considerations for IP Fragment Filtering", RFC 1858, DOI 10.17487/RFC1858, October 1995, <https://www.rfc-editor.org/info/rfc1858>.

[RFC1981] McCann, J., Deering, S., and J. Mogul, "Path MTU Discovery for IP version 6", RFC 1981, DOI 10.17487/RFC1981, August 1996, <https://www.rfc-editor.org/info/rfc1981>.

[RFC2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, DOI 10.17487/RFC2003, October 1996, <https://www.rfc-editor.org/info/rfc2003>.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <https://www.rfc-editor.org/info/rfc2328>.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, DOI 10.17487/RFC2460, December 1998, <https://www.rfc-editor.org/info/rfc2460>.

[RFC2473] Conta, A. and S. Deering, "Generic Packet Tunneling in IPv6 Specification", RFC 2473, DOI 10.17487/RFC2473, December 1998, <https://www.rfc-editor.org/info/rfc2473>.

[RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, DOI 10.17487/RFC2784, March 2000, <https://www.rfc-editor.org/info/rfc2784>.

[RFC3128] Miller, I., "Protection Against a Variant of the Tiny Fragment Attack (RFC 1858)", RFC 3128, DOI 10.17487/RFC3128, June 2001, <https://www.rfc-editor.org/info/rfc3128>.

[RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", RFC 4340, DOI 10.17487/RFC4340, March 2006, <https://www.rfc-editor.org/info/rfc4340>.

[RFC4459] Savola, P., "MTU and Fragmentation Issues with In-the-Network Tunneling", RFC 4459, DOI 10.17487/RFC4459, April 2006, <https://www.rfc-editor.org/info/rfc4459>.

[RFC4890] Davies, E. and J. Mohacsi, "Recommendations for Filtering ICMPv6 Messages in Firewalls", RFC 4890, DOI 10.17487/RFC4890, May 2007, <https://www.rfc-editor.org/info/rfc4890>.

[RFC4960] Stewart, R., Ed., "Stream Control Transmission Protocol", RFC 4960, DOI 10.17487/RFC4960, September 2007, <https://www.rfc-editor.org/info/rfc4960>.

[RFC4963] Heffner, J., Mathis, M., and B. Chandler, "IPv4 Reassembly Errors at High Data Rates", RFC 4963, DOI 10.17487/RFC4963, July 2007, <https://www.rfc-editor.org/info/rfc4963>.

[RFC5326] Ramadas, M., Burleigh, S., and S. Farrell, "Licklider Transmission Protocol - Specification", RFC 5326, DOI 10.17487/RFC5326, September 2008, <https://www.rfc-editor.org/info/rfc5326>.

[RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <https://www.rfc-editor.org/info/rfc5340>.

[RFC5722] Krishnan, S., "Handling of Overlapping IPv6 Fragments", RFC 5722, DOI 10.17487/RFC5722, December 2009, <https://www.rfc-editor.org/info/rfc5722>.

[RFC5927] Gont, F., "ICMP Attacks against TCP", RFC 5927, DOI 10.17487/RFC5927, July 2010, <https://www.rfc-editor.org/info/rfc5927>.

[RFC6346] Bush, R., Ed., "The Address plus Port (A+P) Approach to the IPv4 Address Shortage", RFC 6346, DOI 10.17487/RFC6346, August 2011, <https://www.rfc-editor.org/info/rfc6346>.

[RFC6888] Perreault, S., Ed., Yamagata, I., Miyakawa, S., Nakagawa, A., and H. Ashida, "Common Requirements for Carrier-Grade NATs (CGNs)", BCP 127, RFC 6888, DOI 10.17487/RFC6888, April 2013, <https://www.rfc-editor.org/info/rfc6888>.

[RFC7098] Carpenter, B., Jiang, S., and W. Tarreau, "Using the IPv6 Flow Label for Load Balancing in Server Farms", RFC 7098, DOI 10.17487/RFC7098, January 2014, <https://www.rfc-editor.org/info/rfc7098>.

[RFC7424] Krishnan, R., Yong, L., Ghanwani, A., So, N., and B. Khasnabish, "Mechanisms for Optimizing Link Aggregation Group (LAG) and Equal-Cost Multipath (ECMP) Component Link Utilization in Networks", RFC 7424, DOI 10.17487/RFC7424, January 2015, <https://www.rfc-editor.org/info/rfc7424>.

[RFC7588] Bonica, R., Pignataro, C., and J. Touch, "A Widely Deployed Solution to the Generic Routing Encapsulation (GRE) Fragmentation Problem", RFC 7588, DOI 10.17487/RFC7588, July 2015, <https://www.rfc-editor.org/info/rfc7588>.

[RFC7676] Pignataro, C., Bonica, R., and S. Krishnan, "IPv6 Support for Generic Routing Encapsulation (GRE)", RFC 7676, DOI 10.17487/RFC7676, October 2015, <https://www.rfc-editor.org/info/rfc7676>.

[RFC7739] Gont, F., "Security Implications of Predictable Fragment Identification Values", RFC 7739, DOI 10.17487/RFC7739, February 2016, <https://www.rfc-editor.org/info/rfc7739>.

[RFC7872] Gont, F., Linkova, J., Chown, T., and W. Liu, "Observations on the Dropping of Packets with IPv6 Extension Headers in the Real World", RFC 7872, DOI 10.17487/RFC7872, June 2016, <https://www.rfc-editor.org/info/rfc7872>.

[RFC8086] Yong, L., Ed., Crabbe, E., Xu, X., and T. Herbert, "GRE-in-UDP Encapsulation", RFC 8086, DOI 10.17487/RFC8086, March 2017, <https://www.rfc-editor.org/info/rfc8086>.

[TUNNELS] Touch, J. and M. Townsley, "IP Tunnels in the Internet Architecture", Work in Progress, Internet-Draft, draft-ietf-intarea-tunnels-10, 12 September 2019, <https://tools.ietf.org/html/draft-ietf-intarea-tunnels-10>.

[UDP-OPTIONS] Touch, J., "Transport Options for UDP", Work in Progress, Internet-Draft, draft-ietf-tsvwg-udp-options-08, 12 September 2019, <https://tools.ietf.org/html/draft-ietf-tsvwg-udp-options-08>.

Благодарности

Спасибо Mikael Abrahamsson, Brian Carpenter, Silambu Chelvan, Lorenzo Colitti, Gorry Fairhurst, Joel Halpern, Mike Heard, Tom Herbert, Tatuya Jinmei, Suresh Krishnan, Jen Linkova, Paolo Lucente, Manoj Nayak, Eric Nygren, Fred Templin, Joe Touch за их комментарии.

Адреса авторов

Ron Bonica
Juniper Networks
2251 Corporate Park Drive
Herndon, Virginia 20171
United States of America
Email: rbonica@juniper.net

Fred Baker
Unaffiliated
Santa Barbara, California 93117
United States of America
Email: FredBaker.IETF@gmail.com

Geoff Huston
APNIC
6 Cordelia St
Brisbane 4101 QLD
Australia
Email: gih@apnic.net

Robert M. Hinden
Check Point Software
959 Skyway Road
San Carlos, California 94070
United States of America
Email: bob.hinden@gmail.com

Ole Troan
Cisco
Philip Pedersens vei 1
N-1366 Lysaker
Norway
Email: ot@cisco.com

Fernando Gont
SI6 Networks
Evaristo Carriego 2644
Haedo
Provincia de Buenos Aires
Argentina
Email: fgont@si6networks.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru