Network Working Group
Request for Comments: 3247
Category: Informational

A. Charny
Cisco Systems, Inc.
J.C.R. Bennett
Motorola
K. Benson
Tellabs
J.Y. Le Boudec
EPFL
A. Chiu
Celion Networks
W. Courtney
TRW
S. Davari
PMC-Sierra
V. Firoiu
Nortel Networks
C. Kalmanek
AT&T Research
K.K. Ramakrishnan
TeraOptic Networks

March 2002

Дополнение к новому определению режима ускоренной пересылки EF PHB

Supplemental Information for the New Definition of the EF PHB (Expedited Forwarding Per-Hop Behavior)

Статус документа

В этом документе представлена информация для сообщества Internet. Документ не задаёт каких-либо стандартов Internet и может распространяться свободно.

Аннотация

Этот документ появился в процессе подготовки разъяснений к RFC2598 An Expedited Forwarding PHB, который привёл к публикации пересмотренной спецификации An Expedited Forwarding PHB[1]. Основным назначением документа является дополнительное разъяснение пересмотренного определения режима EF и его свойств. В документе также даны дополнительные примеры реализации и некоторые рекомендации по расчёту значений параметров нового определения для нескольких популярных архитектур планировщиков и маршрутизации.

Оглавление

[1]Режим ускоренной пересылки.

1. Введение

Режим ускоренной поэтапной пересылки (EF PHB[1]) был разработан в качестве основы для организации сервиса с малыми потерями, задержками и вариациями задержек. Потенциальные возможности такого сервиса и, следовательно, EF PHB очень велики. В силу важности этого PHB, весьма важно сделать требования к пересылке и её поведение для соответствующих EF узлов конкретными, измеримыми и однозначными.

К сожалению исходное определение EF PHB в RFC2598 [10] было недостаточно чётким (см. Приложение A и документ [4]). Более точное определение дано в документе [6]. Данный документ предназначен для улучшения понимания свойств нового определения и обеспечивает дополнительную информацию, не включённую для краткости в текст документа [6].

Документ разделен на несколько частей. В разделе 2 воспроизведено сокращённое определение EF PHB из [6]. Приведено дополнительное обсуждение этого определения и описаны некоторые его свойства. В разделах 3 и 4 рассмотрены вопросы, связанные с потерями и задержками на уровне отдельных пакетов. В разделе 5 рассмотрено влияние известных архитектур планирования на критичные параметры нового определения. Обсуждается также влияние отклонения реальных устройств от идеальной модели с буферизацией на выходе на значения критичных параметров в определении.

2. Определение EF PHB

2.1. Формальное определение

Интуитивное разъяснение определения новой группы EF описано в [6]. Здесь дословно приводится формальное определение из [6].

Узел, поддерживающий EF на интерфейсе I с некоторой заданной скоростью R, **должен** удовлетворять следующим уравнениям:

```
d_j <= f_j + E_a for all j > 0                              (eq_1)
```

где f_j определяется итеративно выражением

```
f_0 = 0, d_0 = 0
f_j = max(a_j, min(d_j-1, f_j-1)) + l_j/R,  for all j > 0  (eq_2)
```

В этом определении:

- **d_j** - время, когда последний бит j-ого пакета EF реально уходит с интерфейса I.
- **f_j** - целевое время отправки j-ого пакета EF с интерфейса I - «идеальное» время, до которого последнему биту пакета следует покинуть узел.
- **a_j** - время, когда последний бит j-ого пакета EF, направленного на выход I, реально прибыл на узел.
- **l_j** - размер (в битах) j-ого пакета EF для отправки через I. l_j относится к дейтаграмме IP (заголовок IP и данные) и не включает дополнительных полей нижележащих (например, MAC) уровней.
- **R** - заданная для EF скорость на выходе I (бит/сек).
- **E_a** - временное отклонение для агрегата EF. Отметим, что E_a представляет худший вариант отклонения реального времени отправки пакета EF от идеального времени его отправки (т. е., E_a определяет верхнюю границу разности d_j - f_j для всех j).
- **d_0** и **f_0** не указывают реальное время отправки пакета и используются исключительно для рекурсии. Начало отсчёта времени следует выбирать таким образом, чтобы в начальный момент в системе не было пакетов EF.
- При определении a_j и d_j «последний бит» пакета включает трейлер уровня 2, если тот присутствует, поскольку в общем случае пакет не может считаться готовым к пересылке, пока не будет получен трейлер.

Поддерживающий EF узел **должен** быть способен характеризоваться диапазоном поддерживаемых для каждого интерфейса значений R, при которых он соответствует приведённым выше уравнениям, а также значением E_a для каждого интерфейса. R может совпадать со скоростью среды или быть меньшее её. Значение E_a **может** быть задано, как худший случай для всех возможных значений R, или выражено, как функция R.

Отметим также, что узел может иметь множество входов и сложное внутреннее планирование в результате чего j-ый пакет EF, прибывающий на узел и предназначенный для некого интерфейса может не оказаться j-ым пакетом EF, отправляемым с этого интерфейса. Это говорит о том, что уравнения eq_1 и eq_2 не идентифицируют конкретные пакеты.

В дополнение к сказанному, узел, поддерживающий EF на интерфейсе I с некоторой заданной скоростью R, **должен** удовлетворять следующим уравнениям:

```
D_j <= F_j + E_p for all j>0                                (eq_3)
```

где F_j определяется итеративно выражениями

```
F_0 = 0, D_0 = 0
F_j = max(A_j, min(D_j-1, F_j-1)) + L_j/R,  for all j > 0   (eq_4)
```

В этом определении:

[1]Expedited Forwarding Per-Hop Behavior.

- **D_j** - реальное время отправки отдельного пакета EF, который поступил на узел для передачи через интерфейс I в момент A_j (т. е., для данного пакета, который был j-ым пакетом EF, полученным через любой вход и предназначенным для интерфейса I, значение D_j указывает время, когда последний бит данного пакета реально покинул узел через интерфейс I).
- **F_j** - целевое время отправки отдельного пакета EF, который прибыл на узел для отправки через интерфейс I в момент A_j.
- **A_j** - время, когда последний бит j-ого пакета EF, предназначенного для отправки через интерфейс I, реально прибыл на узел.
- **L_j** - размер (в битах) j-ого пакета EF, прибывающего на узел и предназначенного для отправки через интерфейс I. L_j измеряется для дейтаграммы IP (заголовок IP и данные) и не включает полей нижележащего (например, MAC) уровня.
- **R** - заданная для EF скорость на выходном интерфейса I (бит/сек).
- **E_p** - временное отклонение для агрегата EF. Отметим, что E_p представляет худший вариант отклонения реального времени отправки пакета EF от идеального времени его отправки (т. е., E_p определяет верхнюю границу разности D_j - F_j для всех j).
- **D_0** и **F_0** не указывают реальное время отправки пакета и используются исключительно для рекурсии. Начало отсчёта времени следует выбирать таким образом, чтобы в начальный момент в системе не было пакетов EF.
- При определении **A_j** и **D_j** «последний бит» пакета включает трейлер уровня 2, если тот присутствует, поскольку в общем случае пакет не может считаться готовым к пересылке, пока не будет получен трейлер.

D_j и F_j указывают время отправки для j-ого пакета, что делает уравнения eq_3 и eq_4 «осведомленными о пакетах». В этом заключается существенное отличие данной пары уравнений от двух первых уравнений этого параграфа.

Поддерживающему EF узлу **следует** быть способным характеризоваться диапазоном поддерживаемых значений R для каждого интерфейса, при которых узел соответствует приведённым выше уравнения, и значением E_p для каждого интерфейса. Значение E_p **может** быть задано , как худший случай для всех возможных значения R, или выражено, как функция R. **Может** быть также задано «неопределённое» значение E_p.

Для проверки совместимости с приведёнными здесь уравнениями может потребоваться иметь дело с пакетами, приходящими на разные интерфейсы в достаточно близкое время. Если два или более пакета EF, направленные на один выходной интерфейс прибывают (на разные входы) почти одновременно и разница во времени прибытия не может быть измерена, допускается использовать метод случайного выбора[1] для назначения одного из пакетов «первым».

В документе [6] приведены дополнительные рекомендации для узлов EF, в соответствии с которыми таким узлам **не следует** менять порядок пакетов в микропотоках.

Описанные в этом параграфе определения называют агрегатами и гарантиями масштабирования скорости с идентификацией пакетов (packet-identity-aware packet scale rate guarantee) [4],[2]. Другие варианты математической характеризации гарантий масштабирования скорости для пакетов приведены в Приложении B.

2.2. Связь с гарантией масштабирования скорости

Рассмотрим идеальное устройство с буферизацией EF FIFO на выходе. Для такого устройства i-й пакет на входе будет i-м также на выходе устройства. Следовательно, в рамках этой идеальной модели характеристики агрегирования «осведомлённости о пакетах» будут идентичны и E_a = E_p. По этой причине в данном параграфе задержка будет обозначаться просто E.

Можно показать, что для такого идеального устройства определение параграфа 2.1 строже общеизвестной кривой «скорость-задержка» [2] в том смысле, что планировщик, удовлетворяющий определению EF, будет соответствовать и кривой «скорость-задержка». В результате все известные для кривой «скорость-задержка» свойства применимы и к изменённому определению EF. Однако ниже будет показано, что определение из параграфа 2.1 лучше соответствует предназначению EF PHB, нежели кривая «скорость-задержка».

В работе [2] показано, что кривая «скорость-задержка» эквивалентна приведённому ниже определению кривой RLC[2]:

```
D(j) <= F'(j) + E                                          (eq_5)
```

где

```
F'(0)=0, F'(j)=max(a(j), F'(j-1))+ L(j)/R for all j>0      (eq_6)
```

Легко убедиться в том, что определение EF в параграфе 2.1 строже RLC, поскольку для всех j выполняется F'(j) >= F(j).

Легко видеть, что F'(j) в определении RLC соответствует времени j-го отправления, которое должно произойти при обслуживании агрегата EF с заданной скоростью R. В соответствии с общепринятой трактовкой, мы принимаем F'(j), как «время завершения исхода» при отправке j-го пакета.

Интуитивно понятный смысл кривой «скорость-задержка» для RLC заключается в том, любой пакет обслуживается не более чем на время E позже, чем он был бы обслужен в модели fluid.

Для RLC (и, следовательно, для более строгого определения EF) считается, что в любом интервале (0, t) агрегат EF приближается к желаемой скорости обслуживания R (пока имеется трафик, достаточный для удержания такой скорости). Разница между идеальным и реальным сервисом в этом интервале зависит от задержки E, которая, в свою очередь, зависит от реализации планировщика. При снижении E уменьшается и разница между заданной в конфигурации скоростью и реальной скоростью, обеспечиваемой планировщиком.

[1] В оригинале «random tie-breaking method». *Прим. перев.*
[2] Rate Latency Curve - кривая зависимости между скоростью и задержкой.

Хотя RLC гарантирует для агрегата EF желаемую скорость во всех интервалах (0, t) с точностью до указанной ошибки, тем не менее могут возникать значительные перерывы в обслуживании. Предположим, например, (большое число) N идентичных пакетов EF размера L приходит от разных интерфейсов в очередь EF при отсутствии трафика, не относящегося к EF. В этом случае любой сохраняющий работу (work-conserving) планировщик будет обслуживать все N пакетов со скоростью канала. Если последний пакет передан в момент NL/C, где C указывает пропускную способность выходного канала, F'(N) будет равно NL/R. То есть, планировщик работает не идеально, поскольку NL/C < NL/R для R < C. Предположим сейчас, что в момент NL/C приходит большое число пакетов, не относящихся к EF, за которыми следует один пакет EF. В этом случае планировщик может обоснованно задержать начало передачи пакета EF до момента F'(N+1) = (N+1)L/R + E - L/C. Это означает, что агрегат EF не будет обслуживаться совсем в интервале (NL/C, (N+1)L/R + E - L/C). Интервал этот может быть достаточно большим, если R существенно меньше C. По сути, агрегат EF может быть «наказан» прерыванием обслуживания за получение ускоренного по сравнению с заданным в конфигурации обслуживания в начале.

Новое определение EF смягчает эту проблему за счёт введения в рекурсию элемента min(D(j-1), F(j-1)). Это, по сути, означает, что время завершения потока «сбрасывается», если пакет передаётся до «идеального» момента его отправки. Для случая этого случая при обслуживании агрегата EF в порядке FIFO предположим, что пакет приходит в момент t на сервер, соответствующий определению EF. Пакет тогда будет передан не позднее момента t + Q(t)/R + E, где Q(t) указывает размер очереди EF в момент t (с учётом рассматриваемого пакета) [4].

2.3. Необходимость двойной характеризации EF PHB

В более общем случае, когда выходной планировщик не обслуживает пакеты EF в порядке FIFO или переменная внутренняя задержка в устройстве меняет порядок следования пакетов, i-й пакет при доставке на данный выходной интерфейс может оказаться не i-м при передаче с этого интерфейса. В таких случаях агрегирование и идентификация пакетов уже не будут давать одинаковых результатов.

Определение агрегатного поведения можно рассматривать как действительно совокупную характеристику услуг, предоставляемых пакетам EF. В качестве аналогии рассмотрим «тёмный» резервуар, в который помещаются все поступающие пакеты. Планировщику разрешено случайным образом помечать пакеты в резервуаре, не зная порядка их поступления в резервуар. Совокупная часть определения измеряет точность выходной скорости, обеспечиваемой для агрегата EF в целом. Чем меньше E_a, тем более точна будет гарантия того, что резервуар сливается со скоростью не меньше заданной в конфигурации.

Отметим, что в этой аналогии с резервуаром пакеты агрегата EF могут произвольно менять порядок. Однако в определении EF PHB [6] явно требуется сохранение порядка пакетов внутри микропотока. Это требование ограничивает реализации планировщиков, что в примере с резервуаром будет требовать сохранения порядка выхода пакетов одного микропотока из резервуара, но позволяет менять порядок на уровне агрегата.

Отметим, что изменение порядка внутри агрегата при отсутствии нарушений порядка на уровне потока не обязательно говорит о «плохом» обслуживании. Рассмотрим, например, планировщик, который работает с 10 разными «потоками» EF, имеющими разные скорости. Знающий о потребностях в скорости планировщик может выбрать передачу пакета из более быстрого потока до передачи пакета из более медленного с целью снижения вариаций задержки на уровне потока. В частности, идеальный планировщик WFQ, осведомлённый о «потоках» будет менять порядок внутри агрегата, поддерживая порядок и незначительные вариации задержки на уровне потока.

Интуитивно понятно, что для такого планировщика, а также для более простого планировщика FIFO «точность» скорости обслуживания имеет важнейшее значение для минимизации вариаций задержки на уровне «потока». Для оценки точности скорости обслуживания нужно определение определение с учётом идентификации пакетов.

Однако малое значение E_a не даёт каких-либо гарантий абсолютного значения задержки для пакета. Фактически, если скорость превосходит заданную в настройке, определение агрегатного поведения может приводить к сколь угодно большим задержкам для подмножества пакетов. Это является основным мотивом создания осведомленного об идентификации пакетов определения.

Основной целью характеризации на уровне пакетов для реализации EF в отличие от характеризации поведения агрегата является обеспечение способа нахождение способа сделать задержку на уровне пакета функцией от параметров входного трафика.

Хотя определение агрегатного поведения характеризует точность скорости обслуживания для всего агрегата EF, осведомленная об идентификации пакетов часть характеризует отклонение устройства от идеального сервера, который обслуживает агрегат EF в режиме FIFO со скоростью не ниже заданной в конфигурации.

Значение E_p в осведомленном об идентификации пакетов определении подвержено, следовательно, влиянию двух факторов - точность агрегатной скорости обслуживания и степень разупорядочения пакетов в агрегате EF (порядок пакетов в одном микропотоке не меняется). Следовательно, знающее о субагрегате устройство, которое обеспечивает идеальную скорость обслуживания для агрегата, а также для каждого из субагрегатов, никогда не может иметь очень большого значения E_p (в этом случае E_p должно быть не меньше отношения размера максимального пакета к наименьшей скорости для всех субагрегатов). В результате большое значение E_p не обязательно говорит о плохом обслуживании агрегата EF, а скорее показывает, что сервис хорош, но не является FIFO. С другой стороны, большое значение E_p может также показывать значительную неточность (пики) и, следовательно, в этом случае большое значение E_p говорит о низком качестве реализации.

В результате большое число E_p не обязательно служит мерилом качества реализации EF. Однако малое значение E_p показывает высокое качество реализации FIFO.

Поскольку E_p и E_a относятся к разным аспектам реализации EF, их следует рассматривать совместно для определения качества реализации.

3. Задержки по пакетам

Основной мотив осведомленного об идентификации пакетов определения заключается в возможности привязки к пакету величины задержки. В этом разделе рассматривается расчёт задержки для отдельных пакетов.

3.1. Границы задержки на одном интервале

Если общий трафик, приходящий на выходной порт I со всех входов, ограничивается с помощью механизма leaky bucket[1] с параметрами (R, B), где R указывает настроенную для I скорость, а B определяет размер «ёмкости» (величину пика), то задержка любого пакета, уходящего из I ограничена значением D_p, которое определяется уравнением

```
D_p = B/R + E_p                                          (eq_7)
```

(см. Приложение B).

Поскольку границы задержек зависят от заданной в конфигурации скорости R и уровня всплесков трафика на входе B, желательно обеспечить видимость обоих параметров для пользователя устройства. Для PDB, желающего получить конкретные границы задержек может потребоваться ограничение задаваемой в конфигурации скорости и поддерживаемого уровня всплесков трафика. Уравнение (eq_7) обеспечивает возможность определения приемлемого рабочего диапазона для устройства с заданным E_p. Это может быть полезно также для ограничения общей нагрузки для заданного выхода некой скоростью R_1 < R (например, для ограничения сквозной задержки [5]). Важно понимать, что границы задержки в (eq_7) не зависят от R_1 несмотря на возможность настройки этого параметра. Может оказаться возможным более чёткое задание границ за счёт явного ограничения входной скорости, но ограничения (eq_7) не используют эту информацию.

3.2. Худший случай задержки на многоэтапном участке

Хотя PHB по своей природе определяет локальное поведение, в этом параграфе кратко рассматривается вопрос попакетной задержки при прохождении пакетов через несколько интервалов, реализующих EF PHB. С учётом границ задержки (eq_7) на одном интервале возникает соблазн предположить, что границы задержки при прохождении h интервалов будут определяться произведением h на значение (eq_7). Однако это не всегда верно, если B не представляет худший вариант входных пиков трафика на всех узлах сети.

К сожалению, определить такое наихудшее значение B - задача нетривиальная. Если EF PHB реализовано с использованием агрегатного планирования на основе классов, где все пакеты EF используют общий буфер FIFO, эффект накопления вариаций задержки (jitter) может приводить к росту пиков от интервала к интервалу. В частности, можно показать, без введения некоторых ограничений на использование EF даже при идеальной форме всех потоков EF на входе для любого значения задержки D можно создать сеть, где использование EF на любом канале ограничено так, что не превышает заданное значение, потоки не проходят через превышающее заданных порог число интервалов, но при этом все равно имеются пакеты, которые сталкиваются с задержкой, превышающей D [5]. Этот результат предполагает, что возможность ограничить наихудший вариант пиков трафика и результирующую сквозную задержку на пути через несколько интервалов может потребовать не только ограничения использования EF на всех каналах, но будет также вносить ограничения для глобальной топологии сети. Такие топологические ограничения потребуется задавать в определении любого PDB, создаваемого «поверх» EF PHB, если для такого PDB требуется строгое ограничение для худшего случая задержки.

4. Потеря пакетов

Любому устройству с конечным размером буферов может потребоваться отбрасывать пакеты при достаточно сильных пиках входного трафика. В плане реализации цели EF по снижению уровня потерь узел можно характеризовать рабочим диапазоном, в котором не будет происходить потерь EF по причине насыщения. Это можно задать как «переполняющуюся ёмкость» (token bucket) со скоростью r <= R и величиной пика B, которые могут быть восприняты от всех входов для данного выхода без возникновения потерь.

Однако, как было отмечено в предыдущем разделе, феномен накопления вариаций задержки в общем случае осложняет гарантии того, что входные пики никогда не выйдут за пределы заданного рабочего диапазона для внутренних узлов домена DiffServ. Гарантия отсутствия потерь при прохождении через множество интервалов может потребовать задания ограничений на топологию сети, которые выходят за рамки локального по сути определения PHB. Таким образом, должна быть возможность понять соответствует ли устройство определению EF даже при потере некоторых пакетов.

Это можно сделать с помощью отдельной (off-line) проверки выполнения уравнений (eq_1) - (eq_4). После наблюдения последовательности пакетов, приходящих на узел и уходящих с него, не ушедшие пакеты считаются потерянными и обычно удаляются из входного потока. Остальные пакеты составляют прибывший поток, а ушедшие с узла пакеты - ушедший поток. Соответствие уравнениям может быть, таким образом, проверено путём рассмотрения пакетов, благополучно прошедших через узел.

Отметим, что указание того, какие пакеты теряются в случае возникновения потерь, выходит за рамки определения EF PHB. Однако пакеты, которые не были потеряны, должны соответствовать уравнениям определения EF PHB в параграфе 2.1.

5. Вопросы реализации

Проходящие через маршрутизатор пакеты будут сталкиваться с задержкой по множеству причин. Двумя основными компонентами задержки являются время нахождения пакета в выходном буфере, пока планировщик не решит передать этот пакет, и время время реальной передачи пакета в выходную линию.

Однако в маршрутизаторе могут возникать и другие задержки пакетов. Например, маршрутизатор может затратить некоторое время на обработку заголовка пакета перед его размещением в соответствующем выходном буфере. В другом случае маршрутизатор может включать буфер FIFO (называется очередью передачи в [7]), где пакет находится после выбора выходным планировщиком до момента передачи. В подобных случаях возникающая дополнительная задержка может быть учтена в параметрах задержки E_a и E_p.

Особого внимания требует реализация EF на маршрутизаторах с многоступенчатой матрицей коммутации. Пакет может сталкиваться с дополнительной задержкой по причине его конкуренции за ресурсы пересылки с другим

[1] Переполняющаяся ёмкость.

трафиком в нескольких «точках конкуренции» коммутационного ядра. Задержка пакета EF может возникнуть ещё до того, как он попадёт к планировщику выходного канала и её также следует учитывать. Маршрутизаторы с буферизацией на входе и на входе и выходе на основе перекрёстной модели (crossbar) также могут потребовать изменения параметров задержки. Такие факторы, как коэффициент ускорения (speedup factor) и выбор алгоритмов перекрёстного арбитража, могут существенно воздействовать на задержки.

Задержка в ядре коммутатора может иметь два источника, которые должны быть рассмотрены. Первая часть задержки является фиксированной и не зависит от другого трафика. Эта компонента задержки включает такие аспекты, как время фрагментации и сборки пакетов в ядрах, коммутирующих ячейки, постановку пакетов в очередь и извлечение из очередей на каждом этапе коммутации, а также время передачи между этапами. Вторая часть задержки в ядре коммутатора является переменной и зависит от типа и объёма другого трафика, проходящего через ядро. Эта задержка возникает в тех случаях, когда этапы коммутации включают смешанный трафик, проходящий между разными парами входных и выходных портов. В результате пакетам EF приходиться конкурировать с другим трафиком за ресурсы пересылки в ядре. Часть этого одновременного трафика может даже принадлежать другим агрегатам, не относящимся к EF. Это вносит дополнительную задержку, которая тоже принимается во внимание при учёте времени в определении.

Для учёта этих соображений в данном разделе рассматриваются два упрощённых примера реализации. В первом случае идеальный узел с буферизацией на выходе принимает пакеты от входного интерфейса с незамедлительной доставкой их выходному планировщику. В этой модели свойства выходного планировщика полностью определяют значения параметров E_a и E_p. Будем рассматривать случай когда выходной планировщик реализует агрегатные очереди по классам, где все пакеты EF попадают в одну очередь. Мы рассмотрим значения E_a и E_p для разных широко распространённых планировщиков на основе класса.

Во втором примере рассматривается маршрутизатор, как «чёрный ящик» с известной границей переменной части задержки, которая может возникать с момента прибытия пакета на вход до момента доставки на соответствующий выход. Выходной планировщик предполагается агрегатным, где все пакеты EF используют одну очередь FIFO с известными значениями E_a(S)=E_p(S)=E(S). Эта модель является подходящей абстракцией для большого класса реальных маршрутизаторов.

5.1. Модель буферизованного выхода с EF FIFO

Как было отмечено выше, в этой модели E_a = E_p, поэтому мы будем опускать индексы, обозначая обе задержки E. В оставшейся части этого параграфа рассматриваются значения E для множества реализаций планировщиков.

5.1.1. Очередь со строгим, неупреждающим приоритетом

Строгий планировщик по приоритетам (Strict Priority), где все пакеты EF используют одну общую очередь FIFO, которая обслуживается строго с неупреждающим приоритетом по отношению к другим очередям, соответствует определению EF с параметром задержки E = MTU/C, где MTU указывает максимальный размер пакета, а C - скорость выходного канала.

5.1.2. WF2Q

Другим планировщиком, удовлетворяющим определению EF с малой задержкой, является WF2Q, описанный в [1]. Планировщик по классам WF2Q, в котором для всего трафика EF используется одна общая очередь с весом, соответствующим заданной в конфигурации скоростью агрегата EF, соответствует определению EF с параметром задержки E = MTU/C+MTU/R.

5.1.3. DRR

Для DRR[1] [12] можно показать, что E будет расти пропорционально N*(r_max/r_min)*MTU, где r_min и r_max обозначают наименьшую и наибольшую из скоростей, выделенных для всех очередей в планировщике, а N - число очередей.

5.1.4. SFQ и SCFQ

Для беспристрастных очередей SFQ[2] [9] и SCFQ[3] [8] можно показать, что E будет расти пропорционально числу очередей в планировщике.

5.2. Маршрутизатор с внутренней задержкой и EF FIFO на выходе

В этом разделе мы рассмотрим маршрутизатор, в котором приходящий пакет может столкнуться с переменной задержкой D_v, верхняя граница которой составляет D (т. е., 0<=D_v<=D). На выходе все пакеты EF используют общую очередь одного класса. Планирование для очередей по классам выполняется планировщиком с известным значением E_p(S)=E(S), где E(S) соответствует модели в которой планировщик реализован как идеальное устройство с буферизацией на выходе.

В этом случае расчёт E_p более сложен. Можно показать, что для таких устройств E_p = E(S)+2D+2B/R (см. [13]).

Вернёмся к обсуждению раздела 3, где ограничение входных пиков B может оказаться непростым в обычной топологии. При отсутствии информации о границе B можно сказать, что E_p = E(S) + D*C_inp/R (см. [13]).

Отметим также, что границы в этом разделе определяются с использованием лишь границ переменной части интервальной задержки и границ ошибок выходного планировщика. При наличии дополнительной информации об архитектуре устройства можно рассчитать границы более точно.

[1] Deficit Round Robin.
[2] Start-Time Fair Queuing.
[3] Self-Clocked Fair Queuing.

6. Вопросы безопасности

В этом информационном документе приведена дополнительная информация с целью улучшить понимание EF PHB, описанного в [6]. Документ не добавляет новых функций и в результате не создаёт новых проблем безопасности в дополнение к описанным в упомянутой спецификации.

7. Литература

[1] J.C.R. Bennett and H. Zhang, "WF2Q: Worst-case Fair Weighted Fair Queuing", INFOCOM'96, March 1996.

[2] J.-Y. Le Boudec, P. Thiran, "Network Calculus", Springer Verlag Lecture Notes in Computer Science volume 2050, June 2001 (available online at http://lcawww.epfl.ch).

[3] Bradner, S., "Key Words for Use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[4] J.C.R. Bennett, K. Benson, A. Charny, W. Courtney, J.Y. Le Boudec, "Delay Jitter Bounds and Packet Scale Rate Guarantee for Expedited Forwarding", Proc. Infocom 2001, April 2001.

[5] A. Charny, J.-Y. Le Boudec "Delay Bounds in a Network with Aggregate Scheduling". Proc. of QoFIS'2000, September 25-26, 2000, Berlin, Germany.

[6] Davie, B., Charny, A., Baker, F., Bennett, J.C.R., Benson, K., Boudec, J., Chiu, A., Courtney, W., Davari, S., Firoiu, V., Kalmanek, C., Ramakrishnan, K.K. and D. Stiliadis, "An Expedited Forwarding PHB (Per-Hop Behavior)", RFC 3246, March 2002.

[7] T. Ferrari and P. F. Chimento, "A Measurement-Based Analysis of Expedited Forwarding PHB Mechanisms," Eighth International Workshop on Quality of Service, Pittsburgh, PA, June 2000.

[8] S.J. Golestani. "A Self-clocked Fair Queuing Scheme for Broad-band Applications". In Proceedings of IEEE INFOCOM'94, pages 636-646, Toronto, CA, April 1994.

[9] P. Goyal, H.M. Vin, and H. Chen. "Start-time Fair Queuing: A Scheduling Algorithm for Integrated Services". In Proceedings of the ACM-SIGCOMM 96, pages 157-168, Palo Alto, CA, August 1996.

[10] Jacobson, V., Nichols, K. and K. Poduri, "An Expedited Forwarding PHB", RFC 2598, June 1999.

[11] Jacobson, V., Nichols, K. and K. Poduri, "The 'Virtual Wire' Behavior Aggregate", Work in Progress.

[12] M. Shreedhar and G. Varghese. "Efficient Fair Queuing Using Deficit Round Robin". In Proceedings of SIGCOMM'95, pages 231-243, Boston, MA, September 1995.

[13] Le Boudec, J.-Y., Charny, A. "Packet Scale Rate Guarantee for non-FIFO Nodes", Infocom 2002, New York, June 2002.

Приложение A. Сложности с определением RFC 2598 EF PHB

В работе [10] приведено определение EF PHB.

> «EF PHB определяется, как трактовка пересылки для отдельного агрегата diffserv[1], при которой скорость отправки пакетов данного агрегата с любого узла diffserv не меньше заданного значения. **Следует** обеспечивать гарантию заданной скорости отправки независимо от интенсивности остального трафика, пытающегося проходить через узел. **Следует** обеспечивать среднее значение скорости[2] отправки не менее заданного значения.»

Буквальная интерпретация этого определения приводит к поведению, для которого в двух последующих параграфах показано несоответствие. Определение также излишне ограничивает максимально возможную скорость агрегата EF.

A.1 Синхронизированная пересылка

Рассмотрим поток, пересылаемый с маршрутизатора с заданной в EF скоростью R=C/2, где C - скорость выходной линии. На рисунке ниже E представляет собой пакет EF размера MTU, а x - пакет, не относящийся к EF, или неиспользуемую ёмкость (также размера MTU).

```
E x E x E x E x E x...
 |-----|
```

Интервал между вертикальными линиями на рисунке составляет 3*MTU/C, что превышает MTU/(C/2), и является субъектом определения EF PHB. В течение этого интервала следовало бы переслать 3*MTU/2 битов агрегата EF, но пересылается только MTU битов. Хотя данную картину пересылки следует соответствующей спецификации при любой разумной интерпретации EF PHB, она формально не соответствует определению RFC 2598.

Отметим, что такая картина может возникать в любом сохраняющем работу (work-conserving) планировщике в идеальной архитектуре с выходной буферизацией, где пакеты EF прибывают с равными интервалами, как показано на рисунке, а не относящийся к EF трафик отсутствует.

Пример может показаться тривиальным, но он демонстрирует недостаток математической точности в формальном определении. То, что сохраняющий работу не может формально соответствовать определению, является неприятным фактом, который может служить предупреждением о том, что в определении нужно что-то поменять для решения проблемы.

Причина, лежащая в основе описанной здесь проблемы, достаточно проста - можно считать, что агрегат EF обслуживается с заданной скоростью лишь в некотором интервале, где имеется очередь пакетов EF, достаточная для поддержания этой скорости. В приведённом выше примере пакеты приходят в точности с той же скоростью, с которой они обслуживаются, и поэтому постоянной очереди пакетов нет. Иногда, если скорость поступления пакетов меньше заданной для агрегата EF скорости, очереди нет совсем и возникает отмеченная формальная сложность.

[1] Differentiated Service - дифференцированное обслуживание. *Прим. перев.*

[2] При измерении за период не менее, чем потребуется для передачи в канал с такой скоростью пакетов размера MTU.

По-видимому, простым решением этой сложности будет требование обслуживать агрегат EF с заданной конфигурацией скоростью лишь при наличии заполненной очереди. Однако ниже будет показано, что такого решения не достаточно.

A.2 Внутренняя задержка в маршрутизаторе

Сейчас мы согласны с тем, что приведённый выше пример не так тривиален, как может показаться на первый взгляд.

Рассмотрим маршрутизатор с настроенной скоростью EF, равной R = C/2 как в предыдущем примере, и внутренней задержкой 3T (где T = MTU/C) между временем прихода пакета на маршрутизатор и временем, когда этот пакет становится первым в очереди на отправку в выходной канал. Задержку в маршрутизаторе могут вызывать такие события, как обработка заголовков, поиск маршрута и задержка в многоуровневой матрице коммутации. Предположим, что трафик EF поступает со стабильной скоростью (2/3)R = C/3. Временные параметры прохождения пакетов через маршрутизатор показаны ниже.

```
Номер пакета EF               1  2  3   4   5   6   ...
Прибытие (на маршрутизаторе) 0  3T 6T  9T  12T 15T ...
Прибытие (на планировщике)   3T 6T 9T  12T 15T 18T ...
Отправка                     4T 7T 10T 13T 16T 19T ...
```

И на сей раз выход не соответствует приведённому в RFC 2598 определению EF PHB. Как и в предыдущем примере причина заключается в том, что планировщик не может пересылать трафик EF быстрее, чем тот прибывает. Однако легко заметить, что наличие внутренней задержки обеспечивает постоянное присутствие пакетов в маршрутизаторе. Внешний наблюдатель может сделать правомерное заключение о том, что число поступивших на маршрутизатор пакетов EF всегда превышает число покинувших маршрутизатор пакетов EF хотя бы на 1, следовательно агрегат EF постоянно задерживается. Однако, несмотря на постоянную задержку агрегата EF, наблюдаемая выходная скорость никогда не становится существенно меньше заданной в конфигурации скорости.

Этот пример показывает, что простого добавления условия приёма агрегата EF с заданной в конфигурации скоростью лишь при отсутствии задержки агрегата EF в данном случае не достаточно.

Тем не менее, описанная здесь проблема имеет важное значение на практике. Большинство маршрутизаторов вносит некоторую внутреннюю задержку. Не предполагается, что производители, заявляющие совместимость с EF, будут объявлять также детали внутренних задержек в маршрутизаторах. Следовательно, наличие внутренних задержек может приводить к тому, что полностью соответствующая EF реализация будет демонстрировать поведение, представляющееся не совместимым, что явно нежелательно.

A.3 Максимальная скорость и обеспечение эффективности

Хорошо понятно, что с любым неупреждающим (non-preemptive) планировщиком заданная в соответствии с RFC 2598 скорость для агрегата EF не может превышать C/2 [11]. Это обусловлено тем, что пакет EF размера MTU может поступить в пустую очередь в момент t, когда началась обработка не относящегося к EF пакета размером MTU. Максимальное число битов EF, которые могут быть переданы в течение интервала [t, t + 2*MTU/C], равно MTU. Но если задана скорость R > C/2, размер этого интервала будет превышать MTU/R и для обеспечения соответствия маршрутизатор в течение этого интервала должен обрабатывать больше MTU битов EF. Следовательно R должно быть не более C/2.

Можно показать, что для отличных от PQ планировщиков (например, разных реализаций WFQ) максимальная задаваемая в конфигурации скорость может быть много меньше 50%. Например, для SCFQ [8] максимальная скорость не может превышать C/N, где N - число очередей в планировщике. Для WRR, отмеченного как соответствующий спецификации в параграфе 2.2 документа RFC 2598, это ограничение ещё сильнее. Это обусловлено тем, что в этих планировщиках пакеты, прибывающие в пустую очередь EF, могут быть вынуждены ждать, пока будет обработано по одному (для SCFQ) или несколько (для WRR) пакету из каждой другой очереди.

Хотя часто предполагается, что заданная в конфигурации скорость трафика EF существенно ниже пропускной способности канала, требование, чтобы эта скорость никогда не была выше 50% от пропускной способности канала, представляется неоправданным ограничением. Например в полносвязной (full mesh) сети, где любой поток проходит от источника к получателю только через один канал, не видится разумных причин ограничивать скорость трафика EF значением 50% (или даже меньше для некоторых планировщиков) от пропускной способности канала.

Другим, возможно более ярким, примером является факт, что даже устройство TDM с выделенными временными интервалами не может быть настроено на пересылку трафика EF со скоростью больше 50% от скорости канала без нарушения RFC 2598 (пока весь канал не выделен для EF). Если заданная в конфигурации скорость EF превышает 50% (но меньше скорости канала), всегда будет присутствовать интервал больше MTU/R, в течение которого доступна будет скорость меньше заданной в конфигурации. Например, предположим, что задана скорость для агрегата EF в размере 2C/3. Тогда картина пересылки на устройстве TDM будет иметь вид

```
  E E x E E x E E x ...
    |---|
```

где лишь один пакет обслуживается в отмеченном интервале 2T = 2MTU/C. Но в соответствии с определением RFC 2598 маршрутизатор в течение этого интервала должен обработать не менее 4/3 MTU. Возможность заказать для трафика EF не более половины пропускной способности даже для линии TDM показывает искусственность и ненужность такого ограничения.

A.4 Нетривиальная природа сложностей

Одним из способов решения рассмотренных выше проблем может быть попытка прояснить определение интервалов, к которым определение поведения применяется или усреднять по множеству интервалов. Однако такая попытка связана со множеством аналитических и математических сложностей. Например, попытка связать время начала интервалов с некими «эпохами» пересылаемых потоков потребует глобальной синхронизации часов и на практике может приводить к ложным трактовкам и ошибкам.

Другой подход может заключаться в усреднении скорости за больший интервал времени. Однако нет достаточно понимания величины интервала, которого будет достаточно во всех разумных ситуациях. Кроме того, такой подход может поставить под угрозу краткосрочные гарантии, которые являются сущностью EF PHB.

Была исследована также комбинация двух простых исправлений. Первое заключалось в добавлении условия, в соответствии с которым определяются лишь те интервалы, входящие в период, в течение которого агрегат EF непрерывно задерживается в маршрутизаторе (т. е., когда пакет EF находится в маршрутизаторе). Второе заключалось в учёте ошибок (задержки) при оценке качества анонсируемых услуг EF.

Ниже приведено предлагаемое определение с учётом этих двух изменений.

> В течение любого интервала времени (t1, t2), в котором наблюдается непрерывная задержка трафика EF, должно быть обслужено не менее R(t2 - t1 - E) битов трафика EF, где R - заданная в конфигурации скорость для агрегата EF, а E - зависящий от реализации параметр задержки.

Условие «непрерывной задержки» (continuously backlogged) позволяет обойти сложности с нехваткой пакетов для пересылки, а добавление параметра задержки размером MTU/C решает проблему «синхронизации», отмеченную в параграфе A.1, а также снимает ограничение на задаваемую в конфигурации скорость EF.

Однако ни одно из этих исправлений (и оба вместе) не решает проблему, отмеченную в примере параграфа A.2. Чтобы увидеть это, вернёмся к примеру из параграфа A.2, где агрегат EF непрерывно задерживается, но скорость обслуживания агрегата EF существенно меньше заданной в конфигурации скорости и, следовательно, отсутствует конечное значение задержки, что не позволяет обеспечить соответствие этого примера определению.

Приложение B. Дополнительные характеристики гарантий масштабирования скорости

Доказательства некоторых оценок из этого документа представлены в работе [13]. В этих доказательствах используется алгебраическая характеризация определения агрегата, данного в уравнениях (eq_1) и (eq_2), а также определения осведомлённости о пакетах, данного в уравнениях (eq_3) и (eq_4). Поскольку эта характеризация представляет самостоятельный интерес, она описана в данном приложении.

Теорема B1. Характеризация определения агрегата без f_n.

Рассмотрим систему, где пакеты получают номера 1, 2, ... в порядке их поступления. Как в определении агрегата обозначим a_n время прихода пакета n, d_n - время его отправки и l_n - размер n-го пакета для отправки. Пусть d_0=0. Определение агрегата со скоростью R и задержкой E_a эквивалентно тому, что для всех n и всех 0<=j<= n-1 выполняется условие

```
d_n <= E_a + d_j + (l_(j+1) + ... + l_n)/R                    (eq_b1)
```

или имеется некий номер j+1 <= k <= n, такой что

```
d_n  <= E_a + a_k + (l_k + ... + l_n)/R                       (eq_b2)
```

Теорема B2. Характеризация определения осведомлённости о пакетах без F_n.

Рассмотрим систему, где пакеты получают номера 1, 2, ... в порядке их поступления. Как в определении осведомлённости а пакетах обозначим A_n и D_n время прибытия и отправки пакета n, а L_n - размер этого пакета. Пусть D_0=0. Определение осведомлённости о пакете для скорости R и задержки E_p эквивалентно тому, что для всех n и всех 0<=j<= n-1 выполняется условие

```
D_n <= E_p + D_j + (L_{j+1} + ... + L_n)/R                    (eq_b3)
```

или имеется некий номер j+1 <= k <= n, такой что

```
D_n  <= E_p + A_k + (L_k + ... + L_n)/R                       (eq_b4)
```

Доказательства обеих теорем можно найти в работе [13].

Благодарности

Этот документ не был бы написан без Fred Baker, Bruce Davie и Dimitrios Stiliadis. Их время, поддержка и содержательные комментарии неоценимы.

Адреса авторов

Anna Charny
Cisco Systems
300 Apollo Drive
Chelmsford, MA 01824
EMail: acharny@cisco.com

Jon Bennett
Motorola
3 Highwood Drive East
Tewksbury, MA 01876
EMail: jcrb@motorola.com

Kent Benson
Tellabs Research Center
3740 Edison Lake Parkway #101
Mishawaka, IN 46545
EMail: Kent.Benson@tellabs.com

Jean-Yves Le Boudec

ICA-EPFL, INN
Ecublens, CH-1015
Lausanne-EPFL, Switzerland
EMail: jean-yves.leboudec@epfl.ch

Angela Chiu
Celion Networks
1 Sheila Drive, Suite 2
Tinton Falls, NJ 07724
EMail: angela.chiu@celion.com

Bill Courtney
TRW
Bldg. 201/3702
One Space Park
Redondo Beach, CA 90278
EMail: bill.courtney@trw.com

Shahram Davari
PMC-Sierra Inc
411 Legget Drive
Ottawa, ON K2K 3C9, Canada
EMail: shahram_davari@pmc-sierra.com

Victor Firoiu
Nortel Networks
600 Tech Park
Billerica, MA 01821
EMail: vfiroiu@nortelnetworks.com

Charles Kalmanek
AT&T Labs-Research
180 Park Avenue, Room A113,
Florham Park NJ
EMail: crk@research.att.com

K.K. Ramakrishnan
TeraOptic Networks, Inc.
686 W. Maude Ave
Sunnyvale, CA 94086
EMail: kk@teraoptic.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru

Полное заявление авторских прав

Подтверждение

Финансирование функций RFC Editor обеспечено Internet Society.