Internet Engineering Task Force (IETF)
Request for Comments: 9331
Category: Experimental
ISSN: 2070-1721

K. De Schepper
Nokia Bell Labs
B. Briscoe, Ed.
Independent

January 2023

The Explicit Congestion Notification (ECN) Protocol for Low Latency, Low Loss, and Scalable Throughput (L4S)

Протокол явных уведомлений о перегрузке для архитектуры L4S

Аннотация

Эта спецификация определяет протокол, который будет применяться для новой сетевой услуги, названной L4S[1]. В L4S используется схема явных уведомлений о перегрузке (Explicit Congestion Notification или ECN) на уровне IP, похожая на исходный («классический») подход ECN с некоторыми отличиями. L4S использует расширяемый (Scalable) контроль перегрузок, включающий более частые сигналы управления из сети и реагирующий на эти сигналы более детальными настройками, так что для трафика L4S становится возможной очень малая (обычно доли миллисекунды в среднем) и стабильная задержка в очередях без ущерба для загрузки каналов. Таким образом, даже трафик, которому нужна пропускная способность (подобный TCP), может получить одновременно высокую пропускную способность и очень малые задержки даже в периоды высокой загрузки.

Определённый в этом документе идентификатор L4S отличает трафик L4S от «классического» (например, TCP-Reno-friendly). Узкие места в сети (bottleneck) могут постепенно обновляться, чтобы отличать и изолировать остающийся «классический» трафик без его ухудшения из-за трафика L4S. Эта экспериментальная спецификация определяет правила для транспорта L4S и элементов сети, чтобы потоки L4S не наносили ущерба производительности друг друга и «классического» трафика. Некоторые вопросы остаются не решёнными и требуют дополнительных исследований. Отдельно приведены примеры новых алгоритмов маркировки для активного управления очередями (Active Queue Management или AQM) и нового транспорта (подобного TCP или трафику в реальном масштабе времени).

Статус документа

Документ не относится к категории Internet Standards Track и публикуется для проверки, экспериментальной реализации и оценки.

Документ является результатом работы IETF[2] и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG[3]. Не все документы, одобренные IESG, претендуют на статус стандартов Internet, см. раздел 2 в RFC 7841.

Информацию о текущем статусе документа, ошибках и способах обратной связи можно найти по ссылке https://www.rfc-editor.org/info/rfc9331.

Авторские права

Авторские права (Copyright (c) 2023) принадлежат IETF Trust и лицам, указанным в качестве авторов документа. Все права защищены.

К документу применимы права и ограничения, указанные в BCP 78 и IETF Trust Legal Provisions и относящиеся к документам IETF (http://trustee.ietf.org/license-info), на момент публикации данного документа. Прочтите упомянутые документы внимательно. Фрагменты программного кода, включённые в этот документ, распространяются в соответствии с упрощённой лицензией BSD, как указано в параграфе 4.e документа IETF Trust Legal Provisions, без каких-либо гарантий (как указано в Simplified BSD License).

Оглавление

[1] Low Latency, Low Loss, and Scalable throughput - малые задержки и потери с расширяемой пропускной способностью.
[2] Internet Engineering Task Force - комиссия по решению инженерных задач Internet.
[3] Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

1. Введение

Эта спецификация определяет протокол, который будет применяться для новой сетевой услуги, названной L4S. В L4S используется схема ECN на уровне IP с тем же набором кодов (и переходами), как в исходном ECN [RFC3168]. [RFC3168] требует, чтобы маркировка ECN была эквивалентна отбрасыванию (drop) при использовании как в сети, так и при откликах транспорта. В отличие от классической маркировки i) сеть применяет маркировку L4S более оперативно и часто, нежели отбрасывание, и ii) реакция транспорта на каждый маркер ускоряется и сглаживается по сравнению с реакцией на отбрасывание. Эти два изменения уравновешивают друг друга и пропускная способность для потока L4S будет сопоставима с потоками без L4S при одних условиях. Тем не менее, более частые сигналы управления ECN и ускоренный отклик на эти сигналы приводят к очень малым задержкам в очередях без ущерба для загрузки каналов и такая низкая задержка может обеспечиваться при высокой нагрузке. Например, задержка постановки в очередь при высокой и сильно меняющейся нагрузке ов описанном ниже примере решения DCTCP/DualQ на канале DSL или Ethernet в среднем составляет доли миллисекунды и примерно 1 - 2 мсек при 99-м процентиле без вреда для использования канала [L4Seval22] [DualPI2Linux]. Отметим, что к отмеченному следует добавлять задержку в очереди доступа к среде передачи, такой как беспроводный канал. Эту задачу также требуется решить, но отдельно (см. параграф 6.3 в описании архитектуры L4S [RFC9330]).

L4S полагается на расширяемые (Scalable) средства контроля перегрузок для снижения задержки и сохранения малой задержки при расширении потоков (отсюда и название). Контроль перегрузок в Data Center TCP (DCTCP) служит примером расширяемого контроля, но DCTCP применяется исключительно в контролируемых средах, таких как ЦОД [RFC8257] по причине избыточной для сосуществования с трафиком TCP-Reno-friendly энергичности. Механизм Dual-Queue Coupled AQM, заданный в дополняющей этот документ экспериментальной спецификации [RFC9332], представляет собой платформу AQM, позволяющую расширяемым средствам контроля перегрузок на основе DCTCP, сосуществовать с имеющимся трафиком, предоставляя обоим примерно одинаковую скорость потока в похожих условиях. Отметим, что расширяемый контроль перегрузок остаётся небезопасным для развёртывания в Internet, пока не выполняются требования, указанные в разделе 4. Поведение транспортного уровня (требования Prague L4S).

Архитектура L4S предназначена не только для эластичного трафика (похожего на TCP) и имеется расширяемый контроль перегрузок для потоков в реальном масштабе времени (real-time media), такой как вариант L4S [SCReAM-L4S] из SCReAM [RFC8298] методов предотвращения перегрузки среды (RTP Media Congestion Avoidance Techniques или RMCAT). Поведение трафика L4S отличается от классического в части откликов на перегрузку. Протоколы транспортных соединений, такие как TCP, QUIC, SCTP[1], DCCP[2], RTP/RTCP ортогональны и по этой причине не позволяют отличать пакеты L4S от классических.

[1]Stream Control Transmission Protocol - протокол управления потоковой передачей.
[2]Datagram Congestion Control Protocol - протокол дейтаграмм с контролем перегрузки.

Заданный в этом документе идентификатор L4S является ключевым отличием трафика L4S от «классического» (например, Reno-friendly). Узкие места в сети можно поэтапно изменять для идентификации и изоляции имеющегося классического трафика от L4S с целью предотвратить негативное влияние на классический трафик в результате снижения уровня задержки и потерь для нового масштабируемого транспорта. Хотя для получения преимуществ требуется развёртывание у отправителей и в сети возможные в будущем преимущества послужили стимулом для первоначального внедрения отдельных частей системы.

1.1. Вопросы задержки, потерь и масштабирования

Задержки становятся критически важным фактором производительности для многих (возможно, большинства) приложений Internet, например, интерактивных web-приложений и служб, передачи, видео со звуком, интерактивного видео, удалённого присутствия, мгновенного обмена сообщениями, сетевых игр, удалённых рабочих столов, облачных приложений и служб, а также удалённого управления оборудованием и производственными процессами. Во многих частях мира дальнейший рост битовой скорости сетей доступа ведёт к замедлению отдачи [Dukkipati06] пока задержки остаются проблемой. В результате было многое сделано для сокращения времени распространения за счёт кэширования и переноса серверов ближе к пользователям. Однако очереди остаются основным, хотя и меняющимся компонентом задержки.

Архитектура Diffserv обеспечивает ускоренную пересылку (Expedited Forwarding или EF) [RFC3246], где трафик с малой задержкой может переноситься в очередь другого трафика. Если рост чувствительных к задержке приложений продолжится, периоды лишь с чувствительным к задержкам трафиком будут составлять все большую часть на каналах со слабым агрегированием трафика. Если в течение таких периодов для всего трафика применяется одинаковая обработка, Diffserv становится бесполезным. Каналы со слабым агрегированием также обычно становятся узким местом на пути под нагрузкой, например, это может происходить на каналах доступа, выделенных для отдельных сайтов (дом, небольшое предприятие, мобильное устройство). Таким образом, вместо дифференциации требуется исключить первопричины всех необязательных задержек.

Проект Bufferbloat (раздувание буферов) показал, что чрезмерная буферизация (bufferbloat) ведёт с значительно большей задержке чем базовое время распространения [Bufferbloat]. Эти задержки возникают лишь время от времени, когда ищущий пропускную способность поток (например, TCP) достаточно продолжителен для заполнения буфера очередью, в результате чего каждый пакет из других потоков, использующих этот буфер, должен пройти через очередь.

Для решения этой и других проблем был разработан механизм AQM. В отличие от Diffserv, где низкая задержка для некоторого трафика обеспечивается за счёт другого трафика, AQM контролирует задержку для всего трафика в классе. В общем случае методы AQM повышают уровень отбрасывания из буферов по мере превышения очередью небольшого заданного порогового размера. Это обеспечивает достаточную сигнализацию для потоков, которым нужна пропускная способность (жадных потоков), чтобы сохранить буферное пространство для основной цели - смягчения (поглощения) пиков. Однако случайное раннее обнаружение (Random Early Detection или RED) и другие алгоритмы 1990-х годов были чувствительны к конфигурации и сложны в настройке [RFC7567]. Поэтому механизмы AQM не получили широкого распространения.

Более современные методы AQM, такие как Flow Queue CoDel [RFC8290], PIE[1] [RFC8033], Adaptive RED [ARED01], проще в настройке, поскольку они задают пороги для очередей временем, а не числом байтов и конфигурация не меняется в зависимости от скорости канала. Однако «окно пилы» (sawtoothing window) классического контроля перегрузок создаёт для оператора дилемму - i) настроить неглубокую (shallow) рабочую точку AQM, чтобы верхушки зубьев пилы вызывали минимальную задержку в очереди, но тогда канал в остальном используется не полностью, или ii) задать более глубокую рабочую точку в буфере для более эффективного использования канала, но в этом случае пики будут вызывать большие изменения задержки. Даже при совершенной настройке AQM дополнительная задержка в очереди для пиков пилы будет иметь такой же порядок значений, что и базовый интервал кругового обхода (round-trip time или RTT), фактически удваивая суммарное значение RTT.

Если само поведение отправителя вносит вариации задержки, никакой механизм AQM в сети не сможет «отменить» такую задержку без существенного ущерба для использования канала. Даже постановка потока в очередь, изолирующая его от других потоков (например, [RFC8290]), не может воспрепятствовать вариациям задержки, которые создаёт сам поток. Поэтому приложениям, которым нужна высокая пропускная способность и малая задержка, придётся перейти к расширяемому контролю перегрузок, где не возникает значительных пилообразных вариаций.

Однако простой смены поведения хоста недостаточно. Даже при реализации хостом расширяемых элементов контроля перегрузки с малой задержкой требуется изоляция больших вариаций очереди, вносимых имеющимися классическими элементами контроля перегрузки. Механизмы L4S AQM обеспечивают изоляцию задержки в сети, а идентификатор L4S позволяет AQM отличать пакеты L4S от классических. Изоляцию L4S можно обеспечить организацией очередей по потокам (например, [RFC8290]), до DualQ [RFC9332] вполне достаточно и механизм фактически обеспечивает лучшую задержку в «хвосте» [DCttH19]. В этом документе рассматриваются оба подхода.

Решение DualQ было разработано для обеспечения очень малой задержки без создания очередей по потокам в каждом узком месте. Это было полезно, поскольку очереди по потокам (per-flow queuing или FQ) имеют известные недостатки (не в последнюю очередь необходимость просмотра в сети заголовков транспортного уровня), которые делают это решение несовместимым с подходами к приватности, такими как туннели виртуальных частных сетей IPsec (Virtual Private Network или VPN), или управлением очередями на канальном уровне, где транспортные заголовки могут быть скрыты (например, 5G).

Задержки не являются единственной проблемой L4S. При первоначальной разработке механизма предотвращения перегрузок в TCP было известно, что их не удастся расширить для больших произведений пропускной способности и задержки (bandwidth-delay, см. примечание 6 от Jacobson и Karels [TCP-CA]). С учётом выхода Reno за рамки расширяемости уже при обычных скоростях широкополосных сетей WAN [RFC3649], были внедрены «более расширяемые» варианты TCP CUBIC [RFC8312] и Compound [CTCP]. Однако они тоже не решают задачу расширяемости. К сожалению, полностью расширяемые (fully Scalable) механизмы контроля перегрузок, такие как DCTCP [RFC8257], превосходят Classic ECN лишь при использовании общей очереди, поэтому они развёрнуты лишь в частных ЦОД и испытательных стендах.

[1] Proportional Integral controller Enhanced - улучшенный пропорциональный интегральный контроллер.

Эти расширяемые алгоритмы контроля перегрузок, решающие проблему задержки, могут решить также проблему расширяемости классических элементов контроля перегрузки. Более тонкие (finer) зубья пилы в окне перегрузки (congestion window или cwnd) имеют меньшую амплитуду, поэтому они вызывают очень незначительные вариации задержки, а среднее время восстановления от одного сигнала перегрузки до следующего (средняя продолжительность зубы пилы) остаётся неизменным, что обеспечивает постоянный жёсткий контроль при изменении скорости потока. В статье [L4Seval22] дано полное объяснение этого факта как на простом языке, так и в более точной математической форме. Без математических выкладок объяснение приведено также в разделе 4 описания архитектуры L4S [RFC9330].

1.2. Терминология

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.

Classic Congestion Control - классический контроль перегрузок

Поведение контроля перегрузок, способное сосуществовать со стандартным Reno [RFC5681] без существенного негативного влияния на скорость потока [RFC5033]. Поскольку с 1988 г., когда был разработан механизм контроля перегрузок в TCP, скорости потоков возросли, при классическим контроле перегрузок, таком как Reno или CUBIC, восстановление после сигнала перегрузки (потеря или маркер ECN) может занимать сотни интервалов кругового обхода (и растёт), как показано в параграфе |5.1 описания архитектуры L4S [RFC9330] и в [RFC3649]. Поэтому контроль очередей и загрузки каналов становится очень слабым и малейшие помехи (например, появление нового потока) препятствуют достижению высокой скорости.

Scalable Congestion Control - расширяемый контроль перегрузок

Контроль перегрузок, где среднее время от одного сигнала насыщения до следующего (время восстановления) независимо от скорости потока при прочих равных условиях. Это обеспечивает некоторый контроль над очередями и загрузкой канала, а также гарантирует высокую пропускную способность, устойчивую к нарушениям. Например, DCTCP усредняет 2 сигнала пересылки за интервал кругового обхода, независимо от скорости потока, как и другие недавно разработанные расширяемые механизмы контроля перегрузок, такие как Relentless TCP [RELENTLESS], Prague для TCP и QUIC [PRAGUE-CC] [PragueLinux], BBRv2 [BBRv2] [BBR-CC], L4S-вариант SCReAM для потоков в реальном масштабе времени [SCReAM-L4S] [RFC8298]. Дополнительные сведения можно найти в параграфе 4.3 [RFC9331].

Classic Service - классический сервис

Классический сервис предназначен для всех вариантов поведения контроля перегрузок, сосуществующих с Reno [RFC5681] (например, сам Reno, CUBIC [RFC8312], Compound [CTCP], TFRC [RFC5348]). «Классической очередью» называется очередь, обеспечивающая классический сервис.

Low Latency, Low Loss, and Scalable throughput (L4S) service - сервис с малыми задержками и потерями, а также расширяемой пропускной способностью

Сервис L4S предназначен для трафика с расширяемыми алгоритмами контроля перегрузок, такими как Prague [PRAGUE-CC], который был выведен из DCTCP [RFC8257]. Сервис L4S предназначен для более широкого класса трафика, нежели просто Prague, и позволяет развивать набор средств контроля перегрузок, аналогичных Prague, таких как отмечены выше (Relentless, SCReAM и т. п.). Очередью L4S называется очередь, предоставляющая услуги L4S.

Атрибуты Classic и L4S могут применяться к очередям (queue), кодам (codepoint), идентификаторам (identifier), классификации (classification), пакетам (packet), потокам (flow). Например термин «пакет L4S» относится к пакету с идентификатором L4S, переданному из системы контроля перегрузок L4S.

Оба типа сервиса (Classic и L4S) могут справляться с некоторой долей неотзывчиваго и слабо отзывающегося трафика, но в случае L4S скорость должна быть достаточно плавной или низкой, чтобы не возникала очередь (например, DNS, VoIP, синхронизация игр и т. п.).

Reno-friendly - совместимость с Reno

Часть классического трафика, совместимая со стандартным контролем перегрузок Reno, заданным для TCP в [RFC5681]. Спецификация TFRC [RFC5348] косвенно подразумевает, что дружественностью (совместимостью) считается «нахождение обычно в пределах двухкратного отличия скорости передачи потока TCP при одинаковых условиях». Термин Reno-friendly используется здесь вместо TCP-friendly, поскольку последнее выражение стало неточным, так как протокол TCP сейчас использует много разных вариантов контроля перегрузок, а Reno применяется не только в TCP, но и в других транспортных протоколах (например, QUIC [RFC9000]).

Classic ECN - классический механизм явных уведомлений о перегрузке

Исходный механизм явных уведомлений о перегрузке (ECN) [RFC3168] требует считать сигналы ECN эквивалентом отбрасывания пакетов как при генерации в сети, так и отвечающим хостом.

Для L4S имена кодов 2-битового поля IP-ECN не отличаются от заданных в спецификации ECN [RFC3168] - Not-ECT, ECT(0), ECT(1), CE, где ECT указывает поддержку ECN в транспорте (ECN-Capable Transport), а CE - возникновение перегрузки (Congestion Experienced). Пакеты с кодом CE называют промаркированными ECN (ECN-marked), а иногда - просто маркированными, если наличие ECN ясно из контекста.

Site - сайт

Дом, мобильное устройство, небольшое предприятие или кампус, где узким местом сети является канал доступа. Этому определению соответствуют не все сети, но оно является полезным и широко распространенным.

1.3. Область действия

Новый идентификатор L4S, заданный в этой спецификации, применим к пакетам IPv4 и IPv6 (как и Classic ECN [RFC3168]). Он подходит для индивидуальной (unicast), групповой (multicast) и anycast-пересылки.

Идентификатор L4S ортогонален классификации пакетов по кодам дифференцированного обслуживания (Differentiated Services Code Point или DSCP) [RFC2474]. Практическое значение этого рассмотрено в параграфе 5.4. Взаимодействие идентификаторов L4S с другими идентификаторами.

Этот документ является экспериментальным и не обновляет какие-либо Standards Track RFC и зависит от [RFC8311] со статусом Standards Track, который:

- обновляет ECN Proposed Standard [RFC3168], позволяя Experimental RFC смягчать требование эквивалентности маркировки ECN отбрасыванию пакета (при маркировке в сети или на отвечающем хосте); например, в эксперименте Alternative Backoff with ECN (ABE) [RFC8511] это смягчение позволяет отправителю слабее реагировать на маркировку ECN, нежели на отбрасывание;
- меняет статус спецификации Experimental ECN nonce [RFC3540] на Historic (устарела);
- вносит соответствующие изменения в перечисленные ниже Proposed Standard RFC:
  - ECN для RTP [RFC6679];
  - спецификации контроля перегрузки различных профилей идентификаторов контроля перегрузки DCCP (Congestion Control Identifier или CCID) [RFC4341] [RFC4342] [RFC5622].

Этот документ посвящён идентификаторам, используемым для взаимодействия между хостами и сетями. Документ предназначен для разработчиков AQM для хостов и сетей, а также операторов, которые хотят комбинировать различные идентификаторы, что требует гибкости от разработчиков оборудования.

2. Идентификация пакетов L4S - структура документа

Обработка маркировки L4S ECN является экспериментальной альтернативой обработке Classic ECN [RFC3168], обновлённой в [RFC8311], чтобы можно было проводить эксперименты, подобные описанным здесь. В [RFC4774] рассмотрены некоторые проблемы и критерии оценки для определения дополнительной семантики ECN, которые обсуждаются также в параграфе 4.3.1. Рекомендации по реагированию на перегрузки в документах RFC.

Архитектура L4S [RFC9330] включает 3 основных компонента: поведение передающего хоста, маркировка в сети и протокол L4S ECN для идентификации пакетов L4S.

В разделе 3. Требования к выбору идентификатора пакетов L4S указаны требования, на основании которых был выбран идентификатор L4S. Далее описывается протокол L4S ECN, который i) идентифицирует переданные хостами пакеты, для которых предполагается разнообразное поведение при передаче, и ii) задаёт трактовку маркеров, установка которых ожидается для пакетов L4S на узлах сети.

Чтобы пакет при пересылке трактовался как L4S, отправитель устанавливает в поле ECN заголовка IP код ECT(1). В разделе 4. Поведение транспортного уровня (требования Prague L4S) указаны требования к поведению транспортного уровня, включая обратную связь и реагирование на перегрузку.

Узел сети, реализующий службу L4S, всегда классифицирует пакеты ECT(1) для обработки L4S и по умолчанию классифицирует для такой обработки пакеты CE, если не применяются эвристические методы из параграфа 5.3. Полные требования к поведению элементов сети заданы в разделе 5, включая классификацию, маркировку ECN и взаимодействие идентификаторов L4S с другими идентификаторами, а также поэтапную пересылку.

L4S ECN работает с туннелированием и инкапсуляцией ECN, за исключением одного известного случая, когда нужно особое внимание к конфигурации, рассмотренного подробно в разделе 6. Поведение в туннелях и при инкапсуляции.

Эта спецификация L4S ECN имеет статус экспериментальной. В разделе 7 собраны общие вопросы и проблематика, требующие дальнейшего изучения в ходе экспериментов с L4S. Нерешенные вопросы и проблемы, относящиеся к определенным компонентам, рассмотрены в соответствующих спецификациях, таких как DualQ [RFC9332].

Назначение агентством IANA идентификатора L4S рассмотрено в разделе 8, а раздел 9 посвящён вопросам безопасности, связанным с идентификаторами L4S. Вопросы безопасности системы, такие как правила и приватность, рассмотрены в описании архитектуры L4S [RFC9330].

3. Требования к выбору идентификатора пакетов L4S

В этом разделе кратко описан процесс, в результате которого был выбран идентификатор L4S.

Идентификатору для пакетов L4S следует соответствовать указанным ниже требованиям:

- сквозное прохождение между конечными точками источника и получателя через границы между хостом и сетью, смежными сетями, а также через промежуточные устройства (middlebox);
- видимость на уровне IP;
- эквивалентность для IPv4 и IPv6, независимость от транспорта;
- возможность поэтапного внедрения;
- возможность AQM классифицировать пакеты, инкапсулированные в IP и протоколы нижележащего уровня;
- минимальное число дополнительных кодов (codepoint);
- согласованность для всех пакетов потока транспортного уровня, чтобы пакеты одного потока не обслуживались разными очередями.

Можно было бы учесть фактор восстановления идентификатора при отказе эксперимента. Однако это не требовалось, поскольку давало бы преимущество схемам, где шансы провала превышали вероятность успеха. Было признано, что вряд ли какой-то идентификатор будет соответствовать всем приведённым требованиям, особенно с учётом ограниченности оставшегося пространства в заголовках IP. Поэтому во всех случаях будет нужен компромисс и при указании требования использовать уровень **следует** (SHOULD), а не **необходимо** (MUST).

После тщательной оценки разных схем в качестве компромисса был выбран вариант с кодами ECT(1) и CE. Этот выбор подробно описан ниже, а в Приложении B указаны его плюсы и минусы в соответствии с приведёнными требованиями.

4. Поведение транспортного уровня (требования Prague L4S)

В этом разделе заданы требования к поведению L4S на транспортном уровне, известные как Prague L4S Requirements (происхождение названия объяснено в Приложении A).

4.1. Установка кода

Отправитель, желающий для пакета обработки L4S при его пересылке, должен указать в поле ECN заголовка IP (v4 или v6) код ECT(1).

4.2. Требования к откликам транспорта

Для поддержки расширяемого контроля перегрузки транспортный протокол (4.3. Требования к откликам на перегрузку) **должен** обеспечивать отклики о степени маркировки CE на пути пересылки. При включении ECN в протокол TCP [RFC3168] метод обратной связи сообщал не более, чем об одной маркировке CE за интервал кругового обхода. Некоторые транспортные протоколы, производные от TCP, имитируют это поведение, тогда как другие сообщают более точные сведения о маркировке ECN. Это означает, что некоторые транспортные протоколы требуют обновления для поддержки расширяемого контроля перегрузок. Ниже приведены сведения для нескольких популярных протоколов.

TCP

Поддержка требования точных откликов ECN [RFC7560] (например, как в AccECN [ACCECN]) на обеих сторонах является обязательным условием для расширяемого контроля перегрузки в TCP. Поэтому наличие ECT(1) в заголовках IP даже для одного направления в соединении TCP означает, что обе стороны поддерживают точные отклики ECN. Однако обратное неверно, т. е. даже если обе стороны поддерживают AccECN, любая из них может отказаться от использования расширяемого контроля перегрузок независимо от другой стороны.

SCTP

Подходящий механизм откликов ECN для SCTP мог бы добавить блок (chunk) с информацией о числе полученных маркеров CE (как описано в давно устаревшем документе [SCTP-ECN] и очерчено в Приложении A к уже отменённой спецификации стандарта SCTP [RFC4960]).

RTP по протоколу UDP

Для расширяемого контроля перегрузок обе (все) стороны одного интервала пересылки (hop) на уровне среды (media-level) должны сообщить о поддержке ECN [RFC6679] и применять новый базовый формат откликов RTCP [RFC8888]. Наличие ECT(1) означает, что обе (все) стороны этого media-level hop поддерживают ECN. Однако обратное неверно,т. е. каждая из сторон media-level hop может отказаться от использования расширяемого контроля перегрузок, даже если обе поддерживают ECN.

QUIC

Поддержка достаточно подробных откликов ECN обеспечивается транспортом IETF QUIC v1 [RFC9000].

DCCP

Вектора подтверждения (Acknowledgement или ACK) в DCCP [RFC4340] уже достаточно для информирования о маркировке CE, требуемого для расширяемого контроля перегрузок.

4.3. Требования к откликам на перегрузку

В качестве условия передачи пакетов с идентификатором L4S (ECT(1)) хосту **следует** реализовать поведение контроля перегрузок, которое в установившемся состоянии гарантирует, что средняя продолжительность между индуцированными метками ECN не будет возрастать с ростом скорости потока при прочих равных условиях. Это называется расширяемым (Scalable) контролем перегрузки. Неизменный интервал маркировки гарантирует, что по мере роста скорости потока средняя продолжительность интервала без откликов о пропускной способности не будут расти при прочих равных условиях, а также гарантирует отсутствие значительных вариаций размера очереди без ущерба для загрузки канала.

При контроле перегрузки с использованием зубьев пилы для проверки пропускной способности этот интервал называется временем восстановления, поскольку он указывает среднее время после спада зуба пилы до восстановления прежней верхней точки. При расширяемом контроле перегрузки пилы не возникает, но её могут создавать другие механизмы. Например, для DCTCP [RFC8257], TCP Prague [PRAGUE-CC] [PragueLinux] и L4S-варианта SCReAM [SCReAM-L4S] [RFC8298] среднее время восстановления составляет половину интервала кругового обхода (или половину эталонного RTT) независимо от скорости потока.

Как и для всех вариантов поведения транспорта, предполагается детальная спецификация (возможно, в Experimental RFC) для каждого метода контроля перегрузок в соответствии с рекомендациями по заданию новых алгоритмов контроля перегрузок [RFC5033]. Кроме того, ожидается, что в относящихся к L4S материалам будут документированы, в частности, временные рамки усреднения пропорциональности и контроля за пиками (всплесками) трафика. Для требований к времени восстановления выше использован уровень **следует** (SHOULD), а не **должен** (MUST) для обеспечения разумной гибкости реализаций.

Толкование «при прочих равных условиях» допускает, что время восстановления может различаться для разных RTT при условии, что оно не будет расти с увеличением скорости потока для любого конкретного RTT.

Утверждение о приблизительном постоянстве времени восстановления эквивалентно утверждению о постоянстве числа маркеров ECN CE за интервал кругового обхода, независимо от скорости потока при прочих равных условиях. Например, среднее время восстановления в половину RTT эквивалентно 2 маркерам ECN за время кругового обхода. Применительно к функциям отклика на установившуюся перегрузку можно также говорить, что окно перегрузки сокращается обратно пропорционально числу байтов в пакетах с маркировкой кодом CE (см. раздел 2 в [PI2]).

Для безопасного сосуществования с другим трафиком Internet расширяемому контролю перегрузки не разрешается помечать пакеты кодом ECT(1), пока не выполняются указанные ниже требования и рекомендации.

1. Расширяемый контроль перегрузки **должен** обеспечивать возможность замены классическим контролем перегрузки (приложением и/или средствами администрирования). При активации классического контроля перегрузок он не будет помечать свои пакеты кодом ECT(1) (см. A.1.3. Возможность замены классическим контролем перегрузки).

2. Помимо отклика на маркеры ECN, расширяемый контроль перегрузки **должен** реагировать на потери пакетов так, чтобы безопасно сосуществовать с классическим контролем перегрузки, таким как Reno [RFC5681], в соответствии с [RFC5033] (см. обоснование в A.1.4. Возврат к классическому контролю при потере пакетов).

3. В неконтролируемых средах **должен** быть реализован мониторинг для поддержки обнаружения проблем в AQM с поддержкой ECN на узких местах пути, которые представляются не поддерживающими L4S и могут находиться в общей очереди. Такой мониторинг следует применять к текущему (live), применяющему расширяемый контроль перегрузки. Мониторинг не требуется для текущего трафика при наличии мониторинга тестового трафика, охватывающего соответствующий путь через неконтролируемые среды.

   Должна использоваться функция обнаружения отмеченных выше проблем в AQM с поддержкой ECN. Функции обнаружения **следует** обеспечивать способность заставить средства контроля перегрузки адаптировать свою реакцию на маркеры ECN в реальном масштабе времени для безопасного сосуществования с классическим контролем перегрузки, таким как Reno [RFC5681], в соответствии с [RFC5033]. Это может дополняться более детальным обнаружением потенциальных проблем автономно (offline). Если при использовании лишь автономного обнаружения на некоторых путях видны проблемы с таким AQM, расширяемый механизм контроля перегрузок **должен** заменяться классическим по меньшей мере на проблемных путях.

   Обоснование и разъяснения приведены в 4.3.1. Рекомендации по реагированию на перегрузки в документах RFC, A.1.5. Сосуществование с классическим контролем в узких местах с Classic ECN и рекомендациях по работе с L4S [L4SOPS].

   > Отметим, что не предполагается изменение расширяемым контролем перегрузки установки ECT(0), пока механизмы адаптируются для сосуществования с классическим контролем перегрузки. После замены механизмов контроля ECT(0) будет устанавливаться.

4. В диапазоне между минимальным вероятным и типичным RTT, ожидаемыми в предполагаемом варианте внедрения, расширяемый контроль перегрузки **должен** сходиться к скорости, которая не зависит от RTT, насколько это возможно без ущерба для стабильности и загрузки канала (см. Приложение A.1.6).

5. Расширяемому контролю перегрузок **следует** сохранять ответственность за перегрузку, когда типичные значения RTT при работе через общедоступную сеть Internet становятся значительно меньше, поскольку в ниже больше не вносятся задержки в очередях. Было бы предпочтительно сделать минимальное окно перегрузки при расширяемом контроле перегрузок размером меньше 1 сегмента вместо использования основанного на тайм-ауте подхода, описанного для TCP в параграфе 6.1.2 спецификации ECN [RFC3168] (или эквивалента для иного транспорта). Однако более низкий минимум не задан в качестве формального требования для экспериментов (см. обоснование в Приложении A.1.7).

6. Обнаружению потерь в расширяемом контроле перегрузки **следует** быть устойчивым к изменению порядка в адаптивном интервале времени, который изменяется в зависимости от пропускной способности, и приспосабливается к нарушению порядка (как в Recent ACK (RACK) [RFC8985]), в отличие от учёта лишь числа пакетов (как в правиле 3 Duplicate ACK (DupACK) NewReno [RFC5681] [RFC6675] без поддержки масштабирования). По мере роста скорости передачи данных (например, в результате внедрения новой или улучшенной технологии) контроль перегрузок, обнаруживающий потери путём подсчёта пакетов, с большей вероятностью будет некорректно трактовать события нарушения порядка как потери (см. обоснование в Приложении A.1.8). Это требование не относится к элементам контроля перегрузки, применяемым лишь в контролируемых средах, где сеть практически не меняет порядок пакетов.

7. Предполагается, что расширяемый контроль перегрузки ограничит очереди, вызываемые всплесками (burst) пакетов. Не представляется нужным устанавливать ограничение ниже 10% от минимального RTT, ожидаемого в типовом развёртывании (например, дополнительная очередь примерно на 250 мксек для общедоступной сети Internet). Это будет преобразовываться в число пакетов путём умножения на текущую среднюю скорость пакетов. Тогда очередь, вызываемая каждым всплеском в узком месте, не задержит более чем на 250 мксек (в худшем случая использования потоком всей пропускной способности). Здесь не задано нормативного требования по ограничению всплесков и до обретения достаточного опыта из экспериментов L4S даже непонятно, нужно ли такое требование, хотя представляется, что отправитель L4S заинтересован в этом.

Каждый отправитель в сессии может применять расширяемый контроль перегрузки независимо от используемого получателем контроля перегрузки при передаче им данных. Поэтому могут передаваться пакеты ECT(1) в одном направлении и ECT(0) или Not-ECT - в другом.

Ниже в этом документе (параграф 5.4.1.1) рассматриваются условия для смешивания другого «безопасно неотзывчивого трафика» (например, DNS, LDAP[1], NTP, голос, синхронизация игр) с трафиком L4S. Хотя такой трафик может помещаться в одну очередь с трафиком L4S, отравителям не следует помечать его кодом ECT(1), за исключением (маловероятного) случая, когда он удовлетворяет указанным выше требованиям.

4.3.1. Рекомендации по реагированию на перегрузки в документах RFC

[RFC3168] требует одинаковой реакции на пакет с маркером CE и отбрасывание пакета. [RFC8311] (Standards Track) обновляет [RFC3168] разрешая эксперименты с ECN, включая L4S. [RFC8311] разрешает экспериментальные отклики на маркеры CE, отличающиеся от реакции на отбрасывание пакетов, при условии, что отличия указаны в Experimental RFC, таком как данный документ.

В BCP 124 [RFC4774] представлены рекомендации для разработчиков протоколов при использовании альтернативной семантики поля IP-ECN. В [RFC8311] разъяснено, что описанный в BCP 124 опыт применения не требуется обновлять для смягчения требований к эквивалентности реакции. Хотя в BCP 124 включено требование из [RFC3168], BCP документ не задаёт своих требований на его основе. BCP 124 [RFC4774] описывает три варианта поэтапного внедрения Option 3 (параграф 4.3 в BCP 124 [RFC4774]), наиболее соответствующих L4S. Требования Option 3 к конечным узлам заключаются в реакции на маркер CE «дружественным к потоку способом, соответствующим контролю перегрузок, заданному IETF». Это перекликается с другими базовыми требованиями к контролю перегрузок в RFC Series. Например, в [RFC5033] сказано: «контроллеры перегрузок, оказывающие существенное негативное влияние на

[1]Lightweight Directory Access Protocol - облегченный протокол доступа к каталогам.

трафик, используя стандартный контроль перегрузок, могут быть подозрительными», а в [RFC8085] при рассмотрении контроля перегрузок в UDP сказано: «Приложениям с большим объёмом данных, отказавшимся от реализации TFRC или использования окна в стиле TCP, **следует** реализовать схему контроля перегрузок, обеспечивающую использование пропускной способности (ёмкости), беспристрастно разделяемой с TCP (по порядку значений).»

Нормативный п. 3 в параграфе 4.3 выше (отклик L4S на перегрузку от Classic ECN AQM) нацелен на выполнение требований сосуществования, но при этом нужно идти на некоторые компромиссы. В этом параграфе рассмотрены и обоснованы некоторые компромиссы, а Приложение A.1.5 и эксплуатационное руководство L4S [L4SOPS] содержат подробный анализ, примеры и ссылки (приведённый здесь нормативный текст имеет приоритет, если возникают противоречия). Подход основан на оценке риска причинения ущерба, сочетающей учёт вероятности условий нанесения ущерба и важность возможных последствий.

Учёт вероятности (распространённости) рассматривает три случая.

Отбрасывание хвоста (Drop Tail)

Сосуществование классических потоков с L4S не вызывает сомнений, если в узком месте не поддерживается ECN, что остаётся наиболее распространенным случаем с момента публикации спецификации ECN [RFC3168] в 2001 г.

L4S

Сосуществование не вызывает сомнений, если в узком месте поддерживается L4S.

Classic ECN

Компромиссы связаны со случаями поддержки в узком месте Classic ECN [RFC3168] без поддержки L4S.

Общая очередь с Classic ECN

На момент написания документа членам рабочей группы Transport не было известно внедрений Classic ECN с общей очередью в узком месте Internet. Тем не менее, в масштабе Internet редкий не означает малочисленный, а в будущем может стать и не столь редким.

Очереди по потокам с Classic ECN

В большинстве AQM (в частности, FQ-CoDel [RFC8290] и COBALT [COBALT]) с очередями по потокам, развёрнутых с 2012 г. по умолчанию включён механизм Classic ECN. Но компромиссы применяются лишь ко второму из указанных ниже случаев.

- При очередях по потокам сосуществование классических потоков и L4S обычно не вызывает проблем, поскольку разные потоки не будут попадать в одну очередь (BCP 124 [RFC4774] не предусматривает введение очередей по потокам, которые появились как возможный метод изоляции примерно через 8 лет после публикации BCP).
- Изоляция между классическими потоками и L4S не идеальна в случаях совпадения хэш-значений идентификаторов потоков или при инкапсуляции нескольких потоков внутри Layer 3 VPN с одним идентификатором потока.

Подводя итог, отметим, что проблема сосуществования ограничивается случаями несовершенной изоляции потоков в FQ или развёртыванием классического ECN AQM в общей очереди (см. эксплуатационные рекомендации для L4S [L4SOPS], где вопрос рассмотрен подробно включая недавние исследования с попыткой оценить распространённость). Если один из этих случаев имеет место, проблемы сосуществования не возникает, пока источники классического и L4S трафика (например, разные приложения в одном доме) не используют одновременно общую очередь в узком месте и потоки каждого типа достаточно продолжительны для возникновения дисбаланса. Вопрос частоты возникновения проблемы сосуществования отмечен в разделе 7 как нерешенный, ответ на который должны дать эксперименты с L4S.

Для случая, когда долгосрочные потоки классического трафика и L4S попадают в общую очередь, тестирование одного механизма контроля перегрузок L4S (TCP Prague) показало, что дисбаланс средней пропускной способности может достигать отношения 25:1 в пользу L4S для худшего случая [ecn-fallback]. Однако при более скудной пропускной способности классический поток получает большую долю канала, например для канала 4 Мбит/с отношение будет ниже ~10:1 для путей с базовым RTT менее 100 мсек и падает ниже ~5:1 для базового RTT менее 20 мсек.

Эти соотношения могут явно выходить за рамки текущих рекомендаций RFC по сосуществованию. Однако тенденция к сохранению большей доли канала для классических потоков и очень малая распространённость условий возникновения проблем привели к возможности нарушения требований RFC (на короткое время).

- Рекомендуемым подходом остаётся обнаружение и адаптация к Classic ECN AQM в реальном масштабе времени, что соответствует всем RFC по сосуществованию. Иными словами, **следует** в п. 3 параграфа 4.3 выше предполагает, что отправитель реализует что-то похожее на код проверки концепции (proof-of-concept), который обнаруживает наличие Classic ECN AQM и откатывается к классическим откликам на перегрузку в течение нескольких интервалов кругового обхода [ecn-fallback]. Хотя этот код надёжно обнаруживает Classic ECN AQM, текущий код может ошибочно счесть L4S AQM классическим, чаще всего при малой скорости канала или большом RTT. Хотя это безопасный путь обхода и ожидается, что разработчики смогут улучшить доказательство концепции, высказывались опасения в потере разработчиками веры в такое обнаружение и отказе от него.

- П. 3 параграфа 4.3 выше допускает компромисс, когда сосуществование на короткое время может отходить от требований RFC, но требуется обязательный мониторинг для обнаружения таких случаев и принятия мер. Этот подход допускает кратковременное отклонение от RFC с учётом малой распространённости, а ущерб будет заключаться в замедлении продвижения потока. Эксплуатационные рекомендации L4S [L4SOPS] включают ряд примеров корректировочных действий, включая изменения у отправителя или в сети. Однако финальное нормативное требование п. 3 параграфа 4.3 возлагает окончательную ответственность за принятие мер на отправителя. При обнаружении проблемы сосуществования с Classic ECN AQM (не решённой сетью) отправитель, в соответствии с требованием, **должен** вернуться к классическому контролю перегрузки.

В [L4SOPS] также приведены примеры путей внедрения контроля перегрузок L4S в сценариях с малым риском.

4.4. Фильтрация или сглаживание обратной связи ECN

В параграфе 5.2 ниже указано, что от L4S AQM ожидается незамедлительная сигнализация L4S ECN, чтобы избежать задержки из-за фильтрации или сглаживания. Это отличается от Classic AQM, где изменения в очереди фильтруются перед сигналом с помощью маркера ECN или отбрасывания. В архитектуре L4S [RFC9330] ответственность за сглаживание вариаций очереди перенесена в контроль перегрузок у отправителя. Этот перенос ответственности за

сглаживание позволяет каждому отправителю сглаживать вариации в масштабе времени, пропорционального его RTT. В классическом подходе сеть не знает значений RTT для потоков, поэтому для обеспечения стабильности при сглаживании применяется значение RTT для худшего случая. Для потоков с RTT меньше такого худшего значения контроль перегрузок становится неоправданно медленным.

Перенос ответственности также позволяет отправителю L4S отказываться от сглаживания в зависимости от контекста (запуск, предотвращение перегрузки и т. п.). Поэтому данный документ не задаёт требований к контролю перегрузок L4S в части сглаживания вариаций. Разработчикам рекомендуется открыто публиковать применяемые для сглаживания подходы, а также опыт и результаты, полученные в экспериментах с L4S.

5. Поведение узла сети

5.1. Классификация и перемаркировка

Узел сети, реализующий услуги L4S:

- **должен** классифицировать приходящие пакеты ECT(1) для обработки L4S, если они не будут переопределены другим классификатором (см., например, 5.4.1.2. Исключение трафика из обработки L4S);
- **должен** классифицировать приходящие пакеты CE для обработки L4S, если они не будут переопределены другим классификатором или не возникло указанное ниже исключение.

  Пакеты CE могут быть созданы как ECT(1) или ECT(0), но приведённое выше правило классификации, как будто они созданы как ECT(1), является безопасным выбором (см. обоснование в Приложении B). Исключением является случай наличия в сети знающего о потоках механизма, отличающего пакеты CE, созданные как ECT(0) (см. параграф 5.3), но обязательность такого механизма не предполагается.

Обработка L4S AQM следует тем же правилам смены кодов, что и [RFC3168]. В частности, код ECT(1) **недопустимо** менять на какой-либо код, кроме CE, а CE **недопустимо** заменять любым другим кодом. Пакеты ECT(1) классифицируются как поддерживающие ECN (ECN-capable) и при возникновении перегрузки алгоритм L4S AQM будет все чаще помещать в поле IP-ECN код CE, в остальных случаях просто пересылая пакеты без изменений как ECT(1). Условия для маркировки пакетов в L4S заданы в параграфе 5.2.

При сохраняющейся перегрузке обработка маркировки L4S **должна** начинать отбрасывание трафика L4S, пока перегрузка не спадёт, как рекомендовано для всех методов AQM в параграфе 4.2.1 [RFC7567], который следует похожим рекомендациям раздела 7 в [RFC3168]. Во время перегрузки для трафика L4S **должна** применяться такая же вероятность отбрасывания, как для классического трафика.

Если L4S AQM знает о применяемом транспорте, это требование можно выполнить путём отбрасывания лишь в наиболее перегруженных AQM по потокам. В DualQ с защитой очередей по потокам (например, [DOCSIS-QPROT]) этого можно добиться путём перенаправления пакетов в потоках, вносящих наибольший вклад в перегрузку, из очереди L4S, чтобы они отбрасывались по правилам классических очередей.

Для совместимости с прежними решениями в неконтролируемых средах сетевой узел с поддержкой обработки L4S **должен** реализовать также обработку AQM для классического трафика в соответствии с параграфом 1.2. От этой обработки Classic AQM не требуется маркировать пакеты ECT(0), но если это делается, нужно следовать параграфу 5.2 в части соотношения маркировки и отбрасывания. Прибывающие пакеты ECT(0) и Not-ECT **должны** классифицироваться для обработки этим механизмом Classic AQM (для DualQ Coupled AQM классификация подробно рассмотрена в параграфах 2.3 и 2.5.1.1 [RFC9332]).

При возникновении непредвиденных проблем в экспериментах с L4S **должна** быть возможность отключить в реализации L4S обработку L4S. После отключения пакеты ECT(1) **должны** рассматриваться как Not-ECT.

5.2. Соотношение маркировки L4S CE и отбрасывания

Соотношение в маркировки CE и отбрасывания в L4S не имеет значения, когда применяются механизмы AQM в очередях по потокам приложений, которые затем явно планируются (например, планировщиком FQ в FQ-CoDel [RFC8290]). Тем не менее, это соотношение нужно определять для связывания классических сигналов перегрузки с сигналами L4S в DualQ Coupled AQM [RFC9332], как показано ниже.

Пока узел AQM не планирует потоки приложений явно, вероятность отбрасывания механизмом AQM классического пакета Not-ECT (p_C) **должна** быть примерно пропорциональна квадратному корню вероятности маркировки, если бы это был пакет L4S (p_L). Т. е.

```
p_C ~= (p_L / k)^2
```

Коэффициент пропорциональности (k) не стандартизован для обеспечения функциональной совместимости, но **рекомендуется** значение 2. Термин «вероятность» (likelihood) здесь имеет смысл, позволяющий маркировке и отбрасыванию быть как вероятностными, так и детерминированными.

Эта формула гарантирует, что масштабируемые и классические потоки будут сходиться к примерно одинаковому окну перегрузки для наихудшего случая контроля перегрузки Reno. Это связано с тем, что окно перегрузки в расширяемом и классическом контроле перегрузки обратно пропорционально p_L и sqrt(p_C), соответственно. Поэтому возведение p_C в квадрат в предыдущей формуле уравновешивается квадратным корнем в характеристике потоков, совместимых с Reno.

> Отметим, что вопреки [RFC3168], механизм AQM, реализующий обработку L4S и Classic, не устанавливает для пакета маркер ECT(1) при таких же условиях, когда он отбросил бы пакет Not-ECT, как разрешает [RFC8311], обновляющий [RFC3168]. Однако такой механизм применяет маркировку ECT(0) при тех же условиях, при которых он отбросил бы пакет Not-ECT в соответствии с [RFC3168].

В архитектуре L4S [RFC9330] отправитель, а не сеть, отвечает за сглаживание вариаций очереди. Поэтому механизм L4S AQM **должен** сообщать о перегрузке как можно скорее. Отправитель L4S обычно интерпретирует маркировку CE как несглаженный сигнал.

Это требование не мешает L4S AQM смешивать дополнительные сглаженные сигналы перегрузки, такие как сигналы от сглаженные сигналы классического AQM, с несглаженными сигналами L4S в связанном DualQ [RFC9332], но лишь до тех пор, пока о перегрузке можно сигнализировать незамедлительно и сразу же интерпретировать сигналы у отправителя, что важно для функциональной совместимости.

5.3. Идентификация пакетов L4S узлами, знающими транспортный уровень

Для классификации пакетов L4S узлу сети не требуется идентифицировать потоки транспортного уровня. Тем не менее, если узел L4S классифицирует пакеты по идентификаторам потоков на транспортном уровне и их полям ECN, а все пакеты ECT в потоке имеют код ECT(0), узел может классифицировать пакеты CE в том же потоке, как будто пакеты Classic ECT(0). В остальных случаях узел сети должен классифицировать пакеты CE, как будто ECT(1). Такие случаи возникают когда: i) в потоке ещё не идентифицированы пакеты ECT, ii) для узла нежелательно идентифицировать потоки транспортного уровня, iii) некоторые пакеты ECT в потоке были ECT(1) (это нужно проверить в экспериментах L4S).

5.4. Взаимодействие идентификаторов L4S с другими идентификаторами

Примеры в этом параграфе посвящены использованию идентификаторов, дополняющих идентификатор L4S, для классификации пакетов по очередям на основе классов. В параграфе 5.4.1 рассмотрены 2 очереди (L4S и Classic) как в DualQ Coupled AQM [RFC9332] автономно (параграф 5.4.1.1) или в иерархии очередей (параграф 5.4.1.2). В параграфе 5.4.2 рассматриваются схему, которые могут комбинировать идентификаторы потоков 5-tuple с другими идентификаторами.

5.4.1. Примеры DualQ с идентификаторами, дополняющими идентификатор L4S

5.4.1.1. Включение дополнительного трафика в очередь с L4S

В типичном для общедоступной сети Internet случае элемент сети, реализующий L4S в общей очереди, может захотеть классифицировать некий низкоскоростной, но неотзывчивый (unresponsive) трафик (например, DNS, LDAP, NTP, голос, синхронизация игр) в очередь с малой задержкой, смешивая его с трафиком L4S. В этом случае очередь неуместно называть очередью L4S, поскольку она используется и для другого (не L4S) трафика. Будем называть её очередью с малой задержкой или L-очередью. Для L-очереди возможны 2 варианта работы:

- обработка L4S, сочетающая L4S AQM и планирование с приоритетом;
- обработка с малой задержкой только за счёт планирования с приоритетом без ECN-маркировки в AQM.

Чтобы отобрать пакеты для обработки лишь планировщиком, неуместно применять идентификатор L4S ECT(1), поскольку такой трафик не реагирует на маркировку ECN. Примерами подходящих идентификаторов являются:

- диапазоны адресов конкретных приложений или хостов, настроенных или известных как безопасные, например, жёстко заданные устройства IoT[1], передающих небольшой трафик;
- некоторые приложения или протоколы с небольшим объёмом данных (например, ARP и DNS);
- конкретные коды Diffserv, указывающие трафик с ограниченными пиками, такие как классы EF [RFC3246], VOICE-ADMIT [RFC5865], NQB[2] [NQB-PHB] или эквивалентные локальные кода DSCP (см. [L4S-DIFFSERV]).

Отметим для ясности, что не рекомендуется, например, в целях локальной оптимизации, классифицировать трафик в очередь L на основе идентификации прикладного уровня (скажем, DNS). Приложения не смогут полагаться на такую незапрошенную оптимизацию. Надёжней будет, если отправитель установит подходящий идентификатор уровня IP, такой как один из указанных выше кодов Diffserv.

Таким образом, элемент сети, реализующий L4S в общей очереди, **может** классифицировать дополнительные пакеты в очередь L на основе идентификаторов, не связанных с полем IP-ECN, но этому трафику **следует** быть «безопасным» для смешивания с трафиком L4S (см. 5.4.1.1.1. "Безопасный" неотзывчивый трафик).

Пакет с одним из таких идентификаторов (не ECN) для включения в очередь L не будет получать маркировку L4S ECN, если он явно не содержит код ECT(1) или CE. Спецификация L4S AQM **должна** задавать поведение для пакетов с неожиданными комбинациями кодов, например, классификатор, не связанный с ECN для очереди L и ECT(0) в поле IP-ECN (примеры подходящих комбинаций приведены в параграфе 2.5.1.1 спецификации DualQ [RFC9332]).

Отметим для понимания, что некоторые операторы могут применят не связанные с ECN идентификаторы (такие как указаны выше), которые они считают подходящими для идентификации не относящегося к L4S трафика с целью безопасного смешивания с трафиком L4S. Это не является для хостов альтернативой индикации передачи им пакетов L4S. Лишь код ECT(1) ECN указывает элементам сети, что хост передаёт пакеты L4S (а CE указывает, что они могли быть переданы с кодом ECT(1)). В частности, ECT(1) указывает, что хост заявляет о своём соответствии транспортным требованиям, указанным в разделе 4.

Узлу сети **недопустимо** менять код Not-ECT или ECT(0) в поле IP-ECN на идентификатор L4S для включения не относящихся к L4S пакетов в очередь L. Это гарантирует, что коды сохранятся для возможного последующего использования на пути через сеть. Если несоответствующий или враждебный узел сети заменить ECT(0) на ECT(1), пакет впоследствии может получить маркер ECN от нисходящего L4S AQM, но отправитель отреагирует на индикацию насыщения, считая, что он передал классический пакет. Это может приводить к чрезмерным уступкам в пользу других потоков L4S в том же узком месте нисходящего направления.

5.4.1.1.1. "Безопасный" неотзывчивый трафик

Предыдущий параграф требует, чтобы неотзывчивый трафик был «безопасен» для смешивания с L4S. В идеале это означает, что отправитель никогда не передаёт последовательностей пакетов со скоростью, превышающей доступную в узком месте пропускную способность. Однако неотзывчивый транспорт обычно даже не знает пропускную

[1]Internet of Things - Internet вещей.
[2]Non-Queue-Building - без очередей.

способность узких мест на пути, не говоря уже о её доступности. Тем не менее, приложение можно считать достаточно безопасным, если оно распределяет пакеты (не обязательно с абсолютной регулярностью) так, что максимальная мгновенная скорость остаётся значительно ниже типичной скорости широкополосного доступа.

Это туманное, но полезное определение, поскольку ему соответствуют многие приложения, заинтересованные в малой задержке, такие как DNS, голос, синхронизация игр, RPC, ACK, keep-alive.

Низкоскоростные потоки, такие как голос и синхронизация игр, могут не применять постоянно адаптирующийся контроль перегрузок на основе ECN, но они должны, по меньшей мере, использовать «автоматический выключатель» (circuit-breaker) в качестве отклика на перегрузку [RFC8083]. Если объем трафика от неотзывчивых приложений достаточно велик для перегрузки канала, это хотя бы защитит пропускную способность, доступную реагирующим на перегрузку приложениям. Однако задержка в очереди L в результате вероятно возрастёт до типичного для Classic AQM (на основе отбрасывания) уровня. Если оператор сочтёт, что такого самоограничения недостаточно, он может захотеть управлять очередью L (см. параграф 8.2 описания архитектуры L4S [RFC9330]).

5.4.1.2. Исключение трафика из обработки L4S

Как расширение приведённого выше примера, оператор может захотеть исключить для части трафика обработку L4S из соображений политики, например, безопасности (трафик от враждебных источников) или коммерции (например, ограничить получаемые преимущества L4S некоторым бизнес-клиентам). В таких случаях классификатор **должен** указывать подходящие локальные идентификаторы (например, адреса отправителей) до применения к несоответствующему трафику сквозного идентификатора L4S ECN.

Узлу сети **недопустимо** менять сквозной идентификатор L4S ECN с L4S на Classic, когда решение оператора исключает локально обработку L4S для части трафика. Сквозной идентификатор L4S сохраняется для использования другими операторами или может применяться в политике оператора независимо от заданных локально идентификаторов. Такой подход позволяет любому оператору удалить заданные локально исключения в будущем, например, при распространении преимуществ L4S на всех клиентов. Если несоответствующий или вредоносный сетевой узел меняет код ECT(1) на ECT(0), пакет может быть промаркирован с использованием ECN нисходящим Classic ECN AQM. Отправителям L4S нужно обнаруживать и обрабатывать такую маркировку (п. 3 в параграфе 4.3), но это не делает упомянутую замену кода приемлемой, поскольку обнаружение не будет немедленным и совершенным.

Сетевой узел, поддерживающий L4S, но исключающий некоторые пакеты с идентификатором L4S из обработки L4S, **должен** все равно применять маркировку или отбрасывание, совместимые с откликами L4S на перегрузку. Например, он может отбрасывать такие пакеты с той же вероятностью, что и классические пакеты, или использовать для них маркировку ECN с вероятностью, соответствующей трафику L4S (например, связанной вероятностью DualQ Coupled AQM), но ориентируясь на классическую целевую задержку. **Недопустимо** применять для таких пакетов маркировку ECN с классической вероятностью, поскольку это может запутать отправителя.

5.4.1.3. Обобщённая комбинация L4S и других идентификаторов

Механизм L4S озабочен малой задержкой, которую он может обеспечить для всего трафика без дифференциации и потребности воздействия на распределение пропускной способности. Diffserv обеспечивает дифференциацию как для пропускной способности, та и для задержки, но управление задержкой зависит от управления пропускной способностью. L4S и Diffserv можно сочетать, если оператор сети хочет управлять распределением пропускной способности и обеспечивать малую задержку для любого трафика в рамках одного из выделений пропускной способности (вместо обеспечения малой задержки путём ограничения пропускной способности) [L4S-DIFFSERV].

Приведённые выше примеры DualQ были сформулированы в контексте принятого по умолчанию Best Effort PHB[1] с использованием 2 очередей - очереди с малой задержкой (L) и классической (C) очереди. Предполагается, что наиболее распространённой и полезной будет одна структура DualQ. Но в более общем плане оператор может выбрать управление пропускной способностью через иерархию Diffserv PHB на узле и предлагать один (или несколько) вариантов PHB с использованием пары очередей для очереди с малой задержкой и классическим вариантом PHB.

В первом случае, если предположить, что элемент сети не предлагает PHB, кроме DualQ, для пакетов с кодом ECT(1) или CE, элемент сети будет классифицировать их для обработки L4S независимо от кода DSCP. Если пакет имел, например код EF в поле DSCP, элемент сети может классифицировать его в очередь L независимо от кода ECN. Однако при включении DualQ в иерархию других PHB классификатор будет относить часть трафика к другим PHB на основе поля DSCP до распределения между очередью с малой задержкой и классической очередью (на основе ECT(1), CE и, возможно, EF DSCP или иных идентификаторов, как в примере выше). В [L4S-DIFFSERV] дано много примеров для исполнения различных требований.

В [L4S-DIFFSERV] описано, как оператор может использовать L4S для обеспечения малой задержки вместе с Diffserv для распределения пропускной способности. Выделены два основных подхода. Оператор может разделить тот или иной вариант Diffserv PHB между услугами L4S и Classic или распределить услуги L4S и/или Classic между разными Diffserv PHB. В любом случае пакет будет классифицироваться по его кодам Diffserv и ECN.

Имеется много способов сочетания идентификатора L4S ECN (ECT(1) и CE) с другими идентификаторами для достижения конкретных целей. В приведённой ниже классификации указаны некоторые из таких способов. Пометка «Рекомендовано для стандартного применения» означает, что эту комбинацию можно применять на передающих хостах и в сетях. Пометка «Для локального применения» означает, что комбинация применима лишь внутри сети.

1. Идентификаторы, дополняющие идентификатор L4S.
   a. Включение дополнительного трафика в очередь L (рекомендовано для стандартного и локального применения)
   b. Исключение части трафика из очереди L (для локального применения)
2. Идентификаторы для включения классификации L4S в иерархию PHB (рекомендовано для стандартного и локального применения)

[1] Per-Hop Behaviour - поведение по этапам пересылки.

a. PHB до классификации L4S ECN.

b. PHB после классификации L4S ECN.

5.4.2. Очереди по потокам с добавлением других идентификаторов к L4S

На узле с очередями по потокам (например, FQ-CoDel [RFC8290]) идентификатор L4S может дополняться идентификатором потока на транспортном уровне в качестве дополнительной детализации (т. е. очереди Not-ECT и ECT(0) отделены от ECT(1) и CE). В Приложении B (Риск нарушения порядка классических пакетов CE внутри потока) рассматривается возникающая неоднозначность, когда пакеты, изначально помеченные кодом ECT(0), маркируются кодом CE восходящим AQM до их прибытия на узел, который классифицирует CE как L4S. Указано, что риск нарушения порядка пренебрежимо мал, а последствия возможного нарушения порядка минимальны.

В качестве альтернативы можно предположить, что смешивание идентификаторов Classic и L4S не соответствует интересам самого потока. Тогда механизм AQM может использовать поле IP-ECN для переключения между классическим поведением и L4S AQM внутри очереди для одного потока. Например, для пакетов с поддержкой ECN механизм AQM может состоять из простого порога маркировки, а идентификатор L4S ECN может просто выбрать более низкий порог, чем это сделал бы идентификатор Classic ECN.

5.5. Ограничение всплесков пакетов из каналов

Так же как отправителям нужно ограничивать всплески (burst) пакетов (параграф 4.3), каналы должны ограничивать вносимые ими всплески (burstiness). В обоих случаях (отправители и каналы) это является компромиссом, поскольку групповая (batch) обработка пакетов выполняется осмысленно, например, для повышения эффективности обработки или более эффективного использования задержки получения доступа к среде. Некоторые придерживаются мнения, что не нужно сокращать всплески у отправителя ниже величины всплесков, возникающих в каналах (и обратно). Однако сокращение наибольшей задержки переносит внимание на следующую по величине и т. д.

Этот документ не задаёт количественных требований по ограничению возникающих в каналах пиков задержки, особенно по причине того, что канальные технологии выходят за рамки спецификации L4S. Тем не менее, в двух следующих параграфах очерчены возможности устранения канальных всплесков при реализации и внедрении L4S.

5.5.1. Ограничение всплесков пакетов из каналов с L4S AQM

Было бы бессмысленно внедрять L4S AQM для конкретной технологии канального уровня, не рассмотрев возможностей сокращения задержки всплесков, вносимой этой технологией. Это по меньшей мере ограничило бы всплески, которые иначе внёс бы канал в проходящий трафик, что вызывало бы «скачки» обратной связи у отправителя, а также возможную дополнительную задержку в очереди нисходящего направления. Этот документ не предполагает даже рекомендаций в части целевого значения задержки всплесков, пока в отрасли не будет набрано достаточного опыта применения L4S. Однако, как предложено в параграфе 4.3, не представляется необходимым ограничивать всплески ниже примерно 10% от минимального базового значения RTT в типовом варианте внедрения (например, длительность всплеска 250 мксек в общедоступной сети Internet).

5.5.2. Ограничение всплесков пакетов из восходящих каналов с L4S AQM

Первоначальные эксперименты с L4S заключаются во внедрении L4S AQM в узких местах и контроля перегрузок L4S у отправителей. Предполагается увидеть взаимодействия с восходящими каналами, наиболее подверженными пикам и побудить операторов снизить пики в настраиваемых каналах их сетей или хотя бы найти компромисс в части целевой задержки некоторых L4S AQM. Это может потребовать специальной переработки для наиболее проблемных каналов. Побочные эффекты первоначального внедрения L4S будут частью полученного в экспериментах L4S опыта.

Детали таких изменений для каналов выходят за рамки этого документа. Тем не менее, при внедрении технологии L4S на выходном интерфейсе устройства имеет смысл рассмотреть возможности сокращения всплесков, приходящих из других входных интерфейсов. Например, при реализации L4S AQM на соединении с восходящим интерфейсом WAN в домашнем шлюзе можно рассмотреть варианты изменения профилей Wi-Fi для интерфейсов того же шлюза, чтобы смягчить входные пики агрегированных кадров Wi-Fi от других станций Wi-Fi.

6. Поведение в туннелях и при инкапсуляции

6.1. Неизменность в туннелях и при инкапсуляции

Предполагается, что идентификаторы L4S будут работать через туннели и внутри их без изменений, пока туннель переносит поля ECN любым из способов, определённых с первого варианта 2001 г. [RFC3168]. L4S будет работать (но не полагаться на них) и последующими обновлениями передачи ECN [RFC4301], [RFC6040], [ECN-SHIM]. Однако имеется вероятность, что до сих пор некоторые туннели совсем не передают ECN. В таких случаях L4S будет работать через туннель, но внутри него внешний заголовок L4S будет выглядеть как Classic.

Механизмы AQM обычно реализуются там, где буфер уровня «питает» нижележащий уровень, поэтому они не зависят от инкапсуляции канального уровня. Если узкое место канала не знает об IP, ожидается, что идентификаторы L4S всё равно будут работать без изменений с любой инкапсуляцией канального уровня, пока они распространяются в поле IP-ECN, как задано для этой канальной технологии (например, MPLS [RFC5129] или TRILL[1] [TRILL-ECN-SUPPORT]). В некоторых из таких случаев (например, в коммутаторах L3 Ethernet) AQM обращается к заголовку уровня IP внутри инкапсуляции, поэтому снова ожидается, что идентификатор L4S будет работать без изменений. Тем не менее, программа по определению ECN для других нижележащих уровней продолжается [ECN-ENCAP].

6.2. Поведение VPN для снижения ограничений Anti-Replay

Если смесь пакетов L4S и Classic передаётся в одну защищённую связь (security association или SA) VPN и на выходе VPN реализована необязательная функция предотвращения повторного использования (anti-replay), она может необоснованно отбрасывать классические пакеты (или отвергать записи из них), ошибочно воспринимая их большую

[1]Transparent Interconnection of Lots of Links - прозрачное соединение множества каналов.

задержку в очереди как replay-атаку (см. Dropped Packets for Tunnels with Replay Protection Enabled в [Heist21], где рассматривается возможное влияние на производительность). Эта проблема известна для IPsec [RFC4301] и DTLS [RFC9147] VPN, поскольку в них применяются аналогичные механизмы окна anti-replay, которые способны проверять наличие повтора лишь в рамках окна, поэтому при окне меньше степени нарушения порядка возможны ложные обнаружения атак с отбрасыванием заднего края окна. Спецификации IPsec AH[1] [RFC4302] и ESP[2] [RFC4303] предлагают разработчикам изменять размер окна anti-replay в зависимости от скорости, а в DTLS v1.3 [RFC9147] сказано: «Получателю **следует** выбирать достаточно большое окно, чтобы можно было обработать правдоподобное нарушение порядка, которое зависит от скорости передачи данных.» Однако на практике размер окна предотвращения повторов в VPN не всегда масштабируется должным образом.

Если сеть VPN, участвующая в эксперименте L4S, столкнётся с неподобающим обнаружением повтора, прежде всего нужно убедиться, что выход настроен корректно в соответствии с приведёнными выше требованиями к размеру окна.

Если реализация выхода VPN не поддерживает достаточно большое окно anti-replay, например, из-за аппаратных ограничений, можно выбрать один из временных вариантов, указанных ниже в порядке снижения предпочтений.

- Если VPN можно настроить для классификации в разные SA, индексируемые DSCP, с использованием локально выбранных подходящих DSCP для классических пакетов и L4S. Коды DSCP могут устанавливаться сетью (на основе младшего бита в поле IP-ECN) или передающим хостом и нужны лишь до входа в VPN.
- Если приведённый выше вариант невозможен, а применять L4S нужно, подойдёт любой из 2 вариантов:
  - отключение защиты от повторов на выходе VPN после рассмотрения влияния этого на безопасность (поддержка отключения anti-replay обязательна в IPsec и DTLS);
  - отключение распространения ECN во внешний заголовок на входе туннеля, что приведёт к потере преимуществ L4S и Classic ECN при передаче через VPN.

Изменение реализаций VPN выходит за рамки этого документа, поэтому здесь основное внимание уделено изменению конфигурации. Хотя документ не задаёт требований к реализации VPN, решение вопроса о необходимости таких требований может быть одним из аспектов экспериментов с L4S.

7. Эксперименты с L4S

В этом разделе рассмотрены открытые вопросы, на которых нужно сосредоточится в экспериментах L4S, а также нерешенные проблемы, которые нужно исследовать в рамках экспериментов, чтобы их можно было полностью решить на этапе рабочей группы. В разделе также перечислены показатели, которые нужно отслеживать в экспериментах (сводный текст в других документах L4S), а также возможные направления будущих исследований.

В дополнение к этому разделу i) спецификация DualQ [RFC9332] задаёт требования к эксплуатации и управлению для DualQ Coupled AQM, а ii) общие требования в части эксплуатации и управления для экспериментов с контролем перегрузок L4S приведены выше в разделах 4 и 5 (например, требования к сосуществованию и масштабированию, а также варианты поэтапного развёртывания.

Спецификация каждого элемента расширяемого контроля перегрузок должна будет включать зависящие от протокола требования к настройке и мониторингу производительности в процессе экспериментов. Полезный контрольный список имеется в Приложении A к [RFC5706].

7.1. Нерешённые вопросы

Предполагается, что эксперименты с L4S дадут ответы на приведённые ниже вопросы:

- Были ли внедрены все части L4S и, если да, какая часть путей поддерживает их?
  - Какие типы L4S AQM внедрены (например, FQ, связанные и несвязанные DualQ и т. п.)? Насколько превалировал каждый из внедрённых вариантов?
  - Отличаются ли схемы сигнализации развёрнутых AQM от ожидаемых при задании требований Prague для конечных точек?
- Ведёт ли использование L4S через Internet к большей удовлетворённости пользователей?
- Позволяет ли L4S внедрять новые интерактивные приложения?
- Обеспечивает ли внедрение L4S через Internet к улучшению показателей:
  - задержки в очередях (средней и при 99-м процентиле) при разной нагрузке;
  - загрузки каналов;
  - истощения и беспристрастности;
  - диапазон масштабирования скоростей потоков и RTT?
- Насколько зависит производительность услуг L4S от пропускной способности в узком месте и RTT для пути?
- Насколько каналы со всплесками трафика в Internet влияют на производительность L4S (см. Underutilization with Bursty Links в [Heist21]) и сколь часто такие каналы встречаются? Насколько было необходимо и было реализовано ограничение всплесков у отправителей и на каналах (особенно радио) или насколько потребовалось увеличить целевую задержку L4S, чтобы работать с такими всплесками (см. п. 7 в параграфе 4.3 и параграф 5.5.2)?
- Указывает ли начальный эксперимент с ошибочно идентифицированным скачкообразным (bursty) трафиком при большом значении RTT (см. Underutilization with Bursty Traffic в [Heist21]) похожие проблемы при малых

[1]Authentication Header - заголовок аутентификации.
[2]Encapsulating Security Payload - инкапсулированные защищённые данные.

RTT и, если да, насколько эффективно решение из Приложения A.1 к спецификации DualQ [RFC9332] (или иное решение)?

- Требовалась ли обычно защита очередей по потокам?
  - Насколько хорошо работала защита от перегрузок или защита очереди?
- Как потоки L4S сосуществовали с классическими потоками в узком месте?
  - Как часто возникали проблемы?
  - Что вызывало проблемы сосуществования и были ли проблемы связаны с механизмами Classic ECN AQM с одной очередью (в предположении возможности отличить очередь Classic ECN AQM от FQ)?
- Насколько распространены проблемы сервиса L4S, связанные с туннелями или инкапсуляцией при отсутствии поддержки декапсуляции ECN?
- Насколько легко было внедрение полностью совместимого с L4S контроля перегрузки для различных транспортных протоколов (TCP, QUIC, RMCAT и т. п.)?

В ранних экспериментах с L4S необходимо выполнять мониторинг ущерба для другого трафика, в частности, истощение пропускной способности или избыточные задержки в очередях. Для отдельного потока сложно (или совсем невозможно) измерить его влияние на другие потоки, поэтому мониторинг нужно выполнять с отслеживанием по потокам и/или классам трафика.

7.2. Нерешённые проблемы

- Какой наилучший способ решения задачи для L4S в узких местах с Classic ECN AQM в одной очереди с учётом ошибочного восприятия L4S AQM как ECN AQM? См. эксплуатационные рекомендации для L4S [L4SOPS].
- Решение проблемы недостаточно взаимодействия между текущим контролем перегрузок L4S и CoDel с поддержкой лишь Classic ECN при запуске потока. Исходно это было связано с ошибкой при инициализации среднего уровня перегрузки в Linux-реализации TCP Prague. Ошибка была быстро устранена и исключено соответствующее влияние на производительность, но дальнейшее улучшение будет полезным (например, путём изменение CoDel, расширяемых элементов контроля перегрузки или того и другого сразу).

7.3. Будущий потенциал

Исследователи могут увидеть, что L4S открывает ряд интересных направлений для изучения:

- возможность ускоренного схождения и отслеживает доступной пропускной способности;
- возможность улучшения определённых канальных технологий и межуровневых взаимодействий с ними;
- возможность использования виртуальных очередей, например, для дальнейшего сокращения задержки или сохранения запаса при изменении пропускной способности в радио-сетях;
- разработка и спецификация контроля перегрузок на пути возврата с использованием элементов L4S (например, AccECN или QUIC);
- определение следующего шага после сокращения задержек (кроме скорости света);
- расширение набора имеющихся L4S AQM;
- новые приложения с поддержкой L4S.

8. Взаимодействие с IANA

Этот экспериментальный документ задаёт семантику кода 01 в поле ECN заголовка IP. Для Experimental RFC процесс выделения кода в заголовках IP (v4 и v6) регулируется предложенным стандартом [RFC8311], обновившим [RFC3168].

Агентство IANA обновило запись 01 в реестре ECN Field (Bits 6-7) (https://www.iana.org/assignments/dscp-registry/).

Таблица 1. Реестр ECN Field (Bits 6-7).

Значение	***Обозначение***	***Документ***
01	ECT(1) (ECN-Capable Transport(1))[1]	[RFC8311] [RFC Errata 5399] RFC 9331

9. Вопросы безопасности

Подходы к обеспечению целостности сигналов, использующих новый идентификатор, представлены в Приложении C.1. Соображения безопасности связанные со злоупотреблениями при использовании идентификаторов приведены в описании архитектуры L4S [RFC9330] и так же рассмотрены вопросы управления скоростью и задержки в L4S.

Выбор ECT(1) в качестве идентификатора L4S задаёт разную трактовку кодов ECT(0) и ECT(1), которые в [RFC3168] указаны как различающиеся, но эквивалентные. В L4S ECN узлам сети также не разрешается менять один код на другой, даже при выборе оператором сети локальной обработки в соответствии с другим кодом (см. и параграфы 5.4.1.1 и 5.4.1.2). В указанных параграфах также рассмотрено возможное влияние замены кода несовместимым или вредоносным узлом сети. Эта спецификация не меняет эффекты других неожиданных изменений поля IP-ECN, описанных в разделе 18 [RFC3168].

Если окно anti-replay на выходе VPN слишком мало, различия в задержке будут ошибочно восприниматься как replay-атака с отбрасыванием задержанных пакетов (например, классических) из одной защищённой связи (SA) с менее задержанными пакетами (например, L4S). В параграфе 6.2 рекомендуется настраивать VPN, применяемые в экспериментах L4S, с достаточно большим окном anti-replay, как того требуют соответствующие спецификации, а также рассматриваются иные варианты.

[1] ECT(1) предназначен лишь для экспериментального использования ([RFC8311], параграф 4.2).

Если участвующий в эксперименте L4S пользователь организует VPN без учёта приведённых выше рекомендаций и разрешит кому-либо передавать трафик в свою VPN, он станет уязвим для DoS-атак, где злоумышленник может вынудить механизм предотвращения повторного использования в VPN отбрасывать достаточно большой объем классического (C) трафика (если он имеется) для существенного снижения скорости. Когда пользователь активно загружает трафик C, атакующий может отправлять трафик C в VPN для заполнения канала в узком месте, а затем время от времени передавать пакеты L4S для увеличения шансов выхода за пределы окна предотвращения повторов в VPN. Пользователь может предотвратить такие атаки, следуя рекомендациям параграфа 6.2.

Рекомендации по обнаружению потерь на основе времени предотвращают атаки ACK-splitting, описанные в [Savage-TCP].

10. Литература

10.1. Нормативные документы

[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>.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/info/rfc3168>.

[RFC4774] Floyd, S., "Specifying Alternate Semantics for the Explicit Congestion Notification (ECN) Field", BCP 124, RFC 4774, DOI 10.17487/RFC4774, November 2006, <https://www.rfc-editor.org/info/rfc4774>.

[RFC6679] Westerlund, M., Johansson, I., Perkins, C., O'Hanlon, P., and K. Carlberg, "Explicit Congestion Notification (ECN) for RTP over UDP", RFC 6679, DOI 10.17487/RFC6679, August 2012, <https://www.rfc-editor.org/info/rfc6679>.

10.2. Дополнительная литература

[A2DTCP] Zhang, T., Wang, J., Huang, J., Huang, Y., Chen, J., and Y. Pan, "Adaptive-Acceleration Data Center TCP", IEEE Transactions on Computers, Volume 64, Issue 6, pp. 1522-1533, DOI 10.1109/TC.2014.2345393, June 2015, <https://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6871352>.

[ACCECN] Briscoe, B., Kühlewind, M., and R. Scheffenegger, "More Accurate ECN Feedback in TCP", Work in Progress, Internet-Draft, draft-ietf-tcpm-accurate-ecn-22, 9 November 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-tcpm-accurate-ecn-22>.

[Ahmed19] Ahmed, A.S., "Extending TCP for Low Round Trip Delay", Master's Thesis, University of Oslo, August 2019, <https://www.duo.uio.no/handle/10852/70966>.

[Alizadeh-stability] Alizadeh, M., Javanmard, A., and B. Prabhakar, "Analysis of DCTCP: Stability, Convergence, and Fairness", SIGMETRICS '11: Proceedings of the ACM SIGMETRICS Joint International Conference on Measurement and Modeling of Computer Systems, pp. 73-84, DOI 10.1145/1993744.1993753, June 2011, <https://dl.acm.org/doi/10.1145/1993744.1993753>.

[ARED01] Floyd, S., Gummadi, R., and S. Shenker, "Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management", ACIRI Technical Report 301, August 2001, <https://www.icsi.berkeley.edu/icsi/node/2032>.

[BBR-CC] Cardwell, N., Cheng, Y., Hassas Yeganeh, S., Swett, I., and V. Jacobson, "BBR Congestion Control", Work in Progress, Internet-Draft, draft-cardwell-iccrg-bbr-congestion-control-02, 7 March 2022, <https://datatracker.ietf.org/doc/html/draft-cardwell-iccrg-bbr-congestion-control-02>.

[BBRv2] "TCP BBR v2 Alpha/Preview Release", commit 17700ca, June 2022, <https://github.com/google/bbr>.

[Bufferbloat] The Bufferbloat community, "Bufferbloat", <https://bufferbloat.net/>.

[COBALT] Palmei, J., Gupta, S., Imputato, P., Morton, J., Tahiliani, M. P., Avallone, S., and D. Täht, "Design and Evaluation of COBALT Queue Discipline", IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN), DOI 10.1109/LANMAN.2019.8847054, July 2019, <https://ieeexplore.ieee.org/abstract/document/8847054>.

[CTCP] Sridharan, M., Tan, K., Bansal, D., and D. Thaler, "Compound TCP: A New TCP Congestion Control for High-Speed and Long Distance Networks", Work in Progress, Internet-Draft, draft-sridharan-tcpm-ctcp-02, 3 November 2008, <https://datatracker.ietf.org/doc/html/draft-sridharan-tcpm-ctcp-02>.

[DCttH19] De Schepper, K., Bondarenko, O., Tilmans, O., and B. Briscoe, "'Data Centre to the Home': Ultra-Low Latency for All", Updated RITE project Technical Report, July 2019, <https://bobbriscoe.net/projects/latency/dctth_journal_draft20190726.pdf>.

[DOCSIS-QPROT] Briscoe, B., Ed. and G. White, "The DOCSIS® Queue Protection Algorithm to Preserve Low Latency", Work in Progress, Internet-Draft, draft-briscoe-docsis-q-protection-06, 13 May 2022, <https://datatracker.ietf.org/doc/html/draft-briscoe-docsis-q-protection-06>.

[DualPI2Linux] Albisser, O., De Schepper, K., Briscoe, B., Tilmans, O., and H. Steen, "DUALPI2 - Low Latency, Low Loss and Scalable (L4S) AQM", Proceedings of Linux Netdev 0x13, March 2019, <https://www.netdevconf.org/0x13/session.html?talk-DUALPI2-AQM>.

[Dukkipati06] Dukkipati, N. and N. McKeown, "Why Flow-Completion Time is the Right Metric for Congestion Control", ACM SIGCOMM Computer Communication Review, Volume 36, Issue 1, pp. 59-62, DOI 10.1145/1111322.1111336, January 2006, <https://dl.acm.org/doi/10.1145/1111322.1111336>.

[ECN++] Bagnulo, M. and B. Briscoe, "ECN++: Adding Explicit Congestion Notification (ECN) to TCP Control Packets", Work in Progress, Internet-Draft, draft-ietf-tcpm-generalized-ecn-10, 27 July 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-tcpm-generalized-ecn-10>.

[ECN-ENCAP] Briscoe, B. and J. Kaippallimalil, "Guidelines for Adding Congestion Notification to Protocols that Encapsulate IP", Work in Progress, Internet-Draft, draft-ietf-tsvwg-ecn-encap-guidelines-17, 11 July 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-tsvwg-ecn-encap-guidelines-17>.

[ecn-fallback] Briscoe, B. and A. Ahmed, "TCP Prague Fall-back on Detection of a Classic ECN AQM", Technical Report: TR-BB-2019-002, DOI 10.48550/arXiv.1911.00710, February 2021, <https://arxiv.org/abs/1911.00710>.

[ECN-SHIM] Briscoe, B., "Propagating Explicit Congestion Notification Across IP Tunnel Headers Separated by a Shim", Work in Progress, Internet-Draft, draft-ietf-tsvwg-rfc6040update-shim-15, 11 July 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-tsvwg-rfc6040update-shim-15>.

[Heist21] "L4S Tests", commit e21cd91, August 2021, <https://github.com/heistp/l4s-tests>.

[L4S-DIFFSERV] Briscoe, B., "Interactions between Low Latency, Low Loss, Scalable Throughput (L4S) and Differentiated Services", Work in Progress, Internet-Draft, draft-briscoe-tsvwg-l4s-diffserv-02, 1 November 2018, <https://datatracker.ietf.org/doc/html/draft-briscoe-tsvwg-l4s-diffserv-02>.

[L4Seval22] De Schepper, K., Albisser, O., Tilmans, O., and B. Briscoe, "Dual Queue Coupled AQM: Deployable Very Low Queuing Delay for All", Preprint submitted to IEEE/ACM Transactions on Networking, DOI 10.48550/arXiv.2209.01078, September 2022, <https://arxiv.org/abs/2209.01078>.

[L4SOPS] White, G., Ed., "Operational Guidance for Deployment of L4S in the Internet", Work in Progress, Internet-Draft, draft-ietf-tsvwg-l4sops-03, 28 April 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-tsvwg-l4sops-03>.

[LinuxPacedChirping] Misund, J. and B. Briscoe, "Paced Chirping — Rethinking TCP start-up", Proceedings of Linux Netdev 0x13, March 2019, <https://legacy.netdevconf.info/0x13/session.html?talk-chirp>.

[NQB-PHB] White, G. and T. Fossati, "A Non-Queue-Building Per-Hop Behavior (NQB PHB) for Differentiated Services", Work in Progress, Internet-Draft, draft-ietf-tsvwg-nqb-15, 11 January 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-tsvwg-nqb-15>.

[PI2] De Schepper, K., Bondarenko, O., Tsang, I., and B. Briscoe, "PI^2: A Linearized AQM for both Classic and Scalable TCP", Proceedings of ACM CoNEXT 2016, pp. 105-119, DOI 10.1145/2999572.2999578, December 2016, <https://dl.acm.org/citation.cfm?doid=2999572.2999578>.

[PRAGUE-CC] De Schepper, K., Tilmans, O., and B. Briscoe, Ed., "Prague Congestion Control", Work in Progress, Internet-Draft, draft-briscoe-iccrg-prague-congestion-control-01, 11 July 2022, <https://datatracker.ietf.org/doc/html/draft-briscoe-iccrg-prague-congestion-control-01>.

[PragueLinux] Briscoe, B., De Schepper, K., Albisser, O., Misund, J., Tilmans, O., Kühlewind, M., and A. Ahmed, "Implementing the 'TCP Prague' Requirements for L4S", Proceedings of Linux Netdev 0x13, March 2019, <https://www.netdevconf.org/0x13/session.html?talk-tcp-prague-l4s>.

[QV] Briscoe, B. and P. Hurtig, "Report on Prototype Development and Evaluation of Network and Interaction Techniques", RITE Technical Report, Deliverable 2.3, Appendix C.2: "Up to Speed with Queue View", September 2015, <https://riteproject.files.wordpress.com/2015/12/rite-deliverable-2-3.pdf>.

[RELENTLESS] Mathis, M., "Relentless Congestion Control", Work in Progress, Internet-Draft, draft-mathis-iccrg-relentless-tcp-00, 4 March 2009, <https://datatracker.ietf.org/doc/html/draft-mathis-iccrg-relentless-tcp-00>.

[RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, DOI 10.17487/RFC2474, December 1998, <https://www.rfc-editor.org/info/rfc2474>.

[RFC3246] Davie, B., Charny, A., Bennet, J.C.R., Benson, K., Le Boudec, J.Y., Courtney, W., Davari, S., Firoiu, V., and D. Stiliadis, "An Expedited Forwarding PHB (Per-Hop Behavior)", RFC 3246, DOI 10.17487/RFC3246, March 2002, <https://www.rfc-editor.org/info/rfc3246>.

[RFC3540] Spring, N., Wetherall, D., and D. Ely, "Robust Explicit Congestion Notification (ECN) Signaling with Nonces", RFC 3540, DOI 10.17487/RFC3540, June 2003, <https://www.rfc-editor.org/info/rfc3540>.

[RFC3649] Floyd, S., "HighSpeed TCP for Large Congestion Windows", RFC 3649, DOI 10.17487/RFC3649, December 2003, <https://www.rfc-editor.org/info/rfc3649>.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/info/rfc4301>.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI 10.17487/RFC4302, December 2005, <https://www.rfc-editor.org/info/rfc4302>.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, DOI 10.17487/RFC4303, December 2005, <https://www.rfc-editor.org/info/rfc4303>.

[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>.

[RFC4341] Floyd, S. and E. Kohler, "Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 2: TCP-like Congestion Control", RFC 4341, DOI 10.17487/RFC4341, March 2006, <https://www.rfc-editor.org/info/rfc4341>.

[RFC4342] Floyd, S., Kohler, E., and J. Padhye, "Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 3: TCP-Friendly Rate Control (TFRC)", RFC 4342, DOI 10.17487/RFC4342, March 2006, <https://www.rfc-editor.org/info/rfc4342>.

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

[RFC5033] Floyd, S. and M. Allman, "Specifying New Congestion Control Algorithms", BCP 133, RFC 5033, DOI 10.17487/RFC5033, August 2007, <https://www.rfc-editor.org/info/rfc5033>.

[RFC5129] Davie, B., Briscoe, B., and J. Tay, "Explicit Congestion Marking in MPLS", RFC 5129, DOI 10.17487/RFC5129, January 2008, <https://www.rfc-editor.org/info/rfc5129>.

[RFC5348] Floyd, S., Handley, M., Padhye, J., and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification", RFC 5348, DOI 10.17487/RFC5348, September 2008, <https://www.rfc-editor.org/info/rfc5348>.

[RFC5562] Kuzmanovic, A., Mondal, A., Floyd, S., and K. Ramakrishnan, "Adding Explicit Congestion Notification (ECN) Capability to TCP's SYN/ACK Packets", RFC 5562, DOI 10.17487/RFC5562, June 2009, <https://www.rfc-editor.org/info/rfc5562>.

[RFC5622] Floyd, S. and E. Kohler, "Profile for Datagram Congestion Control Protocol (DCCP) Congestion ID 4: TCP-Friendly Rate Control for Small Packets (TFRC-SP)", RFC 5622, DOI 10.17487/RFC5622, August 2009, <https://www.rfc-editor.org/info/rfc5622>.

[RFC5681] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion Control", RFC 5681, DOI 10.17487/RFC5681, September 2009, <https://www.rfc-editor.org/info/rfc5681>.

[RFC5706] Harrington, D., "Guidelines for Considering Operations and Management of New Protocols and Protocol Extensions", RFC 5706, DOI 10.17487/RFC5706, November 2009, <https://www.rfc-editor.org/info/rfc5706>.

[RFC5865] Baker, F., Polk, J., and M. Dolly, "A Differentiated Services Code Point (DSCP) for Capacity-Admitted Traffic", RFC 5865, DOI 10.17487/RFC5865, May 2010, <https://www.rfc-editor.org/info/rfc5865>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <https://www.rfc-editor.org/info/rfc5925>.

[RFC6040] Briscoe, B., "Tunnelling of Explicit Congestion Notification", RFC 6040, DOI 10.17487/RFC6040, November 2010, <https://www.rfc-editor.org/info/rfc6040>.

[RFC6077] Papadimitriou, D., Ed., Welzl, M., Scharf, M., and B. Briscoe, "Open Research Issues in Internet Congestion Control", RFC 6077, DOI 10.17487/RFC6077, February 2011, <https://www.rfc-editor.org/info/rfc6077>.

[RFC6660] Briscoe, B., Moncaster, T., and M. Menth, "Encoding Three Pre-Congestion Notification (PCN) States in the IP Header Using a Single Diffserv Codepoint (DSCP)", RFC 6660, DOI 10.17487/RFC6660, July 2012, <https://www.rfc-editor.org/info/rfc6660>.

[RFC6675] Blanton, E., Allman, M., Wang, L., Jarvinen, I., Kojo, M., and Y. Nishida, "A Conservative Loss Recovery Algorithm Based on Selective Acknowledgment (SACK) for TCP", RFC 6675, DOI 10.17487/RFC6675, August 2012, <https://www.rfc-editor.org/info/rfc6675>.

[RFC7560] Kuehlewind, M., Ed., Scheffenegger, R., and B. Briscoe, "Problem Statement and Requirements for Increased Accuracy in Explicit Congestion Notification (ECN) Feedback", RFC 7560, DOI 10.17487/RFC7560, August 2015, <https://www.rfc-editor.org/info/rfc7560>.

[RFC7567] Baker, F., Ed. and G. Fairhurst, Ed., "IETF Recommendations Regarding Active Queue Management", BCP 197, RFC 7567, DOI 10.17487/RFC7567, July 2015, <https://www.rfc-editor.org/info/rfc7567>.

[RFC7713] Mathis, M. and B. Briscoe, "Congestion Exposure (ConEx) Concepts, Abstract Mechanism, and Requirements", RFC 7713, DOI 10.17487/RFC7713, December 2015, <https://www.rfc-editor.org/info/rfc7713>.

[RFC8033] Pan, R., Natarajan, P., Baker, F., and G. White, "Proportional Integral Controller Enhanced (PIE): A Lightweight Control Scheme to Address the Bufferbloat Problem", RFC 8033, DOI 10.17487/RFC8033, February 2017, <https://www.rfc-editor.org/info/rfc8033>.

[RFC8083] Perkins, C. and V. Singh, "Multimedia Congestion Control: Circuit Breakers for Unicast RTP Sessions", RFC 8083, DOI 10.17487/RFC8083, March 2017, <https://www.rfc-editor.org/info/rfc8083>.

[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>.

[RFC8257] Bensley, S., Thaler, D., Balasubramanian, P., Eggert, L., and G. Judd, "Data Center TCP (DCTCP): TCP Congestion Control for Data Centers", RFC 8257, DOI 10.17487/RFC8257, October 2017, <https://www.rfc-editor.org/info/rfc8257>.

[RFC8290] Hoeiland-Joergensen, T., McKenney, P., Taht, D., Gettys, J., and E. Dumazet, "The Flow Queue CoDel Packet Scheduler and Active Queue Management Algorithm", RFC 8290, DOI 10.17487/RFC8290, January 2018, <https://www.rfc-editor.org/info/rfc8290>.

[RFC8298] Johansson, I. and Z. Sarker, "Self-Clocked Rate Adaptation for Multimedia", RFC 8298, DOI 10.17487/RFC8298, December 2017, <https://www.rfc-editor.org/info/rfc8298>.

[RFC8311] Black, D., "Relaxing Restrictions on Explicit Congestion Notification (ECN) Experimentation", RFC 8311, DOI 10.17487/RFC8311, January 2018, <https://www.rfc-editor.org/info/rfc8311>.

[RFC8312] Rhee, I., Xu, L., Ha, S., Zimmermann, A., Eggert, L., and R. Scheffenegger, "CUBIC for Fast Long-Distance Networks", RFC 8312, DOI 10.17487/RFC8312, February 2018, <https://www.rfc-editor.org/info/rfc8312>.

[RFC8511] Khademi, N., Welzl, M., Armitage, G., and G. Fairhurst, "TCP Alternative Backoff with ECN (ABE)", RFC 8511, DOI 10.17487/RFC8511, December 2018, <https://www.rfc-editor.org/info/rfc8511>.

[RFC8888] Sarker, Z., Perkins, C., Singh, V., and M. Ramalho, "RTP Control Protocol (RTCP) Feedback for Congestion Control", RFC 8888, DOI 10.17487/RFC8888, January 2021, <https://www.rfc-editor.org/info/rfc8888>.

[RFC8985] Cheng, Y., Cardwell, N., Dukkipati, N., and P. Jha, "The RACK-TLP Loss Detection Algorithm for TCP", RFC 8985, DOI 10.17487/RFC8985, February 2021, <https://www.rfc-editor.org/info/rfc8985>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/info/rfc9000>.

[RFC9001] Thomson, M., Ed. and S. Turner, Ed., "Using TLS to Secure QUIC", RFC 9001, DOI 10.17487/RFC9001, May 2021, <https://www.rfc-editor.org/info/rfc9001>.

[RFC9147] Rescorla, E., Tschofenig, H., and N. Modadugu, "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3", RFC 9147, DOI 10.17487/RFC9147, April 2022, <https://www.rfc-editor.org/info/rfc9147>.

[RFC9330] Briscoe, B., Ed., De Schepper, K., Bagnulo, M., and G. White, "Low Latency, Low Loss, and Scalable Throughput (L4S) Internet Service: Architecture", RFC 9330, DOI 10.17487/RFC9330, January 2023, <https://www.rfc-editor.org/info/rfc9330>.

[RFC9332] De Schepper, K., Briscoe, B., Ed., and G. White, "Dual-Queue Coupled Active Queue Management (AQM) for Low Latency, Low Loss, and Scalable Throughput (L4S)", RFC 9332, DOI 10.17487/RFC9332, January 2023, <https://www.rfc-editor.org/info/rfc9332>.

[Savage-TCP] Savage, S., Cardwell, N., Wetherall, D., and T. Anderson, "TCP Congestion Control with a Misbehaving Receiver", ACM SIGCOMM Computer Communication Review, Volume 29, Issue 5, pp. 71–78, DOI 10.1145/505696.505704, October 1999, <https://dl.acm.org/doi/abs/10.1145/505696.505704>.

[SCReAM-L4S] "SCReAM", commit 140e292, November 2022, <https://github.com/EricssonResearch/scream>.

[SCTP-ECN] Stewart, R., Tüxen, M., and X. Dong, "ECN for Stream Control Transmission Protocol (SCTP)", Work in Progress, Internet-Draft, draft-stewart-tsvwg-sctpecn-05, 15 January 2014, <https://datatracker.ietf.org/doc/html/draft-stewart-tsvwg-sctpecn-05>.

[sub-mss-prob] Briscoe, B. and K. De Schepper, "Scaling TCP's Congestion Window for Small Round Trip Times", BT Technical Report: TR-TUB8-2015-002, DOI 10.48550/arXiv.1904.07598, May 2015, <https://arxiv.org/abs/1904.07598>.

[TCP-CA] Jacobson, V. and M. J. Karels, "Congestion Avoidance and Control", Laurence Berkeley Labs Technical Report, November 1988, <https://ee.lbl.gov/papers/congavoid.pdf>.

[TCPPrague] Briscoe, B., "Notes: DCTCP evolution 'bar BoF': Tue 21 Jul 2015, 17:40, Prague", message to the tcpPrague mailing list, July 2015, <https://www.ietf.org/mail-archive/web/tcpprague/current/msg00001.html>.

[TRILL-ECN-SUPPORT] Eastlake 3rd, D. and B. Briscoe, "TRILL (Transparent Interconnection of Lots of Links): ECN (Explicit Congestion Notification) Support", Work in Progress, Internet-Draft, draft-ietf-trill-ecn-support-07, 25 February 2018, <https://datatracker.ietf.org/doc/html/draft-ietf-trill-ecn-support-07>.

[VCP] Xia, Y., Subramanian, L., Stoica, I., and S. Kalyanaraman, "One more bit is enough", SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 37-48, DOI 10.1145/1080091.1080098, August 2005, <https://doi.acm.org/10.1145/1080091.1080098>.

Приложение A. Обоснование требований Prague L4S

Это приложение является информационным, а не нормативным. Здесь приведён список изменений имеющихся расширяемых средств контроля перегрузки, которые могут быть развёрнуты в общедоступной сети Internet и безопасно сосуществовать с имеющимся трафиком. Список дополняет нормативные требования раздела 4, которые отправитель должен выполнить до начала установки идентификатора L4S в пакетах, передаваемых в Internet. Наряду с обоснованием повышения безопасности (раздел 4) это приложение рассматривает повышение производительности (оптимизация).

Требования и рекомендации раздела 4 называют «требованиями Prague L4S», поскольку они исходно заданы на специальной встрече в рамках конференции IETF 94 в Праге [TCPPrague]. Поначалу их называли требованиями TCP Prague, но они применимы не только к TCP, поэтому название было обобщено для других транспортных протоколов, а TCP Prague применяется для конкретной реализации требований.

На момент создания протокол DCTCP [RFC8257] был наиболее распространенным решением с масштабированием. В своей текущей форме DCTCP предназначен для внедрения лишь в контролируемых средах. Развёртывание в Internet приведёт к многочисленным проблемам в части безопасности и производительности. Указанные в этом приложении изменения и дополнительные механизмы потребуются для внедрения в общедоступной сети Internet. Там, где нужны примеры, DCTCP служит базой, но требования раздела 4 в той же мере применимы к другим элементам расширяемого контроля перегрузок, включая адаптивную передачу в реальном масштабе времени и т. п., а не только приложения, которым нужна пропускная способность.

A.1. Обоснование требований к масштабированию транспорта

A.1.1. Использование идентификатора пакета L4S

Описание

Расширяемому контролю перегрузки нужно отличать свои пакеты от пакетов с классическим контролем перегрузки (см. нормативные требования в параграфе 4.1. Установка кода).

Мотивация

Узлам сети требуется возможность классифицировать пакеты L4S в очередь с применением маркировки L4S ECN без учёта состояния потока и изолировать пакеты L4S от задержки в очередях для классических пакетов.

A.1.2. Точные отклики ECN

Описание

Транспортный протокол для масштабируемого контроля перегрузки должен обеспечивать своевременные и точные отклики о степени маркировки ECN для всех пакетов (параграф 4.2. Требования к откликам транспорта).

Мотивация

Для классического контроля перегрузки требуется лишь отклик о возникновении перегрузки в интервале кругового обхода без точного указания отброшенных или помеченных кодом ECN пакетов. Поэтому при добавлении откликов ECN для TCP в 2001 г. [RFC3168] можно было не информировать отправителя о нескольких маркерах ECN в течение RTT. С тех пол были заданы более точные требования к маркировке ECN для TCP в [RFC7560], а в [ACCECN] задано обновление протокола TCP с учётом этих требований. В большинстве других транспортных протоколов эти требования уже выполняются (параграф 4.2. Требования к откликам транспорта).

A.1.3. Возможность замены классическим контролем перегрузки

Описание

Представляется возможным заменить расширяемый контроль перегрузки классическим (см. нормативные требования в п. 1 параграфа 4.3. Требования к откликам на перегрузку).

Мотивация

L4S является экспериментальным протоколом, поэтому представляется разумной возможность отключить его у отправителя при возникновении непреодолимых проблем, возможно, из-за какого-либо неожиданного взаимодействия с конкретным отправителем, на определённом пути через сеть или неудачи всего эксперимента.

A.1.4. Возврат к классическому контролю при потере пакетов

Описание

Помимо масштабируемой реакции на маркировку ECN расширяемый контроль перегрузок должен реагировать на потерю пакетов совместимым с контролем перегрузки Reno методом [RFC5681] (см. нормативные требования в п. 2 параграфа 4.3. Требования к откликам на перегрузку).

Мотивация

Часть условий безопасности при внедрении расширяемого контроля перегрузки в общедоступной сети Internet заключается в обеспечении надлежащего поведения при организации очереди в узком месте сети, обновляемом для поддержки L4S. Потери пакетов могут возникать по многим причинам, но обычно следует консервативно считать потери признаком перегрузки. Поэтому при обнаружении потери пакета расширяемому контролю перегрузки потребуется вернуться к поведению классического контроля перегрузки. Если этого не происходит, классический трафик может быть ограничен.

Расширяемых контроль перегрузки может применяться для разных видов транспорт, например при передаче в реальном масштабе времени или для гарантированного транспорта, такого как TCP. Поэтому конкретное классическое поведение контроля перегрузки, к которому следует возвращаться, зависит от используемой реализации контроля. В случае DCTCP спецификация протокола [RFC8257] говорит: «Отправитель DCTCP должен реагировать на случаи потерь так же, как традиционный TCP.» Для обеспечения безопасного внедрения расширяемого контроля перегрузок в общедоступной сети Internet п. 2 из параграфа 4.3 в этом документе не требует отклика, в точности совпадающего с Reno TCP, но требует отклика, обеспечивающего безопасное сосуществование с классическим контролем перегрузки, подобным Reno.

Даже при поддержке L4S в узком месте может возникать перегрузка с потерей пакетов. В таких случаях отправитель может получить большое число пакетов с кодом CE, а также столкнуться с потерями. Текущие реализации DCTCP реагируют на такие ситуации по-разному. Одним из подходов является реагирование лишь на сигнал отбрасывания (например, сокращением cwnd вдвое), в другом применяется реакция на оба сигнала с большим сокращением cwnd. Предложен компромисс между этими подходами, где реакция на потерю настраивается так, чтобы привести к сокращению окна вдвое при наличии предшествующего отклика ECN в том же интервале кругового обхода. Представляет разумным проведение дополнительных экспериментов для поиска поведения, наиболее подходящего для общедоступной сети Internet.

A.1.5. Сосуществование с классическим контролем в узких местах с Classic ECN

Описание

Мониторинг должен выполняться так, чтобы AQM с поддержкой ECN, но без L4S можно было обнаружить в узких местах на пути. Если такой механизм AQM реализован в общей очереди, любой долгосрочный расширяемый поток возобладает над долгосрочным классическим потоком в той же очереди. Нормативные требования п. 3 в параграфе 4.3 заданы так, чтобы такие проблемы решались в реальном масштабе времени или вмешательством администратора.

Мотивация

Подобно обсуждению в Приложении A.1.4, требование из параграфа 4.3 является условием безопасного сосуществования контроля перегрузок L4S с классическими потоками при создании очереди в общем узком месте сети, не обновлённом для поддержки L4S. Тем не менее, считается разумным при необходимости решать такие задачи в установленные сроки (возможно, с участием человека), поскольку:

- классический поток продолжает работать без истощения, хотя его пропускная способность может существенно снизиться из-за конкурирующего расширяемого потока;
- реализации Classic ECN AQM в очереди для совместного использования считаются редкими;
- обнаружение таких AQM не всегда однозначно, поэтому для уверенности нужно целенаправленное тестирование по отдельному каналу (out-of-band) или даже обращение в соответствующему оператору.

Поэтому соответствующие нормативные требования (параграф 4.3) разделены на 3 части (этапа).

Мониторинг

Мониторинг включает сбор измерительных данных для анализа. Мониторинг указан как обязательный (**должен**) для неконтролируемых сред, хотя место размещения функции мониторинга не указано. Для мониторинга в реальном масштабе времени указано требование «**следует**», что допускает возможность предпочтения оператором отправителя L4S (например, CDN[1]) проверки по отдельному каналу наличия признаков Classic ECN AQM, возможно, для предотвращения расхода ресурсов на мониторинг «живого» трафика.

Обнаружение

Обнаружение включает анализ данных мониторинга для определения вероятности наличия Classic ECN AQM. Обнаружение может непосредственно выявлять проблемы сосуществования потоков или нацеливаться на идентификацию технологий AQM, которые могут вызывать проблемы сосуществования, на основе сведений о развёрнутых AQM. Рекомендуется выполнять обнаружение в реальном масштабе времени, однако оно может быть и отложенным (например, может включать дополнительное тестирование, нацеленное не конкретные AQM).

Действия

Действия включают переход отправителя к классическому контролю перегрузки, который может выполняться в реальном масштабе времени в рамках контроля перегрузки или с участием администратора, переключающего контроль перегрузки на классический для конкретного интерфейса или набора адресатов.

Оператор отправителя(например, CDN) может вместо самого отправителя попросить оператора сети изменить обработку Classic AQM для пакетов L4S, обеспечивая им обход AQM, или обновить AQM для поддержки L4S (см. эксплуатационные рекомендации для L4S [L4SOPS]). Если потоки L4S больше не проходят через Classic ECN AQM, они уже не будут видеть эти механизмы и требование к действиям теряет актуальность.

Весь набор нормативных требования, относящихся к Classic ECN AQM, в параграфе 4.3 сформулирован так, что они не применяются в контролируемых средах, таких как частные сети и ЦОД. Серверы CDN, размещённые в сети доступа ISP, можно рассматривать как контролируемую среду, но все сети, обслуживаемые этой сетью доступа, включая сети клиентов, вряд ли будут входить в сферу того же координированного контроля. Для мониторинга таких неконтролируемых сегментов пути (например, в домашней сети за сетью доступа ISP) указан уровень «**должен**», поскольку для них имеется вероятность наличия общей очереди с Classic ECN AQM. Тем не менее, цель формулировки заключается в требовании лишь эпизодического мониторинга этих неконтролируемых участков сети без обременения операторов CDN, если мониторинг не выявляет каких-либо возможных проблем.

Более подробное рассмотрение всех вариантов можно найти в руководстве по эксплуатации L4S [L4SOPS].

С учётом отмеченного выше, подход, рекомендованный в параграфе 4.3, заключается в мониторинге, обнаружении и действиях по отношению к трафику в реальном масштабе времени. Алгоритм пассивного мониторинга для обнаружения ECN AQM в узких местах и возврата к классическому контролю перегрузок, описан в обширном техническом отчёте [ecn-fallback], где приведены также ссылки на исходный код Linux и сетевую визуализацию результатов оценок. Вкратце, алгоритм в основном отслеживает вариации RTT с использованием того же метода, который поддерживает среднее отклонение сглаженного RTT в TCP, но выполняет сглаживание в интервале того же порядка, что и при классической пиле. Результат зависит и от других показателей, таких как наличие маркировки CE и стабилизации фазы предотвращения перегрузки. В отчёте также указаны направления дальнейшей работы по улучшению подхода, например, для каналов с малой пропускной способностью или для сочетания измерений с кэшированием того, что было известно о пути из прежних соединений. В отчёте предложены и другие подходы.

Хотя использование пассивных инструментов для «живого» трафика (как указано выше) позволяет обнаружить Classic ECN AQM, гораздо сложнее (или невозможно) определить наличие AQM в общей очереди. Это гораздо проще сделать при активном тестировании вне основного канала, поскольку можно применить 2 потока. В разделе 4 отчёта [ecn-fallback] описан простой метод обнаружения Classic ECN AQM и проверки его размещения в общей очереди, который кратко рассмотрен ниже.

Тестовый сервер с поддержкой L4S можно настроить так, что при обращении к нему тестового клиента будет выполняться сценарий, позволяющий клиенту создать 2 параллельных долгосрочных потока. Один поток может работать с классическим контролем перегрузки (C, с установкой ECT(0)), другой - с расширяемым (L, с установкой ECT(1)). Если ни один из потоков не вызывает маркировки ECN, можно предположить, что на пути нет Classic ECN AQM. Если в каком-либо из потоков возникает маркировка ECN, сервер может измерить относительные скорости потоков и время кругового обхода для них. В таблице 2 показаны механизмы AQM, которые можно предположить в разных случаях (в предположении отсутствия других вариантов поведения AQM, кроме известных при написании).

Таблица 2. Тестирование по отдельному каналу с 2 параллельными потоками.

Скорость	***RTT***	***Предполагаемый AQM***
L > C	L = C	Classic ECN AQM (FIFO)
L = C	L = C	Classic ECN AQM (FQ)
L = C	L < C	FQ-L4S AQM
L ~= C	L < C	DualQ Coupled AQM

L = L4S, C = Classic

Ещё одним мотивом рекомендаций из параграфа 4.3 является то, что в расширяемом контроле перегрузки не предполагается смена на установку ECT(0) при изменении поведения для сосуществования с классическими потоками.

[1]Content Distribution Network - сеть распространения содержимого.

Это обусловлено тем, что отправитель должен продолжать проверку правильности решения о переходе и обратного переключения в случае ошибки или переносе узкого места на другой канал.

- Если отправитель в соответствии с рекомендациями меняет лишь своё поведение на классическое, но не меняет код, его код остаётся совместимым с L4S или Classic AQM. Если в узком месте фактически поддерживаются оба режима, код ECT(1) будет по-прежнему классифицироваться в ту же очередь L4S и отправитель может понять, что переход к классическому поведению был ошибкой и вернуться назад.

- Если же отправитель меняет поведение и код даже при поддержке в узком месте обоих режимов, он будет классифицировать ECT(0) в классическую очередь, форсируя некорректное решения без возможности возврата.

- Кроме того, отказ от смены кода позволяет исключить риск перехода на другой путь через балансировщик нагрузки или маршрутизацию по нескольким путям, в которой хэшируется весь байт прежнего типа обслуживания (Type-of-Service или ToS), что к сожалению ещё является распространённой патологией.

  Отметим, что при настройке потока на применение лишь классического контроля перегрузок вполне уместно не использовать код ECT(1).

A.1.6. Снижение зависимости от RTT

Описание

Расширяемому контролю перегрузки нужно максимально сокращать смещение (bias) RTT по меньшей мере в диапазоне от малых до типичных значений RTT, которые будут взаимодействовать в предполагаемом сценарии внедрения (см. нормативные требования в п. 4 параграфа 4.3. Требования к откликам на перегрузку).

Мотивация

Известно, что пропускная способность при классическом контроле перегрузки обратно пропорциональна RTT, поэтому можно предполагать, что потоки на путях с очень малым RTT будут истощать потоки путей с большими RTT. Однако классический контроль перегрузки не допускает наличия очень малых RTT, поскольку они вызывают большие очереди. Рассмотрим, например, два пути с базовыми RTT в 1 и 100 мсек. Если классический контроль перегрузки создаёт очередь с задержкой 100 мсек, эти значения RTT превратятся в 101 и 200 мсек, что ведёт к отношению пропускной способности потоков примерно 2:1. Если же классический контроль перегрузок вносит задержку в очереди лишь на 1 мсек, отношение будет 2:101, а для пропускной способности около 50:1.

Поэтому при очень коротких очередях потоки с большим RTT будут истощаться, если расширяемый контроль перегрузки не будет выполнять требования параграфа 4.3.

При RTT больше типичных потоки L4S могут применять такое же смещение RTT как в текущем классическом контроле перегрузок и работать по-прежнему удовлетворительно. Поэтому в параграфе 4.3 нет дополнительных требований к потокам L4S с большим RTT в части смещения RTT.

Одним из способов выполнения этих требований в масштабируемом контроле перегрузки является поведение, при котором аддитивный рост происходит как для стандартного потока Reno, но с большим RTT, путём использования виртуального значения RTT (rtt_virt), являющегося функцией от фактического RTT (rtt). Примеры функций даны ниже.

```
rtt_virt = max(rtt, 25 ms)
rtt_virt = rtt + 10 ms
```

Функции в этих примерах выбраны так, что по мере снижения фактического RTT от высокого к низкому виртуальное значение RTT сокращается более медленно (см. [PRAGUE-CC]).

Однако потоки с коротким RTT могут быстрее реагировать на изменения доступной пропускной способности, будь то изменение состава потоков или пропускной способности радиоканалов. Таким образом, было бы ошибкой требовать, чтобы потоки с малым RTT были столь же медлительными, как потоки с большим RTT, поскольку это привело бы к ненужно недогрузке каналов и в результате вызывало бы нестабильность. Поэтому вместо строго требования независимости от RTT в п. 4 параграфа 4.3 сказано: «не зависит от RTT, насколько это возможно без ущерба для стабильности и загрузки канала.» Это позволяет потокам с меньшим RTT использовать свои преимущества в гибкости.

A.1.7. Сокращение окна перегрузки

Описание

Расширяемый контроль перегрузки должен сохранять свою реакцию на перегрузки, когда типичные значения RTT через общедоступную сеть Internet становятся существенно меньше в результате меньшей задержки в очередях (см. нормативные требования в п. 5 параграфа 4.3. Требования к откликам на перегрузку).

Мотивация

В настоящее время предполагается, что минимальное окно перегрузки для TCP (и производных протоколов) с поддержкой ECN составляет 2 максимальных размера сегмента у отправителя (sender maximum segment size или SMSS) или 1 SMSS после тайм-аута повторной передачи. Когда окно перегрузки достигает этого минимума, при наличии дополнительной маркировки ECN протокол TCP должен дождаться тайм-аута повторной передачи, прежде чем отправить новый сегмент,(см. Параграф 6.1.2 в спецификации ECN [RFC3168]). На практике большинство известных алгоритмов контроля перегрузки на основе окна в этот момент перестают реагировать на сигналы ECN и окно перегрузки больше не сокращается независимо от степени маркировки ECN. Отсутствие дополнительной реакции отправителя на перегрузку ведёт к росту очереди, переопределяя любой механизм AQM, и дополнительной задержке в очереди (делая окно достаточно большим для возвращения реакции на перегрузку). Это может приводить к стабильной, но более длинной очереди или вызывать потери в очереди, когда механизм тайм-аута повторной передачи выступает заслоном.

В большинстве элементов контроля перегрузки на основе окна для других протоколов используется похожее минимальное окно, измеряемое в байтах для протоколов, использующих более мелкие пакеты.

Механизмы L4S значительно сокращают задержку в очередях, поэтому на одном и том же пути RTT снижается. Эта проблема становится на удивление распространённой [sub-mss-prob], потому что при одинаковой пропускной способности канала меньшее значение RTT подразумевает меньшее окно. Рассмотрим, например, жилое здание с восходящим каналом широкополосного доступа в Internet со скоростью 8 Мбит/с, предполагая максимальный размер сегмента 1500 байт. Каждый из 2 восходящих потоков будет иметь минимальное окно 2 SMSS, если RTT не больше 6

мсек, что достаточно характерно при доступе к ближайшему ЦОД. При наличии 2 таких потоков TCP перестанет отвечать на ECN и задержка в очередях возрастёт.

Пока расширяемый контроль перегрузки не выполняет с самого начала требования параграфа 4.3, он будет часто показывать невосприимчивость к ECN, сводя на нет преимущества L4S в части малой задержки для себя и других.

Это, по-видимому, подразумевает, что будут требоваться средства расширяемого контроля перегрузки, способные работать с окном перегрузки меньше 1 SMSS. Например, если TCP с поддержкой ECN получает маркер ECN, уже имея окно в 1 SMSS, [RFC3168], он должен отложить передачу до тайм-аута повтора. Менее радикальный, но более сложный механизм может поддерживать окно перегрузки меньше 1 SMSS (при необходимости, значительно меньше), как описано в [Ahmed19]. Вероятно, возможны и другие подходы.

Однако в требованиях параграфа 4.3 указано «**следует**», поскольку наличие минимального окна не представляется плохим. При конкуренции с неотзывчивым потоком такое окно обеспечивает естественную защиту потока от истощения, сохраняя некоторый поток данных.

Указание для перехода к окну меньше 1 SMSS уровня «**следует**» при сохранении требований [RFC3168] позволяет отследить варианты изменения минимального окна в разных масштабируемых контроллерах перегрузки.

A.1.8. Измерение устойчивости к нарушению порядка по времени

Описание

При обнаружении потерь масштабируемый контроль перегрузки должен быть устойчив к изменению порядка в адаптивном интервале времени, который меняется в зависимости от пропускной способности, а не только для фиксированного числа пакетов (см. нормативные требования в п. 6 параграфа 4.3. Требования к откликам на перегрузку).

Мотивация

Основной целью L4S является масштабируемая пропускная способность (отсюда название). Расширяемость во всех измерениях является целью всех технологий IETF. Обратная линейная зависимость реакции (параграф 4.3) необходима, но не достаточна для решения задачи расширяемости контроля перегрузки, указанной в [RFC3649]. Помимо роста частоты сигналов ECN с повышением скорости важно обеспечить отсутствие ограничений на расширение пропускной способности из-за ложного восприятия потерь.

Конечные системы не могут определить причину пропуска пакета - потеря или нарушение порядка. Они могут лишь считать, что произошла потеря, если пропуск в порядковых номерах не был заполнен после поступления определённого числа пакетов (например, правило 3 DupACK стандартного контроля перегрузки TCP [RFC5681]) или по истечении некоторого времени (например, RACK [RFC8985]).

Многолетний рост скорости передачи пакетов, привёл к указанным ниже наблюдениям.

- Даже когда лишь некоторые передающие узлы считают, что произошла потеря на основании подсчёта переупорядоченных пакетов, все сети будут сокращать время, в течение которого они сохраняют порядок пакетов. В некоторых канальных технологиях сохранение примерно неизменным времени, в течение которого изменяется порядок, ведёт с годами к росту числа потерь на этих каналах, предполагаемых хостами.
- Если же все хосты фиксируют потери по времени, интервал времени, в течение которого сеть сохраняет порядок, остаётся практически постоянным.

Поэтому у хостов есть мотивы для обнаружения потерь в интервале времени (чтобы не обманывать себя потерями, которых нет). Для хостов, переходящих на контроль перегрузок L4S, не возникает никакого ущерба при включении кода обнаружения потерь по времени (аппаратное восстановление потерь является исключением, которое будет рассмотрено позже). Поэтому требование к хостам L4S выполнять обнаружение потерь по времени не является обременительным.

Если не применять требования параграфа 4.3 к хостам L4S, даже при отсутствии обременения для хостов все сети столкнулись бы с неопределённостью в части способности некоторых хостов L4S обнаруживать потери по времени. Тогда всем канальным технологиям пришлось бы сокращать время, в течение которого допускается переупорядочение пакетов. Это не является проблемой для некоторых канальных технологий, но для других все сильнее осложняется масштабирование, особенно при использовании группировки каналов (bonding) в таких технологиях как LTE, 5G, DOCSIS[1].

С учётом разных технологий на путях Internet любые ограничения расширяемости для канальных технологий в сетях доступа приведут к ограничению масштабируемости для Internet в целом.

Может возникнуть вопрос осмысленности реализации этого требования к обнаружению потерь лишь на хостах L4S, поскольку сети всё равно будут сталкивать с неопределённостью обнаружения потерь с помощью DupACK для потоков без поддержки L4S. Ответ заключается в том, что канальные технологии, где сложно постоянно сокращать интервал переупорядочивания, должны будут решать вопрос лишь для трафика, не относящегося к L4S (это возможно, поскольку идентификатор L4S виден на уровне IP). Поэтому можно сосредоточить внимание на ресурсах обработки и памяти при расширении классического (не L4S) трафика. Тогда рост доли трафика L4S будет смягчать проблемы масштабирования.

В итоге для хостов L4S не никаких причин быть частью проблемы, а не её решения.

Должно или **следует**? Как отмечено выше, этот тонкая проблема взаимодействия между хостами и сетями, которая **должна** быть решена. Если сети не могут быть уверенными, что все хосты L4S перешли к обнаружению потерь по времени, им приходиться принимать худший вариант, постоянно сокращая интервал переупорядочивания, для хостов, использующих подсчёт. Однако было принято решение использовать уровень «**следует**». Основной причиной послужило то, что сети могут быть достаточно уверены в выполнении требования хостами L4S, поскольку это даёт преимущества и не вызывает проблем.

Детали

Время, затрачиваемое на восстановление потерь, более значимо для краткосрочных потоков, поэтому хорошим компромиссом служит изменение окна переупорядочения от небольшой доли RTT в начале потока до большей части RTT по истечении множества интервалов кругового обхода.

[1] Data Over Cable Service Interface Specification - спецификация передачи данных через интерфейс кабельных сетей (ТВ).

Этот подход в основном принят в RACK [RFC8985]. Однако работа RACK начинается с подхода 3 DupACK, поскольку оценка RTT может быть нестабильной. Пока начальное окно успевает за изменениями, подсчёт 3 DupACK даёт такой же результат, что и обнаружение потерь по времени, поэтому соответствует рекомендациям параграфа 4.3. Это связано с тем, что начальное окно гарантирует, что подсчёт 3 DupACK в начале соединения распространяется на небольшую часть интервала кругового обхода.

Как отмечено выше, имеются аппаратные реализации восстановления потерь с использованием подсчёта DupACK (например, некоторые реализации RoCEv2[1]). При малой задержке эти реализации могут сменить свой контроль перегрузок на L4S, поскольку тот (в отличие от восстановления потерь) реализуется программно, но не удаётся выполнить все требования восстановления потерь. Однако считается, что в этом нет необходимости, поскольку сетевые технологии обеспечивают крайне низкую степень нарушения порядка. Поэтому контролируемые среды, где порядок практически не меняется, исключены из нормативных рекомендаций параграфа 4.3.

Обнаружение потерь по времени также предотвращает атаки ACK-splitting, описанные в [Savage-TCP].

A.2. Оптимизация расширяемого транспортного протокола

A.2.1. Установка ECT в пакетах управления и повторных пакетах

Описание

Этот параграф касается TCP и производных от него протокол (например, SCTP), а также RTP/RTCP [RFC6679]. Исходная спецификация ECN для TCP использование ECN для пакетов управления и повторной передачи, а [RFC6679] исключает применение ECT в дейтаграммах RTCP в случаях изменения пути после его проверки на прохождение ECN. Для повышения производительности масштабируемые транспортные протоколы должны разрешать ECN на уровне IP в пакетах управления TCP (SYN, SYN-ACK, ACK и т. п.) и повторно передаваемых пакетах. Это справедливо и для другого транспорта, например, SCTP и RTCP.

Мотивация для TCP

[RFC3168] запрещает применять ECN в указанных типах пакетов TCP по многим причинам. Это означает, что такие пакеты не защищены ECN от потерь при перегрузке, что существенно снижает производительность, особенно для коротких потоков. В ECN++ [ECN++] предложен эксперимент по использованию ECN со всеми типами пакетов TCP, пока доступны отклики AccECN [ACCECN] (что соответствует требованиям параграфа 4.2 для расширяемого контроля перегрузок).

Мотивация для RTCP

В экспериментах L4S нужно в общем случае соблюдать правило спецификации RTP ECN [RFC6679], исключающее ECT в дейтаграммах RTCP. Тем не менее, применение ECN расширяется и будет полезно выполнить некоторые эксперименты для RTCP с поддержкой ECN, чтобы собрать данные о нужности такой предосторожности.

A.2.2. Рост быстрее аддитивного

Описание

Производительность возросла бы, если бы расширяемый контроль перегрузки не ограничивал расширение окна перегрузки стандартным аддитивным увеличением на 1 SMSS за интервал кругового обхода [RFC5681] во время предотвращения перегрузки. Это верно для контроля перегрузки в TCP и производных протоколах, включая аналогичные подходы, применяемые в реальном масштабе времени.

Мотивация

В соответствии с пекущим определением [RFC8257] протокол DCTCP использует обычное аддитивное увеличение Reno в фазе предотвращения перегрузки. Когда доступная пропускная способность увеличивается внезапно (например, при завершении другого потока или увеличении скорости радио-канала), может потребоваться очень много интервалов кругового подхода, чтобы воспользоваться вновь доступной пропускной способностью. Для решения проблемы был разработан протокол TCP CUBIC [RFC8312], но в результате продолжающегося роста скоростей потоков задержка при разгоне до доступной пропускной способностью снова стала чрезмерной (см., например параграф 5.1 в описании архитектуры L4S [RFC9330]). Даже при выходе из режима совместимости с Reno на каждое 8-кратное увеличение скорости потока CUBIC задержка ускорения вырастает вдвое.

В установившемся состоянии DCTCP устанавливает примерно 2 маркера ECN за интервал кругового обхода, поэтому можно быстро определить, когда эти сигналы исчезли и искать доступную пропускную способность быстрее с минимальным влиянием на другие потоки (классические и расширяемые) [LinuxPacedChirping]. В качестве варианта решения проблемы предложен протокол TCP для ЦОД с адаптивным ускорением (Adaptive-Acceleration Data Center TCP или A2DTCP) [A2DTCP]), который можно внедрить в общедоступной сети Internet.

A.2.3. Быстрое схождение при замедленном старте

Описание

Производительность возросла бы, если бы расширяемый контроль перегрузки сходился (к установившемуся состоянию раздела пропускной способности) быстрее (или, хотя бы не медленней), чем классический контроль перегрузки. Это влияет на поведение при запуске потока во всех средствах контроля перегрузки L4S, выведенных из классического транспорта, использующего процедуру замедленного старта TCP (slow start), включая работающие в реальном масштабе времени средства.

Мотивация

Новому потоку DCTCP требуется для получения своей доли пропускной способности в узком месте с наличием других потоков, использующих пропускную способность, больше времени, чем при классическом контроле перегрузки. В среде ЦОД время схождения для DCTCP больше в 1,5 - 2 раза из-за существенно большего уровня маркировки ECN, вызываемой фоновым трафиком DCTCP, что вынуждает новые потоки раньше выходить из режима замедленного старта [Alizadeh-stability]. В тестах для применения в общедоступной сети Internet время схождения DCTCP по сравнению с обычным TCP slow start на основе потери пакетов оказалось ещё хуже [LinuxPacedChirping] из-за низкого порога маркировки ECN, требуемого для L4S. Это усугубляется обычно большим несоответствием скорости канала передающего хоста и типичного узкого места доступа в Internet. Это пагубно в общем случае и особенно сильно влияет на производительность коротких потоков по сравнению с классическим контролем перегрузок.

[1]Remote Direct Memory Access over Converged Ethernet version 2 - удалённый доступ к памяти через Converged Ethernet версии 2

Приложение B. Компромиссы при выборе идентификатора L4S

Это приложение является информационным, а не нормативным. Как указано в разделе 3, пространства заголовков IP (v4 и v6) не достаточно для полного выполнения требования. Поэтому при выборе идентификатора L4S потребовались компромиссы. В этом приложении приведены аргументы за и против сделанного выбора.

Ниже приведён ненормативный обзор схемы кодов.

Пакеты с кодом ECT(1) и (условно) CE обозначают семантику L4S как альтернативу семантике Classic ECN [RFC3168].

- Код ECT(1) указывает, что пакет передан отправителем, поддерживающим L4S.
- С учётом нехватки кодов L4S AQM и Classic ECN AQM используют общий код CE для индикации встреченной пакетом перегрузки. Если пакета уже получил маркер CE в буфере восходящего направления до поступления в следующий механизм AQM, этот механизм должен угадать для пакета CE классификацию L4S или Classic ECN. Выбор обработки L4S безопасней, поскольку в этом случае несколько классических пакетов могут прибыть раньше, а пакеты L4S не будут задержаны.
- Если классификатор знает о транспорте, может быть доступна дополнительная информация. Тогда классификатор может отнести пакет CE к ECN, если предшествующий пакет ECT из того же потока имел код ECT(0). Однако службе L4S не требуется знать о транспортном уровне.

Возражения

Используется последний код ECN

Служба L4S потенциально может заменить услуги, предоставляемые Classic ECN, поэтому использование ECT(1) для идентификации L4S может в конечном итоге означать, что код ECT(0) потрачен «впустую» лишь для того, чтобы различать две формы ECN.

Сложности ECN на некоторых нижележащих уровнях

Не всегда возможна поддержка эквивалента поля IP-ECN в AQM, работающих в буфере ниже уровня IP [ECN-ENCAP]. В зависимости от схемы нижележащего уровня службе L4S может потребоваться отбрасывать кадр вместо маркировки, даже если она может инкапсулировать пакет с поддержкой ECN.

Риск нарушения порядка классических пакетов CE внутри потока

Отнесение всех пакетов CE к очереди L4S создаёт риск ошибочной классификации пакетов CE, изначально имевших код ECT(0), к L4S. Если в классической очереди имеется задержка, эти некорректно классифицированные пакеты CE придут раньше, вызывая нарушение порядка. Переупорядочение внутри микропотока может вынудить отправителя TCP (и аналогичного транспорта) к ненужным повторам передачи. Однако этот риск чрезвычайно мал в силу указанных ниже причин.

1. Достаточно необычно наличие нескольких узких местам на одном пути (доступная пропускная способность должна быть для этого идентична).
2. Лишь в части таких случаев первое узкое место будет поддерживать маркировку Classic ECN, а второе - L4S ECN. Это будет единственным случаем, где пакеты ECT(0) могут получить маркер CE от AQM, поддерживающего Classic ECN, тогда как далее оставшиеся пакеты ECT(0) будут сталкиваться с дальнейшей задержкой на классической части последующего L4S DualQ AQM.
3. Даже при досрочной доставке нескольких пакетов потребуются весьма необычные условия для вызова ложных повторов в отличие от случаев запоздания некоторых пакетов. В первом узком месте маркеры CE будут применены по меньшей мере для N (размер окна переупорядочивания, разрешаемого транспортным протоколом, прежде, чем он сочтёт пакет потерянным) последовательных пакетов, а второе узкое место внедрит непрерывную последовательность и (по меньше мере) N таких пакетов между двумя пакетами потока, переданными ранее.

   Рассмотрим, например, случай N=3 и последовательность пакетов 100, 101, 102, 103, ... Предположим, что пакеты 150, 151, 152, переданные в потоке позднее, были внедрены в последовательность 100, 150, 151, 101, 152, 102, 103, ... Если бы это было поздним нарушением порядка, даже 1 пакет с нарушением порядка вызвал бы ложный повтор, но здесь этого не происходит, поскольку пакет 101 перемещает счётчик кумулятивных ACK вперёд на 3 пакета, прибывших с нарушением порядка. Позднее, по приходе пакетов 148, 149, 153, … проблем не будет, несмотря на пропуск 3 пакетов, поскольку пакеты для заполнения пропуска уже находятся в приёмном буфере.
4. Даже при текущей рекомендации TCP N=3 [RFC5681] ложные повторы будут маловероятны по указанным выше причинам. По мере внедрения RACK [RFC8985] возникает тенденция к увеличению размера окна разупорядочения (N), что сделает вероятность преждевременного прибытия N пакетов подряд исчезающе малой.
5. Наличие двух маркеров CE в потоке Classic ECN маловероятно с учётом того, что FQ-CoDel является единственным широко развёрнутым AQM с поддержкой маркировки Classic ECN и он очень тщательно разделяет потоки и равномерно распределяет маркировку между потоками.

Крайне маловероятно, что указанные выше 5 весьма необычных ситуаций возникнут одновременно. Но даже в этом случае последствия вряд ли будут ужасными - неоправданный быстрый повтор передачи. Всякий раз, когда источник трафика (классический контроль перегрузки) ошибочно считает нарушение порядка пакетов CE потерей, можно полагать, что он уменьшит своё окно перегрузки и без необходимости отправит пакет повторно. Однако окно перегрузки уже было бы сокращено при получении маркеров CE. Если отправитель использует ABE [RFC8511], он может сократить cwnd немного сильнее при потере, чем при маркировке CE. Но он вернёт сокращение, как только обнаружит, что повтор передачи был ошибочным.

Отметим, в заключение, что влияние раннего нарушения порядка на ненужные повторы передачи из-за неоднозначности CE, как правило, будет исчезающе малым.

Недостаточное окно anti-replay в некоторых имеющихся VPN

Если задержка сокращается для части потоков внутри VPN, функция anti-replay в некоторых VPN может ошибочно считать задержку replay-атакой. В параграфе 6.2 рекомендуется устанавливать достаточно большое окно anti-replay на выходе VPN в соответствии с требованиями спецификации. Однако в некоторых реализациях VPN максимальный размер окна недостаточно велик для учёта большой разницы в задержке при преобладающих скоростях передачи. В параграфе 6.2 предложены обходные пути, но конечным пользователям L4S через VPN нужно уметь распознавать эти проблемы, чтобы воспользоваться обходными путями.

Трудно отличить Classic ECN AQM

В этой схеме источнику, получившему отклик ECN, не вполне ясно, к какому типу AQM относится маркер CE. Это не является проблемой для источников Classic ECN, которые передают пакеты ECT(0), поскольку L4S AQM считает пакеты ECT(0) классическими и применяет для них маркировку Classic ECN.

Однако без явного устранения неоднозначности маркировки CE источникам L4S приходится применять эвристические методы для выбора реакции на перегрузку (см. Приложение A.1.5). В противном случае при прохождении долгосрочных классических потоков через узкое место с Classic ECN AQM вместе с долгосрочными потоками L4S и применением для потоков L4S отклика L4S на сигналы Classic CE, потоки L4S будут превосходить классические. Эксперименты показали, что потоки L4S могут занять примерно в 20 раз больше общей пропускной способности, чем эквивалентные классические потоки. При сокращении пропускной способности канала (например, до 4 Мбит/с) неравенство уменьшается и классические потоки продолжаются без истощения.

Когда L4S предложили впервые (2015 г., на 14 позже публикации Classic ECN [RFC3168]), считалось, что классические Classic ECN AQM не были внедрены из-за того, что измерения при исследовании не выявили практически никаких признаков маркировки CE. Позднее Classic ECN включили в развёртывание FQ, однако планировщики FQ тормозят потоки L4S, конкурирующие с классическими, поскольку они обеспечивают равенство скоростей потоков. Неизвестно, были ли не связанные с FQ внедрения Classic ECN AQM в последующие годы и будет ли это впредь.

Был предложен алгоритм обнаружения Classic ECN AQM при стабилизации потока после его запуска [ecn-fallback] (см. Приложение A.1.5). Тесты для второй версии алгоритма показали достаточно хорошее обнаружение Classic ECN AQM в разных обстоятельствах, однако алгоритм часто работал некорректно при малой пропускной способности канала и/или больших RTT. Хотя этот обход является безопасным, имеется риск того, что он будет препятствовать применению алгоритма.

Услуги без L4S для пакетов управления

Исключительно для TCP в RFC Classic ECN [RFC3168] и [RFC5562] требуют очистки отправителем поля IP-ECN (Not-ECT) при повторной передаче и в некоторых пакетах управления (в частности, «чистых» ACK, пробы окна, SYN). При классификации пакетов L4S по полю IP-ECN эти пакеты управления TCP не будут отнесены к очереди L4S и в результате могут быть задержаны относительно других пакетов потока. Это не вызовет нарушение порядка (повторные передачи уже нарушают порядок, а пакеты управления обычно не содержат данных), однако повысит уязвимость пакетов управления TCP к потерям и задержке. Для решения проблемы в ECN++ [ECN++] предложен эксперимент по поддержке ECN для всех пакетов управления и повторов TCP, пока доступны отклики ECN.

Поддержка

Нужна сквозная работа

Поле IP-ECN обычно распространяется через Internet без удаления и искажения, по крайней мере в стационарных сетях. В отличие от DSCP, поле ECN пересылается без изменений сетями, не поддерживающими ECN.

Нужна работа в туннелях

Идентификаторы L4S работают через туннели (и в них), передающие поле IP-ECN любым из возможных способов, заданных с момента спецификации туннелирования ECN в 2001 г. [RFC3168]. Однако вполне возможно, что некоторые туннели до сих пор не реализуют распространения ECN.

Нужна работа с разными канальными технологиями

В большинстве (но не во всех) узких местах пути известен протокол IP и L4S AQM могут размещаться там, где возможны манипуляции с полем IP-ECN. Узкие места на узлах нижнего уровня, не знающих IP, будут отбрасывать пакеты для сигнализации перегрузки или применять свои уведомления, заданные канальной технологией, включая обмен сигналами с полем IP-ECN. Программы по определению таких сигналов разрабатываются и схемы, уже определённые для ECN, наследуют и поддержку L4S (6.1. Неизменность в туннелях и при инкапсуляции).

Возможен переход к одному коду

Если все отправители Classic ECN в конечном итоге перейдут на услуги L4S, код ECT(0) можно будет отдать на другие цели, когда прежнее использование ECT(0) прекратится или сильно сократится (возможно, этого не будет).

L4 не требуется

Будучи основанной на поле IP-ECN, эта схема не нуждается в доступе к идентификаторам потоков транспортного уровня. Тем не менее, она не исключает таких решений.

Приложение C. Возможные конфликты при использовании кода ECT(1)

Код ECT(1) в поле IP-ECN уже был выделен спецификацией ECN nonce [RFC3540], которая признана устаревшей (Historic) [RFC8311]. ECN вероятно является единственным среди протоколов Internet полем, общим для IPv4 и IPv6, сохраняя возможность сквозного применения с туннелями и нижележащими уровнями. Поэтому код ECT(1) не следует переназначать для другого экспериментального использования (L4S) без тщательной оценки возможных конфликтов. Эти вопросы рассматриваются ниже.

C.1. Целостность откликов на перегрузку

Принимающие хосты могут обманывать отправителя для ускорения загрузки, подавляя отклики в маркерах ECN (или потерях, если повтор передачи не требуется или доступен иным способом).

Устаревшая спецификация ECN nonce [RFC3540] предполагала, что отправитель TCP может устанавливать до ECT(0) или ECT(1) в каждом пакете потока и запоминать последовательность установки. Если какой-либо пакет будет потерян или промаркирован при перегрузке, получатель не увидит соответствующий бит. Получатель ECN nonce отправляет назад младший бит суммы, поэтому он не может скрыть потерю или маркировку без 50% вероятности ошибки в сумме.

Крайне маловероятно, что ECT(1) потребуется в качестве nonce для защиты целостности уведомлений. Спецификация ECN nonce [RFC3540] признана устаревшей отчасти из-за появления других способов (без использования кода в заголовке IP) защиты целостности откликов TCP и другого транспорта [RFC8311]. Ниже дано несколько примеров.

- Отправитель может проверить целостность небольшой случайной выборки откликов получателя, время от времени указывая в поле IP-ECN значение, которое обычно устанавливает сеть. Затем он проверяет, верно ли отклики получателя передают ожидаемое (см. п. 2 в параграфе 20.2 спецификации ECN [RFC3168]). Это работает для потерь и подходит для точных откликов ECN [RFC7560], предназначенных для L4S. Как и (устаревшая) спецификация ECN, этот метод не защищает от некорректного поведение отправителя, но позволяет добросовестному отправителю проверить корректность откликов о перегрузке от получателя.

- Сеть может проверить корректность передачи маркеров ECN (или потери пакетов) по всему контуру обратной связи, отслеживая раскрытие перегрузки (Congestion Exposure или ConEx) [RFC7713]. Это предполагает целостность уведомлений о перегрузке и откликов обратной связи. Сведения ConEx доступны в любой точке пути через сеть, поэтому они могут применяться для принудительных откликов на перегрузку. Независимо от подавления получателем или нисходящим направлением в сети откликов о перегрузке или некорректной реакции отправителя на них, ConEx позволяет нейтрализовать любые преимущества, которые можно получить за счёт такого обмана.

- Поля откликов о перегрузке в заголовках транспортного уровня передаются без изменений насквозь и их целостность может быть защищена. Это сохраняет целостность откликов получателя, но не защищает от некорректного поведения получателя или отправителя. Опция аутентификации TCP (TCP Authentication Option или TCP-AO) [RFC5925], сквозная защита QUIC [RFC9001] или защита целостности IPsec [RFC4303] позволяют обнаружить вмешательство в отклики о перегрузке (случайные или враждебные) в TCP, QUIC и ином транспорте. TCP-AO по умолчанию учитывает основной заголовок TCP и опции TCP, но часто не обеспечивает достаточной прочности для использования на многих сквозных путях, где промежуточные устройства могут приводить к отрицательному результату проверки из-за их попыток улучшить производительность или безопасность (например, путём повторной сегментации или сдвига пространства порядковых номеров).

На момент написания этого документа защита целостности транспортных откликов с использованием QUIC становилась обычным делом. Однако по-прежнему не принято защищать целостность более широкого контура обратной связи на основе потерь или Classic ECN. Если это положение измениться в процессе экспериментов L4S, может потребоваться разработка одного или нескольких методов из числа указанных выше.

C.2. Уведомление о менее значимой по сравнению с CE перегрузкой

Исследователи предлагали использовать код ECT(1) в качестве уведомления о менее серьёзной перегрузке, нежели CE, в частности, для того, чтобы потоки могли быстрее заполнить доступную пропускную способность после периода бездействия, при завершении или запуске других потоков, например, как в VCP[1] [VCP] и Queue View (QV) [QV].

Перед назначением ECT(1) на роль идентификатора L4S нужно было решить, не лучше ли сохранить код для будущей стандартизации быстрого ускорения потоков, которая считается важной и долговременной проблемой [RFC6077].

Уведомление до перегрузки (Pre-Congestion Notification или PCN) является ещё одним вариантом альтернативной семантики IP-ECN. Здесь ECT(1) служит для уведомления о менее серьёзной перегрузке, чем CE [RFC6660], однако поле IP-ECN принимает семантику PCN лишь для пакетов с кодом Diffserv, заданным для указания маркировки PCN в контролируемой среде. PCN требуется применять лишь к внешнему заголовку туннеля через контролируемую область, чтобы не было конфликтов с иным сквозным применением поля ECN. Поэтому регион PCN на пути не помешает идентификатору L4S, заданному в разделе 2.

Благодарности

Спасибо Richard Scheffenegger, John Leslie, David Täht, Jonathan Morton, Gorry Fairhurst, Michael Welzl, Mikael Abrahamsson, Andrew McGregor за обсуждения, которые привели к созданию этой спецификации. Ing-jyh (Inton) Tsang был одним из авторов предварительных версий документа. Спасибо Mikael Abrahamsson, Lloyd Wood, Nicolas Kuhn, Greg White, Tom Henderson, David Black, Gorry Fairhurst, Brian Carpenter, Jake Holland, Rod Grimes, Richard Scheffenegger, Sebastian Moeller, Neal Cardwell, Praveen Balasubramanian, Reza Marandian Hagh, Pete Heist, Stuart Cheshire, Vidhi Goel, Mirja Kühlewind, Ermin Sakic, Martin Duke за помощь и рецензирование документа. Спасибо thanks Ingemar Johansson за рецензирование и предоставление содержательного текста. Спасибо также рецензентам направления: Valery Smyslov, Maria Ines Robles, Bernard Aboba, Lars Eggert, Roman Danyliw, Éric Vyncke. Спасибо Sebastian Moeller за выявление взаимодействия м VPN anti-replay и Jonathan Morton за выявление связанных с этим атак. Отдельная благодарность руководителям tsvwg Gorry Fairhurst, David Black, Wes Eddy за терпеливую помощь для этого и других документов L4S при прохождении процесса IETF. Приложение A с требованием Prague L4S основано на тексте, написанном Marcelo Bagnulo Braun изначально в качестве приложения к [RFC9330]. Этот текст, в свою очередь, был основан на коллективных результатах участников «круглого стола» DCTCP Evolution в рамках IETF 94 [TCPPrague].

Работа авторов частично финансировалась European Community по программе Seventh Framework в рамках проекта Reducing Internet Transport Latency (RITE) (ICT-317700). Работа Koen De Schepper частично финансировалась в рамках проектов 5Growth и DAEMON EU H2020. Работу Bob Briscoe частично финансировалась Research Council of Norway в рамках проекта TimeIn, а также CableLabs и Comcast Innovation Fund. Представленные здесь мнения принадлежат исключительно авторам.

Адреса авторов

Koen De Schepper
Nokia Bell Labs
Antwerp
Belgium
Email: koen.de_schepper@nokia.com
URI: https://www.bell-labs.com/about/researcher-profiles/koende_schepper/

Bob Briscoe (editor)
Independent
United Kingdom
Email: ietf@bobbriscoe.net
URI: https://bobbriscoe.net/

Перевод на русский язык

Николай Малых (nmalykh@protokols.ru)

[1] Variable-structure congestion Control Protocol - протокол контроля перегрузки с переменной структурой.