

# Протоколы IntServ и DiffServ. Поддержка QoS в маршрутизаторах

Технологии стека TCP/IP были разработаны для эластичного трафика, который достаточно терпим к задержкам и вариациям задержек пакетов. Поэтому основное внимание разработчиков протоколов TCP/IP было сосредоточено на обеспечении надежной передачи трафика с помощью TCP. Однако со временем для борьбы с перегрузками на медленных линиях доступа в IP-маршрутизаторы были встроены многие механизмы QoS, в том числе механизмы приоритетных и взвешенных очередей, профилирования трафика и обратной связи. Однако эти механизмы использовались каждым сетевым администратором по своему усмотрению, без какой-либо стройной системы. И только в середине 90-х годов начались работы по созданию стандартов QoS для IP-сетей, на основе которых можно было бы создать систему поддержки параметров QoS в масштабах составной сети и даже Интернета.

В результате были разработаны две системы стандартов QoS для IP-сетей:

система **интегрированного обслуживания (Integrated Services, IntServ)** ориентирована на предоставление гарантий QoS для потоков *конечных пользователей* «из конца в конец»;

система **дифференцированного обслуживания (Differentiated Services, DiffServ)** предоставляет гарантии QoS в агрегированной форме *для классов трафика*.

Обе системы включают в себя все базовые элементы поддержки QoS:

- кондиционирование трафика;
- сигнализация, обеспечивающая координацию маршрутизаторов;
- резервирование пропускной способности интерфейсов маршрутизаторов для потоков и классов;
- приоритетные и взвешенные очереди.

## Система интегрированного обслуживания

Система **IntServ** начала разрабатываться в IETF в начале 90-х годов, она была первой моделью, в рамках которой проблема обеспечения параметров QoS в сетях TCP/IP начала решаться систематически. Модель IntServ предполагает интегрированное взаимодействие маршрутизаторов сети по обеспечению требуемого качества обслуживания *вдоль всего пути потока* между конечными компьютерами.

Ресурсы маршрутизаторов (пропускная способность интерфейсов, размеры буферов) распределяются в соответствии с QoS-запросами приложений в пределах, разрешенных политикой QoS для данной сети. Эти запросы распространяются по сети сигнальным **протоколом резервирования ресурсов (Resource reSerVation Protocol, RSVP)**. Этот протокол подобен *сигнальным протоколам телефонных сетей*, с помощью которых вызывающий абонент сети запрашивает соединение с вызываемым абонентом. Однако специфика дейтаграммных пакетных сетей естественно накладывает свой отпечаток. Так, с помощью RSVP соединение в сети не устанавливается, так как IP-пакеты в любом случае (при резервировании или без него) будут передаваться маршрутизаторами между конечными узлами на основе записей таблиц маршрутизации. Резервируется же пропускная способность и запрашиваются желаемые параметры QoS для некоторого потока между двумя конечными узлами сети. Подобное резервирование является *односторонним*, так что если гарантированное качество обслуживания и пропускная способность должны быть обеспечены для двустороннего обмена, потребуются две операции резервирования, при этом параметры резервирования могут быть разными для каждого направления.

Протокол RSVP обеспечивает резервирование соединений как с двухточечной, так и с древовидной топологиями. Древовидная топология соединения соответствует случаю передачи пакетов с групповым IP-адресом, когда один узел передает данные сразу нескольким получателям и копии пакета распространяются вдоль ветвей древовидного маршрута. Мультимедийные приложения, которым требуются гарантии пропускной способности и параметров QoS и которые, следовательно, являются одним из потенциальных пользователей протокола RSVP, могут задействовать групповое вещание для рационального расходования ресурсов сети. Поэтому разработчики протокола RSVP не могли не учесть этого варианта топологии соединений.

Рассмотрим работу протокола RSVP на примере сети, показанной на рис. 1. В этой сети имеется маршрут группового вещания, соединяющий передающий узел C1 с двумя принимающими узлами C2 и C3. Рассмотрим основные этапы процесса резервирования для нашего примера.

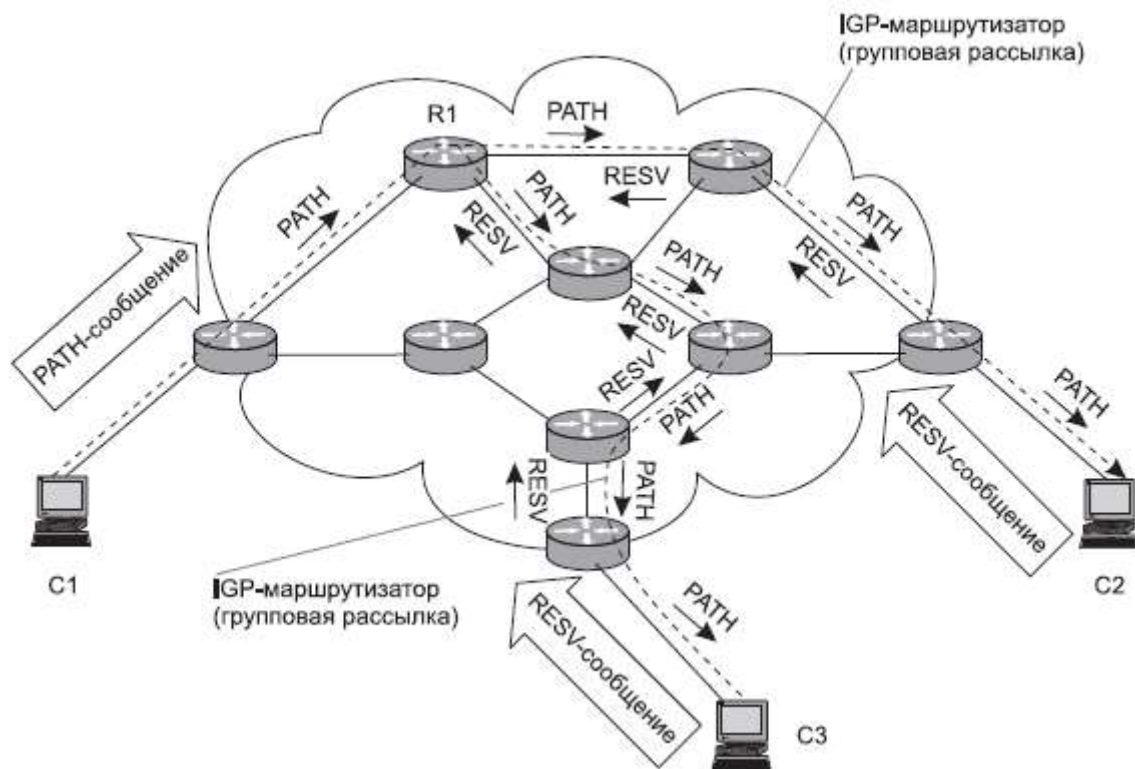


Рис. 1. Резервирование ресурсов по протоколу RSVP

1. Источник данных (компьютер C1) посылает получателям специальное сообщение **PATH**, по групповому адресу. В этом сообщении источник указывает параметры, рекомендуемые для качественного приема трафика, который он будет отправлять по данному маршруту: верхние и нижние границы пропускной способности, а также максимальные значения задержки и вариации задержки. Эти параметры составляют **спецификацию трафика источника**. Сообщение PATH передается маршрутизаторами сети в направлении ко всем указанным в групповом адресе получателям. В качестве параметров трафика применяются параметры алгоритма ведра маркеров, то есть средняя скорость и глубина ведра.
2. Каждый маршрутизатор, поддерживающий протокол RSVP, получив сообщение PATH, фиксирует «состояние пути», которое включает предыдущий адрес источника сообщения, то есть последний по времени шаг в обратном направлении (ведущий к источнику). Это необходимо для того, чтобы ответ приемника прошел по тому же пути, что и сообщение PATH. Заметим, что резервирование ресурсов маршрутизатора пока не происходит.
3. После получения сообщения PATH каждый приемник (то есть компьютеры C2 и C3) отправляет в обратном направлении маршрутизатору, от которого он получил это сообщение, **запрос на резервирование ресурсов**, то есть сообщение **RESV**. В этом сообщении содержится **спецификация запроса приемника на резервирование**, в которой указываются нужные *приемнику* параметры требуемой пропускной способности и качества обслуживания. Эта спецификация вырабатывается приемником *на основе* спецификации трафика источника, но она не обязательно повторяет указанные источником параметры. Приемник использует спецификацию источника только как рекомендацию, но окончательное решение о том, какую пропускную способность и какие параметры качества обслуживания нужно резервировать, принимает приемник. Например, приемник может решить принимать только аудио-поток от источника, а видео не принимать, тогда ему не потребуется резервировать пропускную способность, учитывающую передачу обоих потоков. Вместе со спецификацией запроса на резервирование приемник помещает в сообщение **спецификацию фильтра**, которая и определяет, к каким пакетам сеанса нужно применять данное резервирование (например, по типу транспортного протокола и номеру порта он может выделить только аудио-пакеты). Вместе спецификации запроса и фильтра представляют собой **дескриптор потока**, для которого выполняется резервирование.
4. Каждый маршрутизатор, получив сообщение RESV, проверяет, во-первых, имеются ли у маршрутизатора ресурсы, необходимые для поддержания запрашиваемой пропускной

способности и уровня QoS, а во-вторых, имеет ли пользователь право на резервирование ресурсов. Если запрос не может быть удовлетворен (из-за недостатка ресурсов или ошибки авторизации), маршрутизатор возвращает сообщение об ошибке отправителю. Если запрос принимается, то маршрутизатор посылает сообщение RSVP далее вдоль маршрута следующему маршрутизатору, а данные о требуемом уровне QoS передаются тем же механизмом маршрутизатора, которые ответственны за управление трафиком.

5. Протокол RSVP не определяет способ, с помощью которого маршрутизатор проверяет, достаточно ли у него ресурсов для принятия запроса на резервирование. Предполагается, что такая проверка может быть реализована программным обеспечением маршрутизатора, а ее детали определяются производителем маршрутизатора индивидуально. Например, если у маршрутизатора сконфигурирована очередь для обслуживания приоритетного трафика, он может вести учет выделения пропускной способности этой очереди различным потокам и при поступлении очередного запроса RSVP сравнивать наличие свободной пропускной способности с запрашиваемой. При положительном результате проверки маршрутизатор запоминает новые параметры резервирования и вычитает их из счетчиков соответствующих свободных ресурсов.
6. Когда последний в обратном направлении маршрутизатор получает сообщение RSVP и принимает запрос, то он посылает подтверждающее сообщение узлу-источнику. При групповом резервировании учитывается тот факт, что в точках разветвления дерева доставки несколько резервируемых потоков сливаются в один. Так, в маршрутизаторе R1 в рассматриваемом примере сливаются сообщения RSVP от приемников C2 и C3. Если для всех резервируемых потоков запрашивается одинаковая пропускная способность, то она требуется и для общего потока, а если запрашиваются различные величины пропускной способности, то для общего потока выбирается максимальная.
7. После установления состояния резервирования в сети источник начинает отправлять данные, которые обслуживаются на всем пути к приемнику (приемникам) с заданным качеством обслуживания.

Для того чтобы параметры резервирования можно было применить затем к трафику данных, необходимо, чтобы сообщения RSVP и пакеты данных следовали через сеть *одним и тем же маршрутом*. Это можно обеспечить, если передавать сообщения RSVP на основе тех же записей таблиц маршрутизации, которые применяются для пользовательского трафика.

Резервирование можно отменить прямо или косвенно. Прямая отмена выполняется по инициативе источника или приемника с помощью соответствующих сообщений протокола RSVP. Неявная отмена происходит по тайм-ауту: состояние резервирования имеет срок жизни, как, например, и динамические записи в таблицах маршрутизации, и приемник по протоколу RSVP должен периодически подтверждать резервирование. Если же подтверждающие сообщения перестают поступать, то резервирование отменяется по истечению его срока жизни. Такое резервирование называется мягким.

Модель резервирования ресурсов IntServ, опирающаяся на протокол RSVP, не нашла широкого применения. Основной причиной было опасение поставщиков услуг Интернета за работоспособность своих маршрутизаторов. Понятно, что применение модели IntServ в масштабах Интернета привело бы к необходимости хранения маршрутизаторами информации о состоянии резервирования для миллионов отдельных пользовательских потоков, и это намного больше, чем требует сегодняшняя модель, в соответствии с которой маршрутизаторы хранят данные только об агрегированных префиксах сетей назначения (даже эта намного более экономная модель привела в середине 2010-х годов к необходимости хранения магистральными маршрутизаторами Интернета более полумиллиона записей в таблицах маршрутизации).

Другим недостатком модели IntServ является игнорирование ею структуры Интернета, состоящей из сетей различных провайдеров. Запрос на резервирование должен пройти в общем случае через сети различных провайдеров, при этом каждый провайдер должен согласиться с тем, что его маршрутизаторы должны выделять часть своих ресурсов потоку данных неизвестного пользователя. Модель IntServ игнорирует эту проблему, не предлагая механизмов аутентификации пользователей и политики выделения ресурсов, а без этого механизм резервирования легко может использоваться злоумышленниками и привести к исчерпанию ресурсов сети.

Для преодоления этих недостатков была разработана другая модель, названная моделью дифференцированного обслуживания.

## Система дифференцированного обслуживания

Дифференцированное обслуживание (DiffServ) опирается на те же механизмы QoS, что и

интегрированное обслуживание, однако в качестве объектов обслуживания рассматриваются не отдельные потоки, а классы трафика.

**Классом трафика** называется совокупность поступающих на обработку пакетов, обладающих общими признаками, например, все пакеты голосовых приложений или все пакеты с MTU в определенных пределах.

В отличие от потока, в классах трафика пакеты не различаются по их маршрутам; это отличие иллюстрирует рис. 2. Так, маршрутизатор R1 относит все пакеты, например, пакеты, требующие приоритетного обслуживания, к одному классу. Для интерфейса i1 они представлены одним и тем же входящим агрегированным потоком, несмотря на то, что в этот поток входят пакеты с разными маршрутами: как пакеты, направляемые через маршрутизатор R3, так и пакеты, направляемые через маршрутизатор R2. Так как пакеты приоритетного класса направляются маршрутизатором по разным маршрутам, то агрегированный поток пакетов данного класса разделяется на два агрегированных потока. Маршрутизатор R2 оперирует уже другим составом приоритетного класса, поскольку в него вошли не все потоки интерфейса i1 маршрутизатора R1.

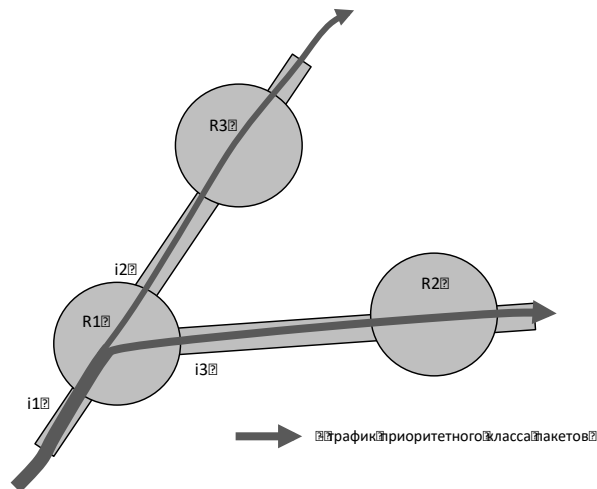


Рис. 2. Передача класса трафика маршрутизаторами

Таким образом, в агрегированный поток некоторого класса входит некоторое (возможно большое) число пользовательских потоков, иногда называемых микропотоками, подчеркивая их отличие от агрегированного потока.

Предполагается, что в класс объединяются пользовательские потоки, которые имеют близкие требования к качеству обслуживания — задержкам пакетов и вариациям задержек. Требования к пропускной способности для каждого индивидуального потока в этой модели не обеспечиваются, здесь можно говорить только о пропускной способности для класса в целом.

Обычно в сети DiffServ поддерживается дифференцированное обслуживание небольшого количество классов трафика, например двух (чувствительного к задержкам и эластичного) или трех (к первым двум прибавляется класс, требующий гарантированной доставки пакетов с определенным минимумом скорости трафика). Небольшое количество классов определяет масштабируемость этой модели, так как маршрутизаторы не должны запоминать состояния каждого пользовательского потока. Высокая степень масштабируемости Diffserv обеспечивается также тем, что каждый маршрутизатор самостоятельно принимает решение о том, как он должен обслуживать тот или иной класс трафика, не согласуя свои действия с другими маршрутизаторами. Такой подход назван *независимым поведением маршрутизаторов* (Per Hop Behavior, PHB). Так как в модели DiffServ маршруты пакетов не отслеживаются, то здесь не используется сигнальный протокол резервирования ресурсов, подобный протоколу RSVP в модели IntServ. Вместо этого маршрутизаторы сети выполняют статическое резервирование ресурсов для каждого из поддерживаемых сетью классов. Например, если маршрутизатор поддерживает несколько очередей с алгоритмом взвешенного обслуживания, и за каждой очередью закреплен определенный класс трафика, то выделяя очередям некоторый процент пропускной способности выходного интерфейса маршрутизатора, мы тем самым резервируем пропускную способность для классов трафика.

В качестве признака принадлежности IP-пакета к определенному классу в DiffServ используется метка, переносимая в поле приоритета IP-пакета (байт ToS), которое с появлением стандартов DiffServ было переопределено и названо байтом DS. Эта метка должна переносить между маршрутизаторами информацию о принадлежности пакета к определенному классу.

Как показано на рис. 3, байт DS переопределяет значения битов байта ToS, определенных ранее в соответствующих спецификациях (RFC 791, RFC 1122, RFC 1349).

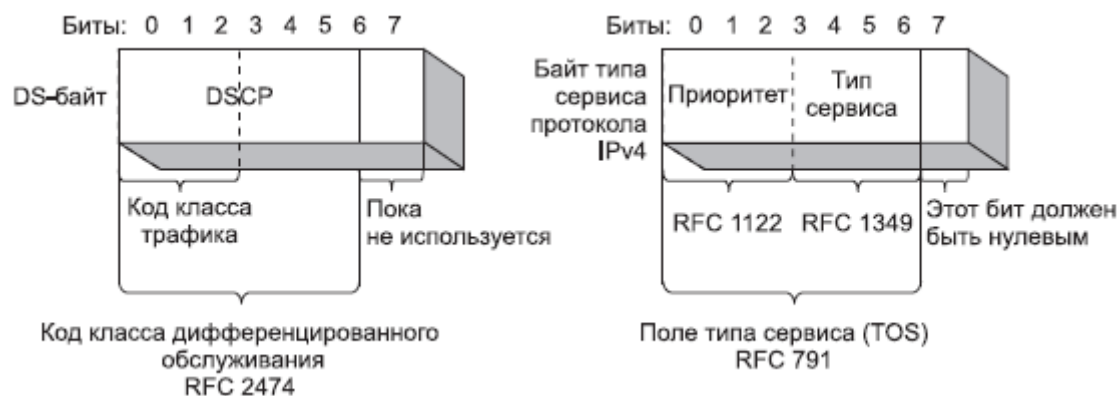


Рис. 3. Соответствие битов байта DS битам поля типа сервиса

В настоящее время используются только старшие шесть битов байта DS, причем из них только старшие три требуются для определения класса трафика (что дает не более восьми различных классов). Младший бит (из используемых шести) байта DS обычно переносит признак IN — индикатор того, что пакет «вышел» из профиля трафика и его можно отбросить или перевести в другой класс. Промежуточные два бита обычно описывают различные варианты обслуживания пакетов внутри одного класса трафика.

Маршрутизатор, поддерживающий модель DiffServ, должен обеспечивать классификацию, маркирование, измерение и кондиционирование трафика, его обслуживание в приоритетной или взвешенной очереди и сглаживание.

Хотя маркировкой пакетов может заниматься каждый маршрутизатор сети, в модели дифференцированного обслуживания основным вариантом считается маркировка пакетов на границе сети, поддерживающей эту модель и находящейся под административным контролем одной организации. Такая сеть называется DiffServ-доменом. При выходе пакетов за пределы DiffServ-домена маркировка снимается, так что другой домен может назначить ее заново. Пограничные маршрутизаторы DiffServ-домена исполняют роль контрольно-пропускных пунктов домена, проверяя входящий трафик и определяя, имеет ли он право на дифференцированное обслуживание.

Модель DiffServ подразумевает существование соглашения об уровне обслуживания (SLA) между доменами с общей границей. Это соглашение определяет критерии политики предоставления сервиса, профиль трафика, а также гарантируемые параметры QoS. Ожидается, что трафик будет формироваться и сглаживаться в выходных точках домена в соответствии с SLA, а во входной точке домена кондиционироваться в соответствии с правилами политики. Любой трафик «вне профиля» (например, выходящий за верхние границы полосы пропускания, указанной в SLA) не получает гарантий обслуживания (или же оплачивается по повышенной стоимости в соответствии с SLA). Правила политики предоставления сервиса могут включать время дня, адреса источника и приемника, транспортный протокол, номера портов. В том случае, когда соблюдаются правила политики, и трафик удовлетворяет оговоренному профилю, DiffServ-домен должен обеспечить при обслуживании этого трафика параметры QoS, зафиксированные в SLA.

На сегодняшний день в IETF разработано два стандарта пошагового продвижения пакетов для схемы PNB, которые представляют два разных варианта обслуживания.

**Быстрое продвижение** (Expedited Forwarding, EF) характеризуется значением кода 10111 в байте DS (десятичное значение 46) и представляет собой высший уровень качества обслуживания, обеспечивая минимум задержек и вариаций задержек.

**Гарантированная доставка** (Assured Forwarding, AF) характеризуется четырьмя классами трафика и тремя уровнями отбрасывания пакетов в каждом классе — всего получается 12 различных вариантов трафика. Каждому классу трафика выделяется определенные минимум пропускной способности и

размер буфера для хранения его очереди. Трафик, параметры которого превышают указанные в профиле, доставляется с меньшей степенью вероятности, чем трафик, удовлетворяющий условиям профиля. Это означает, что качество его обслуживания может быть понижено, но он не обязательно будет отброшен.

На основе этих **пошаговых спецификаций** и соответствующих соглашений об уровне обслуживания (SLA) могут быть построены **сервисы** для конечных пользователей «из конца в конец» — это EF- и AF-сервис соответственно. Предполагается, что провайдер наряду с EF- и AF-услугами предоставляет и стандартные услуги IP-сети, то есть услуги «по возможности» без гарантий пропускной способности и параметров QoS.

Основное назначение EF-сервиса — обеспечение качества обслуживания, сопоставимого с качеством обслуживания выделенных каналов, поэтому этот сервис называется также *сервисом виртуальных выделенных каналов*.

Поскольку EF-сервис допускает полное вытеснение другого трафика (например, при реализации EF-сервиса с помощью приоритетной очереди), то его реализация должна включать некоторые средства ограничения влияния EF-трафика на другие классы трафика, например, путем ограничения скорости EF-трафика на входе маршрутизатора по алгоритму ведра маркеров. Максимальная скорость EF-трафика и, возможно, величина пульсаций должны устанавливаться сетевым администратором.

Четыре класса AF-сервиса ориентированы на гарантированную доставку, но без минимизации уровня задержек пакетов, как это оговорено для EF-сервиса. Гарантированная доставка выполняется в том случае, когда входная скорость трафика не превышает отведенной данному классу минимальной пропускной способности. Реализация классов AF-трафика хорошо сочетается с EF-сервисом — EF-трафик может обслуживаться по приоритетной схеме, но с ограничением интенсивности входного потока. Оставшаяся пропускная способность распределяется между классами AF-трафика в соответствии с алгоритмом взвешенного обслуживания, который обеспечивает необходимую пропускную способность, но не минимизацию задержек. Реализация AF-сервиса предполагает (но не требует) взвешенного обслуживания для каждого класса с резервированной полосой пропускания, а также применения обратной связи (в форме RED).

Относительная простота определяет недостатки дифференцированного обслуживания. Главным недостатком является сложность предоставления количественных гарантий качества обслуживания «из конца в конец». Поясним это на примере сети, изображенной на рис. 4.

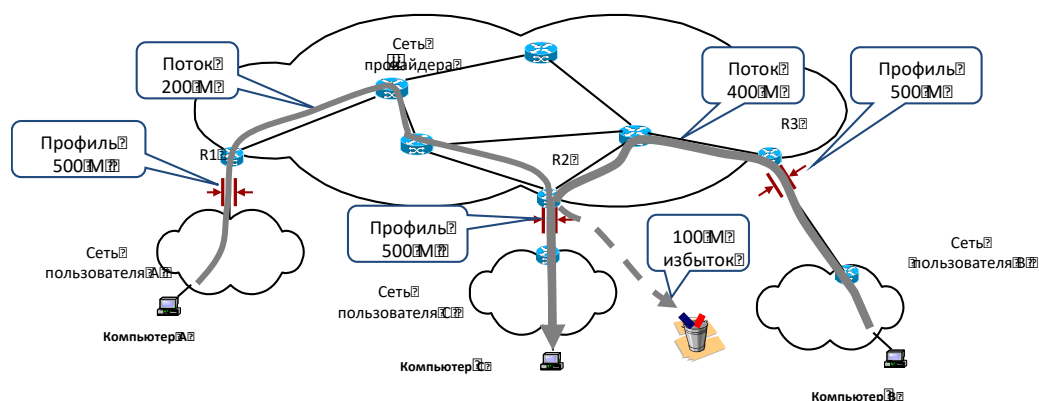


Рис. 4. Неопределенность уровня обслуживания в модели DiffServ

В этом примере сеть провайдера соединена с тремя сетями корпоративных пользователей: А, В и С. Сеть провайдера предоставляет пользователям услуги двух классов DiffServ: быстрого продвижения EF и обслуживания «с максимальными усилиями». У провайдера заключено соглашение SLA со своими клиентами, в котором он гарантирует услугу EF своим пользователям в том случае, если скорость такого трафика от клиента не превышает 500 Мбит/с. Трафик EF-услуги обслуживается маршрутизаторами провайдера с помощью приоритетных очередей. Для того чтобы приоритетный трафик клиентов оставил достаточно ресурсов маршрутизаторов для обслуживания трафика класса «с максимальными усилиями», провайдер применил профилирование трафика на интерфейсах, соединяющих его маршрутизаторы с маршрутизаторами сетей пользователей. Профилирование проводится с уровнем скорости трафика CIR = 500 Мбит/с, точно в соответствии с соглашениями SLA. Профилирование применяется в обоих направлениях, чтобы защитить как свою сеть, так и сети пользователей от избыточного (не предусмотренного SLA) приоритетного трафика.

Как видно из рисунка, такое профилирование не всегда работает так, как задумано. В нашем примере компьютер А посылает на компьютер С поток данных класса EF со средней скоростью 200 Мбит/с. Эта скорость ниже, чем оговоренный предел 500 Мбит/с, поэтому профилирующий элемент входного интерфейса маршрутизатора R1 пропускает все пакеты потока. Компьютера В также посылает поток данных класса EF компьютеру С, его скорость равна 400 Мбит/с, что также устраивает профилирующий элемент маршрутизатора R3.

Однако когда эти два потока поступают на интерфейс маршрутизатора R2, соединяющий его с сетью пользователя С, выясняется, что их суммарная скорость пакетов класса EF уже превышает порог профилирования, так как она становится равной 600 Мбит/с. Если профилирующий элемент маршрутизатора R2 настроен стандартным образом, то он просто будет отбрасывать пакеты, не удовлетворяющие условиям алгоритма ведра маркеров с параметром скорости 500 Мбит/с. Поток отброшенных пакетов будет иметь скорость 100 Мбит/с.

Провайдер может попытаться смягчить ситуацию, изменив конфигурацию профилирующего элемента маршрутизатора R2 и настроив его не на отбрасывание пакетов-нарушителей, а на маркирование их как пакетов класса «по возможности» и обслуживание их соответствующим образом. Однако это будет нарушением соглашения SLA, хотя каждый из пользователей его не нарушал — ведь и пользователь А, и пользователь В направляли трафик класса EF со скоростью, не превышающей 500 Мбит/с. Отказ от профилирования приоритетного трафика в направлении к пользовательским сетям также не является решением проблемы, так как может привести к полному вытеснению из сети трафика «по возможности» (например, при сложении многих потоков приоритетного трафика на одном из интерфейсов маршрутизатора).

Описанная проблема в рамках модели DiffServ неразрешима. Она является следствием того, что в этой модели нет процедуры проверки наличия ресурсов вдоль пути следования трафика, той процедуры, которую в модели IntServ выполняет протокол RSVP. Таким образом, мы видим, что решив проблему масштабируемости (за счет агрегирования потоков данных в классы), разработчики модели DiffServ лишили систему обеспечения качества обслуживания определенности — ее работа зависит от того, насколько удачно в сети провайдера распределились потоки приоритетного трафика. Провайдер может смягчить эту проблему за счет мониторинга распределения трафика клиентов по различным маршрутам в пределах своей сети и соответствующим подбором скоростей интерфейсов и профилей.

Недостатки обеих моделей — IntServ и DiffServ — объясняют отсутствие услуг QoS, предоставляемых в масштабах всего Интернета. Отдельные провайдеры предлагают такие услуги, но только в пределах своей сети, где они могут применять как IntServ, так и отдельные механизмы QoS в соответствии с собственной нестандартной моделью.

**(S) О поддержке функций QoS маршрутизаторами смотрите также на сайте раздел «Вопросы реализации маршрутизаторов. Функциональная схема. Операционная система Cisco IOS. Классификация маршрутизаторов»**