Министерство связи Российской Федерации
Санкт-Петербургский Государственный Университет Телекоммуникаций имени проф. М. А. Бонч-Бруевича

КАФЕДРА СЕТЕЙ СВЯЗИ

ДИПЛОМНАЯ РАБОТА

“Анализ и сравнение подходов к обеспечению гарантированного качества обслуживания в мультисервисных сетях”

Студентка-дипломница
5курс, СК-85 **Осипова Н.А.**

Руководитель **Гольдштейн А.Б.**

Санкт-Петербург
2003г.

Содержание:

Введение....4
Глава 1. Основы технологии VoIP. Проблемы обеспечения QoS....7
1.1. Технология Voice over IP....7
1.1.1. Актуальность темы дипломной работы....7
1.1.2. Конвергенция сетей связи....8
1.1.3. Качество сервиса....9
1.1.4. IP-телефония....11
1.1.4.1. Экономическая эффективность IP-телефонии....11
1.1.4.2. Основные сценарии IP-телефонии....12
1.1.4.2.1. Компьютер – компьютер....12
1.1.4.2.2. Компьютер – телефон (телефон – компьютер)....12
1.1.4.2.3. Телефон – телефон....13
1.1.5. Различные подходы к построению сетей IP-телефонии....14
1.1.5.1. Построение сети по рекомендации H.323....14
1.1.5.2. Сеть на базе протокола SIP....18
1.1.5.3. Сеть на базе MGCP....19
1.2.Оценки качества обслуживания....20
1.2.1. Параметры качества работы сети IP-телефонии....20
1.2.1.1. Задержка речевых пакетов....22
1.2.1.2. Джиттер задержки пакетов....23
1.2.1.3. Потери речевых пакетов....23
1.2.1.4. Готовность сети....24
1.2.1.5. Эхо....24
1.2.2. Методы оценки качества передачи речи при IP-телефонии....26
1.2.2.1. Метод MOS-Mean Opinion Score....26
1.2.2.2. Метод Quality Rating....27
1.2.2.3. Метод PSQM-Perceptual Speech Quality Measurement....27
1.2.2.4. Метод ICPIF-Calculated Planning Impairment Factor....28
1.2.3. Кодеки и оценка качества кодеков....30
1.2.4. Механизмы обеспечения качества сервиса....33
1.2.4.1. Обслуживание очередей....34
1.2.4.1.1. Стратегия FIFO....34
1.2.4.1.2. Очередь с приоритетами....34
1.2.4.1.3. Class-Based Queuing (CBQ)....35
1.2.4.1.4. Взвешенные справедливые очереди - Weighted Fair Queuing(WFQ)....36
1.2.5. Сетевые аспекты обеспечения качества обслуживания IP-телефонии....36
1.2.6. Дифференциальное и интегральное обслуживание....38
1.2.6.1. Архитектура QoS Int-Serv....38
1.2.6.2. Архитектура QoS Diff-Serv....41
Глава 2. Технология MPLS....44
2.1. Принцип коммутации....46
2.2. Метки и правила их использования....49
2.3. Стек меток....52
2.4. Привязка и распределение меток....53
2.5. Построение коммутируемого маршрута....54
2.6. Качество обслуживания в сетях MPLS....56
Глава 3. Протокол RSVP....57
3.1. Работа протокола RSVP....57
3.2. RSVP-компоненты....59
3.3. RSVP-сообщения....60
3.4. Стили резервирования....62

3.4.1. Индивидуальное резервирование 62
3.4.2. Общее резервирование....63
3.5. Типы услуг 65
3.5.1. Регулируемая нагрузка 65
3.5.2. Гарантированная битовая скорость 66
Глава 4. Сравнение технологий MPLS и RSVP 67
4.1. Затраты на внедрение технологий 67
4.2. Масштабируемость 67
4.3. Перегрузки 69
4.4. Время установления соединения 70
4.5. Время пересылки пакетов....71
4.6. Затраты ресурсов сети на обслуживание 71
4.7. Пропускная способность 71
4.8. Классификация пакетов 72
4.9. Установление маршрута 72
4.10. Совместимость с технологиями коммутации каналов....72
4.11. Построение сети 73
4.12. Безопасность 73
4.13. Обслуживание мультикастных приложений 76
4.14. Действия протоколов RSVP и MPLS при обрыве связи 77
4.15. Вывод....78
Список литературы: 80

Введение

Вопросы качества обслуживания – одни из наиболее актуальных при предоставлении любых услуг связи. Высокое качество обслуживания представляет интерес не только для конечного пользователя, но и для самого поставщика услуг. Пользователи стремятся использовать высококачественные сети, а это означает увеличение годовых доходов операторов. Задача оператора сети связи – найти оптимальное решение для удовлетворения, как собственных интересов, так и интересов потребителей услуг.

Согласно Рекомендации МККТ G.106 *качество обслуживания* абонента (Quality of Service) можно определить как степень удовлетворенности абонента обслуживанием на сети.

Расплывчатость определения связана с тем, что оно в значительной мере определяется типом приложения. В каждом конкретном случае определение качества обслуживания становится гораздо более четким. При передаче голосового трафика через сеть под качеством обслуживания чаще всего понимают гарантии того, что голосовые пакеты будут доставляться сетью с задержкой не более N мс, при этом вариация задержки не превысит M мс, и эти характеристики станут выдерживаться сетью с вероятностью 0,95 на определенном интервале времени.

Где *задержка (delay)* – промежуток времени, затрачиваемый на то, чтобы речевой сигнал прошел от говорящего до слушающего.

Вариация задержки – различие интервалов между моментами прибытия пакетов.

Качество обслуживания определяется совокупностью таких свойств, как обеспеченность обслуживанием, эксплуатационная пригодность обслуживания, эффективность обслуживания, целостность обслуживания.

Обеспеченность обслуживанием (service support) - возможность органов связи предоставлять абоненту различные виды обслуживания и оказывать помощь в их использовании.

Эксплуатационная пригодность обслуживания (service operability) - характеризует возможность абонента успешно и легко управлять процессом обслуживания.

Эффективность обслуживания (serve ability) – способность сети в определенных условиях эксплуатации по требованию абонента предоставлять обслуживание на требуемое время в пределах заданных допусков на показатели качества передачи сигналов без его существенного ухудшения.

Из перечисленных свойств основным свойством, характеризующим качество обслуживания на сети как системе массового обслуживания, является эффективность обслуживания. Составляющими этого свойства являются доступность и непрерывность обслуживания.

Доступность обслуживания (service accessibility) – способность сети в определенных условиях эксплуатации по требованию абонентов предоставлять обслуживание с допустимым качеством передачи сигналов без перерывов в течение требуемого промежутка времени.

В свою очередь доступность обслуживания складывается из таких свойств, как доступность сети и доступность соединения.

Доступность сети (network accessibility) – характеризует возможность вызова абонентом местной станции (в которую он подключен) и получения сигнала готовности станции к приему адресной информации.

Доступность соединения (connection accessibility) – характеризует способность сети в определенных условиях эксплуатации после получения от абонента необходимой адресной информации устанавливать соединение с допустимым качеством передачи сигналов.

В соответствии с Рекомендацией G.180 доступность соединения оценивается таким показателем, как вероятность отказа в соединении на сети электросвязи.

Вероятность отказа в соединении на сети электросвязи (network connection failure probability) – это вероятность того, что по требованию абонента коммутируемое соединение не может быть установлено по правильному адресу при заданном качестве передачи сигналов электросвязи по соединительному тракту.

Непрерывность установленного соединения (retain ability of an established connection) характеризует возможность продолжения этого соединения в определенных условиях эксплуатации в течение заданного периода времени без перерывов с заданным качеством передачи сигналов.

Одним из основных показателей непрерывности соединения (cut off call probability) является вероятность того, что установленное соединение будет нарушено по причинам, отличным от преднамеренного отбоя абонентами, участвующими в соединении, например по причине отказа оборудования АСК или систем передачи, участвующих в соединении.

Приведем еще несколько определений, которые впоследствии будут встречаться в данной дипломной работе:

Сеть – распределенная среда, состоящая из большого количества устройств для поддержки различных технологий и протоколов.

Из конца в конец (End-to-end (e2e)) – путь потока данных от одного узла до другого.

Поток данных (flow) – последовательность пакетов, имеющих некоторые общие признаки, например, адрес узла-источника.

Интернет – всемирная сеть компьютерных сетей. Используя интернет можно обмениваться цифровой информацией.

Пропускная способность (throughput) – максимально возможная скорость передачи данных.

Компьютерная телефония - это отрасль, специализирующаяся на приложении компьютерного интеллекта к осуществлению и приему телефонных вызовов, а также, к другим сложным взаимодействиям. В первую очередь это реализация голосового соединения по каналам вычислительных сетей.

IP-телефония - это общий термин, обозначающий передачу голоса и факса (а также связанные с этим сервисы) частично или полностью через пакетные сети на основе протокола IP. Понятие «IP-телефония» распространяется также и на те случаи, когда голос и факс передаются вместе с другими видами информации, в частности с текстом и изображением».

Internet-телефония - более узкий термин, соответствующий случаю, когда услуги IP-телефонии частично или полностью осуществляются через Internet.

Глава 1. Основы технологии VoIP. Проблемы обеспечения QoS

1.1. Технология Voice over IP

1.1.1. Актуальность темы дипломной работы

Данная дипломная работа рассматривает новую технологию, все более завоевывающую популярность в мире - технологию передачи речевой информации по сетям с маршрутизацией пакетов IP - Voice over IP (VoIP) или IP-телефонии.

Данная технология представляет собой компрессию голосового сигнала с последующей передачей по цифровым каналам передачи данных с использованием протокола IP. Частным случаем IP-телефонии является Интернет-телефония, при которой в качестве передающей сети используется сеть Интернет. При том, что IP-телефония - сравнительно новая технология, она уже успела зарекомендовать себя вполне перспективной и жизнеспособной. Согласно прогнозам аналитиков, в мире ожидается значительный (в ближайшие годы в десятки раз) рост речевого трафика, переносимого по сетям с маршрутизацией пакетов.

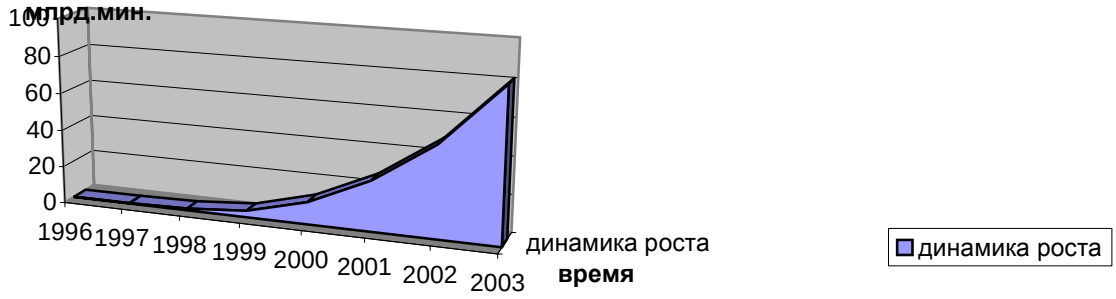


рис.1.1 Прогнозируемая динамика развития мирового трафика IP-телефонии

Объем рынка услуг IP-телефония сегодня составляет примерно миллиард долларов, а в течение ближайших нескольких лет, по разным прогнозам, вырастет до величин порядка 8,5-24 млрд. долл. Несмотря на разброс оценок величины рынка, большинство аналитиков сходятся на том, что IP-телефония является магистральным путем развития телекоммуникационной индустрии. Дополнительно IP-телефония вносит новые услуги в сферу телекоммуникаций:

речевые и видеоконференции, одновременный доступ к приложениям, быстрый поиск абонента и другие.

IP - всего лишь протокол, который используется для передачи по сети Интернет оцифрованной и сжатой в «пакеты» определенного объема электронной информации. В принципе, после соответствующей обработки таким способом на любые досягаемые для паутины Интернета расстояния может передаваться любой тип информации (голосовая, видео, компьютерные данные), так что собственно телефония является лишь одной составляющей общего процесса, однако, материал данной дипломной работы ограничивается только ею.

Преимущества IP-телефонии неоспоримы. В первую очередь это низкая стоимость передачи информации. А также это универсальность обработки информации независимо от ее исходного вида, а, следовательно, использование одних и тех же каналов для передачи информации разного типа. При этом популярность технологий пакетной передачи речи растет, причем как среди операторов, так и среди корпоративных пользователей. В последние годы немало компаний успешно внедрили эти технологии для организации каналов дальней телефонной связи.

Все вышеизложенное показывает, что выбранная тема дипломной работы актуальна и важна.

1.1.2. Конвергенция сетей связи

В 1996 году в США трафик передачи данных впервые превысил речевой. Это послужило толчком к началу эры интегрированных решений и конвергенции всех видов связи.

К конвергенции подталкивали следующие обстоятельства:

- Желание упростить и удешевить создание и эксплуатацию сетей, в которых нуждаются сотрудники организаций;
- Стремление использовать для аудиопотоков более широкий набор типов среды передачи;
- Стремление снизить стоимость телефонных переговоров, особенно международных.

Традиционные сети передачи данных базировались на магистралях с коммутацией каналов, предназначенных для телефонного трафика. При новом подходе - все наоборот: телефония будет надстраиваться над инфраструктурой сети передачи данных. Протокол IP “де-факто” стал стандартом для передачи мультимедийной информации.

Смещение центра тяжести в область передачи данных поставило вопрос о поиске удобного встраивания речи в мультимедийный цифровой поток. Причина популярности IP как раз и заключается в его восприимчивости к требованиям со стороны не только услуг передачи данных, но и приложений реального времени. Примером может служить успешно реализованная технология передачи речевой информации по сетям с маршрутизацией пакетов - Voice over IP (VoIP) или IP- телефония.

Рынок IP-телефонии неуклонно растет, так как интеграция голоса и данных позволяет создавать единую сеть коммуникаций, обслуживание которой может производить один администратор. А также потому, что IP-телефония существенно сокращает расходы за междугородние и международные разговоры.

1.1.3. Качество сервиса

Традиционная телефонная сеть была создана таким образом, чтобы гарантировать высокое качество услуги даже при больших нагрузках. IP-телефония, напротив, не гарантирует качества, причем при больших нагрузках оно значительно падает. Отсутствие гарантированного качества обслуживания при передачи речи по сетям с маршрутизацией пакетов компенсируется появлением таких технологий, как многопротокольная коммутация по меткам – Multiprotocol Label Switching (MPLS), протокол резервирования ресурсов – Resource Reservation Protocol (RSVP), дифференциальное обслуживание разнотипного трафика – Differentiated Services (DiffServ).

Поддержка механизмов качества обслуживания – Quality of Service (QoS) позволяет предоставлять ресурсы сети тем приложениям, которым они нужны в наибольшей степени. Например, можно резервировать определенную полосу пропускания под голосовые пакеты, а данным, менее критичным к задержкам (передача файлов по сети), назначать меньший приоритет.

Избежать заторов в IP-сетях, вызванных разнородным трафиком, можно лишь за счет дифференциации качества обслуживания.

Для реализации механизмов QoS в заголовке IP-пакета предусмотрено поле типа сервиса размером 8 бит (Type of Service, ToS), которое задает характер обработки пакета в процессе транспортировки последнего.

Рис.1.2 Поле типа сервиса

Первые три бита этого поля образуют подполе приоритета пакета (precedence). Приоритет может быть от самого низкого-0 до самого высокого-7. Поле тип сервиса содержит также три бита, определяющие критерий выбора маршрута. Выбор осуществляется между тремя альтернативами: малой задержкой, высокой достоверностью и высокой пропускной способностью. Установленный бит D (delay) говорит о том, что маршрут должен выбираться для минимизации задержки доставки данного пакета, бит T – для максимизации пропускной способности, а бит R – для максимизации надежности доставки. Во многих сетях улучшение одного из этих параметров связано с ухудшением другого. На практике приходится делать выбор между малыми задержками, высокой пропускной способностью и высокой надежностью. Редко имеет смысл устанавливать хотя бы два из этих трех критериев выбора маршрута. Зарезервированные биты имеет нулевое значение.

Распределение разрядов в поле типа сервиса представлено в табл.1, а назначение различных комбинаций — табл.2.

Таблица 1

Распределение разрядов в поле типа сервиса IP-пакета

Разряды	Назначение
0-2	Приоритет (Precedence)
3	Задержка (Delay)
4	Пропускная способность (Throughput)
5	Надежность (Reliability)
6-7	Зарезервировано

Таблица 2

Назначение различных комбинаций в поле типа сервиса IP-пакета

Разряды	Значение	Описание
0-2	111	Управление сетью (Network Control)
	110	Межсетевое управление (Internetwork Control)

Разряды	Значение	Описание
	101	CRITIC/ECP
	100	Быстрее, чем мгновенно (Flash Override)
	11	
	10	Немедленно (Immediate)
	1	Мгновенно (Flash)
	0	Обычно (Routine)
3	1	Малая задержка
	0	Нормальная задержка
4	1	Высокая пропускная способность
	0	Нормальная пропускная способность
5	1	Высокая надежность
	0	Обычная надежность

1.1.4. IP-телефония

IP-телефония позволяет установить телефонную связь через IP-сеть по выделенному виртуальному каналу. Наиболее часто в качестве IP-сети используют Интернет. Основной принцип IP телефонии заключается в оцифровке и компрессии голоса (обычно по алгоритмам G711, G729, G729a или G723.1) с последующей пакетизацией и передачей по IP сетям.

Для организации связи можно воспользоваться компьютером со специальным программным обеспечением, IP-телефоном или обычным телефоном.

1.1.4.1. Экономическая эффективность IP-телефонии

Определяющим фактором привлекательности IP-телефонии для конечных потребителей безусловно является дешевизна междугородных и международных телефонных переговоров через IP-сети по сравнению с традиционной телефонной связью. Так, например, одна минута обычного телефонного разговора с США для конечного потребителя в Москве стоит сегодня порядка $1,5 - $2,3, а через Интернет обходится ему $0,25 - $0,8, т.е. на каждом пятиминутном разговоре потребитель услуг IP-телефонии экономит от 6 до 7,5 долларов США в зависимости от расстояния и используемой сети.

Кроме того, процесс вызова абонента для конечного потребителя услуг IP-телефонии практически не отличается от традиционной телефонной связи. Человек, который хочет позвонить через IP-сеть, на обыкновенном телефонном аппарате вместо традиционной восьмерки набирает местный номер (который является номером ближайшего телефонного сервера) и слышит голосовое

приглашение с предложением набрать телефонный номер вызываемого абонента (или сначала идентификационный номер, если это оговорено договором). После набора номера система IP-телефонии (через второй сервер) соединяет потребителя с телефоном вызываемого абонента. Если разговор не может состояться, звонящий будет голосом проинформирован о причине невозможности соединения (например, «вызываемый номер занят», «все линии на удаленном телефонном сервере заняты», «удаленный телефонный сервер недоступен», «неверно набран номер» и т.п.) Система также передает звонящему абоненту такие телефонные сигналы, как сигнал «вызов», «занято» и пр. Таким образом, звонящий слышит привычные ему реальные сигналы телефонной сети.

1.1.4.2. Основные сценарии IP-телефонии

Могут быть установлены различные типы телефонных соединений:

- Соединение PC-PC;
- Соединение PC- телефон;
- Соединение телефон-телефон.

1.1.4.2.1. Компьютер – компьютер

Два компьютера, подключенные к сети Интернет, могут общаться без посредников. Из общей схемы исчезнет шлюз, поскольку необходимость преобразования сигнала отпала (если быть точнее, в качестве шлюза выступает некая программа – интернет-телефон, запушенная на обоих компьютерах). Данные сразу передаются по стандартным протоколам Интернета, поэтому помехи проникнуть в пакет не могут. Единственное негативное воздействие помех – задержка пакетов. Повлиять на качество звука можно, лишь купив более быстрый модем и выбрав провайдера с мощными каналами связи.

1.1.4.2.2. Компьютер – телефон (телефон – компьютер)

Установив на свой компьютер программу интернет-телефонии, пользователь может связаться с человеком, не владеющим компьютером. Компьютер существенно расширит возможности связи и облегчит дозвон. Этот

сценарий находит применение в разного рода справочно-информационных службах Интернет, в службах сбыта товаров или в службах технической поддержки.

1.1.4.2.3. Телефон – телефон

Звонок по такой схеме внешне мало, чем отличается от обычного телефонного звонка по телефонной сети общего пользования (ТфОП). Последовательность действий такова: сначала набирается телефонный номер ближайшего шлюза интернет-телефонии (предварительно нужно подписаться на его услуги); затем после переключения телефонного аппарата в тоновый режим, набирается номер абонента; вводится идентификационный номер – и далее следует разговор.

До шлюза сигнал передается, как и обычные телефонные звонки. При этом в него (как и в любой другой телефонный сигнал) могут примешаться помехи. На уровень задержек, а, следовательно, на комфортность и качество разговора в режиме телефон-телефон влияние оказывает лишь пропускная способность (предел суммарной скорости передачи и приема информации по всем речевых и видео канала) линий связи провайдера интернет-телефонии и загруженность сети Интернет на маршруте следования пакетов. В настоящий момент, технологически проблема качества звука решается путем оптимизации задержек на пути следования сигнала. Из нескольких возможных система выбирает наименее загруженные маршруты; где это допустимо, повышается приоритет голосовых пакетов. За счет этих мер паузы в разговоре удается сделать практически незаметными.

В общем виде схема связи выглядит так:
Абонент 1 – [локальная телефонная сеть 1] – [шлюз 1] – **Интернет** –
[шлюз 2] –[локальная телефонная сеть 2] – Абонент 2

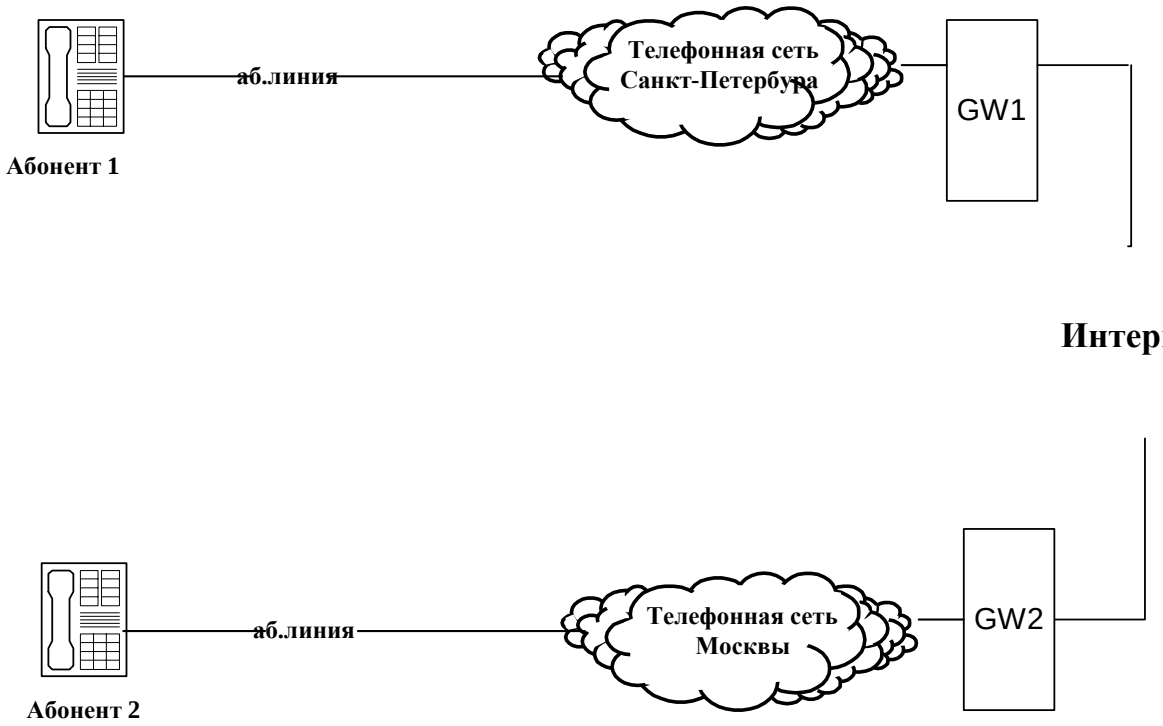


Рис1.3 Схема соединения абонентов ТФОП по сети IP

1.1.5. Различные подходы к построению сетей IP-телефонии

1.1.5.1. Построение сети по рекомендации H.323

Первый в истории подход к построению сетей IP-телефонии на стандартизованной основе был предложен Сектором Международного Союза Электросвязи, разрабатывающим рекомендации в области телекоммуникаций, (International Telecommunications Union Telecommunication Standardization Sector-ITU-T) в рекомендации H.323.

Рекомендация H.323 предусматривает довольно сложный набор протоколов, который предназначен не просто для передачи речевой информации по IP-сетям с коммутацией пакетов. Его цель – обеспечить работу мультимедийных приложений в сетях с негарантированным качеством обслуживания. Речевой трафик – это только одно из приложений H.323, наряду с видеоинформацией и данными.

Вариант построения сетей IP-телефонии, предложенный Международным союзом электросвязи в рекомендации H.323, хорошо подходит тем операторам

местных телефонных сетей, которые заинтересованы в использовании сети с коммутацией пакетов (IP-сети) для предоставления услуг междугородной и международной связи. Рекомендация H.323 основывается на абонентской сигнализации Q.931. Процедура установления соединения в сетях H.323 базируется на рек. Q.931 и практически идентична той же процедуре в сетях ISDN. Протокол RAS, входящий в семейство протоколов H.323, обеспечивает контроль использования сетевых ресурсов, поддерживает аутоинтефикацию пользователей и может обеспечить начисление платы за услуги.

Основными устройствами сети на базе рекомендации H.323 являются: терминал (Terminal), шлюз (Gateway), привратник (Gatekeeper) и устройство управления конференциями (Multiprotocol Control Unit – MCU). В зависимости от технической реализации эти элементы размещаются на отдельных компьютерах под управлением ОС Windows или Unix, равно как и могут быть интегрированы внутри сетевого оборудования.

Терминал H.323 – оконечное устройство пользователей сети IP-телефонии, которое обеспечивает двухстороннюю речевую (мультимедийную) связь с другим терминалом H.323, шлюзом или устройством управления конференциями.

В общем случае *шлюз IP* представляет собой систему голосовых и интерфейсных плат, сообщающихся между собой посредством специальной шины SCBus. Голосовые (DSP) платы отвечают за преобразование цифрового или аналогового сигнала голоса в IP-пакеты, сжатие и передачу их в сеть (и обратное преобразование), а интерфейсные платы служат «мостом» между телефонной сетью и сетью передачи данных. Таким образом, в функции шлюза входит:

- Обеспечение взаимодействия между оконечными пользователями;
- Преобразование (оцифровка) голосовой информации;
- Сжатие, восстановление, кодировка потоков цифровой информации;

Для обеспечения приемлемого качества связи шлюз IP-телефонии следует рассматривать по следующим параметрам:

- Быстродействие;
- Качество и скорость сжатия/восстановления;

- Возможности восстановления утерянных пакетов.

Сервер доступа (gatekeeper), или *привратник*, — ключевое звено в архитектуре сети IP-телефонии. Именно он отвечает за установку и разрыв соединений, обработку сигнала вызывающего абонента, выбор вызываемого (терминируемого) VoIP-шлюза, маршрутизацию вызова в сети. Привратник осуществляет связь с базой данных учета абонентов. Обычно gatekeeper выполнен в виде программного модуля, устанавливаемого на сервер, который работает под управлением ОС Unix или Windows. Таким образом, привратник выполняет следующие функции:

- Регистрация оконечных и других устройств;
- Контроль доступа пользователей системы к услугам IP-телефонии при помощи сигнализации RAS;
- Преобразование alias-адреса вызываемого пользователя (объявленного имени абонента, телефонного номера, адреса электронной почты) в транспортный адрес сетей с маршрутизацией пакетов IP (IP адрес + номер порта TCP);
- Контроль, управление и резервирование пропускной способности сети;
- Ретрансляция сигнальных сообщений H.323 между терминалами.

Устройство управления конференциями обеспечивает возможность связи между тремя или более участниками. Рекомендация H.323 предусматривает три вида конференций: централизованная (т.е. управляемая MCU, с которым каждый участник конференции соединяется в режиме точка-точка), децентрализованная (когда каждый участник конференции соединяется с остальными ее участниками в режиме точка-группа точек) и смешанная.

Существует еще один элемент сети H.323 – *прокси-сервер H.323*, т.е. сервер-посредник. Этот сервер функционирует на прикладном уровне и может проверять пакеты с информацией, которой обмениваются два приложения. Прокси-сервер может определять, с каким приложением (H.323 или другим) ассоциирован вызов, и осуществлять нужное соединение. Прокси-сервер выполняет следующие ключевые функции:

- Подключение через средства коммутируемого доступа или локальные сети терминалов, не поддерживающих протокол резервирования ресурсов

(RSVP). Два таких прокси-сервера могут образовать в IP-сети туннельное соединение с заданным качеством обслуживания;

- Маршрутизацию трафика H.323 отдельно от обычного трафика данных;
- Обеспечение совместимости с преобразователем сетевых адресов, поскольку допускается размещение оборудования H.323 в сетях с пространством адресов частных сетей;
- Защита доступа - доступность только для трафика H.323.

H.323 - первый стандарт для передачи голоса и видео по пакетным сетям, который представляет собой сложный набор протоколов, описывающий разные аспекты этой задачи. Взаимодействие H.323 и IP можно условно изобразить в виде следующего протокольного стека:

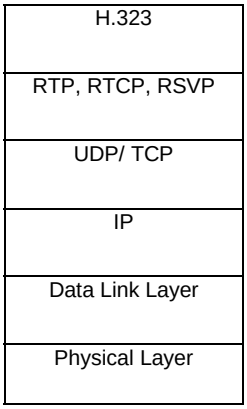


Рис.1.4 Протокольный стек взаимодействия H.323 и IP

В настоящий момент существует два противоположных подхода к внедрению систем IP-телефонии. Один из них (революционный, как его принято называть) заключается в том, что заказчику требуется отказаться от традиционной ТфОП и использовать только локальную сеть, которая по своим каналам обеспечивает передачу голоса между абонентами. Второй подход (эволюционный), наоборот, предполагает сохранение существующей структуры и одновременное добавление нового оборудования для расширения функциональности телекоммуникационной системы. В каждом конкретном случае следует анализировать внешние условия, а также требования к

комплексу оборудования, с той целью, чтобы обеспечить выбор правильной стратегии при внедрении систем IP-телефонии.

1.1.5.2. Сеть на базе протокола SIP

Второй подход к построению сетей IP-телефонии, предложенный рабочей группой MMUSIC комитета IETF (Internet Engineering Task Force- инженерная группа, которая занимается решением ближайших технических проблем Internet) в документе RFC 2543, основан на использовании протокола SIP-Session Initiation Protocol. SIP представляет собой текст-ориентированный протокол, который является частью глобальной архитектуры мультимедиа, разработанной комитетом Internet Engineering Task Force (IETF).

Подход SIP к построению IP-телефонии намного проще в реализации, чем H.323, но меньше подходит для организации взаимодействия с телефонными сетями. В основном это связано с тем, что протокол SIP плохо согласуется с системами сигнализации, используемыми в ТфОП. Поэтому протокол SIP более подходит поставщикам услуг IP-телефонии, причем эта услуга будет являться всего лишь частью пакета услуг.

Сеть SIP содержит основные элементы трех видов: агенты пользователя, прокси-сервера и сервера переадресации.

Агенты пользователя (User Agent или SIP client) являются приложениями терминального оборудования и включают в себя две составляющие: агент пользователя - клиент (User Agent Client-UAC) и агент пользователя – сервер (User Agent Server-UAS), иначе известные как *клиент* и *сервер* соответственно. Клиент UAC инициирует SIP запросы, т.е. выступает в качестве вызывающей стороны. Сервет UAS принимает запросы и возвращает ответы, т.е. выступает в качестве вызываемой стороны.

Кроме того, существует два типа сетевых серверов переадресации. Серверы SIP могут работать как в режиме с сохранением состояний текущих соединений (statefull), так и в режиме без сохранения состояний текущих соединений (stateless). Сервер SIP, функционирующий в режиме stateless, может обслуживать сколь угодно большое количество пользователей , в отличие от привратника H.323, который может одновременно работать с ограниченным количеством пользователей.

Прокси-сервер (Proxy-server) действует “от имени других клиентов” и содержит функции клиента (UAC) и сервера (UAS).Этот сервер интерпретирует и может перезаписывать заголовки запросов перед отправкой их к другим серверам. Ответные сообщения следуют по тому же пути обратно к прокси-серверу, а не к клиенту.

Сервер переадресации (Redirect server) определяет текущее местоположение вызываемого абонента и сообщает его вызывающему пользователю. Для определения текущего местоположения вызываемого абонента сервер переадресации обращается к *серверу определения местоположения*, принципы работы которого в документе RFC 2543 не специализированы.

1.1.5.3. Сеть на базе MGCP

Третий подход к построению сетей IP-телефонии, основанный на использовании протокола MGCP, также предложен комитетом IETF, рабочей группой MEGACO.

При разработке этого протокола рабочая группа MEGACO опиралась на сетевую архитектуру, содержащую основные функциональные блоки трех видов:

Шлюз - Media Gateway (MG), который выполняет функции преобразования речевой информации, поступающей со стороны ТфОП с постоянной скорость передачи, в вид, пригодный для передачи по сетям с маршрутизацией пакетов IP(кодирование и упаковку речевой информации в пакеты RTP/UDP/IP, а также обратное преобразование);

Контроллер шлюзов - Call Agent, который выполняет функции управления шлюзами;

Шлюз сигнализации – Signaling Gateway (SG), который обеспечивает доставку сигнальной информации, поступающей со стороны ТфОП, к котроллеру шлюзов и перенос информации в обратном направлении.

Протокол MGSP является внутренним протоколом для обмена информацией между функциональными блоками распределенного шлюза, который извне представляется одним шлюзом. Протокол MGSP является master/slave протоколом. Это означает, что контроллер шлюзов является

ведущим, а сам шлюз – ведомым устройством, которое должно выполнять все команды, поступающие от контроллера Call Agent.

Третий подход, предлагаемый организацией IETF, хорошо подходит для развертывания глобальных сетей IP-телефонии, приходящих на смену традиционным телефонным сетям.

1.2. Оценки качества обслуживания

1.2.1. Параметры качества работы сети IP-телефонии

Разнородность трафика в современных IP-сетях ставит вопрос о дифференциальном подходе к обеспечению различных приложений сетевыми ресурсами. Так, при передаче данных, как правило, задержка передачи и ее вариация не являются критичными, чего нельзя сказать о достоверности передачи. В случае передачи голоса наиболее важны характеристики задержки (и в первую очередь ее вариация) и в меньшей степени достоверность.

Традиционно IP-трафик передается по методу “best effort” – “с максимальными усилиями ”. Сеть старается обработать поступающий трафик как можно быстрее, но при этом никаких гарантий относительно результата своих усилий не дает. Не гарантируется ни проверка сети обеспечить поток данных сетевыми ресурсами, ни приоритезация. Другими словами, безотносительно к какому типу трафика относятся информационные пакеты (голос, видео, FTP и т.д.), они обрабатываются по принципу “первый пришел-первый получил обслуживание”. Очевидно, что эта модель не подходит для передачи трафика со специфическими требования к задержке, производительности или надежности передачи данных. Для перехода к новым приложениям на сетях с коммутацией пакетов нужен свой механизм обеспечения качества передачи. В такой структуре сети речевые пакеты должны отличаться от пакетов данных и это отличие должно фиксироваться узлами сети.

Хотя причины обеспечения качества услуг в сетях IP-телефонии взаимосвязаны, можно выделить четыре основные составляющие, значения которых можно определить независимо. Эти параметры в настоящее время могут контролироваться оборудованием сетей IP-телефонии, и они чаще всего

включаются в соглашения провайдера услуг с клиентом о заданном уровне качества SLA (Service Level Agreement). В этом договоре сервис-провайдер и потребитель определяют:

- Параметры качества обслуживания трафика, которые интересуют потребителя и которые согласен поддерживать провайдер – например, средняя пропускная способность, максимальные задержки и вариации задержек, максимальная интенсивность потерь данных, коэффициент готовности сервиса, максимальное время восстановления сервиса после отказа;
- Методы измерения этих параметров;
- Определение платы за обслуживание. Система оплаты может быть достаточно сложной, особенно если соглашение предусматривает несколько уровней качества обслуживания, которые оплачиваются по разным тарифам;
- Санкции за нарушение обязательств провайдера по обеспечению надлежащего качества обслуживания, а также за отклонение параметров трафика пользователя от оговоренных значений. Эти санкции могут выражаться в виде штрафов, либо в иной форме, например, в форме предоставления сервиса в течении некоторого времени бесплатно или по пониженному тарифу;
- Как и любой договор, соглашение SLA по взаимному согласию провайдера и клиента может включать большое количество различных дополнительных статей. Например, статью, оговаривающую условия перехода к более качественному обслуживанию или обслуживанию с разным уровнем качества в зависимости от дня недели или времени суток;
- Соглашение может включать также правила “кондиционирования” пользовательского трафика, то есть обработки трафика, который выходит за оговоренные пределы. Также правила “кондиционирования” могут определять условия отбрасывания или маркировки пакетов-нарушителей (помеченные пакеты будут отбрасываться сетью не всегда, а только в том случае, когда сетевые устройства испытывают перегрузки).

Многие сервис-провайдеры предлагают своим клиентам типовые контракты SLA. Типовые контракты облегчают жизнь сервис-провайдерам, так как для их

реализации можно обойтись без средств гарантированного поддержания качества обслуживания. Требуется только поддерживать примерно постоянный Уровень запаса пропускной способности и предлагать в SLA те значения параметров QoS, которые демонстрирует работающая сеть. Для предоставления различным пользователям индивидуального уровня качества обслуживания такой подход не работает, и в этом случае требуется реализовать в сети различные механизмы управления QoS, такие как RSVP, DiffServ.

Параметры качества работы IP-телефонии:

1.2.1.1. Задержка речевых пакетов

Задержка (delay) является неотъемлемой чертой любой сети передачи данных с пакетной коммутацией. Сети с коммутацией пакетов были созданы для передачи данных, и возможность их использования для передачи голосового или факсимильного трафика в реальном времени, по аналогии с традиционной телефонией, в значительной степени зависит от вносимой задержки. Здесь под задержкой понимается промежуток времени, за который пакет пересекает сеть IP-телефонии от отправителя до получателя.

Экспериментально установлены следующие градации численных величин задержек:

- 1-й уровень - до 200 мс - отличное качество связи. Для сравнения, в сети ТФОП допустимы задержки до 150-200 мс.
- 2- й уровень - до 400 мс - считается хорошим качеством связи. Но если сравнивать с качеством связи по сетям ТФОП, разница будет видна. Если задержки постоянно удерживаются на верхней границе 2-го уровня (на 400 мс.), то не рекомендуется использовать эту связь для деловых переговоров.
- 3-й уровень - до 700 мс - считается приемлемым качеством связи для ведения неделовых переговоров. Такое качество связи возможно также при передаче пакетов по спутниковой связи.

Общая задержка при IP-телефонии складывается из задержек на оцифровку, сжатие, формирование голосового пакета, а также задержек при передаче по каналам, обработке и коммутации пакета в промежуточных узлах, локальной коммутацией в приемном узле, декомпрессии и преобразовании к аналоговому виду. Основные средства для минимизации задержки – использование в сети

высокопроизводительных голосовых коммутаторов и приоритезация голосового трафика над трафиком данных.

1.2.1.2. Джиттер задержки пакетов

Джиттер или вариация задержки – это различие во времени прохождения в сети последовательных пакетов одного соединения. Чем больше джиттер, тем сильнее будет отличаться задержка при передаче одного пакета от задержки при прохождении другого. Джиттер возникает в сети из-за очередей и маршрутизации пакетов одного сегмента речи по разным путям. При сборке пакетов на приемном конце их последовательность может быть нарушена. Джиттер приводит к специфическим нарушениям передачи речи, слышимым как трески или щелчки. Джиттер подавляют путем включения в приемную часть шлюза буфера статической или динамической памяти, который восстанавливает исходную последовательность пакетов. Пакеты, джиттер которых превышает время их “удержания” в буферной памяти, не воспринимаются приемным устройством. Таким образом, буфер подавляет джиттер ценой увеличения, как общего времени удержания, так и потери пакетов; регулировка времени удержания (размера буфера) представляет собой компромисс между ними. По разным данным и в зависимости от типа кодека не воспринимается джиттер не более 15-50 мс.

1.2.1.3. Потери речевых пакетов

Поскольку голосовые пакеты не повторяются, при их потере (или искажении) в сети на приемной стороне возникает короткая пауза в речи. Частые потери голосовых пакетов, вызванные плохим качеством каналов связи, могут привести к ухудшению разборчивости речи, а иногда и к полной невозможности общения. Искажения от потери пакетов также зависят от применяемых в шлюзах типов кодеков. Качество речи при использовании кодеков типа G.729 и G.723.1 в большей степени зависит от потери пакетов, по сравнению с высокоскоростными кодеками типа G.711. Приближенно можно считать, что при IP-телефонии хорошего качества допустимый уровень потерь пакетов должен составлять 1-3%, причем меньшая величина относится к низкоскоростным кодекам, а большая - к высокоскоростным.

Измерение указанных выше параметров производится на определенном интервале времени. Чем меньше этот временной интервал, тем более жесткие требования предъявляются к сети, а, следовательно, ко всем ее элементам, поскольку обеспечение QoS “из конца в конец” требует взаимодействия всех узлов на пути трафика и определяется надежностью, функциональностью и производительностью самого “слабого звена”.

1.2.1.4. Готовность сети

Под готовностью сети (service availability) понимается надежность соединения пользователя с информационным сервисом. Применительно к сети IP-телефонии это означает надежность установления телефонного соединения между двумя абонентами. Телефонные сети общего пользования имеют подтвержденную десятилетиями репутацию исключительно надежной инфраструктуры. Их коэффициент готовности составляет 99,999% или 5 минут отказа в год. В то же время сеть интернет со всей ее непредсказуемостью обладает низкой степенью надежности. Надежность в сетях IP-телефонии должна обеспечиваться аппаратными, программными и сетевыми средствами. Если в сети IP-телефонии используется технология динамической маршрутизации, для обхода могут использоваться даже каналы ТФОП. Современные шлюзы IP-телефонии имеют достаточно высокие показатели надежности. Коэффициент готовности с учетом резервирования составляет 99,999%, среднее время между повреждениями (MTBF) – не менее 80-100 тысяч часов.

1.2.1.5. Эхо

Также нельзя не сказать о влиянии эха на качество разговора. Феномен эха вызывает затруднения при разговоре и у говорящего, и у слушающего. Говорящий слышит с задержкой свой собственный голос. Если сигнал отражается дважды, то слушающий дважды слышит речь говорящего (второй раз - с ослаблением и задержкой).

Эхо может иметь электрическую и акустическую природу.

Отражения в дифсистеме является неотъемлемым свойством ТФОП. Поэтому они проявляются при взаимодействии ТФОП и IP-сетей.

С целью экономии кабеля в ТФОП для подключения абонентских терминалов с давних пор используются двухпроводные линии, по которым речевые сигналы передаются в обоих направлениях. Более того, во многих телефонных сетях передача сигналов обоих направлений по двум проводам используется и в соединительных линиях между электромеханическими АТС. Для разделения сигналов разных в терминалах абонентов и на АТС применяются простые мостовые схемы, называемые дифсистемами (hybrid). Работа этих мостовых схем основывается на согласовании импедансов в плечах моста, одним из плеч которого является двухпроводная абонентская линия. Так как абонентские линии могут очень сильно различаться по своим параметрам (длине, диаметру жил кабеля), то достичь точного согласования невозможно. Вместо этого администрация связи вынуждена ориентироваться на некоторую среднюю величину импеданса для всех абонентских линий своей национальной сети. Это приводит к тому, что сигналы прямого и обратного направления в большинстве случаев не разделяются полностью, и в дифсистеме возникает частичное отражение сигналов.

Если задержка распространения сигнала в сети невелика, такой отраженный сигнал попросту незаметен и не вызывает неприятных ощущений. Если задержка достигает величины 15-20 мс, возникает эффект “огромного пустого помещения”. При дальнейшем увеличении задержки субъективная оценка качества разговора резко ухудшится, вплоть до полной невозможности продолжать разговор. Задержки, свойственный процессам передачи речи по IP-сетям, таковы, что не оставляют выбора и делают механизмы, ограничивающие эффект эха, обязательными в любом оборудовании IP-телефонии.

Акустическое эхо возникает при пользовани терминалами громкоговорящей связи, независимо от того, какая технология используется в них для передачи информации. Акустическое эхо может обладать значительной длительностью, а особенным неприятным бывает изменение его характеристик при изменении, например, взаимного расположения терминала и говорящего, или даже других людей в помещении. Эти обстоятельства делают построение устройств эффективного подавления эха очень непростой задачей.

Существуют два типа устройств, предназначенных для ограничения вредных эффектов эха: эхозаградители и эхокомпенсаторы.

Принцип работы эхозаградителей состоит в отключении канала передачи, когда в канале приема присутствует речевой сигнал. Недостаток такой техники в том, что перебить говорящего становится невозможным, т.е. связь становится полудуплексной.

Эхокомпенсатор – это более сложное устройство, которое моделирует эхосигнал для последующего его вычитания из принимаемого сигнала. Эхо моделируется как взвешенная сумма задержанных копий входного сигнала или, иными словами, как свертка входного сигнала с оцененной импульсной характеристикой канала. Оценка импульсной характеристики происходит в тот момент, когда говорит только удаленный корреспондент, для чего используется детектор речевой активности. Эхокомпенсация достигается вычитанием синтезированной копии эхосигнала из сигнала обратного направления .

1.2.2. Методы оценки качества передачи речи при IP-телефонии

Первые три рассмотренные выше параметра качества работы сети IP-телефонии (задержка, джиттер и потери речевых пакетов) непосредственно влияют на качество передачи речевой информации. Эти параметры не характерны для обычных телефонных сетей, поэтому для оценки качества передачи пакетной речи требуются критерии, которые отличаются от тех, которые используются для нормирования аналоговых и цифровых телефонных каналов.

1.2.2.1. Метод MOS-Mean Opinion Score

Ввиду различной природы передачи информации по каналам коммутируемой и IP-сети наиболее надежным способом сравнительной оценки качества передаваемой речи является субъективный метод общего мнения (Mean Option Score-MOS). Оценки MOS рассчитываются после прослушивания группой людей тестируемого тракта передачи речи по пятибалльной шкале.

Оценки 3,5 баллов и выше соответствуют стандартному и высокому телефонному качеству, 3,0…3,5-приемлемому, 2,5…3,0- синтезированному звуку. Для передачи речи с хорошим качеством целесообразно ориентироваться на MOS не ниже 3,5 баллов.

1.2.2.2. Метод Quality Rating

Другим субъективным методом оценки является использование единиц рейтинга R (Quality Rating) по сто балльной шкале.

Таблица 3

Метод оценки качества на основе использования единиц рейтинга R

Диапазон R	Категория качества речи	Удовлетворенность пользователей
90≤R<100	Наилучшая (best)	Удовлетворены в высшей степени
80≤R<90	Высокая (high)	Удовлетворены
70≤R<80	Средняя (medium)	Некоторые не удовлетворены
60≤R<70	Низкая (low)	Многие не удовлетворены
50≤R<60	Плохая (poor)	Почти все не удовлетворены

Единицы MOS связаны с R сложной нелинейной зависимостью. Высшему качеству R =100 соответствует MOS = 4,5. На практике для быстрого пересчета в наиболее важном диапазоне 2,5< MOS < 4,4 удобна простая линейная аппроксимация: MOS=R/20. Ее погрешность менее 5%, что вполне допустимо, учитывая разбросы при субъективной оценке. Таким образом, для соединения хорошего качества желательно ограничиться первыми тремя категориями, т.е. обеспечить R>70 или MOS>3,5.

Недостатками указанных способов измерения качества передачи речи являются их субъективизм и неэффективность. Эти методы не могут быть использованы на практике для управления сетью, так как они не учитывают влияние параметров работы IP-сети на общую величину качества передачи речи.

1.2.2.3. Метод PSQM-Perceptual Speech Quality Measurement

Кроме субъективных методов имеется также автоматический метод измерения качества передачи речи, названный PSQM (Perceptual Speech Quality Measurement). Этот метод основан на сравнении эталонного речевого сигнала и сигнала, поступающего из кодека или IP-сети. Метод PSQM может быть использован для сравнительной оценки качества работы различных речевых кодеков или сетей, но он также не позволяет учитывать влияние отдельных параметров IP-сети на качество передачи речи.

1.2.2.4. Метод ICPIF-Calculated Planning Impairment Factor

Наиболее удобным для оценки качества работы реальных сетей IP-телефонии является метод “рассчитываемого планируемого параметра ухудшения” ICPIF (Calculated Planning Impairment Factor). Основная идея метода состоит в расчете величин различных параметров ухудшения качества передачи речи на каждом участке соединения в сети связи и сложения этих величин для получения общего параметра. Существуют различные факторы ухудшения качества передачи речи в сетях связи (шум, задержка, эхо и т.д.). Величина общего параметра ухудшения *Itot* определяется по формуле:

$$Itot = Io + Iq + Idte + Idd + Ie,$$

где *Io*- параметр ухудшения качества, обусловленный неоптимальным уровнем громкости и/или высоким шумом в канале;
Iq-параметр ухудшения качества, обусловленный шумами квантования в ИКМ;
Idte- параметр ухудшения качества, обусловленный акустическим эхо;
Idd- параметр ухудшения качества, обусловленный передачей речи на большое расстояние (задержки);
Ie- параметр ухудшения качества, обусловленный специальными устройствами, в частности низкочастотными кодеками.

Для сравнения различных сетей IP-телефонии можно не учитывать параметры *Io* и *Iq*, а значение *Idte* принять равным нулю. Зависимость величины

параметра *Idd* от задержки передачи речевого сигнала в сети приведены в рекомендации G.113 (таблице 4).

Таблица 4

Зависимость параметра Idd от задержки речевого сигнала в сети

Задержка (мс)	Параметры Idd
150	0
200	3
250	10
300	15
400	25
500	30
600	35
800	40
>800	40

Параметр *Ie* используется для оценки качества работы сложных устройств обработки речевых сигналов, например низкочастотных кодеков. В рекомендации G.113 каждый тип кодека характеризуется специфическим параметром *Ki* для оценки ухудшения качества передачи речи. Когда в соединении IP-телефонии используется несколько различных кодеков, то общая величина параметра ухудшения определяется суммированием индивидуальных значений параметра *Ki* для каждого кодека.

В таблице 5 приведены величины параметра *Ki* для некоторых наиболее распространенных кодеков, часть из которых применяется в IP-телефонии.

Таблица 5

Параметр Ki оценки ухудшения качества передачи речи для некоторых кодеков

Тип кодека	Скорость передачи (Кбит/с)	Параметр Ki
PCM(G.711)	64	0
APDCM(G.726,G.727)	40	2
	32	7
	24	25
	16	50
CS-ACELP/CA-ACELP(G.729/G.729a)	8	10
LD-CELP(G.728)	16	7
	12.8	20
VSELP(IS 54,USA)	8	20
RPE-LTP(GSM)	13	20

1.2.3. Кодеки и оценка качества кодеков

Речевая информация более критична к продолжительным задержкам, чем данные. Следовательно, рассматривая возможность реализации IP-телефонии, необходимо оценить параметры задержки и потери пакетов информации на всей протяженности сети. При этом приемлемость или неприемлемость результатов такой оценки, будет зависеть от того, какое качество передачи речи необходимо, и какие ресурсы полосы пропускания на это требуются. Существуют также жесткие требования к типу кодека и к значениям задержки и потери пакетов при их прохождении из конца в конец.

Таблица 6

Сравнение кодеков

Кодеки	Полоса, Кбит/с	Оценка MOS	Задержка, мс
G.711	64	4,1	0,75
G.726	32	3,85	1
G.728	16	3,61	3…5
G.729	8	3,9	10
G.729a	8	3,85	10
G.723.1	6,3/5,3	3,8/3,75	30

В таблице представлены типы кодеков и их параметры. Также представлен параметр MOS (Mean Opinion Score), который определяет среднюю оценку качества голоса, полученную экспертным путем.

На рис.1.5 показана зависимость субъективной оценки качества речи при

R>50 от задержки сигнала. Верхняя кривая дает оценку при задержке без использования кодека, а две другие - соответственно при дополнительном включении между абонентами кодеков G.711(64Кбит/с) и G.723.1 (6,3 Кбит/с).

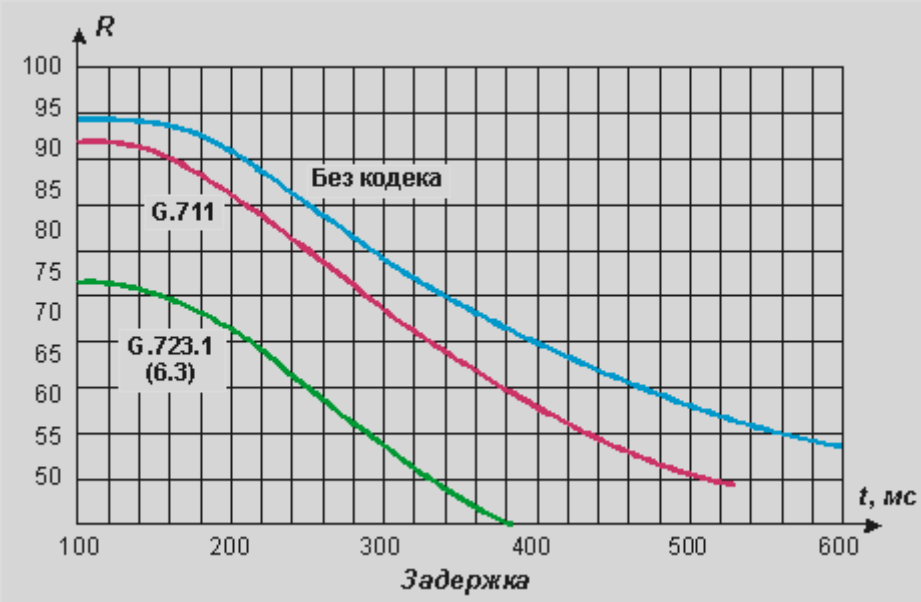


Рис.1.5 Влияние задержки на качество речи

Не изображенные на графике кривые для всех других указанных выше кодеков, в том числе кодеков G.726 и G.728, располагались бы между двумя последними. Следовательно, в зависимости от используемых кодеков и без учета других факторов, качество с рейтингом R> 70 достижимо в случае, если задержка будет не более 200 - 350 мс. Эта оценка справедлива в предположении "нулевых прочих условий", т.е. отсутствия потери пакетов и джиттера.

Искажения от потери пакетов также зависят от применяемых в шлюзах типов кодеков. Как ясно из физических принципов, качество речи при использовании низкоскоростных кодеков должно в большей степени зависеть от потери пакетов, по сравнению с высокоскоростными типа. На графике рис.1.6 кривые показывают, при каком % потери пакетов качество речи понижается до величин R = 70 и R =80, соответствующих нижним границам 3-ей и 2-ой категориям качества табл.3. Эти оценки также предполагают "нулевые условия", т.е. отсутствие задержки и джиттера.

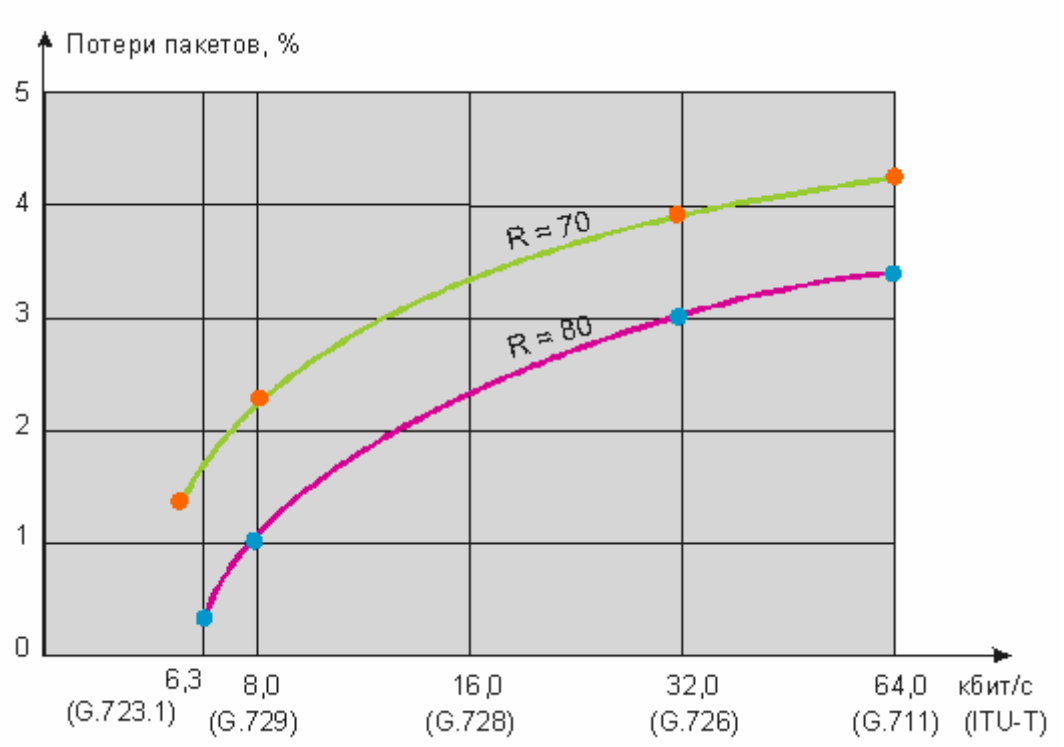


Рис.1.6 Зависимость качества речи от потери пакетов и типов кодеков

Более сложные случаи, когда факторы задержки и потери пакетов действуют одновременно, исследованы только выборочно. На рис.1.7 показано действие этих факторов на популярные у российских провайдеров IP-телефонии кодеки. Две верхние кривые кодеков без потерь приведены для сравнения. Кривые для кодеков G.729 с 2% и G.723.1 (6,3 Кбит/с) с 1% потерь пакетов практически совпадают. Как видно из графиков, при потерях 1 - 2% пакетов и задержках более 150 мс качество речи в IP канале падает ниже порога R = 70.

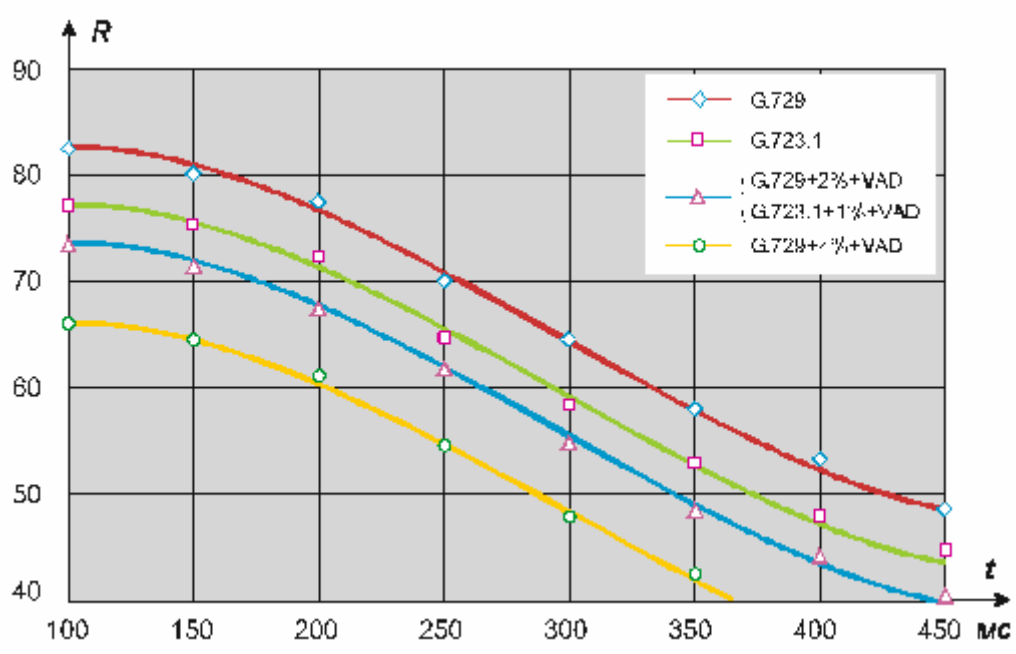


Рис.1.7 Зависимость качества речи от суммарного действия задержки и потери пакетов на кодеки G 729 и G.723.1 (6,3 Кбит/с). VAD - Voice Activity Detection - устройство детектирования (обнаружения) речи, включение которого несколько понижает качество речи.

1.2.4. Механизмы обеспечения качества сервиса

Существуют следующие механизмы, используемые для обеспечения заданного качества сервиса в IP-сетях:

- Механизмы классификации трафика в очереди обрабатываемых пакетов (выделения трафика, нуждающегося в определенном качестве сервиса, и его маркировка)
- Механизмы управления очередями
- Механизмы фильтрации (продвижение приоритетного трафика в случае накопления его в очереди)

Важнейшими механизмами обеспечения QoS являются механизмы обслуживания очередей и кондиционирования трафика. Механизмы первого типа могут поддерживать различные алгоритмы обработки пакетов (от простейшего FIFO, «первый пришел - первый обслужен», до сложных алгоритмов взвешенного обслуживания).

1.2.4.1. Обслуживание очередей

1.2.4.1.1. Стратегия FIFO

Алгоритм обслуживания очередей FirstIn. FirstOut (FIFO), также называемый First Come First Served является самым простым. Пакеты обслуживаются в порядке поступления без какой-либо специальной обработки.

Такая схема приемлема, если исходящий канал имеет достаточно большую свободную полосу пропускания. Алгоритм FIFO относится к так называемым неравноправным схемам обслуживания очередей, так как при его использовании одни потоки могут доминировать над другими и захватывать несправедливо большую часть полосы пропускания. В связи с этим применяются равноправные схемы обслуживания, предусматривающие выделение каждому потоку отдельного буфера и равномерное разделение полосы пропускания между разными очередями.

Поскольку этот подход неприемлем для дифференциального обслуживания потоков - совокупности пакетов имеющих общие признаки, дополнительно используются алгоритм приоритетной обработки и алгоритм взвешенного обслуживания. Комбинацией указанных алгоритмов является алгоритм «взвешенного справедливого обслуживания»(WFQ).

1.2.4.1.2. Очередь с приоритетами

Очередь с приоритетами (Priority Queuing) - это алгоритм, при котором несколько очередей образуют одну систему очередей. В случае простейшего приоритетного обслуживания трафик определенных классов имеет безусловное преимущество перед трафиком других классов. Например, если все IPX-пакеты имеют более высокий приоритет, чем IP-пакеты, то какова бы ни была ценность IP данных, IPX данные будут иметь первоочередной приоритет при разделе доступной полосы. Такой алгоритм гарантирует своевременную доставку лишь наиболее привилегированного трафика.

Назначение разным потокам нескольких разных приоритетов производится по ряду признаков, таких как источник и адресат пакета, транспортный

протокол, номер порта. Пакет каждого потока помещается в очередь, имеющую соответствующий приоритет.

Хотя трафик более высокого приоритета получает лучшее обслуживание, чем он мог бы получить при использовании FIFO, некоторые фундаментальные проблемы остаются нерешенными. Некоторые прикладные программы пытаются использовать весь доступный ресурс. Если им предоставлена наиболее приоритетная очередь, то очереди с низким приоритетом будут блокированы в течение длительного времени или низкоприоритетный трафик встретит настолько большую задержку в результате следования по окружному пути, что станет бесполезным. Это может привести к прекращению менее приоритетных сеансов связи или, по крайней мере, сделает их практически непригодными.

1.2.4.1.3. Class-Based Queuing (CBQ)

При использовании этого механизма трафику определенных классов гарантируется требуемая скорость передачи, а оставшийся ресурс распределяется между остальными классами.

Обработка очередей по алгоритму Class-Based Queuing, CBQ предполагает, что трафик делится на классы. Определение класса трафика в значительной степени произвольно. Класс может представлять весь трафик, проходящий через данный интерфейс, трафик определенных приложений, трафик, направленный к заданному подмножеству получателей, трафик с качеством услуг, гарантированным протоколом RSVR. Каждый класс имеет собственную очередь, и ему гарантируется, по крайней мере, некоторая доля пропускной способности канала. Драйвер интерфейса обходит все очереди по кругу и передает некоторое количество пакетов из каждой очереди. Если какой-либо класс не исчерпывает предоставленный ему лимит пропускной способности, то доля полосы пропускания, выделяемая каждому из остальных классов, пропорционально увеличивается.

Как и в предыдущем случае, используются очереди FIFO, следовательно, для потоков, разделяющих одну очередь, остаются проблемы, присущие FIFO, но гарантируется некоторая справедливость распределения ресурсов в сети между разными очередями. Кроме того, в отличие от приоритетного

обслуживания, CBQ не допускает блокировки очереди и дает возможность учитывать использование сети разными классами.

1.2.4.1.4. Взвешенные справедливые очереди - Weighted Fair Queuing(WFQ)

Если необходимо обеспечить для всех потоков постоянное время задержки, и не требуется резервирование полосы пропускания, то можно воспользоваться алгоритмом Weighted Fair Queuing.

Взвешенная справедливая очередь (Weighted Fair Queuing, WFQ) является частным случаем CBQ, когда каждому классу соответствует свой. Как и в случае CBQ, каждому классу WFQ отводится одна очередь FIFO и гарантируется некоторая часть пропускной способности канала, в соответствии с весовым коэффициентом потока. Если некоторые потоки не используют предоставленную им полосу пропускания полностью, то другие потоки соответственно увеличивают свою долю. Так как в данном случае каждый класс - это отдельный поток, то гарантия пропускной способности эквивалентна гарантии максимальной задержки. Зная параметры сообщения, можно по известным формулам вычислить его максимальную задержку при передаче по сети. Выделение дополнительной пропускной способности позволяет уменьшить максимальную задержку. Алгоритм WFQ гарантирует, что очереди не будут лишены своей доли полосы пропускания, и что трафик получит предсказуемое QoS. Трафик, не использующий целиком свою долю полосы, будет обслуживаться в первую очередь, а оставшаяся полоса будет разделена между остальными потоками.

Определение веса потока производится по полю precedence головка IP-пакета. Значение данного поля лежит в пределах от 0 до 7. Чем выше значение, тем большая полоса выделяется потоку от 0 до 7.

1.2.5. Сетевые аспекты обеспечения качества обслуживания IP-телефонии

IP-телефонию часто считают частью пакета услуг Интернет-провайдера, что не верно. Уже известно множество примеров внедрения IP-телефонии в корпоративные сети и даже построения выделенных сетей IP-телефонии. Причем проблемы при построении сети могут быть самые разные, ведь это

может быть и корпоративная сеть, и сеть традиционного оператора, и отдельная выделенная сеть, или что-то еще. В каждом случае нужны свои решения и подходы. Немаловажен и тип передаваемой по этой сети информации, сеть можно использовать только для передачи речи, а могут передаваться и данные. Важно учитывать характер взаимодействия различных узлов IP-телефонии и обеспечивать минимальные задержки и минимальный уровень потерь. Иногда приходится сокращать полосу пропускания.

Упрощенно резюмируя материал данной главы можно сказать, что полная временная задержка речевого трафика делиться на две основные части: задержки на кодирование и декодирование на шлюзах, и задержки вносимые самой сетью.

Уменьшить общую задержку можно двумя путями, во-первых, спроектировать инфраструктуру сети таким образом, чтобы задержка в ней была минимальной, а, во-вторых, уменьшить время обработки данных в речевом шлюзе.

Для уменьшения задержки в сети нужно сокращать число транзитных участков между маршрутизаторами, а в наиболее важных местах сети использовать высокоскоростные каналы. А для уменьшения разброса задержек можно использовать эффективные методы управления трафиком, например механизмы резервирования.

Как правило, с корпоративными сетями все выглядит достаточно просто. Они имеют ограниченные размеры, контролируемую топологию, а характер трафика обычно заранее известен. Однако, возьмем простой пример: речь передается по существующей ЛВС, которая слишком загружена, чтобы обеспечивать приемлемое качество обслуживания. Решением этой проблемы будет изоляция серверов и клиентов данного типа трафика и ресегментация сети. Разбить сеть на сегменты можно или установив коммутатор Ethernet, или добавив порты в маршрутизатор.

Выделенные сети IP-телефонии обычно используются для междугородной и международной связи. Такие сети лучше строить по принципу многоуровневой иерархической сети, где на каждый уровень возлагаются свои определенные функции. На входе в сеть главное обеспечить подключение речевых шлюзов, а внутри сети – высокоскоростную пересылку трафика. В такой сети очень просто осуществляется расширение и внедрение новых услуг и служб. Проблема проектирования также не доставляет особых

хлопот: характер трафика определен, полоса пропускания также легко рассчитывается. Трафик однотипный, а значит не нужна приоритизация пакетов.

В сетях традиционных операторов передается трафик различных видов, по этому для обеспечения приемлемого качества передачи предлагается использование модели Diff-Serv.

1.2.6. Дифференциальное и интегральное обслуживание

Существует две архитектуры IP-QoS, утвержденные комитетом IETF:

- Архитектура с интеграцией сервисов (Integrated Service Architecture, Int-Serv)
- Архитектура с дифференциацией сервисов (Differentiated Services Framework, Diff-Serv)

1.2.6.1. Архитектура QoS Int-Serv

В основе архитектуры Int-Serv лежит протокол резервирования ресурсов — RSVP (Resource ReSerVation Protocol). Данный протокол позволяет зарезервировать определенную долю сетевых ресурсов, необходимую информационному потоку, на протяжении всего маршрута от станции отправителя до станции получателя.

Для трафика реального времени вводятся два класса обслуживания:

- Контролируемой загрузки сети
- Гарантированного обслуживания

Классу гарантированного обслуживания предоставляется определенная полоса пропускания, а также гарантируются задержка в определенных пределах и отсутствие потерь при переполнении очередей.

Класс контролируемой загрузки сети идентичен традиционному подходу "best effort", но уровень QoS для уже обслуживаемого потока данных остается неизменным при увеличении нагрузки в сети.

Основными компонентами модели Int-Serv являются система резервирования ресурсов, система контроля доступа, классификатор и диспетчер очередей.

Спецификация потока (flow specification) нужна для определения необходимого уровня качества обслуживания потока.

Система контроля доступа, получив запрос сеанса связи, в зависимости от наличия требуемых ресурсов, либо допускает этот запрос к дальнейшей обработке, либо дает отказ. Классификатор определяет класс обслуживания на основе содержания поля приоритета в заголовке. Диспетчер определяет способ организации и механизм обслуживания очереди. Система резервирования ресурсов использует специальный протокол сигнализации, который служит для запроса приложением нужного ему уровня качества обслуживания координации обработки этого запроса всеми устройствами.

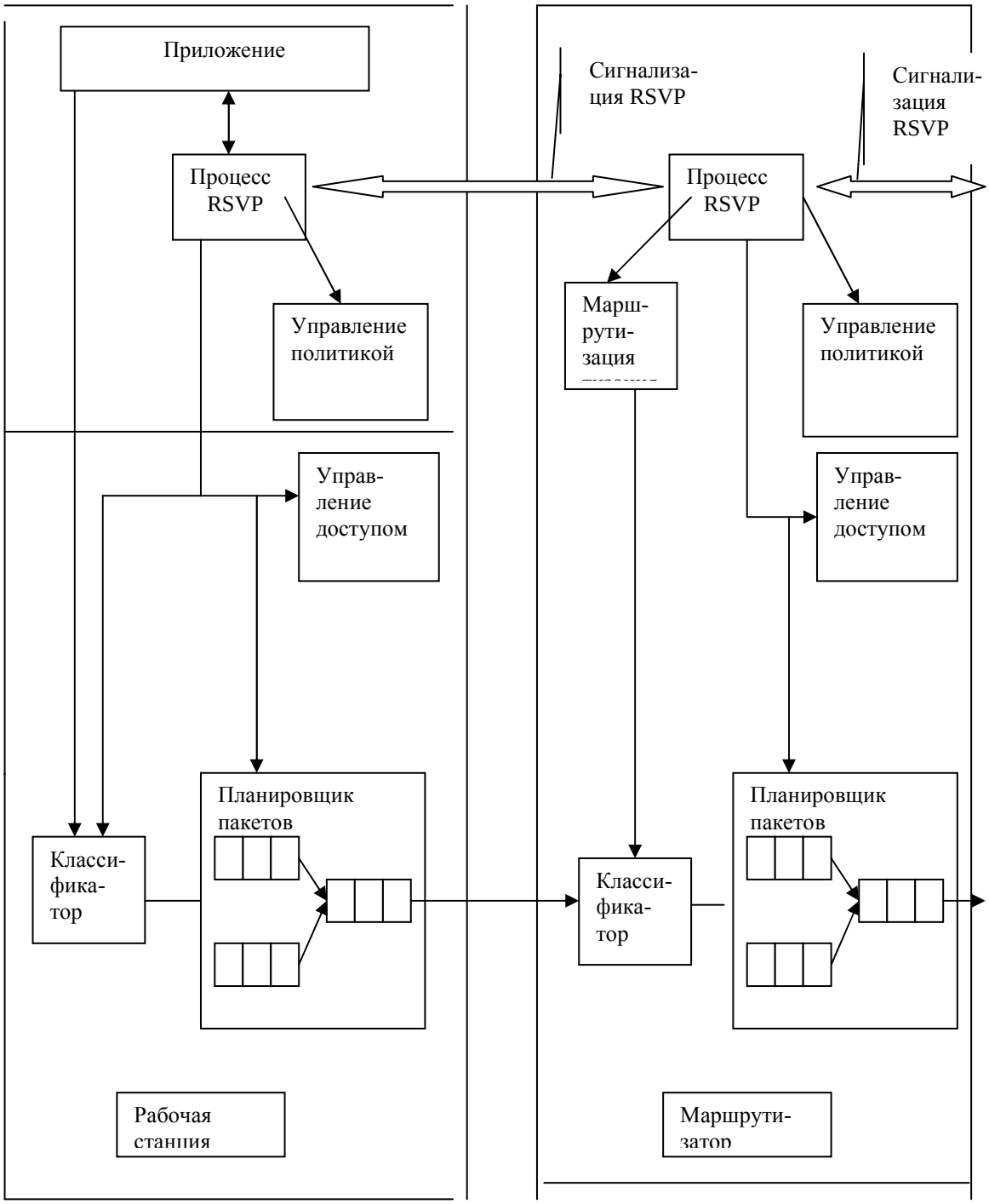


Рис.1.8 Модель Int-Serv

Перед тем как зарезервировать ресурсы, процесс RSVP маршрутизатора соединяется с двумя локальными модулями принятия решения – модулем управления доступом (admission control) и модулем управления политикой (policy control). Модуль управления доступом определяет, имеет ли узел достаточно свободных ресурсов для обеспечения запрошенного уровня QoS. Модуль управления политикой определяет, есть ли у пользователя администраторские права, для того чтобы произвести резервирование. Если какая либо из проверок не прошла, процесс RSVP отправляет сообщение об ошибке процессу приложения, которое создало запрос. Если обе проверки прошли нормально, процесс RSVP устанавливает параметры классификатора пакетов (packet classifier) и планировщика пакетов (packet scheduler) для получения нужного QoS. Классификатор пакетов определяет класс QoS для каждого пакета, а планировщик пакетов управляет передачей пакетов, основываясь на их классе QoS.

Во время процесса принятия решения модулем управления доступом резервирование затребованной полосы пропускания производится только в том случае, если для запрашиваемого класса трафика достаточно оставшейся части. В противном случае запрос на доступ отклоняется, но трафик все равно передается с качеством обслуживания, определенным по умолчанию для данного класса трафика.

1.2.6.2. Архитектура QoS Diff-Serv

Идея архитектуры с дифференциацией сервисов состоит в минимизации служебного трафика, чтобы исключить задержки, возникающие в Int-Serv на начальном этапе взаимодействия.

Архитектура с дифференциацией сервисов является прозрачной для приложений, здесь основная тяжесть реализации приходится на маршрутизаторы и коммутаторы поставщиков сетевых услуг.

Идея состоит в том, чтобы сформировать "облако", поддерживающее качество сервиса. Граничные маршрутизаторы по предопределенным правилам классифицируют входной трафик, нормализуют, маркируют его и передают транзитным устройствам для дальнейшего продвижения. Нормализация трафика предусматривает измерение его параметров, проверку соответствия

заданным правилам предоставления услуг, профилирование (при этом пакеты, не укладывающиеся в рамки установленных правил, могут быть отсеяны) и другие операции.

В результате однотипным приложениям соответствует одинаковый приоритет, и обрабатываются соответствующие пакеты сходным образом.

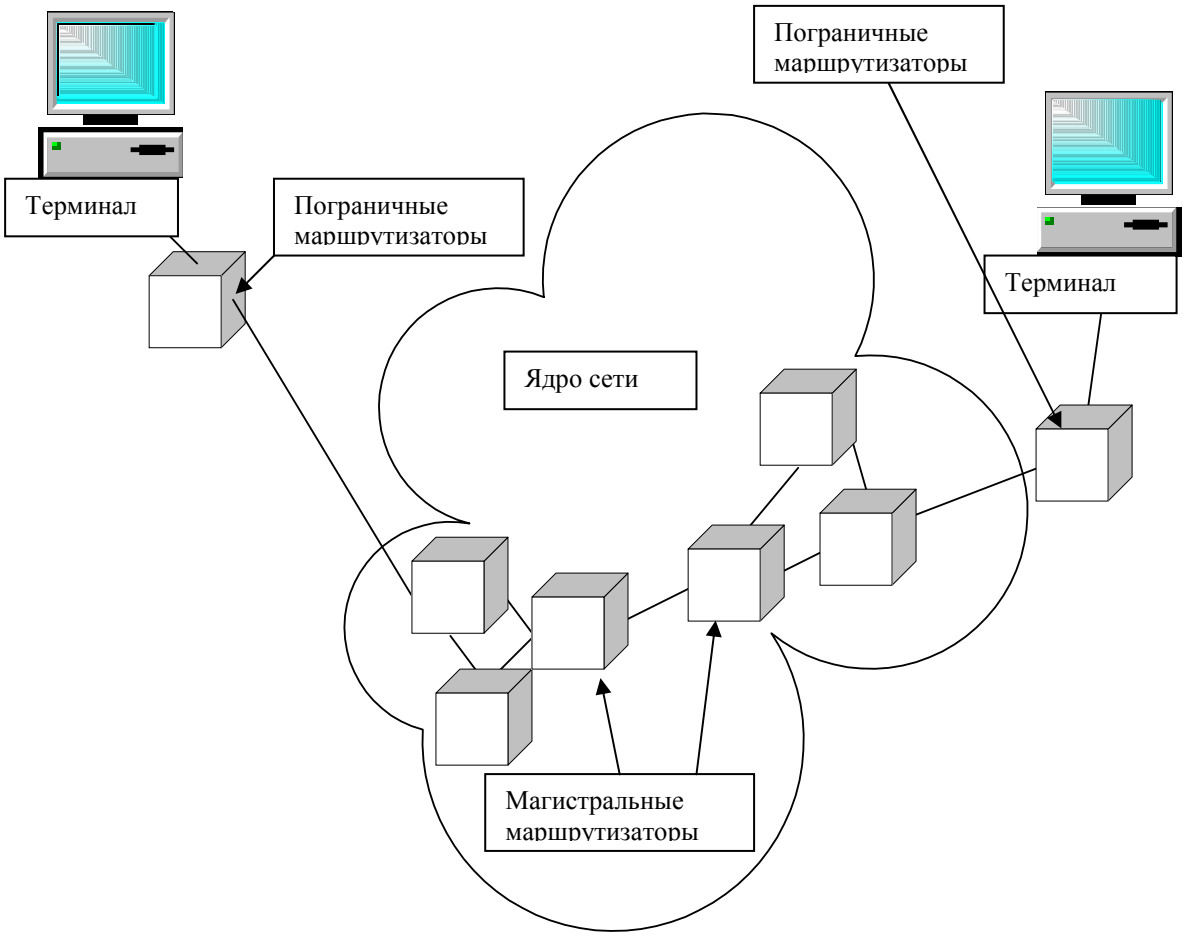


Рис.1.9 Модель Diff-Serv

Для маркировки пакетов используется поле типа сервиса (ToS), которое в данной архитектуре называется DS (Differentiated Services). Базируясь на значении поля DS, транзитный маршрутизатор помещает IP-пакет в очередь, соответствующую заданному приоритету. Движение внутри каждой очереди обеспечивается механизмом управления, ключевыми функциями которого является распределение пропускной способности канала связи и правила обработки в случае накопления пакетов.

Достоинства модели Diff-Serv состоят в том, что она, во-первых, обеспечивает единое понимание того, как должен обрабатываться трафик

определенного класса, а во-вторых, позволяет разделить весь трафик на относительно небольшое число классов и не анализировать каждый информационный поток отдельно. К настоящему времени для Diff-Serv определено два класса трафика:

- *Класс срочной пересылки пакетов* (Expedited Forwarding PHB Group)
- *Класс гарантированной пересылки пакетов* (Assured Forwarding PHB Group)

Хотя эта модель и не гарантирует качество обслуживания на 100%, у нее есть серьезные преимущества. Например, нет необходимости в организации предварительного соединения и в резервировании ресурсов. А так как в модели Diff-Serv используется небольшое, фиксированное количество классов и трафик абонентов распределяется по общим очередям, не требуется высокая производительность сетевого оборудования.

Архитектура с дифференциацией сервисов предлагает базовый уровень, обеспечивающий поддержку качества обслуживания, предоставляет гибкий механизм для разработчиков активного сетевого оборудования. Однако при использовании данной архитектуры возникают определенные сложности.

Каждый поставщик сетевых услуг обеспечивает качество сервиса в своей "зоне ответственности", однако нет гарантии, что поддержка окажется сквозной, как это имеет место в архитектуре с интеграцией сервисов.

Глава 2. Технология MPLS

Аббревиатура MPLS расшифровывается как “многопротокольная коммутация с использованием меток” (Multiprotocol Label Switching, MPLS). “Многопротокольность” в название технологии означает, что средства MPLS применимы к любому протоколу сетевого уровня.

Первый, физический уровень (physical layer) содержит функции, обеспечивающие использование физической среды для двусторонней передачи *битов* (с такой достоверностью, какую обеспечивает эта среда) по прямому тракту, связывающему два узла сети, второй уровень – уровень звена данных (data link layer) - содержит функции, обеспечивающие формирование в этом тракте надежного логического звена связи, по которому обеспечивается двусторонний обмен между этими узлами *информационными блоками*, гарантируя при этом заданную достоверность передачи путем обнаружения и исправления ошибок, а третий, сетевой уровень содержит функции, обеспечивающие транспортировку информационных блоков по сети от отправителя к получателю через несколько узлов, выбирая подходящий маршрут транспортировки, который составляется из звеньев второго уровня. Общая идея протоколов всех уровней (кроме физического) состоит в том, что информационный блок каждого уровня содержит заголовок и информационное поле, и в том, что блок протокола вышележащего уровня помещается в информационное поле блока протокола расположенного сразу под ним нижележащего уровня.

При традиционной транспортировке пакета через сеть с использованием в уровне 3 протокола, не предусматривающего создания виртуальных соединений, каждый маршрутизатор на пути следования пакета самостоятельно принимает решение о том, к какому маршрутизатору переслать этот пакет дальше (такой способ транспортировки по-английски называется hop-by-hop). Иначе говоря, в каждом маршрутизаторе на пути следования пакета анализируется его заголовок и выполняется алгоритм сетевого уровня.

Однако в заголовке пакета содержится гораздо больше информации, чем нужно для того, чтобы выбрать следующий маршрутизатор. Этот выбор можно представить себе как выполнение двух групп функций. Одна группа разделяет все множество прибывающих пакетов на классы, которые удобно называть

классами эквивалентности пересылки (Forwarding Equivalence Classes - FECs). Вторая группа ставит в соответствие каждому FEC определенное направление пересылки. С точки зрения выбора следующего маршрутизатора все пакеты, принадлежащие одному FEC, неразличимы.

При *традиционной IP-маршрутизации* конкретный маршрутизатор может считать, что два пакета принадлежат одному и тому же FEC, если в его таблицах маршрутизации используется некий адресный префикс X, идентифицирующий “направление”, в котором предполагаемые маршруты транспортировки этих двух пакетов совпадают наиболее долго. По мере продвижения пакета по сети каждый следующий маршрутизатор анализирует его заголовок и приписывает этот пакет к такому из “своих” FEC, который соответствует тому же “направлению”.

При использовании *многопротокольной коммутации по меткам* (MPLS) определенный пакет приписывается к определенному FEC только один раз, когда он “входит” в сеть. Этому FEC присваивается *метка* - идентификатор фиксированной длины, передаваемый вместе с пакетом, когда тот пересылается к следующему маршрутизатору. Существенно, что в остальных маршрутизаторах заголовок сетевого уровня не анализируется. Принятая маршрутизатором вместе с пакетом метка используется как указатель входа таблицы, которая определяет очередной маршрутизатор для пересылки к нему пакета, а также новую метку для FEC, к которому относится пакет.

Метод пересылки пакетов, принятый в MPLS, имеет ряд преимуществ перед методами, основанными на анализе заголовка блоков сетевого уровня. В частности, пересылку по методу MPLS могут выполнять маршрутизаторы, которые способны читать и заменять метки, но при этом либо вообще не способны анализировать заголовки блоков сетевого уровня, либо не способны делать это достаточно быстро.

Некоторые маршрутизаторы, обеспечивающие традиционную коммутацию пакетов, анализируют заголовки блоков сетевого уровня не только с целью выбрать следующий маршрутизатор, но и с целью определить приоритет пакета или присвоенный ему класс обслуживания. MPLS позволяет получить такого рода сведения, анализируя метку (но при этом не исключает и традиционного способа их получения).

2.1. Принцип коммутации

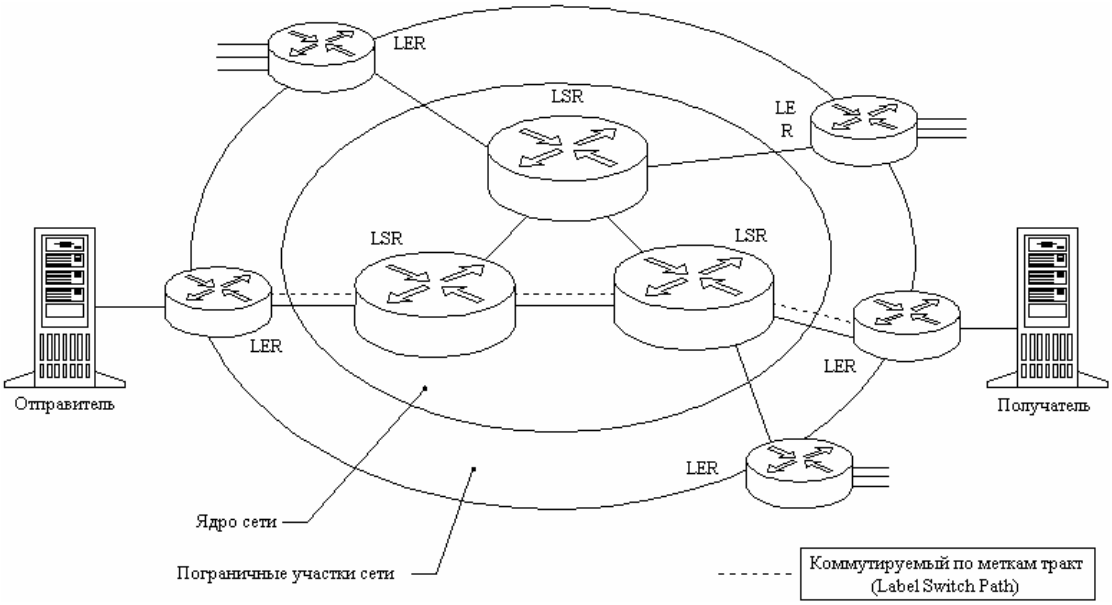


Рис.2.1 Сеть, построенная на базе технологии MPLS

Протокол MPLS предполагает четкое разделение всех функций на две компоненты: пересылка пакетов и управление. Управляющая компонента (control component) задействует протокол распространения меток для обмена информацией с другими маршрутизаторами. На основе этой информации формируется и модифицируется сначала таблица маршрутизации, а затем, с учетом информации о смежных системах на каждом интерфейсе, — таблица пересылки пакетов. Когда система получает новый пакет, пересылающая компонента (forwarding component) анализирует информацию, содержащуюся в его заголовке, ищет соответствующую запись в таблице пересылки и направляет пакет на выходной интерфейс.

Отделение управляющей компоненты от пересылающей позволяет разрабатывать и модифицировать каждую из них независимо. Единственное обязательное требование состоит в том, чтобы управляющая компонента могла передавать информацию пересылающей компоненте через таблицу пересылки пакетов. Благодаря этому становится возможным применение очень простых алгоритмов пересылки, например алгоритма, базирующегося на использовании последовательных меток.

Распространение трафика в сети MPLS происходит по следующему сценарию. Первый пограничный коммутатор – Label Edge Router (LER) на

основании IP адреса пункта назначения и/или другой информации заголовка пакета определяет соответствующее политике обеспечения QoS значение метки, принадлежность пакета определенному классу FEC и выходной интерфейс для пакета. Предположим, входной коммутатор сети получил непомеченный пакет с адресом назначения 192.4.2.1. Он классифицирует этот пакет (относит к классу FEC 192.4/16), присваивает ему метку 5 и передает смежному устройству. Следующий, транзитный маршрутизатор – Label Switching Router (LSR) использует метку для продвижения пакета, сопоставляя с находящейся на нем базой информацией о метках (Label Information Base — LIB), определяет следующий LSR на пути к пункту назначения и заменят метку на новую. Каждый маршрутизатор LSR содержит таблицу, которая ставит в соответствие паре «входной интерфейс, входная метка» тройку «префикс адреса получателя, выходной интерфейс, выходная метка». Получая пакет, LSR по номеру интерфейса, на который пришел пакет, и по значению привязанной к пакету метки определяет для него выходной интерфейс. (Значение префикса применяется лишь для построения таблицы и в самом процессе коммутации не используется.) Старое значение метки заменяется новым, содержавшимся в поле «выходная метка» таблицы, и пакет отправляется к следующему устройству на маркированном пути LSP (Label-Switched Path).

Последний пограничный маршрутизатор снимает метку и отправляет на выходной интерфейс в обычном виде. Таким образом, после прохождения первого пакета вдоль маршрутизаторов LSR создается виртуальный коммутируемый с помощью меток путь LSP. Маршрут LSP функционально эквивалентен виртуальному каналу, поскольку определяет путь через всю сеть — от входа в нее до выхода из нее. По этому пути следуют все пакеты, отнесенные к определенному классу FEC. Первый из поддерживающих метки коммутаторов на этом пути называется входным (ingress, или head-end); а последний коммутатор, завершающий данный LSP, — выходным (egress, или tail-end).

Вся операция пересылки пакетов требует лишь одноразовой идентификации значений полей в одной строке таблицы. Это занимает гораздо меньше времени, чем сравнение IP-адреса отправителя с наиболее длинным адресным префиксом в таблице маршрутизации, которое используется при традиционной маршрутизации.

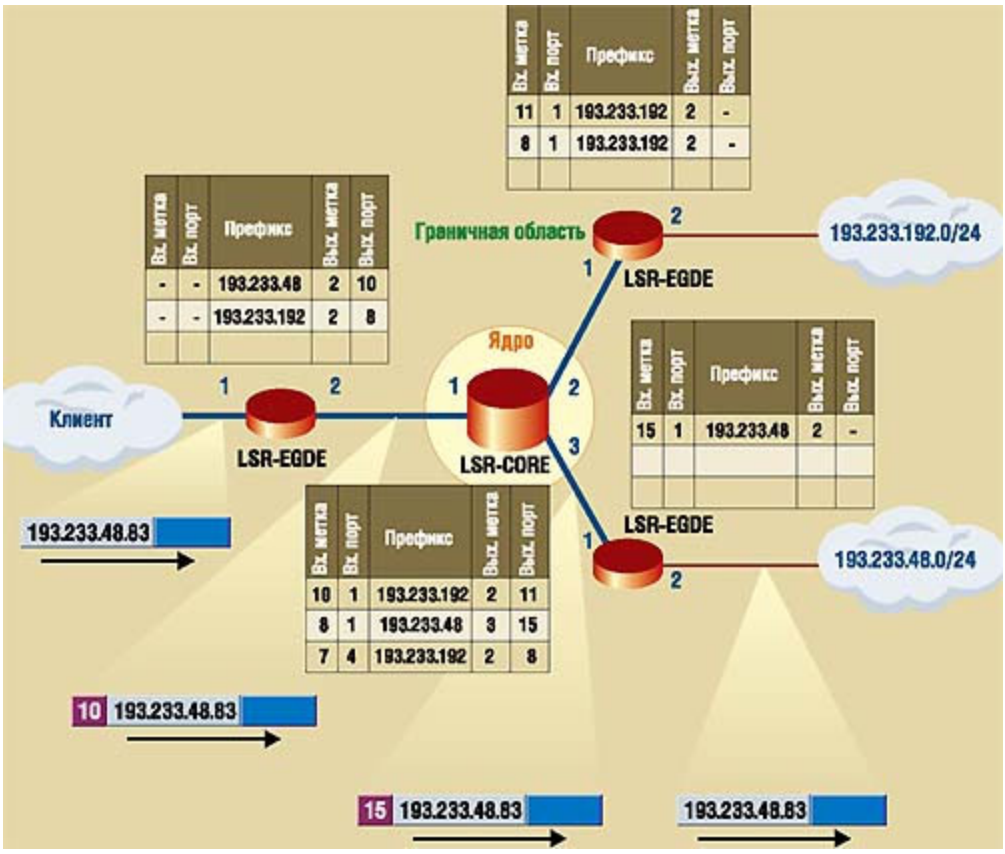


Рис.2.2 Схема коммутации MPLS

Сеть MPLS делится на две функционально различные области — ядро и граничную область. Ядро образуют устройства, минимальным требованием к которым является поддержка MPLS и участие в процессе маршрутизации трафика для того протокола, который коммутируется с помощью MPLS. Маршрутизаторы ядра занимаются только коммутацией. Внутри ядра сети коммутаторы игнорируют информацию сетевого уровня в заголовках пакетов и определяют дальнейший маршрут их следования исключительно на основе меток. Когда коммутатор получает помеченный пакет, его пересылающая компонента ищет в таблице пересылки нужную запись (по номеру входного порта и входной метке), извлекает из нее выходную метку, выходной интерфейс и адрес следующего смежного устройства. Затем коммутатор заменяет входную метку выходной (эта процедура называется label swapping) и передает пакет на выходной интерфейс для дальнейшего продвижения по маршруту LSP. Когда пакет достигает конца маршрута LSP, выходной коммутатор тоже обращается к своей таблице пересылки. Но, поскольку на следующем шаге пакет должен быть передан уже на устройство, не поддерживающее метки, коммутатор удаляет метку и отправляет пакет, используя обычный алгоритм IP-

маршрутизации. Все функции классификации пакетов по различным FEC, а также реализацию таких дополнительных сервисов, как фильтрация, явная маршрутизация, выравнивание нагрузки и управление трафиком, берут на себя граничные LSR. В результате интенсивные вычисления приходятся на граничную область, а высокопроизводительная коммутация выполняется в ядре, что позволяет оптимизировать конфигурацию устройств MPLS в зависимости от их местоположения в сети.

Таким образом, главная особенность MPLS — отделение процесса коммутации пакета от анализа IP-адресов в его заголовке, что открывает ряд привлекательных возможностей.

Каждый из классов FEC обрабатывается отдельно от остальных — не только потому, что для него строится свой путь LSP, но и в смысле доступа к общим ресурсам (полосе пропускания канала и буферному пространству). В результате технология MPLS позволяет очень эффективно поддерживать требуемое качество обслуживания, не нарушая предоставленных пользователю гарантий. Применение в LSR таких механизмов управления буферизацией и очередями, как WFQ или CBQ, дает возможность оператору сети MPLS контролировать распределение ресурсов и изолировать трафик отдельных пользователей.

Итак, при использовании технологии MPLS соответствие между пакетом и потоком устанавливается один раз, на входе в сеть MPLS, в то время как в обычных IP-сетях любой маршрутизатор, находящийся на пути следования пакета, анализирует заголовок пакета, чтобы определить, к какому потоку этот пакет относится, и выбрать направление для пересылки к следующему маршрутизатору. Следовательно, протокол MPLS упрощает процесс продвижения пакетов в сети, поскольку на промежуточных LSR происходит не обычная маршрутизация, а высокоскоростная коммутация на основании информации в метке.

2.2. Метки и правила их использования

Метка определяет принадлежность каждого пакета тому или иному классу эквивалентности пересылки – Forwarding Equivalence Class (FEC). К одному FEC относятся пакеты всех потоков, пути следования которых через сеть (или

часть сети) совпадают. Метка имеет локальное значение – она действительна на участке между двумя соседними маршрутизаторами, являясь исходящей меткой определенного FES для одного из них и входящей - для второго. При переходе потока пакетов в другой FEC, метка нового FEC помещается поверх метки прежнего FEC и используется для коммутации, а прежняя метка сохраняется под ней, но не используется до тех пор, пока не восстановится прежний FEC. Если FEC пакета меняется несколько раз, в стеке накапливается несколько меток.

Технологии MPLS и DiffServ схожи – оба стандарта используют маркировку пакетов во входных точках сети, то есть анализ, классификация трафика происходит на границе доменов. В отличие от DiffServ, использующего для DS уже существующее поле типа сервиса ToS в пакете IP, в MPLS метка может быть помещена в пакет разными способами: специальная 32-разрядная информационная метка вписывается в специальный заголовок, помещаемый либо между заголовками второго/ третьего уровня (уровня звена данных и сетевого), либо в свободное и доступное поле заголовка какого- то одного из этих двух уровней, если таковое имеется. Метка используется для определения следующего маршрутизатора на пути к пункту назначения. Кодовое же слово DS в механизме DiffServ не несет в себе информацию, которая влияет на выбор маршрута для продвижения пакетов, а определяет уровень качества обслуживания пакетов в промежуточных узлах.

Этот специальный заголовок содержит поле, куда записывается значение метки, и несколько специальных полей, среди которых имеется и поле QoS (три бита, т. е. до восьми классов качества обслуживания). Очевидно, что вопрос о том, куда нужно помещать заголовок, содержащий метку, должен согласовываться между объектами, ее использующими. Заголовок этот имеет следующий вид.

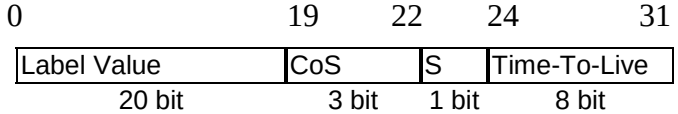


Рис.2.3 Метка MPLS

Label Value – значение метки

CoS – класс обслуживания

S – указатель дна стека меток

Time-To-Live (TTL) – “время жизни”

Каждый маршрутизатор уменьшает время метки на 1, пока оно не достигнет 0. При достижении нулевого значения действия LSR-маршрутизатора зависят от метки (то есть пакет не всегда отбрасывается).

Механизм последовательных меток обладает целым рядом преимуществ по сравнению с традиционной пошаговой маршрутизацией на сетевом уровне. Начнем с того, что использование меток предоставляет поставщику сетевых услуг чрезвычайную гибкость в классификации пакетов, так как метка должна быть уникальной лишь для каждой пары смежных LSR. Поэтому одна и та же метка может быть связана с несколькими FEC, если разным FEC принадлежат пакеты, идущие от разных маршрутизаторов, и имеется возможность определить, от которого из них пришел пакет с данной меткой. В связи с этим вероятность того, что пространство меток будет исчерпано очень мало.

В простейшем случае входной коммутатор сети можно сконфигурировать так, чтобы он относил пакет к тому или иному классу FEC исключительно на основе адреса назначения. Однако для выбора FEC можно использовать и множество других критериев: адрес источника пакета, тип приложения, точку входа в сеть с поддержкой меток и точку выхода из нее, класс обслуживания (CoS), указанный в заголовке IP-пакета, или любое сочетание этих параметров.

Поставщики сетевых услуг могут конструировать специальные LSP-маршруты, удовлетворяющие требования тех или иных приложений. Маршруты можно построить таким образом, чтобы, например, минимизировать число транзитных узлов, обеспечить определенную полосу пропускания или обойти потенциальные точки перегрузки.

Важное преимущество алгоритма пересылки с использованием последовательных меток состоит в том, что он позволяет выделить любой тип пользовательского трафика, ассоциировать его с определенным классом FEC и направить весь трафик этого класса по LSP-маршруту, специально построенному так, чтобы удовлетворить требованиям данного типа трафика.

Протокол MPLS является во многом протоколом “конструирования трафика”, а не протоколом QoS. Маршрутизация MPLS используется для образования виртуальных каналов в IP-сетях, причем предполагается, что для этих каналов

маршрутизаторы сети выделяют определенные ресурсы. При этом потоку трафика, следующему вдоль виртуального пути, гарантируются параметры QoS, Такие как пропускная способность или максимальный уровень задержек. Однако сам способ резервирования и поддержки качества обслуживания остается за пределами протокола MPLS, он только создает виртуальный канал и может переносить в поле метки требования QoS. Резервирование пропускной способности для виртуального канала MPLS может выполнять как администратор, так и другой протокол, например RSVP.

2.3. Стек меток

В рамках архитектуры MPLS вместе с пакетом разрешено передавать не одну метку, а целый их стек. Операции добавления/изъятия метки определены как операции на стеке (push/pop). Результат коммутации задает лишь верхняя метка стека, нижние же передаются прозрачно до операции изъятия верхней. Такой подход позволяет создавать иерархию потоков в сети MPLS и организовывать туннельные передачи. Стек состоит из произвольного числа элементов, каждый из которых имеет длину 32 бита: 20 бит составляют собственно метку, 8 отводятся под счетчик времени жизни пакета, один указывает на нижний предел стека, а три не используются. Метка может принимать любое значение, кроме нескольких зарезервированных.

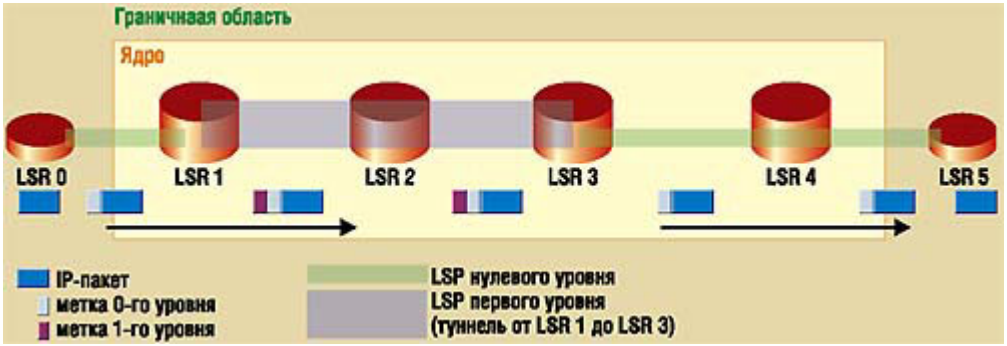


Рис.2.4 Компоненты коммутируемого маршрута.

Коммутируемый путь (LSP) одного уровня состоит из последовательного набора участков, коммутация на которых происходит с помощью метки данного уровня. Например, LSP нулевого уровня проходит через устройства LSR 0, LSR 1, LSR 3, LSR 4 и LSR 5. При этом LSR 0 и LSR 5 являются, соответственно, входным (ingress) и выходным (egress) маршрутизаторами для пути нулевого

уровня. LSR 1 и LSR 3 играют ту же роль для LSP первого уровня; первый из них производит операцию добавления метки в стек, а второй — ее изъятия. С точки зрения трафика нулевого уровня, LSP первого уровня является прозрачным туннелем. В любом сегменте LSP можно выделить верхний и нижний LSR по отношению к трафику. Например, для сегмента «LSR 4 — LSR 5» четвертый маршрутизатор будет верхним, а пятый — нижним.

2.4. Привязка и распределение меток

Управляющий компонент отвечает за создание привязок меток и их последующее распространение между LSR-маршрутизаторами.

Под привязкой понимают соответствие между определенным классом FEC и значением метки для данного сегмента LSP. Привязку всегда осуществляет «нижний» маршрутизатор LSR (расположенный ближе к адресату), поэтому и информация о ней распространяется только в направлении от нижнего LSR к верхнему (расположенному ближе к отправителю). Вместе с этими сведениями могут предаваться атрибуты привязки.

Обмен информацией о привязке меток и атрибутах осуществляется между соседними LSR с помощью протокола распределения меток. Архитектура MPLS не зависит от конкретного протокола, поэтому в сети могут применяться разные протоколы сетевой сигнализации. Очень перспективно в данном отношении — использование RSVP для совмещения резервирования ресурсов и организации LSP для различных потоков.

Существуют два режима распределения меток: независимый и упорядоченный. В первом случае LSR может уведомить вышестоящий LSR о привязке метки к FES еще до того, как получит информацию о привязке “метка-FES” от нижестоящего маршрутизатора. Во втором случае высылать подобное уведомление разрешается только после получения таких сведений “снизу”. Метки могут выдаваться нижним маршрутизатором как по собственной инициативе – спонтанно (unsolicited downstream), так и по запросу верхнего (downstream on-demand). Наконец, возможен “либеральный” и “консервативный” режим распределения меток. В либеральном режиме нижний LSR раздает метки вышестоящим LSR, как имеющим с ним прямую связь, так и

доступным лишь через промежуточные LSR. В консервативном режиме вышестоящий LSR обязан принять метку, если ее выдает смежный LSR, но может отказаться от метки пришедшей к нему транзитом.

2.5. Построение коммутируемого маршрута

Рассмотрим, как система MPLS автоматически создает путь LSP в простейшем случае — с помощью протокола LDP. Архитектура MPLS не требует обязательного применения LDP, однако, в отличие от других возможных вариантов, он наиболее близок к окончательной стандартизации.

Сначала посредством многоадресной рассылки сообщений UDP коммутирующие маршрутизаторы определяют свое «соседство» (adjacency) в рамках протокола LDP. Кроме близости на канальном уровне, LDP может устанавливать связь между «логически соседними» LSR, не принадлежащими к одному каналу. Это необходимо для реализации туннельной передачи. После того как соседство установлено, LDP открывает транспортное соединение между участниками сеанса поверх TCP. По этому соединению передаются запросы на установку привязки и сама информация о привязке. Кроме того, участники сеанса периодически проверяют работоспособность друг друга, отправляя тестовые сообщения (keepalive message)..

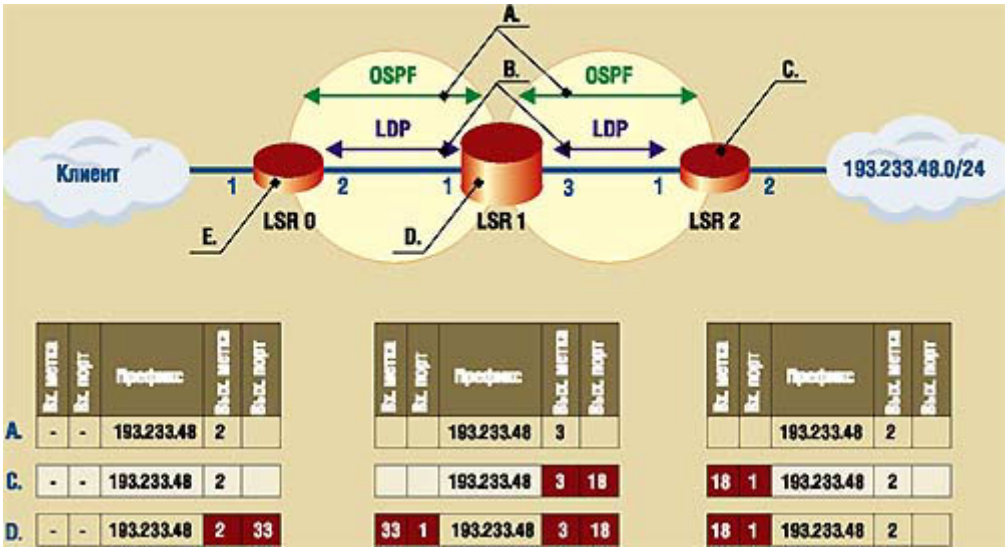


Рис.2.5 Построение коммутируемого пути по протоколу LDP

Рассмотрим на примере, как происходит заполнение таблиц меток по протоколу LDP. Предположим, что выбран упорядоченный режим распределения меток LSP со спонтанным распространением сведений о привязке.

На стадии A каждое из устройств сети MPLS строит базу топологической информации, задействуя любой из современных протоколов маршрутизации (на схеме — OSPF). На стадии B маршрутизаторы LSR применяют процедуру нахождения соседних устройств и устанавливают с ними сеансы LDP.

Далее (стадия C) LSR 2 на основе анализа собственных таблиц маршрутизации обнаруживает, что он является выходным LSR для пути, ведущего к IP-сети 193.233.48.0. Тогда LSR 2 ассоциирует класс FEC с пакетами, адрес получателя которых соответствует префиксу данной сети, и присваивает этому классу случайное значение метки — в нашем случае 18. Получив привязку, протокол LDP уведомляет верхний маршрутизатор LSR (LSR 1) о том, что потоку, адресованному сети с префиксом 193.233.48, присвоена метка 18. LSR 1 помещает это значение в поле выходной метки своей таблицы.

На стадии D устройство LSR 1, которому известно значение метки для потока, адресованного на префикс 193.233.48, присваивает собственное значение метки данному FEC и уведомляет верхнего соседа (LSR 0) об этой привязке. Теперь LSR 0 записывает полученную информацию в свою таблицу. После завершения данного процесса все готово для передачи пакетов из сети «клиента» в сеть с адресом 193.233.48.0, т.е. по выбранному пути LSP.

Спецификация класса FEC может содержать несколько компонентов, каждый из которых определяет набор пакетов, соответствующих данному классу. На сегодняшний день определены два компонента FEC: адрес узла (host address) и адресный префикс (address prefix). Пакет классифицируется как принадлежащий к данному классу FEC, если адрес получателя точно совпадает с компонентом адреса узла либо имеет максимальное совпадение с адресным префиксом. В нашем примере узел LSR 0 выполняет в процессе передачи классификацию пакетов, поступающих к нему из сети клиента, и (если адрес получателя в них совпадает с префиксом 193.233.48), присвоив пакету метку 33, отправляет его через интерфейс 2.

2.6. Качество обслуживания в сетях MPLS

Обеспечение функций QoS – это важнейший компонент технологии MPLS. В MPLS сети QoS-информация передается в поле CoS заголовка MPLS-метки. То есть, MPLS QoS базируется на CoS-битах MPLS-метки.

MPLS QoS достигается посредством выполнения двух главных логических шагов:

Таблица 7

MPLS QoS

Шаг	Место применения	Подходящие функции QoS	Действия QoS
1	Маршрутизатор на входе в MPLS-облако(пограничный маршрутизатор)	Согласование скорости доступа (Committed Access Rate - CAR)	Вариант 1.Механизм CAR ограничивает трафик на входном маршрутизаторе для всего поступающего в MPLS-облако IP-трафика. Он устанавливает для трафика значение IP-приоритета исходя из профиля трафика и существующих политик. Значение поля IP-приоритета пакета копируется в поле MPLS QoS. Вариант 2. Механизм CAR ограничивает трафик на входном маршрутизаторе для всего поступающего в MPLS-облако IP-трафика. Он устанавливает для трафика значение поля MPLS CoS исходя из профиля трафика и существующего контракта. В отличие от варианта 1, значение приоритета в IP-заголовке остается неизменным.
2	Вся MPLS-сеть	Взвешенный алгоритм равномерного обслуживания очередей (Weighted Fair Queuing - WFQ)	Дифференциация трафика в MPLS-магистрали на основании значения поля MLPS CoS с помощью функций IP QoS WFQ

Иными словами, в MPLS-сети используются следующие QoS функции:

- Механизм CAR, применяющийся к входящему трафику для установки поля MPLS QoS.
- В ядре сети используется предоставление дифференцированных услуг с использованием механизма WFQ и анализа битов поля MPLS CoS пакета.

Глава 3. Протокол RSVP

В главе 1 рассматривалась архитектура интегрированных услуг Int-Serv. Теперь, рассматривая архитектуру Int-Serv более подробно, обсудим способ информирования сети о нуждах различных потоков трафика. Для этой цели в архитектуре Int-Serv используется сигнальный протокол качества обслуживания – протокол резервирования ресурсов (Resource Reservation Protocol-RSVP). RSVP – это сигнальный протокол QoS, который позволяет конечным приложениям, требующим определенные гарантированные услуги, проводить сквозную сигнализацию своих QoS-требований.

Протокол RSVP сигнализирует о запросах резервирования ресурсов по доступному маршрутизируемому пути в сети. При этом RSVP не производит собственную маршрутизацию; напротив, этот протокол был разработан для использования других, более мощных, протоколов маршрутизации. При определении пути для данных и управляющего трафика RSVP полагается на используемый в сети протокол маршрутизации. После того как информация протокола маршрутизации адаптируется к изменениям в топологии сети, запросы резервирования протокола RSVP переносятся на новый путь. Подобная модульность помогает протоколу RSVP эффективно функционировать совместно с любой службой предоставления информации о маршрутах.

3.1. Работа протокола RSVP

Конечные системы используют протокол RSVP для запрашивания у сети определенного уровня QoS от имени потока данных приложения. RSVP-запросы передаются по сети при прохождении каждого узла, который используется для передачи потока. Протокол RSVP пытается зарезервировать ресурсы для потока данных на каждом из этих узлов.

RSVP-совместимые маршрутизаторы помогают доставить нужные потоки данных в нужную точку назначения.

Резервирование всегда должно следовать по одному и тому же одноадресному пути или по многоадресному дереву. В случае выхода из строя

линии связи маршрутизатор должен сообщить об этом RSVP-демону, чтобы генерируемые им RSVP-сообщения передавались по новому пути.

Процесс установки резервирования можно разбить на пять отдельных шагов.

1. Отправитель данных посылает управляющие сообщения RSVP PATH по тому же пути, по которому они отправляют обычный трафик с данными. В этих сообщениях описываются данные, которые уже отправляются или только будут отправляться. Сообщение-PATH содержит IP-адрес отправителя и получателя, а также информацию, характеризующую качество сервиса для потока и называемую FlowSpec.

2. Каждый RSVP-маршрутизатор перехватывает PATH-сообщения, сохраняет IP-адрес предыдущей точки назначения, записывает вместо него свой собственный адрес и отправляет обновленное сообщение дальше по тому же пути, по которому передаются данные приложения. Таким образом, сообщение PATH доходит до получателя.
3. Получатель посылает заявку на резервирование QoS с помощью RSVP RESV-сообщения, которое идет от получателя к отправителю в противоположном направлении по маршруту, пройденному RSVP PATH-сообщением.
4. RSVP-маршрутизаторы определяют, могут ли они удовлетворить эти RESV-запросы. Если нет, они отказываются в резервировании. Если да, то они отсылают запрос предыдущему маршрутизатору.
5. Отправители, получив запросы на резервирование ресурсов считают резервирование ресурсов состоявшимися. Таким образом реальное резервирование ресурсов осуществляется RESV-сообщениями.

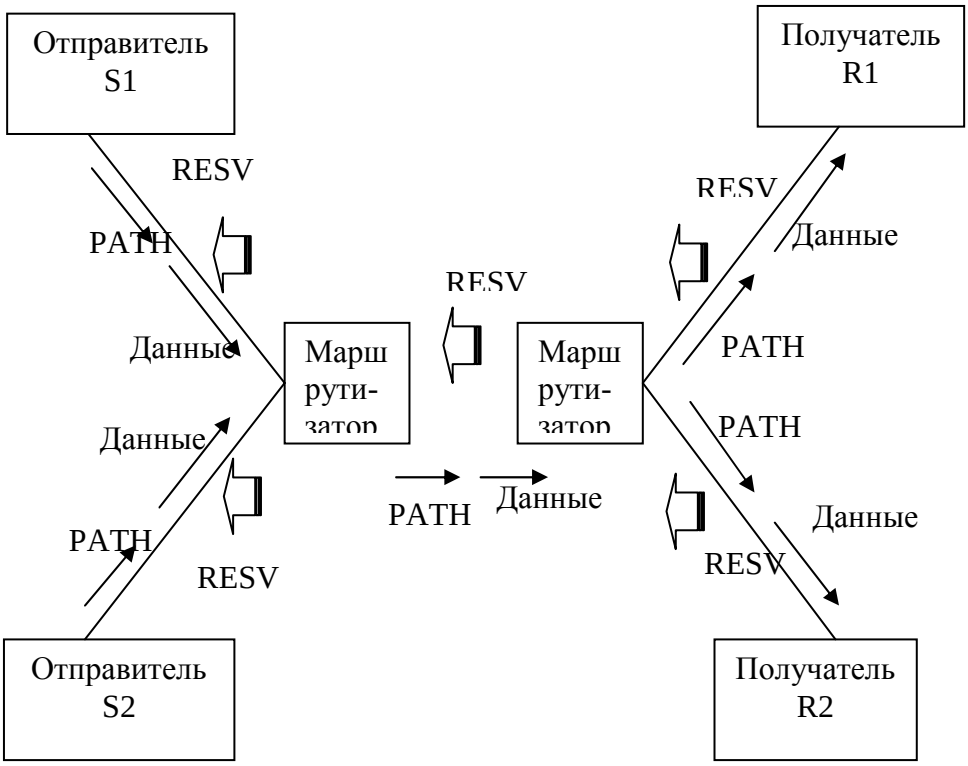


Рис.3.1 Механизм RSVP-резервирования ресурсов

Пакеты, идущие от приложения на машине-источнике к приложению на машине-получателе, формируют отдельный поток. С целью управления доступом требования потока представляются в виде параметров объекта FlowSpec.

3.2. RSVP-компоненты

- RSVP-отправитель (RSVP-sender) – это приложение, инициирующее отправку трафика в RSVP-сеансе;
- RSVP-получатель (RSVP-receiver) – это приложение, которое получает трафик в RSVP-сеансе. Во время конференций или при передаче голоса по протоколу IP приложение может играть роль и RSVP-отправителя, и RSVP-получателя;
- Сети, состоящие из RSVP-совместимых маршрутизаторов (RSVP-enabled router network), находящихся на пути от получателя к отправителю;

- Потоки (совокупность IP-пакетов, посылаемых отправителем одному или более получателям, с соответствующим потоку идентификатором — FlowLabel).

3.3. RSVP-сообщения

В протоколе RSVP используется семь типов сообщений: два обязательных – PATH И RESV – и пять опциональных – PATH ERROR, PATH TEARDOWN, RESV ERROR, RESV CONFIRM и RESV TEARDOWN. RSVP-маршрутизаторы и клиенты используют эти сообщения для создания и поддержки состояний резервирования.

RSVP-сообщения помогают создавать в маршрутизаторах гибкие состояния, которые необходимо периодически обновлять.

Типы сообщений отправителя:

- Отправители периодически отправляют PATH-сообщения, в которых указываются желательные характеристики качества обслуживания трафика – верхняя и нижняя границы полосы пропускания, величина задержки и вариация задержки. Сообщения-PATH должны нести в себе шаблон данных отправителя (Sender Template), описывающий тип этих данных. Шаблон специфицирует фильтр, который отделяет пакеты данного отправителя от других пакетов в пределах сессии (по IP-адресу отправителя и, возможно, по номеру порта). Кроме того, сообщение PATH должно содержать спецификацию потока данных отправителя Tspec, которая определяет характеристики этого потока. Спецификация Tspec используется, чтобы предотвратить избыточное резервирование.

  PATH-сообщения отправляются многоадресным группам или одноадресной точке назначения потока, для которого производится резервирование. При приходе PATH-сообщений маршрутизатор создает блок состояния пути (Path State Block – PSB).

  В PATH-сообщениях содержится периодический hello-интервал, указывающий на частоту посылки этих сообщений отправителем. По умолчанию hello-интервал равен 30 с. Очень важно иметь небольшой hello-интервал или быструю схему повторной передачи, поскольку из-за потери

PATH-сообщения может ухудшиться производительность VoIP-приложений вследствие задержки при установке RSVP-резервирования на пути VoIP-вызова.

- При обнаружении в PATH-сообщении ошибки (ошибок) получателем или маршрутизатором отправляется необязательное сообщение PATH ERROR, оповещающее отправителя о возникшей проблеме. Обычно это происходит из-за ошибки при проверке целостности сообщения.
- Сообщения PATH TEARDOWN, содержащее адрес источника отправителя, посылаются многоадресной группе в случае необходимости удаления пути из базы данных, при выходе из строя канала или если отправитель ликвидирует состояние прохода.

Типы сообщений получателя:

- Получатели периодически отправляют RESV-сообщения. Сообщения RESV несет в себе спецификацию FlowSpec, содержащую два набора параметров: спецификацию потока данных отправителя – Tspec, которая определяет характеристики потока, и спецификацию запроса - Rspec, в которой указываются нужные получателю параметры качества обслуживания, и спецификацию фильтра – FilterSpec, определяющую к каким пакетам сессии относится данная процедура (по IP-адресу отправителя и, возможно, по номеру порта). RESV-сообщения проходят через все RSVP-маршрутизаторы по маршрутизируемому пути к отправителю, для которого производится резервирование ресурсов. По приходе RESV-сообщений (FlowSpec, FilterSpec) маршрутизаторы создают блоки состояния резервирования (Reservation State Block –RSB).

  В RESV-сообщениях содержится периодический hello-интервал, указывающий на частоту посылки этих сообщений получателем.
- При обнаружении в RESV-сообщении ошибки (ошибок) отправителем или маршрутизатором отправляется сообщение RESV ERROR, оповещающее получателя о возникшей проблеме. Обычно это происходит из-за фундаментальной ошибки формата, ошибки при проверке целостности или из-за недостатка свободных ресурсов для предоставления запрашиваемых гарантий.

- Если RESV-сообщение используется для сквозного резервирования ресурсов и получатель запрашивает подтверждение резервирования, то получателям или маршрутизаторам, расположенным в точках объединения запросов, отправляется сообщение RESV CONFIRM.
- Сообщения RESV TEARDOWN отправляются, когда блок RSB должен быть удален из базы данных вследствие выхода из строя линии связи или когда отправитель уничтожает состояние резервирования.

3.4. Стили резервирования

Запрос резервирования включает в себя набор опций, которые в совокупности называются *стилем*. Одна опция резервирования определяет способ резервирования различными отправителями в пределах одной сессии. Другая опция резервирования контролирует выбор отправителей.
RSVP-резервирование ресурсов для потока можно разбить на два главных типа: индивидуальное и общее.

3.4.1. Индивидуальное резервирование

Индивидуальное резервирование (distinct reservations) применяется в тех приложениях, в которых несколько источников данных могут отправлять информацию одновременно. В видеоприложениях каждый отправитель генерирует индивидуальный поток данных, для которого необходимо осуществлять отдельное управление доступом и планирование очереди на всем пути к получателю. Следовательно, для такого потока необходимо осуществлять отдельное резервирование ресурсов для каждого отправителя и для каждого канала в пути.

Индивидуальное резервирование происходит для отправителя и устанавливается с помощью стиля резервирования с фиксированным фильтром (Fixed Filter –FF). Стиль FF использует опции: "четкое" (distinct) резервирование и "явный" (explicit) выбор отправителя. Символически запрос на резервирование в стиле FF можно представить как FF(S{Q}), где S-это отправитель, а Q-объект FlowSpec; эта пара параметров образуют дескриптор потока. RSVP позволяет применение нескольких простых стилей

резервирования FF одновременно, при этом формируется список дескрипторов потоков:

FF(S1{Q1}, S2{Q2}, ...)

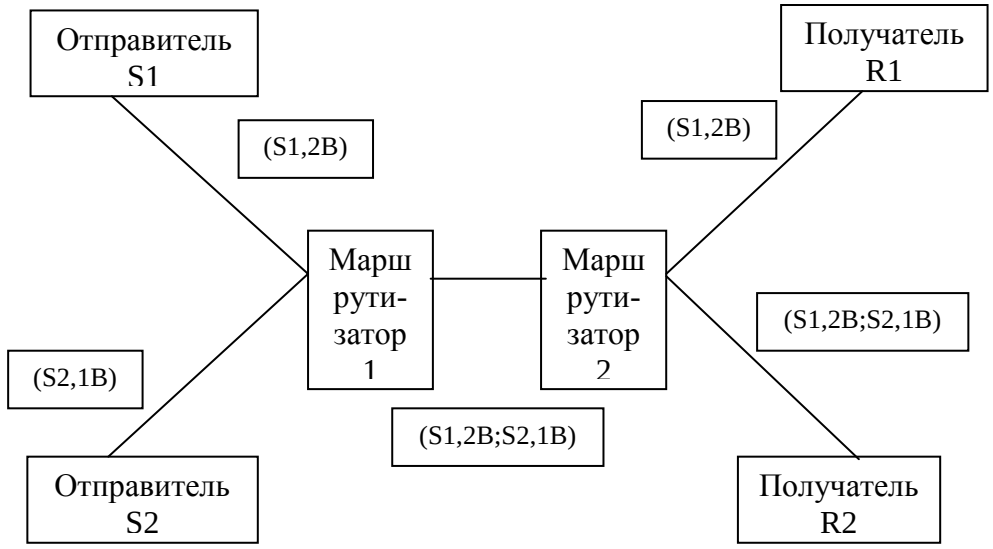


Рис.3.1 Фиксированный фильтр

Самый простой случай индивидуального резервирования ресурсов наблюдается на примере приложения с одноадреным трафиком, где есть только один отправитель и один получатель. Таким образом, простой запрос со стилем FF создает точно заданное резервирование для информационных пакетов от определенного отправителя, без совместного использования ресурса с другими отправителями в пределах одной и той же сессии.

3.4.2. Общее резервирование

Общее резервирование (shared reservation) применяется в тех приложениях, в которых несколько источников данных не склонно передавать одновременно, например цифровые аудиоприложения, такие, как приложения VoIP. В этом случае, поскольку в любой отдельно взятый промежуток времени разговор ведет небольшое число людей, информация передается лишь небольшим ограниченным количеством отправителей. Такой поток не нуждается в отдельном резервировании ресурсов для каждого отправителя, для него необходимо всего лишь одно резервирование, которое при необходимости можно будет применить к любому отправителю в группе.

В терминах протокола RSVP такой поток называется общим потоком (shared flow); он устанавливается с помощью общего явного или группового резервирования.

При общем явном (Shared Explicit – SE) резервировании потоки, которые резервируют сетевые ресурсы указываются отдельно. Стиль SE использует опции: "разделенное" (shared) резервирование и "явный" (explicit) выбор отправителя. Таким образом, стиль резервирования SE формирует одно резервирование, которое совместно используется несколькими отправителями.

Символически запрос на резервирование в стиле SE можно представить как SE((S1,S2) {Q}), где S1,S2,…-отдельные отправители, требующие резервирования ресурсов, а Q-объект FlowSpec.

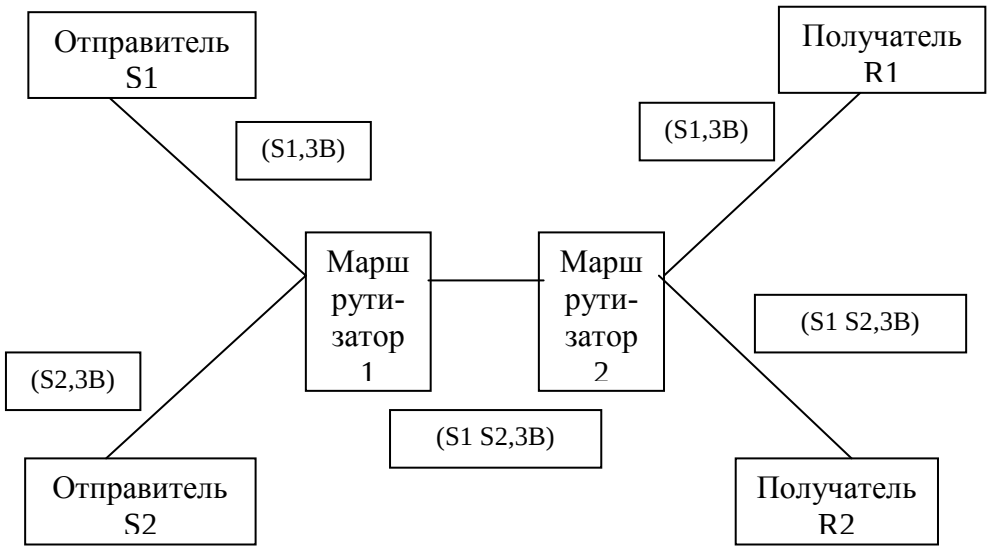


Рис.3.2 Общий явный фильтр

- С помощью группового фильтра (Wildcard Filter – WF) полоса пропускания и характеристики задержки могут быть зарезервированы для любого отправителя. Такой фильтр не позволяет указать отправителей отдельно – он принимает всех отправителей, на что указывает установка адреса источника и порта в ноль.

  Символически запрос на резервирование в стиле WF можно представить как WF(*{Q}), где символ “*” представляет собой групповой символ выбора отправителей, а Q – объект FlowSpec.

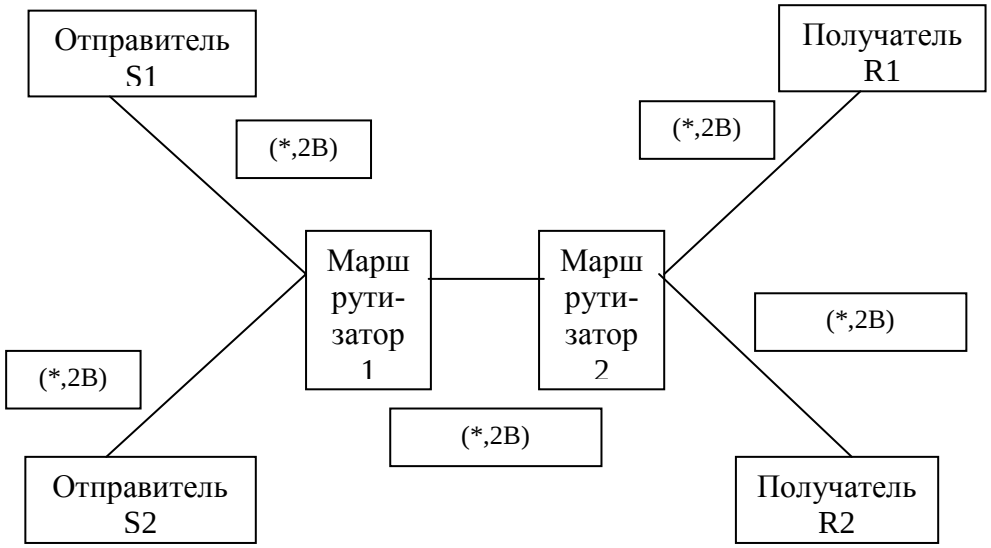


Рис.3.3 Групповой фильтр

Протокол RSVP не позволяют объединять индивидуальное резервирование с общим резервированием, так как эти модели абсолютно несовместимы. Не допускается также объединение явного и произвольного выбора отправителей, так как это может вызвать предоставление не заказанных услуг получателю, который указал тип услуг явно. Таким образом, стили WF, SE и FF не совместимы.

3.5. Типы услуг

Протокол RSVP предоставляет два типа интегрированных услуг, которые получатели могут запрашивать с помощью сообщений RSVP RESV: службу регулируемой нагрузки и службу гарантированной битовой скорости.

3.5.1. Регулируемая нагрузка

Служба регулируемой нагрузки (control load service) обеспечивает гарантию того, что зарезервированный поток достигнет своего пункта назначения с

минимальным вмешательством со стороны трафика, доставляемого без гарантий.

Как правило, служба регулируемой нагрузки применяется при передаче трафика Internet-приложений, чувствительных к перегрузкам в сети. Такие приложения отлично работают в незагруженных сетях, но при перегрузке “сразу приходят в негодность”.

3.5.2. Гарантированная битовая скорость

Служба гарантированной битовой скорости (guaranteed bit rate service) обеспечивает ограничение задержки без отбрасывания датаграмм, удовлетворяющих параметрам трафика, в условиях отсутствия сбоев в работе сетевых компонентов или изменений в информации о маршрутах во время жизни потока. Эта служба гарантирует минимальное вмешательство со стороны трафика, доставляемого без гарантий, изоляцию зарезервированных потоков и числовое выражение максимальной задержки очереди.

Максимальная задержка очереди – это задержка передачи PATH-сообщения от источника до получателя.

Служба гарантированной битовой скорости лучше всего подходит для тех приложений масштаба реального времени, которые позволяют воспроизводить аудио- и видеофайлы. Подобные приложения используются для нормальной работы буфер с целью компенсации неравномерности прибытия пакетов. Определяя максимальную задержку очереди, служба гарантированной битовой скорости помогает оценить необходимый размер буфера.

Глава 4. Сравнение технологий MPLS и RSVP

В главах 2 и 3 был проведен анализ технологий MPLS и RSVP. В этой главе сравним эти протоколы по ряду параметров. А также сделаем выводы о целесообразности их использования.

4.1. Затраты на внедрение технологий

RSVP предполагает весьма значительные накладные расходы, так как каждый узел вдоль пути следования пакетов должен согласиться предоставить запрошенное качество. Следовательно, внедрение RSVP потребует обновления всех маршрутизаторов, чтобы резервирование ресурсов могло производиться на всем протяжении открываемого соединения. Хотя, RSVP способен обеспечивать прозрачность операций для маршрутизаторов, которые его не поддерживают.

При внедрении многопротокольной коммутации с использованием меток MPLS также требуется обновление маршрутизаторов, так как они должны понимать “меченые потоки” и уметь соответствующим образом реагировать на них. Однако, сложные операции классификации, маркировки, определения правил обслуживания и формирования трафика необходимо выполнять только на границах сети. А транзитные коммутаторы должны осуществлять только функцию коммутации. Следовательно, оборудование для транзитных коммутаторов проще, а значит внедрение сети на базе технологии MPLS связано с меньшими материальными затратами, чем при внедрении сети RSVP.

Стоит сказать, что стоимость маршрутизаторов без поддержки MPLS в 4 раза дешевле, чем с поддержкой. Но при такой разнице в цене использование MPLS все равно оправдывает себя.

4.2. Масштабируемость

Масштабируемость(scalability) означает, что сеть позволяет наращивать количество узлов и протяженность связей в очень широких пределах, при этом производительность сети не ухудшается.

Протокол RSVP для резервирования ресурсов производит большую вычислительную обработку и потребляет память. Объем ресурсов, которые необходимы маршрутизатору для обработки и хранения информации RSVP,

увеличиваются пропорционально количеству резервирований. Таким образом, поддерживая много RSVP резервирований можно получить отрицательный результат, так как одновременная активность большого количества потоков может вызвать перегрузку маршрутизаторов, что влечет блокировку и задержку передачи данных. Следовательно, существует ограничение на число одновременных сеансов, которые маршрутизаторы способны эффективно обслуживать. Протокол RSVP с резервированием ресурсов для каждого потока хорошо масштабируется в корпоративных сетях среднего размера и магистралях поставщиков услуг Internet (Internet Service Provider-ISP) протокол RSVP хорошо масштабируется при условии использования больших многоадресных групп или объединения потоков на границе сети. Агрегирование RSVP-резервирований предполагает слияние нескольких сквозных резервирований, имеющих общие маршрутизаторы входа и выхода, в одно большое сквозное резервирование. Другой подход к решению проблемы масштабируемости протокола RSVP в ядре крупной сети заключается в использовании протокола RSVP на границах сети и реализации DiffServ-услуг в ее магистрали.

Из выше сказанного можно сделать вывод, что RSVP имеет проблемы с масштабированием. Протокол RSVP лучше всего работает при разумном числе одновременно поступающих заказов. Таким образом, основная проблема заключается в том, чтобы ограничить число одновременных соединений.

В будущем сети поставщиков услуг и Internet в большинстве случаев будут иметь достаточно широкую полосу пропускания для передачи обычного телефонного трафика. При проектировании сети, имеющей достаточный запас полосы пропускания, весь телефонный трафик можно выделить в отдельный класс. Ширина канала телефонного трафика зависит от ширины доступной полосы пропускания сети. Поскольку этот канал будет являться частью общей полосы пропускания, ему не нужно будет выделять ресурсы для каждого отдельного вызова.

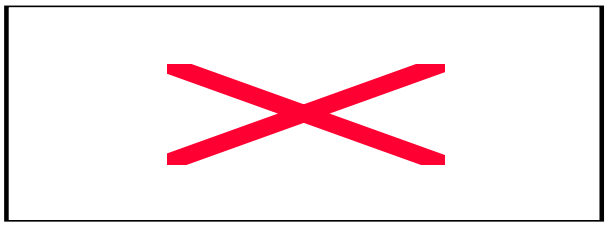
Теперь рассмотрим масштабируемость протокола MPLS. Считав метку, каждый маршрутизатор узнает информацию о следующем адресате на пути, не анализируя заголовок пакета. Это экономит время и ресурсы. Пакеты с метками MPLS могут, следовательно, передаваться от отправителя до получателя без

задержек на обработку. Подобный механизм делает MPLS гораздо более масштабируемым, чем RSVP.

4.3. Перегрузки

Протокол RSVP определяет для трафика кратчайший маршрут.

Рис.4.1 Неэффективность загрузки ресурсов сети, путями определенными протоколом RSVP



Примером неэффективности такого подхода служит сеть с топологией, приведенной на рисунке 4.1.. Несмотря на то, что между маршрутизаторами A и E имеется два пути: верхний, через маршрутизатор B, и нижний, через маршрутизаторы C и D, — весь трафик от A к E в соответствии с принципами маршрутизации, принятыми в протоколе RSVP, направляется по верхнему пути. Только потому, что нижний путь немного длиннее, чем верхний (в нем на один транзитный узел больше), он игнорируется, хотя мог бы задействоваться параллельно с верхним путем.

Заметим, однако, что при наличии в сети нескольких альтернативных маршрутов равной стоимости (метрики), трафик делится между ними, и нагрузка на маршрутизаторы и каналы связи распределяется более сбалансировано. Но когда стоимость альтернативных маршрутов даже незначительно хуже, чем у кратчайшего маршрута, этот инструмент не работает. В результате сеть работает неэффективно. Следовательно, RSVP не решает проблему перегрузок.

Теперь предположим, что сетевой администратор, проанализировав статистику загрузки сети, решил установить правила управления трафиком для того, чтобы уменьшить нагрузку на маршрутизатор B. Для этого ему необходимо перенаправить часть трафика по другим маршрутам, скажем, часть трафика от маршрутизатора A к маршрутизатору E перевести на нижний путь (а часть оставить на прежнем пути). Осуществить такое разделение средствами RSVP маршрутизации было бы невозможно, поскольку она принимает во внимание только адрес назначения пакета, одинаковый в обоих случаях. Но при

использовании протокола MPLS в ядре сети, можно реализовать такие правила достаточно просто. Для этого нужно сконфигурировать два маркированных маршрута так, чтобы маршрутизатор A направлял часть трафика от A к E по верхнему пути, а другую часть - по нижнему пути. Возможность классифицировать трафик по множеству параметров и направить трафик каждого класса по выбранному и, возможно, специально оптимизированному пути позволяет администратору точно управлять потоками трафика.

Таким образом, можно сделать вывод о том, что протокол MPLS, в отличие от RSVP, позволяет направлять трафик через менее загруженные узлы и каналы связи. Коммутируемые по меткам пути (Label Switched Path, LSP) могут изменяться в зависимости от состояния сети и загруженности отдельных ее узлов или каналов. MPLS поддерживает механизм управления трафиком TE (traffic engineering), позволяющий строить оптимальные пути прохождения IP-пакетов через сеть. Таким образом, с помощью MPLS решается проблема непредсказуемости задержек в IP-сети. При надлежащем планировании маршрутов и правил технология MPLS обеспечивает высокий уровень контроля над трафиком. Это означает более эффективную работу сетей, более предсказуемое качество услуг и большую гибкость, позволяющую адаптироваться к изменяющимся потребностям пользователей.

4.4. Время установления соединения

Исходя из описания процесса резервирования ресурсов с помощью протокола RSVP, можно сделать вывод, что на организацию резервирования тратятся большие интервалы времени:

- Отправитель посылает сообщение RSVP PATH;
- Маршрутизаторы пересылают его, запоминая маршрут;
- Получатель посылает RSVP RESV-сообщение;
- Каждый маршрутизатор на пути резервирования определяет, может ли он удовлетворить запрос;
- Отправитель, получив RSVP RESV-сообщение, считает резервирование законченным.

В MPLS пути всегда устанавливаются заранее, до передачи трафика. Метки создаются и согласовываются между смежными узлами до начала поступления пользовательских данных. Поэтому, если требуемый маршрут содержится в

таблице пересылки, то ему уже присвоена метка и поступающий трафик может быть немедленно передан с этой меткой (исключается начальная пауза).

4.5. Время пересылки пакетов

Протокол RSVP после процедуры резервирования присваивает каждому маршруту определенный индекс. Это исключает просмотр таблицы маршрутизации.

MPLS дает возможность ускоренного продвижения пакетов, так как при использовании протокола MPLS маршрутизация на всем пути заменяется коммутацией – более скоростным и экономичным способом продвижения. Основная экономия достигается за счет того, что решение о дальнейшем продвижении пакета принимается после просмотра не гигантской таблицы маршрутизации, которая может состоять из сотен тысяч записей, а таблицы коммутации, включающей только действующие пути коммутации меток.

4.6. Затраты ресурсов сети на обслуживание

Протокол RSVP каждый поток обслуживает отдельно. Для каждого потока производит отдельное резервирование. В процессе обслуживания для каждого потока служебная информация передается отдельно. Это увеличивает затраты ресурсов сети.

Протокол MPLS позволяет осуществлять мультиплексирование, объединять однотипные потоки. А затем весь трафик с одинаковыми метками рассматривается единым образом. Это уменьшает затраты ресурсов сети на обслуживание.

4.7. Пропускная способность

Ни RSVP, ни MPLS не позволяют увеличить пропускную способность сети. Протокол RSVP позволяет лишь перераспределять сетевые ресурсы. RSVP выбирает кратчайший путь до получателя, следовательно, не решает проблему с нехваткой пропускной способности.

Но при использовании MPLS IP-трафик можно направлять по множеству, а не по одному маршруту. Таким образом, можно избежать проблем, связанных с недостаточной пропускной способностью каналов.

4.8. Классификация пакетов

Протокол RSVP присваивает определенный индекс каждому маршруту. Поэтому если каждый пользователь начнет открывать сеансы RSVP, то в конце концов таблицы потоков станут настолько большими, что по сравнению с ними даже таблицы маршрутизации будут гораздо более эффективными, а сама сеть начнет давать сбои.

При использовании технологии MPLS входной маршрутизатор классифицирует пакет и присваивает ему метку. Метка должна быть уникальной лишь для каждой пары смежных LSR – транзитных маршрутизаторов. В связи с этим обстоятельством вероятность того, что пространство меток будет исчерпано, очень мала.

4.9. Установление маршрута

Протокол RSVP устанавливает маршрут при поступлении каждого нового потока трафика.

При использовании MPLS установление и переустановление маркированных маршрутов происходит только при изменении топологии сети или получении соответствующей управляющей информации.

4.10. Совместимость с технологиями коммутации каналов

RSVP может использоваться в паре с Frame Relay или ATM. Сеанс RSVP применяется для управления параметрами качества обслуживания коммутируемых соединений ATM или Frame Relay.

MPLS позволяет применять метки в сетях, где используются технологии канального уровня, и прокладывать маркированные маршруты через неоднородные сетевые инфраструктуры. MPLS позволяет сервис-провайдерам использовать имеющуюся ATM-магистраль в качестве базовой инфраструктуры и пополняет набор предлагаемых ими сервисов за счет множества свойств интеллектуальной технологии третьего уровня. В коммутаторах ATM, поддерживающих MPLS, обычный ATM-трафик и MPLS-трафик обрабатываются раздельно — эта пара никогда не пересечется. MPLS-трафик передается по своим виртуальным соединениям и путям отдельно от

классического ATM-трафика. Для передачи MPLS- и ATM-трафика может совместно использоваться один и тот же порт. В этом случае нужно разделить между ними его ресурсы (пропускную способность). Это разделение должно базироваться на потребностях вышележащих служб или протоколов, информация которых транспортируется с помощью MPLS и классических механизмов ATM.

MPLS также совместим и с Frame Relay. Организация IETF уже разработала проект документа, где описано, как трафик MPLS должен обрабатываться маршрутизаторами Frame Relay.

4.11. Построение сети

При организации резервирования с использованием протокола RSVP на каждый маршрутизатор ложиться ряд задач. При передаче отправителем сообщения RSVP PATH каждый маршрутизатор, поддерживающий протокол RSVP, анализирует это сообщение и запоминает маршрут передачи. При передаче получателем сообщения RSVP RESV каждый маршрутизатор определяет возможность удовлетворения запроса (правомерен ли запрос и имеются ли у маршрутизатора ресурсы, необходимые для поддержки необходимого качества обслуживания). Таким образом, для успешного резервирования маршрутизаторы должны обладать большими вычислительными ресурсами, большими объемами памяти и функциями, поддерживающими RSVP.

Протокол MPLS упрощает построение сети за счет интеграции функций второго (коммутация) и третьего (маршрутизация) уровней. Маршрут определяется только входным маршрутизатором LER. Транзитные маршрутизаторы LSR лишь осуществляют коммутацию по меткам. Следовательно, им не требуются ни большие объемы памяти, ни вычислительные ресурсы. Это значительно упрощает построение транзитных коммутаторов и сети в целом.

4.12. Безопасность

Следующая сложность, связанная с использованием RSVP, — несанкционированный захват или сокрытие сетевых ресурсов. Такая ситуация возможна в случае, если злоумышленник прослушивает сеть с целью перехвата сообщений RSVP PATH. В IETF разрабатывается документ RSVP Cryptographic Authentication для обеспечения криптозащиты передаваемых служебных данных; в будущем он, вероятно, обеспечит кардинальное решение.

Протокол MPLS может обеспечить безопасность. В частности благодаря этой возможности протокол MPLS используют для организации виртуальных частных сетей (VPN). Виртуальная частная сеть моделирует работу корпоративной территориально распределенной сети с помощью инфраструктуры общего пользования Интернет. Технология MPLS обеспечивает безопасность, поскольку она предусматривает маршрутизацию пакетов на основе меток, а не на основе адреса назначения. MPLS позволяет поставщику услуг организовать предоставление услуг VPN, используя простой, гибкий и мощный механизм туннелирования. Виртуальная частная сеть строится как совокупность маркированных маршрутов между различными физическими сегментами VPN. Система маршрутизаторов провайдера распределяет по всей сети информацию о масках подсетей, существующих внутри каждого сегмента. Входные LSR-маршрутизаторы сети направляют трафик VPN по соответствующим LSP-маршрутам, исходя из совокупности адреса назначения пакета и его принадлежности к определенной VPN. Фактически LSP являются туннелями, хорошо защищенными от проникновения в них трафика пользователей других IP VPN, а также хакеров.

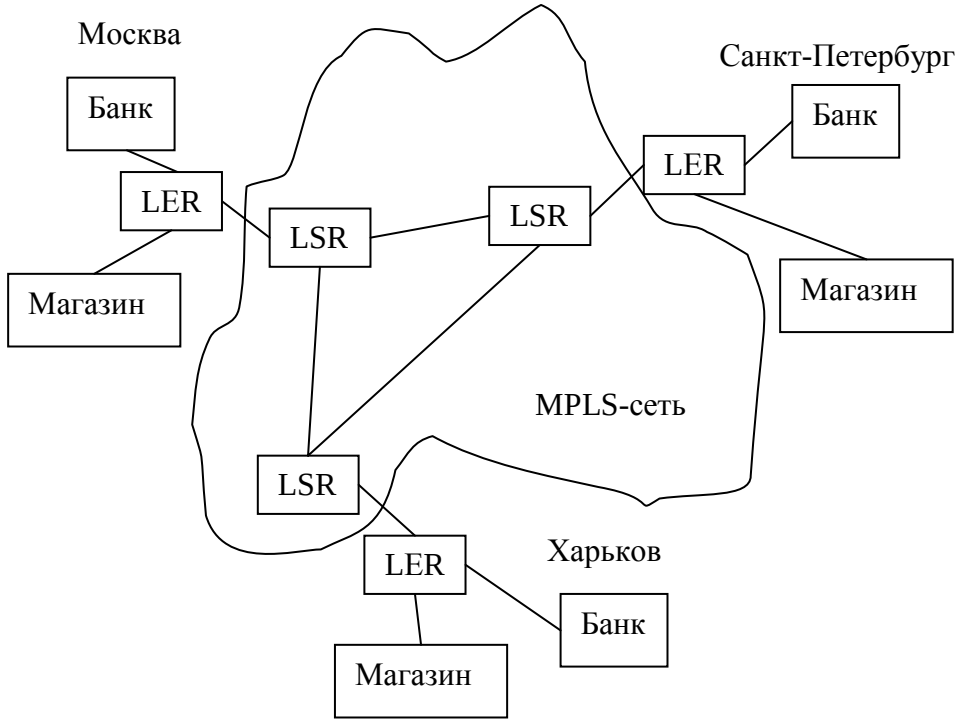


Рис.4.2 Пример объединения банка и магазина с филиалами в разных городах в виртуальные частные сети на основе технологии MPLS

Допустим банк имеет филиалы в разных городах. Сотрудникам банка необходимо ежедневно обмениваться конфиденциальными данными. Следовательно, необходимо построение своей сети. Для этого прокладываются либо свои каналы, либо каналы арендуются. Это требует больших капитальных вложений как на постройку (аренду), так и на обслуживание.

Другим решением данной проблемы является обращение к поставщику услуг, который по всей стране имеет точки присутствия. Это решение более экономически выгодно.

Особая конфигурация пограничных маршрутизаторов не позволяет маршрутизаторам клиентских сетей получать информацию друг о друге, а следовательно, и обмениваться трафиком. На пограничных маршрутизаторах создаются таблицы маршрутизации и продвижения пакетов (VPN Routing and Forwarding, VRF) для каждой обслуживаемой IP VPN. Фактически на пограничном маршрутизаторе функционируют несколько виртуальных маршрутизаторов, каждый из которых работает с таблицей VRF определенной виртуальной сети. К каждой такой таблице можно получить доступ только с клиентских маршрутизаторов, относящихся к одной и той же VPN.

Безопасность между виртуальными частными сетями обеспечивается за счет того, что маршрутные объявления блокируются в пограничных маршрутизаторах LER и рассылаются только филиалам своего офиса. Маршрутное объявление сообщает о подключении компьютера. Филиалы других офисов не увидят этого подключения.

Однако информация может быть взломана, так как между LER пакеты разных виртуальных частных сетей передаются по одному пути. Повысить степень защищенности IP VPN на базе MPLS можно, например, с помощью средств шифрования, контроля доступа и аутентификации (авторизации) пользователей. Эти средства могут находиться под управлением как оператора, так и пользователя VPN.

В частности, шифрация информации производится по следующему принципу: на одну метку накладывают другую метку – стек меток. При перехвате получается сложение этих меток. Таким способом обеспечивается безопасность.

Итак, технология MPLS позволяет четко обособлять друг от друга виртуальные корпоративные IP-сети.

4.13. Обслуживание мультикастных приложений

Протокол RSVP позволяет выполнять резервирование как для уникастных, так и для мультикастных приложений. Процесс резервирования распространяется от получателей к отправителям, от узла к узлу. Для мультикастной доставки данных необходимо объединение RSVP-резервирований. Запросы резервирования, поступающие от получателей мультикастинг-дерева должны объединяться по мере продвижения процесса резервирования в направлении отправителя данных. В каждой точке объединения запросов резервирования RSVP должен объединять запросы резервирования от узлов путем выбора максимума их спецификаций FlowSpecs. В каждом узле требования резервирования объединяются и сопоставляются с имеющимися возможностями. Это продолжается до тех пор, пока запрос не достигнет отправителя или пока не возникнет конфликт перегрузки.

Базовая модель резервирования RSVP является однопроходной: получатель посылает запрос резервирования вдоль мультикастинг-дерева отправителю

данных и каждый узел по пути воспринимает или отвергает этот запрос. Результаты доставляются протоколом RSVP в ЭВМ получателя.

Технология MPLS также предусматривает обслуживание мультикастных приложений. Но специально это не оговорено. В технологии это заложено изначально. Пакеты всех потоков, пути следования которых через сеть (или часть сети) совпадают, относятся к одному классу эквивалентности пересылки FEC и обслуживаются одинаково.

4.14. Действия протоколов RSVP и MPLS при обрыве связи

Рассмотрим действия протоков RSVP и MPLS при обрыве связи между двумя соседними маршрутизаторами, через которые производилась передача данных.

Как описывалось раньше, в протоколе RSVP сначала устанавливается путь с резервированием ресурсов, а потом идет обмен данными строго по этому пути. RSVP использует подход "soft state" (гибкое состояние) для управления состоянием резервирования в маршрутизаторах и персональных компьютерах. Поэтому, в то время как осуществляется передача данных, состояние резервирования периодически освежается посредством сообщений Path и Resv, а соседние маршрутизаторы обмениваются сообщениями “Hello” через каждые 30 миллисекунд. Если происходит разрыв связи, то соседние маршрутизаторы не получают сообщение “Hello”. Тогда каждый маршрутизатор ждет еще 120 миллисекунд (4 пропущенных сообщения “Hello”) и только тогда прекращает передачу данных. За это время происходит потеря данных. Так как маршрут становится недоступным, следующее сообщение Path инициализирует состояние прохода для нового маршрута, а последующие сообщения Resv установят для него резервирование. Состояние же на неиспользованном в данный момент сегменте маршрута будет аннулировано по таймауту. Таким образом, RSVP автоматически адаптируется к изменениям в маршруте.

В протоколе MPLS все пути прописаны заранее. Также в каждом маршрутизаторе заложены сведения о запасных путях. Таким образом, если основной путь перегружен или если на нем произошел обрыв, то маршрутизатор автоматически передает данные по запасному маршруту.

Поэтому в этом случае потеря данных минимальна. Задержки и потеря данных происходит только при “переключении” на дополнительный путь. Задержки составляют микросекунды.

Таким образом, время реакции на “ошибку” в протоколе RSVP измеряется в МИЛЛИсекундах (0,001), а в MPLS в МИКРОсекундах (0,000001).

4.15. Вывод

Выводом из всего выше написанного является то, что применение протокола RSVP вряд ли выйдет за пределы корпоративных сетей. Передача мультимедиа приложений с использованием RSVP в пределах частных сетей оправдана более. RSVP больше пригоден для тех случаев, когда число приоритетных потоков невелико и они требуют умеренной полосы пропускания. А реализация технологии MPLS в сети крупного предприятия или по каналам глобальной сети оказывается более реальной задачей.

Качество обслуживания протокола RSVP лучше при разумном количестве абонентов. А при их увеличении качество значительно ухудшается. Протокол MPLS может благополучно функционировать при значительно большем количестве абонентов.

Приведем графики, наглядно показывающие сравнение технологий RSVP и MPLS.

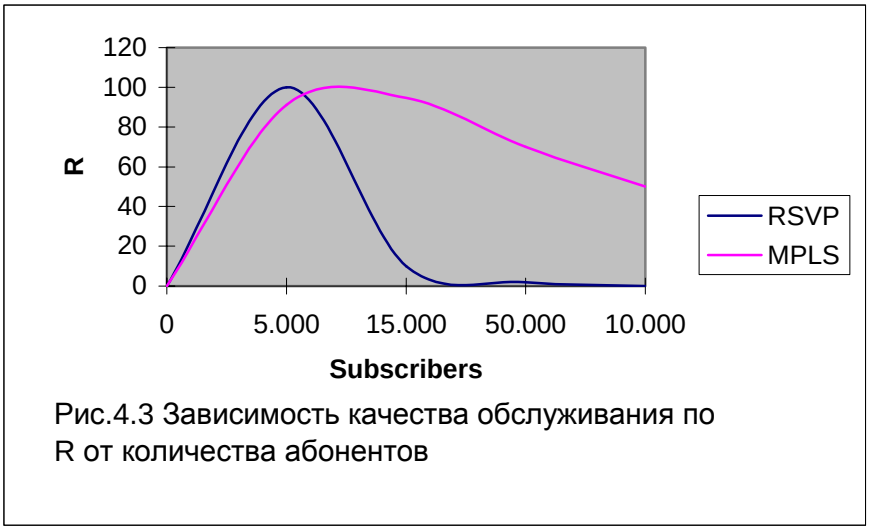


Рис.4.3 Зависимость качества обслуживания по R от количества абонентов

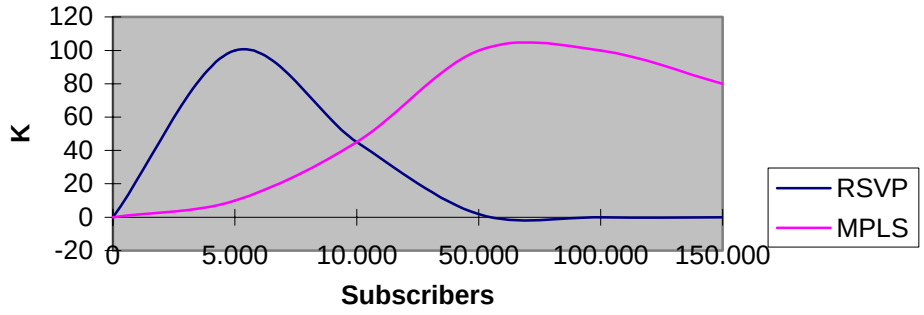


Рис.4.4 Целесообразность внедрения технологий RSVP и MPLS от количества абонентов

Где К - целесообразность внедрения технологий RSVP и MPLS по стобальной шкале

Следовательно, внедрение протокола RSVP целесообразнее при небольшом количестве абонентов, тогда как внедрение технологии MPLS возможно в более крупных сетях.

Список литературы:

1. Шринивас Вегешна. Качество обслуживания в сетях IP. - М.: Вильямс, 2003. – 368с.
2. Гольдштейн Б.С. и др. IP-телефония. – М.: Радио и связь, 2001, - 336с.
3. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, ротоколы – СПб.: Питер, 2002. – 672с.
4. RFC 2205. Resource Reservation Protocol (RSVP). Ver.1. Functional Specification. – September 1997.

   Статьи из журналов:
5. Современные аспекты IP-сетей. – Jet Info, №6 (73), 1999.
6. Тао Ма, Бингксин Ши. Контроль качества в сетях IP. Журнал сетевых рещений/LAN, 2001, №1.
7. Томас Нолл. Резервирование по сценарию RSVP. Сети, 1996, №9.
8. Игорь Алеексеев. Интеллектуальные услуги нового поколения Internet. Сети, 1996, №10.
9. Олифер В.Г., Олифер Н.А., Петрусов Д. ATM и MPLS – враги или союзники? NSI - Newbridge Systems Integration, 25.12.2002.
10. MPLS: много разговоров, мало внедрения. Телеком вести, 12.11.2001.
11. Олифер В.Г., Олифер Н.А. Искусство оптимизации трафика. NSI - Newbridge Systems Integration,17.01.2002.
12. Спенсер Жакалоне. Коммутация меток. Computerword, 2002, №33.
13. Том Нолле. MPLS: новый порядок в сетях IP? Журнал сетевых рещений/LAN, 1999, №5.
14. Реализация протокола MPLS в мультисервисных сетях операторского класса. Lucent Technologies, 06.2002.
15. Анита Карве. Качество услуг для трафика реального времени. Журнал сетевых решений/LAN, 03.1999, №3, с.107.
16. Питер Эшвуд-Смит и др. MPLS: лиха беда начало. Журнал сетевых решений/LAN,2000, №1.
17. Защита информации, виртуальные частные сети (VPN). Технология VIPNet. Служба рассылок Subscriber.ru, 18.12.2002, №6.
18. Технология MPLS. Технологии, стандарты, протоколы,28.05.2002.

19. Российские опыты с MPLS. Телеком вести,02.09.1999.
20. MPLS и QoS. Телеком вести, 09.04.2001.
21. MPLS будет поддерживать различные уровни QoS. Телеком вести, 09.07.2001.

ПРИЛОЖЕНИЕ

Технология MPLS

Общие сведения

Аббревиатура MPLS расшифровывается как Multi-Protocol Label Switching, то есть как “многопротокольная коммутация по меткам”. “Многопротокольной коммутацией” MPLS называется потому, что ее средства применимы к *любому* протоколу сетевого уровня.

Примечание. Можно встретить другие переводы словосочетания label switching - “коммутация меток”, “коммутация на базе (или, что то же, - на основе) меток”. Первый из этих переводов просто неверен - ведь коммутируются не метки, а пакеты. Второй несколько искажает действительное положение вещей - базу (или основу) коммутации составляют далеко не только метки; вернее всего было бы сказать “коммутация с использованием меток” (и такой перевод тоже встречается), но это слишком длинно.

Напомним, что первый, *физический уровень* (physical layer) содержит функции, обеспечивающие использование физической среды для двусторонней *передачи битов* (с такой достоверностью, какую обеспечивает эта среда) по прямому тракту, связывающему два узла сети, второй уровень – *уровень звена данных* (data link layer) - содержит функции, обеспечивающие формирование в этом тракте надежного логического звена связи, по которому обеспечивается двусторонний обмен между этими узлами *информационными блоками*, гарантируя при этом заданную достоверность передачи путем обнаружения и исправления ошибок, а третий, *сетевой уровень* содержит функции, обеспечивающие транспортировку информационных блоков по сети от отправителя к получателю через несколько узлов, выбирая подходящий маршрут транспортировки, который составляется из звеньев второго уровня. Общая идея протоколов всех уровней (кроме физического) состоит в том, что информационный блок каждого уровня содержит заголовок и информационное поле, и в том, что блок протокола вышележащего уровня помещается в информационное поле блока протокола расположенного сразу под ним нижележащего уровня.

При традиционной транспортировке пакета через сеть с использованием в уровне 3 протокола, не предусматривающего создания виртуальных соединений, каждый маршрутизатор на пути следования пакета самостоятельно принимает решение о том, к какому маршрутизатору переслать этот пакет дальше (такой способ транспортировки по-английски называется hop-by-hop). Иначе говоря, в каждом маршрутизаторе на пути следования пакета анализируется его заголовок и выполняется алгоритм сетевого уровня.

Примечание. Здесь и далее, говоря о “пересылке” пакета, мы имеем в виду его передачу к ближайшему маршрутизатору из тех, что расположены на предполагаемом (или возможном) пути следования этого пакета (то есть мы используем слово “пересылка” как эквивалент английского слова forwarding).

Однако в заголовке пакета содержится гораздо больше информации, чем нужно для того, чтобы выбрать следующий маршрутизатор. Этот выбор можно представить себе как выполнение двух групп функций. Одна группа разделяет все множество прибывающих пакетов на классы, которые удобно называть *классами эквивалентности пересылки* (Forwarding Equivalence Classes - FECs). Вторая группа ставит в соответствие каждому FEC определенное “направление” пересылки (слово “направление” написано в кавычках, поскольку в сети используется режим hop-by-hop, и разные пакеты одного и того же FEC могут пересылаться к разным маршрутизаторам, то есть физические направления пересылки могут быть разными). С точки зрения выбора следующего маршрутизатора все пакеты, принадлежащие одному FEC, неразличимы.

При *традиционной IP-маршрутизации* конкретный маршрутизатор может считать, что два пакета принадлежат одному и тому же FEC, если в его таблицах маршрутизации используется некий адресный префикс X, идентифицирующий “направление”, в котором предполагаемые маршруты транспортировки этих двух пакетов совпадают наиболее долго. По мере продвижения пакета по сети каждый следующий маршрутизатор анализирует его заголовок и приписывает этот пакет к такому из “своих” FEC, который соответствует тому же “направлению”.

При использовании *многопротокольной коммутации по меткам* (MPLS) определенный пакет приписывается к определенному FEC только один раз, когда он “входит” в сеть. Этому FEC присваивается *метка* - идентификатор фиксированной длины, передаваемый вместе с пакетом, когда тот пересылается к следующему маршрутизатору. Существенно,

что в остальных маршрутизаторах заголовок сетевого уровня не анализируется. Принятая маршрутизатором вместе с пакетом метка используется как указатель входа таблицы, которая определяет очередной маршрутизатор для пересылки к нему пакета, а также новую метку для FEC, к которому относится пакет. Заметим попутно, что маршрутизатор, поддерживающий MPLS и способный, кроме того, анализировать заголовки и производить пересылку пакетов (т.е. блоков сетевого уровня), не содержащих меток, называется *маршрутизатором коммутации по меткам* (сокращенно LSR - Label Switching Router).

Метод пересылки пакетов, принятый в MPLS, имеет ряд преимуществ перед методами, основанными на анализе заголовка блоков сетевого уровня. В частности, пересылку по методу MPLS могут выполнять маршрутизаторы, которые способны читать и заменять метки, но при этом либо вообще не способны анализировать заголовки блоков сетевого уровня, либо не способны делать это достаточно быстро. Другие преимущества MPLS станут ясны из дальнейшего.

Некоторые маршрутизаторы, обеспечивающие традиционную коммутацию пакетов, анализируют заголовки блоков сетевого уровня не только с целью выбрать следующий маршрутизатор, но и с целью определить приоритет пакета или присвоенный ему класс обслуживания. MPLS позволяет получить такого рода сведения, анализируя метку (но при этом не исключает и традиционного способа их получения).

Представим себе логически завершенный (и, в определенном смысле, автономный) фрагмент сети с коммутацией пакетов. Завершенность этого фрагмента выражается в том, что он имеет вполне определенную замкнутую границу, вдоль которой размещено N так называемых *пограничных узлов MPLS* (MPLS edge nodes). Помимо этих узлов, внутри фрагмента (для удобства мы будем называть его MPLS-сетью *i*-го уровня, а в тех случаях, когда это не вызывает двоякого толкования, - просто MPLS-сетью) имеется множество маршрутизаторов, каждый из которых имеет с остальными маршрутизаторами (в том числе и с пограничными узлами) либо прямые, либо коммутируемые связи. В последнем случае коммутация, необходимая для создания такой связи, производится другими маршрутизаторами из этого множества, которые не обязательно являются пограничными узлами MPLS и могут не иметь функций LSR. Более того, некоторые коммутируемые связи между LSR могут проходить через подсети, встроенные в рассматриваемую MPLS-сеть, но не содержащие в себе функции MPLS. К тому же, транспортировка пакетов от узла к узлу может производиться в режиме hop-by-hop.

MPLS-сеть *i*-го уровня представляет собой транзитную сеть, которая коммутирует однонаправленные потоки пакетов. Пакеты, поступающие в нее из смежной сети (которая может быть MPLS-сетью более высокого уровня, то есть содержать в себе рассматриваемую сеть *i*-го уровня), принимаются пограничным узлом MPLS, который является по отношению к этим пакетам *входным MPLS-узлом*. Пакеты, направляемые сетью в другую смежную сеть, передаются туда другим пограничным узлом, который является по отношению к этим пакетам *выходным MPLS-узлом*. В общем случае, все пакеты, транспортируемые через MPLS-сеть от входного MPLS-узла A к выходному MPLS-узлу B, принадлежат одному FEC и следуют по одному и тому же виртуальному *коммутируемому по меткам тракту* (Label Switched Path - LSP), который может проходить через несколько LSR и маршрутизаторов без функций LSR.

К выходному узлу MPLS поступают потоки пакетов от нескольких входных узлов. В промежуточных маршрутизаторах некоторые из этих потоков могут “сливаться”, то есть объединяться в один общий поток пакетов, которые приобретают в точке “слияния” общий FEC. Таким образом, множество трактов LSP, идущих к одному выходному узлу, образует ветвящееся дерево, корень которого находится в этом выходном узле.

Каждый из N пограничных узлов выполняет, в общем случае, функции и входного, и выходного узла, так что в MPLS-сети существует N такого рода деревьев, которые вместе содержат N(N-1) трактов LSP. Ясно, что через один промежуточный маршрутизатор может проходить несколько LSP, в том числе, LSP, принадлежащих разным деревьям. Если учесть, к тому же, что физическая топология сети отличается от топологии виртуальной сети LSP (и еще раз вспомнить про режим hop-by-hop), то станет ясно, что на практике могут возникать случаи “закольцовывания” путей прохождения пакетов, и, следовательно, в MPLS-сетях нужно предусматривать меры обнаружения и/или предотвращения таких случаев.

Метки и правила их использования

Метка представляет собой короткий идентификатор локального значения, используемый для идентификации класса эквивалентности пересылки FEC, то есть метка, помещенная в некоторый пакет, представляет FEC, к которому этот пакет относится. Как правило,

отнесение пакета к определенному классу производится на основе сетевого адреса пункта его назначения. Метка может быть помещена в пакет разными способами - вписываться в специальный заголовок, “вставляемый” либо между заголовками уровня звена данных и сетевого уровня, либо в свободное и доступное поле заголовка какого-то одного из этих двух уровней, если, конечно, таковое имеется. Очевидно, что вопрос о том, куда нужно помещать заголовок, содержащий метку, должен согласовываться между объектами, ее использующими. Заголовок этот имеет следующий вид.

Label	CoS	B	TTL

Label Значение метки, 20 битов

CoS Класс обслуживания, 3 бита

B Нижняя метка в стеке, 1 бит

TTL Время жизни пакета, 8 битов

Смысл содержимого двух последних полей будет ясен из дальнейшего.

Если два маршрутизатора, один из которых расположен ближе к отправителю пакета (его принято обозначать символом Ru - upstream router), а второй - ближе к получателю пакета (обозначаемый Rd - downstream router), являются маршрутизаторами коммутации по меткам LSR, между ними может быть согласовано, что когда передается пакет от Ru к Rd, то Ru снабжает этот пакет меткой L при условии, что он относится к FEC F. Иными словами, Ru и Rd имеют договоренность о том, что метка L связана с классом F для пакетов, проходящих от Ru к Rd, и является с точки зрения Ru *исходящей меткой*, представляющей FEC F, а с точки зрения Rd - *входящей меткой*, представляющей тот же FEC F. Подчеркнем, что эта договоренность имеет локальное значение, то есть существует только между Ru и Rd. Подчеркнем также, что Rd не может согласиться с Ru1 связать метку L с FEC F1, если он уже договорился с Ru2 о связи этой метки с FEC 2 (кроме

случая, когда Rd имеет возможность отличать пакеты, в которые метку L поместил Ru1, от пакетов, в которые эту же метку поместил Ru2).

Архитектурой MPLS предусматривается, что назначение метки, то есть ее привязку к определенному FEC, производит LSR, к которому приходит поток пакетов, принадлежащих этому FEC. По-английски такой LSR называется downstream LSR, то есть расположенный ниже по течению; для краткости мы будем называть его *нижним LSR*, а расположенный выше по течению upstream LSR будем называть *верхним LSR*. Таким образом, назначение меток всегда производится *снизу*. Затем нижний LSR информирует соответствующие верхние LSR о том, какие метки привязаны к каждому FEC поступающих к нему пакетов. Этот процесс называется *распределением меток*, и производится оно *снизу вверх*. Процесс поддерживается *протоколом распределения меток LDP* (*Label Distribution Protocol*).

Протокол распределения меток представляет собой набор процедур, при помощи которых производится обмен информацией о привязке меток к FEC между двумя (нижним и, всякий раз, одним из верхних) LSR. Речь идет именно об *обмене* информацией, поскольку протокол предусматривает двусторонний диалог взаимодействующих LSR, в ходе которого каждый из них получает сведения о возможностях другого.

Хотя “раздает” метки всегда нижний LSR, инициатором их распределения не обязательно должен быть он; процесс может инициировать и верхний LSR, направив к нижнему LSR соответствующий запрос. В той или иной сети может использоваться распределение меток либо только по запросам сверху, либо только по инициативе нижнего LSR, либо и то, и другое вместе.

Заметим, что нижний LSR распределяет метки не только по тем верхним LSR, которые имеют с ним прямые связи. Протокол распределения меток может быть использован и для диалога двух LSR, между которыми существует лишь коммутируемая связь, однако результат распределения в этом случае зависит от того, в каком из двух режимов, “либеральном” или “консервативном”, работает верхний LSR. Либеральный режим предусматривает, что метка, выданная тем нижним LSR, с которым нет прямой связи, запоминается и используется для пакетов того FEC, для которого она назначена. Такой режим удобен тем, что при реконфигурации сети соответствие между меткой и FEC сохраняется, даже если связь с LSR, определившим это соответствие, стала не коммутируемой, а прямой. Недостаток либерального режима состоит в том, что в верхнем LSR приходится хранить и обрабатывать заметно больше информации о соответствии

метка-FEC. Консервативный режим предусматривает, что метка, выданная тем нижним LSR, с которым нет прямой связи, игнорируется. Этот режим лишен вышеназванного преимущества либерального режима, но зато он исключает необходимость работать с большим количеством меток.

Как уже говорилось, метка всегда локальна, то есть обозначает некоторый FEC только для пары маршрутизаторов, между которыми имеется прямая или коммутируемая связь, и используется при пересылке пакетов этого FEC от того из маршрутизаторов данной пары, который является в ней верхним (Ru), к тому, который является нижним (Rd). Для пересылки пакетов того же FEC к следующему маршрутизатору используется другая метка, идентифицирующая этот FEC для новой пары маршрутизаторов, в которой маршрутизатор, бывший в предыдущей паре нижним, приобретает статус верхнего, а статус нижнего получает второй маршрутизатор этой новой пары. Отсюда ясно, что каждый маршрутизатор MPLS-сети должен хранить соответствие между всеми входящими и всеми исходящими метками, которыми он оперирует.

До сих пор молчаливо предполагалось, что пакет, транспортируемый через MPLS-сеть, имеет только одну метку. В действительности это не так: пакет может содержать несколько меток, которые образуют стек, организованный по принципу last-in, first-out, то есть метка, полученная последней, уничтожается первой и находится наверху стека. При пересылке пакета обрабатывается всегда верхняя метка, причем не имеет значения то обстоятельство, что эта метка могла прежде не быть верхней (над ней могли находиться другие метки), и что в данный момент другие метки имеются в стеке под этой меткой.

Примечание. Чтобы имелась возможность определить, что стек имеет *глубину*, превышающую “1”, используется бит B в заголовке метки. Из двух возможных значений этого бита одно помещается в поле B самой нижней метки, а второе - в поле B всех остальных меток.

Одним из источников появления в пакете дополнительной метки, помещаемой в стек, может быть то, что некий LSR*i*, находящийся на пути следования пакета с определенным FEC*j*, имеет (полные или частичные) сведения о том, куда будет передан (и/или к какому FEC*k* будет отнесен и с какой меткой L*l* будет пересылаться) этот пакет тем LSR*n*, который для текущего FEC*j* является последним (то есть конечным). В этом случае LSR*i* может поместить в пакет метку L*l*, а над ней - метку, которая в нем связана с FEC*j*.

Но основное назначение стеков меток состоит в том, чтобы поддерживать в MPLS-сети упоминавшуюся выше “древовидность” множества трактов LSP, заканчивающихся в одном входном LSR, и в том, чтобы использовать метки при создании так называемых LSP-туннелей. С поддержкой “древовидности” дело обстоит, в принципе, довольно просто. Если в одном LSP сливается несколько потоков пакетов (каждый поток - со своим FEC и со своей меткой), то этот LSP не заменяет метки, связанные с этими потоками, а оставляет их, помещая сверху метку нового FEC, который соответствует объединенному потоку пакетов, образующемуся в результате слияния. Поскольку дерево ветвится многократно, в каком-то другом LSR (находящемся ближе к корню) происходит слияние нескольких объединенных потоков, и в стеке появляется еще одна метка.

Что же касается LSP-туннелей, то мы вернемся к этому вопросу после того, как рассмотрим некоторые другие аспекты использования меток.

Под коммутируемым по меткам трактом LSP *уровня* ***m*** понимается LSP, образуемый последовательностью <R1, R2, ... , R*n*> маршрутизаторов, обладающей следующими свойствами:

Входным маршрутизатором R1 является LSR, который помещает в стек меток обрабатываемого пакета такую по счету метку, что стек приобретает *глубину* ***m***.

При всех *i* (1<*i*<*n*) пакет, поступающий к LSR R*i*, имеет стек меток глубины ***m***

В процессе транспортировки пакета от R1 к R(*n*-1) глубина его стека меток никогда не бывает меньше ***m***.

При всех *i* (1<*i*<*n*) R*i* передает пакет к R(*i*+1) средствами MPLS.

Иными словами, LSP *уровня* ***m*** представляет собой последовательность маршрутизаторов, начинающуюся с входного LSR, который вставляет в пакет метку уровня ***m***, и содержащую промежуточные LSR, каждый из которых принимает решение о пересылке пакета на основе метки уровня ***m***, а заканчивающуюся выходным LSR, где решение о пересылке принимается на основе метки уровня (***m*** - ***k***), ***k***>0, или на основе обычных (не MPLS) процедур пересылки.

Отметим, что в таком LSP имеется возможность передавать от R(*n*-1) к R*n* пакеты со стеком меток глубины (***m***-1), поскольку метка уровня ***m*** выходному LSR не требуется. Это позволяет избавить выходной LSR от операций анализа ненужной ему метки и не требует никаких дополнительных операций (кроме простого уничтожения верхней метки в стеке) в предпоследнем LSR.

Выше упоминалось о выборе “направления” для пересылки пакета; там же объяснялось, почему слово *направление* взято в кавычки. В действительности протокол MPLS поддерживает два способа определения маршрута, по которому будет следовать пакет. Первый из них аналогичен способу hop-by-hop, используемому сегодня в IP-сетях, и предполагает, что каждый маршрутизатор самостоятельно выбирает, куда переслать принятый им пакет, так что маршрут, по которому транспортируется пакет, оказывается, вообще говоря, случайным. Второй способ основан на том, что маршрутизаторы на пути следования пакета действуют не самостоятельно, а в соответствии с инструкциями, полученными от одного из LSR данного тракта LSP (обычно - от нижнего LSR, что позволяет совместить процедуру “раздачи” этих инструкций с процедурой распределения меток). Таким образом, маршрут следования пакетов однозначно определяется заранее, и слово *направление* применительно к пересылке пакета не требует кавычек.

Применение способа hop-by-hop имеет ряд преимуществ, связанных со сравнительной простотой его реализации, но, как уже отмечалось, не исключает феномена “закольцовывания” маршрутов. Простейший метод борьбы с этим феноменом дает использование в заголовке пакета (а для MPLS - в заголовке, содержащем метку) поля TTL - “время жизни” пакета. Входной MPLS-узел помещает в это поле определенное целое число, а каждый маршрутизатор, через который проходит пакет, уменьшает это число на единицу; если очередной (но не последний) маршрутизатор обнаруживает, что поле TTL содержит “0”, он отбрасывает пакет, и тот прекращает свое существование в сети.

Во многом из того, что изложено выше, просматривается “иерархичность” архитектуры MPLS. Она будет просматриваться более отчетливо после того, как мы опишем (столь же кратко, как и все остальное) идею формирования внутри LSP так называемых *LSP-туннелей*.

Идея “туннелирования” не является новой. Образование в виртуальном канале (или тракте) туннелей, по которым проходят другие виртуальные каналы (тракты), реализовано во многих современных протоколах и базируется на инкапсуляции “туннелируемых” пакетов в пакеты, следующие по “основному” виртуальному каналу и имеющие тот же адрес назначения, что и вставляемые (инкапсулируемые) в них пакеты. При этом туннели могут создаваться как по принципу hop-by-hop, так и по принципу использования заранее определенных маршрутов.

Применительно к MPLS мы будем говорить об LSP-туннелях, образуемых не путем инкапсуляции пакетов, а с помощью средств коммутации по меткам. LSP-туннель

представляет собой LSP <R1, R2, ... , R*n*>, в котором R1 является передающим конечным пунктом туннеля, а R*n* - приемным конечным пунктом туннеля. Пакеты, подлежащие транспортировке через LSP-туннель, относятся к одному FEC, и каждый LSR туннеля назначает метку для этого FEC, то есть метку для туннеля. Чтобы направить пакет в LSP-туннель, маршрутизатор передающего конечного пункта туннеля помещает метку, назначенную для этого туннеля, поверх существующего в пакете стека меток (заметим, что и в данном случае предпоследний маршрутизатор LSP-туннеля может уничтожить верхнюю метку в стеке до передачи пакета к приемному конечному пункту).

LSP-туннель создается внутри LSP. Существенно, что начало и/или конец туннеля, как правило, не совпадают с началом и/или концом этого LSP (туннель обычно бывает короче LSP, в котором он создан). В одном LSP может быть создано несколько LSP-туннелей одного уровня с несовпадающими передающими и/или приемными конечными пунктами. Более того, внутри любого из этих туннелей можно создавать LSP-туннели следующего уровня. Количество таких уровней, по тем или иным причинам, не может быть сколь угодно велико, однако “иерархичность” архитектуры MPLS в данном случае вполне очевидна.

Завершая краткое изложение принципов MPLS, нужно отметить, что в нем не были затронуты вопросы, связанные с решением в рамках MPLS задач обеспечения работы в режиме так называемой многоадресной рассылки пакетов (multicast mode). Разработчики MPLS убеждены, что такой режим вполне может быть реализован, однако в настоящее время эта проблема находится в стадии изучения.

Заключение

При подготовке данного материала преследовалась цель дать в сжатом виде общую характеристику MPLS и разъяснить, пусть очень кратко, наиболее характерные и важные, с точки зрения авторов, свойства и особенности этой системы. Мы надеемся, что, ознакомившись с этим материалом, специалист, которому приходилось заниматься проблемами распределения потоков при проектировании сетей связи или при их эксплуатации, сможет представить себе, какие огромные возможности в этой сфере несет в себе технология MPLS.

Для тех, кому понадобится (или захочется) получить об MPLS более глубокое представление, может оказаться полезным перечень литературных источников, которые были использованы авторами:

T.Li. MPLS and the evolving Internet architecture. IEEE Communications Magazine, Deccember 1999.

D.O.Awduche. MPLS and traffic engineering in IP networks. IEEE Communications Magazine, Deccember 1999.

A.Ghanwani, B.Jamoussi, D.Fedyk, P.Ashwood-Smith, L.Li, N.Feldman. Traffic engineering standards in IP networks using MPLS. IEEE Communications Magazine, December 1999.

G.Swallow. MPLS advantages for traffic engineering. IEEE Communications Magazine, Deccember 1999.

T/M/Chen, T.H.Oh. Reliable services in MPLS. IEEE Communications Magazine, Deccember 1999.

Y.Ohba. Issues on loop prevention in MPLS networks. IEEE Communications Magazine, Deccember 1999.

http://www.ietf.org/internet-drafts/draft-ietf-mpls-aech-07/txt

A.Vismanathan, N.Feldman, Zh.Wang, R.Callon. Evolution of Multiprotocol Label Switching. IEEE Communications Magazine, May 1998.

Разумеется, в этот перечень вошло далеко не всё, что уже написано про MPLS, но поскольку перечисленные источники сами содержат литературные ссылки, наш перечень будет нетрудно расширить.

Представляется также полезным привести расшифровку встреченных нами в литературе аббревиатур, хотя в данном материале из них использована лишь небольшая часть.

ARIS	Aggregate Route-based IP Switching - IP-коммутация на базе объединенных маршрутов
ARP	Address Resolution Protocol - протокол преобразования адресов
BGP	Border Gateway Protocol - протокол пограничного шлюза
BUS	Broadcast and Unknown Server - протокол вещательной рассылки
CIP	Classical IP - классический IP

CLNP Connectionless-mode Network Protocol - сетевой протокол с режимом без создания соединения

CoS Class of Service - класс обслуживания

CSR Cell Switch Router - маршрутизатор коммутации конвертов ATM

FEC Forwarding Equivalency Class - класс эквивалентных маршрутов

IETF Internet Engineering Task Force - рабочая группа инженерных сетевых задач Интернет

ILMI Integrated Layer Management Interface - интегрированный интерфейс эксплуатационного управления уровнями

IPX Internet Protocol Extended - расширенный протокол Интернет

ISP Internet Service Provider - поставщик услуг Интернет

LANE LAN Emulation - эмуляция локальной вычислительной сети

LDP Label Distribution Protocol - протокол распределения меток

LE_ARP LANE Address Resolution Protocol - протокол преобразования адресов LANE

LECS LANE Configuration Server - сервер конфигурации LANE

LES LAN Emulation Server - сервер эмуляции LANE

LIS Logical IP Subnet - логическая подсеть IP

LLC Logical Link Control - управление логическим звеном

LSP Label Switched Path - коммутируемый по меткам тракт

LSR Label Switching Router - маршрутизатор коммутации по меткам

MAC Medium Access Protocol - протокол доступа к среде

MARS Multicast Address Resolution Server - сервер преобразования адресов при многоадресной рассылке

MPLS Multiprotocol Label Switching - многопротокольная коммутация по меткам

MPOA Multiprotocol Over ATM - многопротокольная транспортировка поверх ATM

MTU Maximum Transmission Unit - транспортный блок максимального размера

NBMA Nonbroadcast Multi-Assess - множественный доступ без вещательного режима

NHC NHRP Client - клиент NHRP

NHRP Next Hop Resolution Protocol - протокол определения следующего узла для пересылки

NHS NHRP Server - сервер NHRP

OSPF Open Shortest Path First - протокол маршрутизации по кратчайшему пути

PIM-SM	Protocol Independent Multicast Sparse Mode - режим независимой от протокола многоадресной рассылки
PNNI	Private Network-to-Network Interface - межсетевой интерфейс для сетей ведомственной связи
PPP	Point-to-Point Protocol - протокол “точка-точка”)
QoS	Quality of Service - качество обслуживания (или услуги)
RP	Rendesvous Point - пункт “встречи”
RSVP	Resource Reservation Protocol - протокол резервирования ресурсов
TDP	Tag Distribution Protocol - протокол распределения этикеток
ToS	Type of Service - тип обслуживания (или услуги)
TTL	Time To Live - время жизни (существования) пакета

Речь.

1.Основы технологии VoIP. Проблемы обеспечения QoS.

1.1.Толчком к эре конвергированных решений послужило то, что в 1999 году в Европе трафик передачи данных превысил речевой. Конвергенция позволила упростить и удешевить создание и эксплуатацию сетей за счет объединения голосового трафика с трафиком передачи данных, т. е. за счет создания мультимедийного трафика.

1.2.Смещение центра тяжести в область передачи данных поставило вопрос о поиске удобного встраивания речи в мультимедийный цифровой поток. Для этих целей специалисты стали использовать протокол IP, т. к. он восприимчив к требованиям со стороны не только услуг передачи данных, но и приложений реального времени. Примером может служить успешно реализованная технология передачи речевой информации по сетям с маршрутизацией пакетов – IP-телефония. Также протокол IP позволяет конечному абоненту осуществлять интегрированный доступ, т. е. использовать один канал для транслирования речевой информации и данных.

1.3.Т. к. изначально сеть передачи данных представляла услугу только “наилучшей попытки”, то в условиях преобразования сети обеспечение QoS стало важнейшей задачей.

Можно выделить 4 основные составляющие, по которым оценивается качество услуги в сетях IP телефонии: задержка, джиттер, потери, готовность сети. В ТФОП допустимы задержки до 200 мс. и 5 минут отказа в год. Сети с коммутацией пакетов должны обеспечивать не худшее качество связи.

На сегодня разработан ряд технологий, призванных обеспечивать требуемое QoS. Анализу и выбору условий применения технологий MPLS и RSVP посвящена данная дипломная работа.

1.4.Аналитическое сравнение проводится по ряду параметров. Прежде чем проводить сравнение, рассмотрим каждую из этих технологий в отдельности.

2.MPLS

Многопротокольная коммутация с использованием меток – это технология быстрой коммутации пакетов, работающая с любым протоколом сетевого уровня. MPLS позволяет быстро направлять пакеты центальными маршрутизаторами – LSR. В каждый пакет, который поступает в MPLS-сеть, входящим граничным маршрутизатором – LER инкапсулируется метка MPLS. Внутри MPLS-сети маршрутизация производится по метке, т.е. решение о продвижении пакета принимается после просмотра таблицы коммутации, а не маршрутизации, что значительно сокращает время передачи пакета по сети. При выходе пакета из MPLS-сети метка исключается и пакет маршрутизируется с помощью

применяемых в данной сети методов маршрутизации. Значение метки уникально для каждой пары смежных LSR.

3.RSVP

Протокол резервирования ресурсов был разработан как сигнальный протокол , предназначенный для предварительного бронирования определенных сетевых ресурсов с целью их последующего использования для установления определенного сеанса. Протокол функционирует следующим образом: источник посылает сообщение PATH приемнику, в котором специфицируются параметры трафика, планируемого для передачи. Каждый маршрутизатор, находящийся на пути сообщения PATH перенаправляет это сообщение следующему узлу. При получении сообщения PATH приемник отправляет сообщение RESV. RESV использует тот же маршрут, что и PATH. Маршрутизаторы определяют могут ли они удовлетворить эти RESV-запросы. Если нет, они отказываются от резервирования. Если да, то они отсылают запрос следующему маршрутизатору. Отправитель, получив запрос на резервирование считает резервирование состоявщимся. Для поддержания ресурсов в резервированном состоянии сообщения PATH и RESV периодически повторяются, а соседние маршрутизаторы обмениваются сообщениями Hallo.

4.Аналитическое сравнение.

4.1.Объемы ресурсов, которые необходимы маршрутизатору для обработки и хранения информации RSVP, увеличиваются пропорционально числу резервирований. Т.о. поддерживая много RSVP резервирований можно получить отрицательный результат. Следовательно, RSVP имеет проблемы с масштабированием.

При MPLS маршрутизатор не анализирует заголовок пакета, а узнает о следующем адресате по метке. Это экономит время и ресурсы. Следовательно, MPLS более масштабируем.

4.2,6.RSVP определяет для трафика кротчайший маршрут. В результате сеть работает неэффективно и не решает проблем, связанных с нехваткой пропускной способности.

MPLS позволяет направлять трафик через менее загруженные маршруты. Это означает более эффективную работу сетей, а также позволяет избежать проблем с нехваткой пропускной способности.

4.3.При RSVP на организацию резервирований тратятся большие интервалы времени за счет большого объема служебной информации.

В MPLS, если путь устанавливается впервые, то время установления соединения соизмеримо с временем при использовании технологии RSVP. А если данные передаются

по уже существующему маршруту, то время установления соединения значительно сокращается.

4.4. RSVP присваивает каждому маршруту определенный индекс. Это исключает просмотр таблицы маршрутизации.

MPLS дает возможность ускоренного продвижения пакетов, т.к. маршрутизация заменяется коммутацией – более скоростным способом продвижения.

4.5.RSVP каждый поток обслуживает отдельно, что увеличивает затраты ресурсов сети.

MPLS позволяет объединить однотипные потоки и рассматривать их единым образом. Это уменьшает затраты ресурсов сети.

4.7.Протокол RSVP присваивает определенный индекс каждому маршруту. Таблицы потоков могут стать настолько большими, что сеть начнет давать сбои.

MPLS классифицирует пакет и присваивает ему метку. Метка должна быть уникальной лишь для каждой пары смежных транзитных маршрутизаторов. В связи с этим пространство меток не будет исчерпано.

4.8.Протокол RSVP устанавливает маршрут при поступлении каждого нового потока трафика.

При MPLS установление маршрутов происходит только при изменении топологии сети или получении соответствующей управляющей информации.

4.9. RSVP и MPLS может использоваться в паре с Frame Relay или ATM.

4.10.При RSVP маршрутизаторы должны обладать большими вычислительными ресурсами, большими объемами памяти и функциями, поддерживающими RSVP.

При MPLS транзитным маршрутизаторам не требуются ни большие объемы памяти, ни вычислительные ресурсы. Это значительно упрощает построение сети в целом.

4.11. При RSVP возможен несанкционированный захват или сокрытие сетевых ресурсов.

Протокол MPLS может обеспечить безопасность с помощью средств шифрования, контроля доступа и аутентификации (авторизации) пользователей.

4.12. RSVP и MPLS позволяет обслуживать мультикастные приложения.

4.13.Время реакции на обрыв связи в протоколе RSVP измеряется в МИЛЛИсекундах (0,001), а в MPLS в МИКРОсекундах (0,000001).

4.14. В настоящее время технологии RSVP и MPLS находятся в стации разработки, поэтому конкретные цены на оборудование трудно назвать.

5.Вывод.

Внедрение протокола RSVP целесообразнее в пределах частных сетей, тогда как внедрение технологии MPLS возможно в более крупных сетях.

В настоящее время проводятся работы над возможностью совместного использования технологий RSVP и MPLS. Это делается в целях улучшения качества обслуживания абонентов. Из-за различных функций данных технологий специалисты отводят место MPLS в ядре сети, а RSVP – в сети доступа.

Дипломная работа.

“Анализ и сравнение подходов к обеспечению гарантированного качества обслуживания в мультисервисных сетях.”

Студенка	**Осипова Н. А.**
Руководитель	**Гольдштейн А.Б.**
Факультет	**СС,СК и ВТ**
Группа	**СК-85**

Основы технологии VoIP.
Проблемы обеспечения QoS.

- ***Конвергенция***

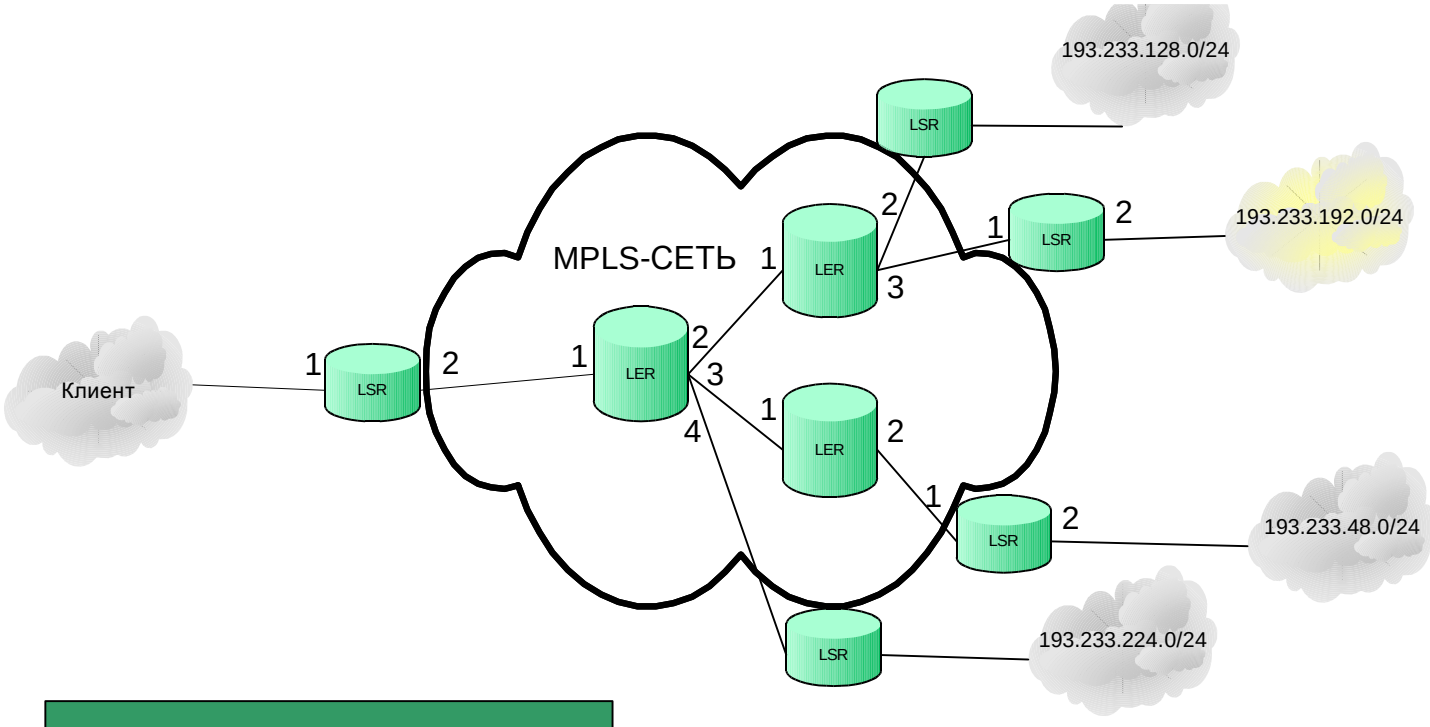
- Интернет
- Мультимедиа

- ***VoIP***

- IP-телефония
- Интегрированный доступ

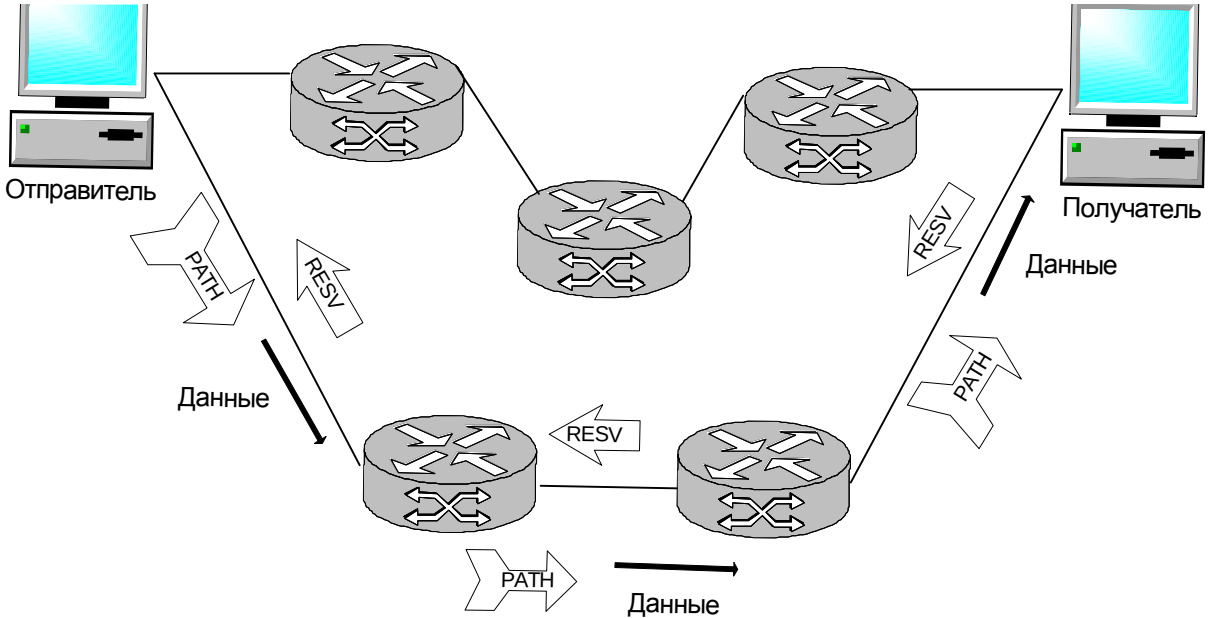
- ***QoS***

- Оценки и нормы качества обслуживания
- MPLS
- RSVP



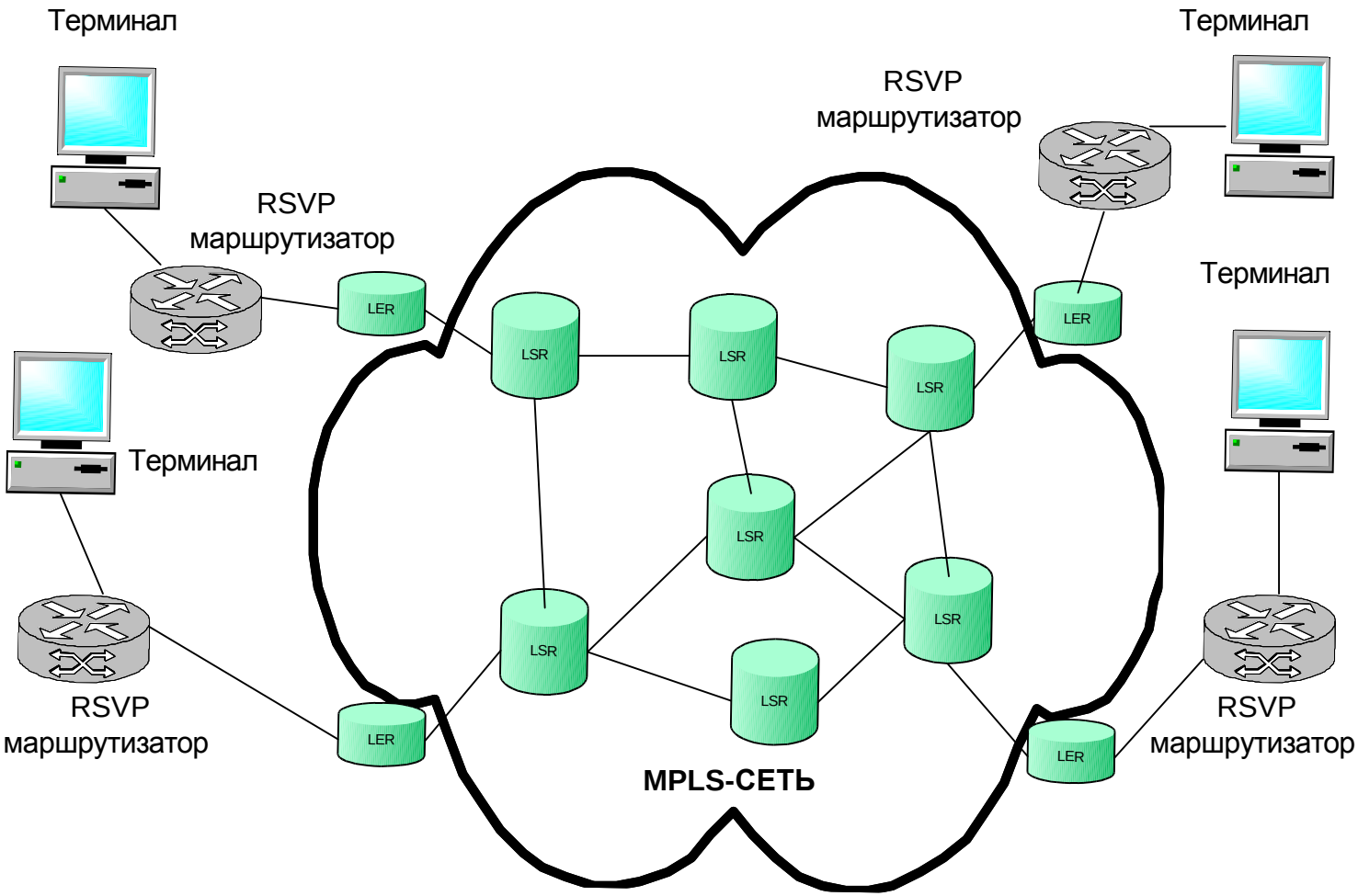
Префикс -193.233.192				
№	Вх.метка	Вх.порт	Вых.метка	Вых.порт
1	-	1	5	2
2	5	1	7	2
3	7	1	3	3
4	3	1	-	2

Префикс -193.233.48				
№	Вх.метка	Вх.порт	Вых.метка	Вых.порт
1	-	1	8	2
2	8	1	5	3
3	5	1	11	2
4	11	1	-	2


Отправитель
Получатель
PATH
RESV
Данные
RESV
Данные
PATH
RESV
PATH
Данные

Аналитическое сравнение.

- **Масштабируемость**
- **Перегрузки**
- **Время установления соединения**
- **Время пересылки пакетов**
- **Затраты ресурсов сети на обслуживание**
- **Пропускная способность**
- **Классификация пакетов**
- **Установление маршрута**
- **Совместимость с технологиями коммутации каналов**
- **Построение сети**
- **Безопасность**
- **Обслуживание мультикастных приложений**
- **Действия протоколов при обрыве связи**
- **Затраты на внедрение технологий**


Терминал
Терминал
RSVP
маршрутизатор
RSVP
маршрутизатор
Терминал
Терминал
RSVP
маршрутизатор
RSVP
маршрутизатор
LER
LER
LER
LER
LSR
LSR
LSR
LSR
LSR
LSR
LSR
MPLS-СЕТЬ

Рецензия на дипломную работу Осиповой Надежды Александровны, студенки СПбГУТ им. проф. М.А. Бонч-Бруевича, факультета СС, СК и ВТ, гр. СК – 85 на тему: «Анализ и сравнение подходов к обеспечению гарантированного качества обслуживания в мультисервисных сетях».

Актуальность темы обусловлена бурным развитием сетей на базе коммутации пакетов, и следовательно, поиском технологий для обеспечения качественной передачи информации.

Главными вопросами работы, в которых автор добилась наиболее существенных результатов, является сравнение механизмов, обеспечивающих гарантированное качество обслуживания.

В ходе выполнения дипломной работы произведено сравнение технологий обеспечения качества RSVP и MPLS, описаны методы оценки качества передачи речи и механизмы обеспечения качества сервиса.

Таким образом, рассматриваемая работа представляет серьёзную практическую ценность. Автор проявила умение разбираться в поставленных задачах, нашла и систематизировала необходимый материал.

Вместе с тем следует отметить, допущенные автором длинноты, некоторое отклонение от общепринятой структуры работы и недостаточную яркость при изложении. Отмеченные недостатки имеют локальный характер и не влияют на конечные результаты работы, не снижают ее высокого уровня, их скорее можно считать пожеланиями к дальнейшей работе автора.

Работа может быть оценена отлично, а ее автор заслуживает звания инженера.

Рецензент _____

Подпись рецензента _____

ОТЗЫВ

на дипломную работу студенки
факультета СС, СК и ВТ
СПбГУТ им. проф. М.А. Бонч-Бруевича

Осиповой Надежды Александровны.

тема работы: ***"Анализ и сравнение подходов к обеспечению гарантированного качества обслуживания в мультисервисных сетях"`***

Бурный рост использования сетей на базе коммутации пакетов для передачи речевой информации приводит к необходимости поиска технологий для обеспечения качества обслуживания. Одними из них являются протоколы RSVP и MPLS. В данной дипломной работе приводится сравнение технологии с точки зрения целесообразности их внедрения.

Отличительной особенностью работы следует считать последовательное прохождение всех этапов выполнения технического задания. Дипломница, опираясь на богатую информационную базу, проанализировала протоколы и сделала выводы о целесообразности их внедрения.

Предоставленный материал свидетельствует, что на этапах работы Осипова Надежда Александровна умело и грамотно применила большое количество специальной литературы. Данный подход позволил ей решить поставленные задачи на высоком уровне и продемонстрировать качество своей специальной подготовки.

Пояснительная записка выполнена аккуратно, в ней учтены основные требования к составу и структуре дипломной работы. Требования технического задания выполнены полностью.

В целом дипломная работа отвечает всем предъявленным требованиям и заслуживает отличной оценки, а Осипова Надежда Александровна достойна присвоения квалификации инженера по специальности "Сети связи и системы коммутации".

Начальник сектора ЛОНИИС ***Гольдштейн А. Б.***