Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»

Кафедра «Телекоммуникационные системы»
Специальность 6М071900 «Радиотехника, электроника и телекоммуникации»

ДОПУЩЕН К ЗАЩИТЕ
Зав. кафедрой
к.т.н., Шагиахметов Д.Р.
(ученая степень, звание, ФИО) (подпись)
«____»_____2014 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ
пояснительная записка

на тему: Исследование показателей качества обслуживания (quality of service) в IP-сетях

Магистрант Капасов Ж.К. _____ группа ИТСп-12-1
(Ф.И.О.) (подпись)
Руководитель к.х.н., ст преподаватель _____ Данько Е.Т.
(ученая степень, звание) (подпись) (Ф.И.О.)

Технический консультант _____ _____
(подпись) (Ф.И.О.)
Рецензент_____ _____ _____
(ученая степень, звание) (подпись) (Ф.И.О.)

Консультант по ВТ к.х.н., ст..преп. _____ Данько Е.Т.
(ученая степень, звание) (подпись) (Ф.И.О.)

Нормоконтроль ст.преп. _____ Абрамкина О. А.
(ученая степень, звание) (подпись) (Ф.И.О.)

Алматы, 2014

Аңдатпа

Бұл диссертациялық жұмыста IP желілерінде асыра жүктемеден сақтап қалу механизмдерінің жұмысы зерттеледі.

Бірінші бөлімде, телекоммуникациялық жүйелердің жалпы дамуы және конвергенциялық желісіне ауысу себептері қарастырылады. Екінші бөлімде негізгі мақсаттары мен сапаның қызметтерін анықталған.

Үшінші бөлімде жұмыстың негізгі принциптері және асыра жүктеменің сақтап қалуға арналған хаттамаларының түрлері қарастырылған.

Төртінші бөлімде пакеттардың өлшенген ерте лақтырып тастау механизміні туралы эксперименттік зерттеу өткізілген.

Бесінші бөлімде, арнаның өткізгіш қабілеттілігіне мәліметтерді тарату үшін қолданылатын хаттамалар қалай әсер ететіні туралы зерттеулер жүргізілген.

Аннотация

В данной диссертационной работе исследуется работа механизма предотвращения перегрузок в IP-сетях.

В первой главе рассматривается общее развитие телекоммуникационных систем. Причины перехода к конвергированным сетям.

Во второй главе определены основные цели и функции качества обслуживания.

В третьей главе рассматриваются основные принципы работы и типы протоколов предотвращения перегрузок.

В четвертой главе проводится экспериментальное исследование механизма взвешенного раннего отбрасывания пакетов.

В пятой главе произведены исследования, как влияет тот или иной используемый протокол для передачи данных, на пропускную способность канала.

Содержание

Введение

В настоящее время идёт увеличение спроса на инфокоммуникационные услуги, этот фактор является катализатором развития телекоммуникационных и информационных технологий. В основном на телекоммуникационном рынке потребители требовали от сервис провайдеров автоматизированную обработку данных с использованием средств вычислительной техники, как на входящем, так и на исходящем конце соединения с возможностью передачи многокомпонентной информации (речь, данные, видео, аудио) в режиме реального времени с гарантированными параметрами качества обслуживания.

Сервис провайдеры для решения выше указанных задач начали интеграцию различных сетей образовывая единую сетевую инфраструктуру на базе IP, которая обеспечивает предоставление услуг передачи данных, видео-трафика и IP-телефонии. Такой инфраструктурой в современном периоде является NGN.

В начальный период использования сети Интернет главными и основными достоинствами пакетной передачи информации была возможность создавать надежные сети которые способные передавать нагрузку на большие расстояния, то сейчас на первый план выходит способность современных пакетных технологий обеспечить заданное качество обслуживание QoS.

В данной диссертационной работе изучается процесс предотвращения перегрузок в сетях передачи данных и использование одного из алгоритмов предотвращения перегрузок - алгоритма WRED. Причиной такого интереса является следующее: первое, то что автор работает инженером передачи данных у крупного сервис провайдера Казахстана АО «Казактелеком» и постоянно сталкивается с проблемой перегрузки у наших корпоративных клиентов, а вторая причина то, что этот механизм реализован во всех современных сетевых устройствах.

Несмотря на то, что теме предотвращения перегрузок уделяется большое внимание различными публикациями [1,2,3], остается проблема настроек оптимальных значений параметров. По этому исследования в данной области попрежнему является актуальными и могут иметь практическую значимость для многих сервис провайдеров.

Основной целью работы является экспериментальное исследование влияния параметров алгоритма WRED на качество передачи и нахождение оптимальных значений параметров для предотвращения перегрузок в сетях передачи данных.

В работе проводится исследование влияния параметров на работу алгоритма WRED, а также даются рекомендации по выбору оптимальных настроек алгоритма для данного уровня перегрузки. Исследование проводилось на реальном оборудовании компании Cisco Systems с использованием программного генератора трафика IxChariot.

Для подтверждения достоверности полученных результатов были проведены математические расчёты, анализ которых показал истинность

полученных значений. Проведённый расчёт позволяет оценить требования к сети, необходимые для качественного предоставления услуг телефонии. и очень важен на этапе разработки политик качества обслуживания.

1 Этапы развития телекоммуникационных систем

1.1 Основные услуги в телекоммуникационных сетях

Одним из важнейших параметров, изменяющихся в процессе развития организма телекоммуникаций, являются обеспечиваемые им услуги. По изменению спектра услуг можно последить и более глубокие изменения – касающиеся конкретных технологий, положенных в основу функционирования всей отрасли.

Наиболее удобный для моей работы подход к классификации услуг заключается в разделении их на две группы – основные и дополнительные.

Услуги, появившиеся на заре развития телекоммуникаций, остаются востребованными, с теми или иными изменениями, и по сей день. К ним следует отнести, в первую очередь, телефонную связь (предоставление канала тональной частоты между оконечными устройствами на время жизни вызова), радиосвязь и передачу данных, которая включала себя лишь телеграф. Со временем эти услуги дополнились передачей движущегося изображения. Перечисленные услуги образовали базовый набор, который успешно прошёл естественный отбор.

Революционной вехой в развитии телекоммуникаций стали успехи в области цифровой техники. С её появлением и развитием существенно изменилась структура трафика в сторону увеличения доли передачи данных. Кроме того, выросла доля цифрового коммутационного оборудования, позволяющего предоставлять значительно более широкий спектр услуг.

Именно с развитием цифровой техники появилась идея объединения способов предоставления услуг – концепция цифровой сети интегрального обслуживания (ISDN).

Концепция ISDN, определенная рекомендациями ITU серии I, предполагает:

- стандартизацию предоставляемых абонентам услуг для обеспечения их совместимости при международной связи;
- стандартизацию интерфейса между пользователем услуги и сетью для обеспечения взаимозаменяемости терминального оборудования;
- стандартизацию свойств и возможностей сети связи.

ISDN представляет собой сеть, которая предусматривает сквозные цифровые соединения между оконечными устройствами и обеспечивает предоставление пользователям широкого спектра речевых и неречевых услуг, доступных им через ограниченный набор стандартизованных интерфейсов.

Доведение цифрового интерфейса до абонента СТОП позволила в рамках концепции ISDN интегрировать разные услуги:

- передачу речи;
- передачу текста;
- передачу данных;
- передачу изображений;

- передачу видео.

Доступ пользователя к услугам обеспечивается через единый унифицированный интерфейс, расположенный в помещении пользователя.

В ISDN цифровой доступ организуется с использованием на абонентском участке уже существующих медных физических пар. Концепцию ISDN можно рассматривать как конечный этап цифровизации СТОП. Сеть ISDN – является наложенной сетью по отношению к СТОП.

Сеть ISDN не получила широкого развития по некоторым объективным и субъективным причинам. ISDN не устроила пользователей по соотношению цена/качество. Для перехода на ISDN необходимо было полностью поменять аналоговые терминалы на терминалы ISDN. Услуги ISDN в полной мере можно использовать лишь при тотальной цифровизации – при взаимодействии аналогового и цифрового абонентов спектр услуг резко сужался. Скорости тоже перестали удовлетворять пользователей – в базовом доступе 128 кбит/с было слишком много для простой телефонной связи и слишком мало для организации, допустим, видеоконференций.

Ещё одним важным моментом в развитии способов предоставления услуг была идея разделения функций распределения информации и предоставления услуг. Данная концепция получила название Intelligent Network. Разделение функций в пределах основных элементов системы электросвязи может рассматриваться как периодически повторяющийся процесс. Примером может служить разделение функций передачи сообщений системы сигнализации и полезной информации, который привел к формированию концепции системы общеканальной сигнализации.

1.2 Развитие техники коммутации

Можно выделить пять видов систем коммутации, характерных для сетей телефонной связи:

- декадно-шаговые АТС;
- координатные АТС;
- квазиэлектронные АТС;
- цифровые АТС;
- оборудование, основанное на технологии «коммутация пакетов».

Первые три вида систем коммутации можно объединить в одну группу: аналоговые АТС. Такой подход объясняется общностью основных процессов модернизации этих коммутационных станций. Цифровые АТС и оборудование, основанное на технологии «коммутации пакетов», требуют отдельного анализа.

1.2.1 Аналоговые АТС

Первая АТС была изобретена в 1892 г. А. Строунджером. Коммутация в этих декадно-шаговых АТС производится под непосредственным управлением

сигналов набора номера вызывающим абонентом без использования каких бы то ни было централизованных управляющих устройств.

Однако эффективное развитие городских телефонных сетей сдерживалось главным образом малой емкостью контактного поля искателей. Автоматизация междугородной телефонной связи выявила низкое качество разговорного тракта из-за нестабильности скользящих контактов искателей, приводившей к недопустимо высокому уровню шумов.

Недостатки декадно-шаговых АТС были устранены в станциях следующего поколения – координатных. Емкость контактного поля коммутационных приборов таких АТС значительно больше, чем емкость поля декадно-шаговых искателей, а контакты скольжения заменены в них контактами давления, имеющими стабильное сопротивление и гораздо больший срок службы.

Кроме замены скользящего контакта «щетка – ламель» на МКС координатные АТС принесли новый обходной принцип управления станциями, при котором сам коммутационный прибор не участвует в выборе направления и поиске свободной линии.

По мере развития технологий стали появляться заменители традиционных электромеханических коммутационных элементов – электронные и магнитные устройства, в которых отсутствовали подвижные части.

На первом этапе достижения электроники стали применяться только в управляющих устройствах АТС, что привело к появлению квазиэлектронных АТС, сочетавших в себе электронное управление и электромеханические коммутационные элементы.

Название квазиэлектронные АТС предполагает сохранение пространственной аналоговой коммутации с применением механических контактов и одновременно использование электронных программируемых управляющих устройств.

1.2.2 Цифровые АТС

Цифровые ГТС первого поколения (1990-1995 гг.) представляют собой следующий комплекс аппаратно-программных средств:

– коммутационное поле, построенное на принципах временной и пространственной коммутации;

– периферийное оборудование: модули аналоговых абонентских линий, аналоговых и цифровых соединительных линий;

– оборудование сигнализации: абонентская, внутристанционная и межстанционная подсистемы;

– система программного управления, предоставляющая конечным пользователям наборы услуг и сервисов.

Абонентам цифровых АТС доступны многие современные услуги. Все основные показатели качества функционирования и надежности коммутационного оборудования (речь не идет о сети в целом), как правило,

обеспечиваются. Можно и дальше перечислять известные преимущества цифровых АТС, но с точки зрения их эволюции существенно другое. Сама СТОП, даже построенная только на базе цифровых АТС, перестает отвечать требованиям инфокоммуникационной системы. Впрочем, этот факт не исключает необходимости проведения работ по модернизации цифровых АТС. Перечень подобных работ очень похож на тот, что был предложен для аналоговых систем коммутации. Решение некоторых задач, как правило, оказывается более простым и эффективным, что обеспечивается управлением АТС по записанной программе, а также применением цифровых технологий для передачи, коммутации и обработки информации [3].

1.2.3 Пакетная коммутация

Следующим этапом развития коммутационной техники становится коммутация пакетов. Изначально предназначенная для передачи данных, сегодня она находит применение и для передачи голосового трафика. При этом говорить о «классической» пакетной коммутации не приходится – для обеспечения качества обслуживания вводятся дополнительные механизмы, реализация которых на практике зачастую весьма и весьма проблематична.

Идея пакетной коммутации как способа распределения информации для сетей следующего поколения пока еще не облечена в форму международных стандартов. Более того, некоторые специалисты считают, что для сети следующего поколения необходима новая технология распределения информации, сочетающая в себе свойства коммутации каналов и пакетов. Пока концепцию NGN ассоциируют с коммутацией IP-пакетов. В конечном счете, способ распределения информации, который будет принят для сетей следующего поколения в качестве международного стандарта, не столь существенно скажется на оборудовании коммутации. Радикальные изменения определяются основной идеей NGN. Эту идею часто называют конвергенцией, хотя определения, содержащиеся в большинстве словарей, свидетельствуют о не совсем удачном выборе термина. Основной смысл концепции NGN – интеграция, но этим термином многие специалисты предпочитают не пользоваться, памятуя о неудаче концепции широкополосной ЦСИО [4].

1.3 Этапы развития транспортной сети

Понятие «транспортная сеть» как правило, связывают с используемой технологией систем передачи информации. Появление систем передачи в качестве самостоятельного элемента сети электросвязи относится к 1870 году, когда в коммерческую эксплуатацию была введена аппаратура для обмена телеграфными сообщениями, которая имела в своем составе электромеханические регенераторы. Интересным фактом может считаться то обстоятельство, что упомянутая аппаратура была разработана как ЦСП с временным разделением каналов.

Электромеханические принципы регенерации не могли быть использованы в телефонии. По этой причине дальность телефонной связи была ограничена несколькими сотнями километров. Развитие электронной промышленности привело в 1915 году к возможности создания аналоговых систем передачи (АСП). Появление систем передачи обеспечило техническую возможность междугородной и международной телефонной связи. Существенными моментами использования систем передачи могут считаться:

– уменьшение стоимости оборудования, реализующего функции по переносу информации между коммутационными станциями (узлами) вторичных сетей;

– поддержку показателей качества передачи информации в соответствии с заданными нормами.

Использование аналоговых систем передачи было обусловлено, на первых порах, отсутствием подходящей элементной базы для организации цифровых каналов. Аналоговые системы передачи обладают рядом недостатков, которые делают их нежизнеспособными. В первую очередь к ним следует отнести невозможность регенерации сигнала.

На сегодняшний день идеи частотного разделения каналов вылились в весьма перспективные технологии волнового уплотнения каналов в оптическом волокне.

Все цифровые системы передачи в том или ином виде реализуют временное разделение источников нагрузки. Изначально появилось асинхронное разделение, использовавшееся в телеграфии. В телефонии долгое время единственным способом временного разделения каналов оставалось синхронное разделение, которое все по традиции называют просто «TDM» или «коммутация каналов».

Асинхронное временное разделение получило развитие несколько позже, и не в телефонии, а в сетях передачи данных. Асинхронное временное разделение принято называть «коммутацией пакетов». Для коммутации пакетов классификационным признаком обычно служит название технологии (X.25, Frame Relay, ATM, MPLS и т.д.). Кроме того, обычно выделяют два режима коммутации пакетов: дейтаграммный и с предварительным установлением виртуальных каналов.

Современные телефонные сети построены на технологии цифровой коммутации каналов. Тракт, установленный через совокупность цифровых коммутационных станций, можно рассматривать как виртуальный канал, по которому передаются пакеты длиной 8 бит. Конечно, такая трактовка весьма условна, но практически может оказаться полезной. В каждой цифровой коммутационной станции пакет из 8 бит задерживается. Это свойство делает цифровую коммутационную станцию похожей на коммутатор пакетов. Различие состоит в том, что в цифровой коммутационной станции дисперсия задержки пакетов равна нулю. В любом коммутаторе пакетов дисперсия может достигать существенных величин, значительно снижая качество обслуживания.

Итак, развитие технологии коммутации каналов привело к тому, что в ней появились некоторые свойства, близкие к коммутации пакетов. Модернизация метода коммутации каналов продолжается. Появились технологии быстрой коммутации каналов (Fast Circuit Switching) и динамического синхронного режима переноса (Dynamic Synchronous Transfer Mode). Более того, некоторые специалисты отмечали, что для NGN потребуется разработка нового метода распределения информации, который, по всей видимости, будет более похож на коммутацию каналов. Поэтому безапелляционные высказывания о пакетной коммутации, как о единственном методе распределения информации в NGN, не стоит принимать за аксиому. Методы коммутации пакетов для информации, критичной ко времени задержки, основаны на установлении виртуальных каналов. Сама природа этого процесса близка к коммутации каналов. Конкурируя между собой, оба метода распределения информации стали заимствовать друг у друга некоторые черты [5].

1.4 Конвергенция в телекоммуникациях

Из вышеприведённого анализа следует, что наблюдается коренные, глубинные изменения в потребности в услугах, а, следовательно, и в способе их предоставления. Одним из доминирующих процессов в телекоммуникациях стала конвергенция.

Конвергенция – возникновение сходства в строении и функциях у систем, изначально далеких по происхождению и назначению.

В телекоммуникационных системах компания Cisco выделяет три аспекта конвергенции:

- конвергенция сетей;
- конвергенция управления;
- конвергенция приложений.

Укрупнение телекоммуникационного бизнеса и появление холдингов, объединяющих сети нескольких специализированных операторов (фиксированной связи, мобильной связи и передачи данных), а также острая конкурентная борьба за абонента, обусловили появление нового класса услуг. Они обеспечивают, прозрачно для абонента, взаимопроникновение сетей и услуг, специфичных для одной, определённой телекоммуникационной среды, в другую за счёт шлюзования.

Процесс конвергенции, зачастую, нуждается в привлечении интеллекта, с помощью которого согласуются протоколы (SDH, ATM, FR, IP) и технологии передачи (VoIP, VoATM, VPN). Роль таких согласующих устройств на сетях операторов обычно выполняют мультисервисные сетевые устройства, обычно поддерживающие ATM и MPLS, IP и FR.

При конвергенции сетей несколько в корне различающихся сетей сливаются в одну. При этом повышается эффективность сети. Операторы переходят от множества наложенных сетей, требующих отдельного управления

и техобслуживания, к одной сети, ядром которой, как правило, является сеть IP/MPLS.

Сетевая конвергенция позволяет вводить современные услуги. Следующая волна Интернета будет формироваться конечным пользователем, которому потребуются эти инновационные приложения и услуги.

Конвергенция управления заключается в том, что оператор должен предоставлять услуги, осуществлять биллинг и управлять услугами, предоставляемыми во всех средах доступа.

При этом повышаются требования к интеллектуализации прикладного уровня и оборудования абонента. Хорошим примером этому может служить концепция «triple play services», предполагающая передачу речи, видео и данных в рамках одной сети. При использовании беспроводных средств передачи данных получается «triple play в движении», или «quadruple play».

Рост числа сетей различного назначения привёл к появлению большого числа различных сетевых приложений и информационных услуг. Специфика деятельности абонентов требует минимизации времени, затрачиваемого потребителем на выбор правильного терминала и алгоритма доступ к необходимой услуге.

В конечном итоге все современные терминалы, в той или ной степени, являются специализированными или универсальными персональными компьютерами. А цифровые коммутаторы различного назначения – это специализированные или универсальные серверы телекоммуникационных услуг. Таким образом, цифровизация существующих телекоммуникационных сетей является отправным пунктом на пути эволюции разнородных сетей в единую, прозрачную для пользователя, мультисервисную среду.

Понятие конвергентные сети ассоциируется в общественном сознании, в основном, со средой Ethernet и IP-приложениями. Несколько лет назад велась массированная реклама интегральных решений на базе этих технологий, которые демонстрировали техническую реализуемость слияния любых услуг в «универсальной» цифровой среде, обеспечивающую доставку произвольного трафика. Однако, специфика операторской деятельности заключается в необходимости сохранения инвестиций и невозможности отказа от существующей телекоммуникационной инфраструктуры в пользу новых сред и технологий. Помимо этого, для оператора очень важным фактором при принятии решений является не только запас по масштабированию производительности, но и маркетинговый запас решения, то есть развитый механизм контроля качества предоставляемых услуг, возможность прогнозирования пиковых нагрузок и планирования модернизации узлов сети.

2 Основные принципы качества обслуживания в IP - сетях

Вплоть до настоящего времени поставщикам услуг Internet и крупным компаниям приходилось создавать и поддерживать отдельные сети для передачи голосовой информации, видеоизображения, трафика, необходимого для решения критически важных задач, и всего остального сетевого трафика. Тем не менее нельзя не отметить сложившуюся в последнее время ярко выраженную тенденцию к объединению всех этих сетей в одну сеть с пакетной передачей данных на основе протокола Internet Protocol (IP).

Наиболее крупная IP-сеть — это, естественно, глобальная сеть Internet. За последние несколько лет рост Internet, передаваемого по Сети трафика и количества существующих Internet-приложений приблизился к экспоненциальному. В то время как Internet и корпоративные интрасети продолжают свой рост, многие аналитики предсказывают появление приложений, ориентированных на передачу нетрадиционных типов информации, например, передачу голоса по сетям IP (Voice over IP — VoIP) или передачу трафика видео-конференций. Поскольку количество пользователей Internet и различных сетевых приложений увеличивается с каждым днем, Сеть нуждается в средствах, которые бы обеспечили поддержку как существующих, так и появляющихся приложений и служб. Тем не менее на сегодняшний день Internet может обеспечить всего лишь негарантированную доставку данных (best effort service). Негарантированная доставка данных не предполагает предоставление каких-либо гарантий, касающихся времени и самого факта прибытия пакета в пункт назначения. При этом нельзя не отметить, что отбрасывание пакетов может произойти только в момент перегрузки сети. Как правило, передаваемые по сети пакеты различаются на основе пяти полей заголовка IP, которые однозначно определяют поток данных, — адрес источника IP-пакета, адрес назначения IP-пакета, поле протокола IP, порт источника и порт назначения. Поток информации состоит из пакетов, сгенерированных приложением, выполняющемся на компьютере-источнике, и предназначающихся для передачи приложению, выполняющемуся на компьютере-приемнике. Пакеты, принадлежащие одному потоку, имеют одинаковые значения всех пяти полей в заголовке IP-пакета.

С целью поддержки передачи голоса, видео и трафика данных приложений с различными требованиями к пропускной способности, системы ядра IP-сети должны обладать возможностью дифференцирования и обслуживания различных типов сетевого трафика в зависимости от предъявляемых ими требований. Негарантированная доставка данных не предполагает проведения какого-либо различия между тысячами потоков информации в ядре IP-сети. Следовательно, IP-сеть не может обеспечить никакой гарантии надежной доставки трафика приложений. Другими словами, негарантированная доставка данных препятствует передаче трафика, требующего выделения заданного минимума сетевых ресурсов и гарантии предоставления определенных услуг. Для разрешения описанной выше

проблемы и было введено такое понятие, как качество обслуживания (quality of service — QoS) в сетях IP.

Функции качества обслуживания в сетях IP (IP QoS) заключаются в обеспечении гарантированного и дифференцированного обслуживания сетевого трафика путем передачи контроля за использованием ресурсов и загруженностью сети ее оператору. QoS представляет собой набор требований, предъявляемых к ресурсам сети при транспортировке потока данных. QoS обеспечивает сквозную гарантию передачи данных и основанный на системе правил контроль за средствами повышения производительности IP-сети, такими, как механизм распределения ресурсов, коммутация, маршрутизация, механизмы обслуживания очередей и механизмы отбрасывания пакетов.

Ниже перечислены некоторые из основных преимуществ качества обслуживания в сетях IP.

– Обеспечение поддержки существующих и появляющихся мультимедийных служб и приложений. Некоторые новые приложения, такие, как передача голоса по сетям IP (VoIP), предъявляют определенные требования к качеству обслуживания.

– Передача контроля за ресурсами сети и их использованием сетевому оператору.

– Обеспечение гарантии обслуживания и дифференцирование сетевого трафика. Это условие является необходимым для объединения аудио-, видеотрафика и трафика приложений в пределах одной IP-сети.

– Позволяет поставщикам услуг Internet предлагать клиентам дополнительные услуги наряду со стандартной услугой негарантированной доставки данных (другими словами, предоставлять услуги в соответствии с так называемым классом обслуживания — Class of Service (CoS)). Поставщик услуг Internet может определить несколько классов дополнительных услуг (например, "платиновый", "золотой" и "серебряный" классы) и настроить сетевые правила, позволяющие обрабатывать трафик каждого класса в соответствии с заданными параметрами.

– Дает возможность организовать обслуживание сетевого трафика в зависимости от сгенерировавшего этот трафик приложения, информация о котором содержится в заголовке IP-пакета.

– Играет значительную роль в развитии новых сетевых технологий, таких, как виртуальные частные сети (Virtual Private Networks — VPNs).

2.1 Уровни качества обслуживания

Сетевой трафик состоит из множества потоков, сгенерированных приложениями конечных станций. Эти приложения отличаются друг от друга различными требованиями к обслуживанию и к рабочим характеристикам сети. По сути, требование к обслуживанию каждого потока целиком и полностью определяется требованиями сгенерировавшего этот поток приложения. Следовательно, для того чтобы выяснить структуру существующих в сети

запросов на качество обслуживания, необходимо определить типы сетевых приложений.

Способность сети обеспечивать различные уровни обслуживания, запрашиваемые теми или иными сетевыми приложениями, наряду с проведением контроля за характеристиками производительности— полосой пропускания, задержкой/дрожанием и потерей пакетов — может быть классифицирована по трем перечисленным ниже категориям.

– Негарантированная доставка данных (best-effort service). Обеспечение связности узлов сети без гарантии времени и самого факта доставки пакета в точку назначения. Следует отметить, что отбрасывание пакета может произойти только в случае переполнения буфера входной или выходной очереди маршрутизатора. На самом деле негарантированная доставка пакетов не является частью QoS вследствие отсутствия гарантии качества обслуживания и гарантии обеспечения доставки пакетов. Следует отметить, что негарантированная доставка пакетов является на сегодняшний день единственной услугой, поддерживаемой в Internet. Несмотря на некоторое снижение производительности, для большинства приложений, ориентированных на передачу информации (например, приложений, обеспечивающих взаимодействие по протоколу передачи файлов (File Transfer Protocol), эта услуга является вполне достаточной. В целом же оптимальные условия функционирования всех приложений включают в себя требования к выделению определенных сетевых ресурсов в терминах полосы пропускания, задержки и уровня потери пакетов.

– Дифференцированное обслуживание (differentiated service). Дифференцирование обслуживание предполагает разделение трафика на классы на основе требований к качеству обслуживания. Каждый класс трафика дифференцируется и обрабатывается сетью в соответствии с заданными для этого класса механизмами QoS. Подобная схема обеспечения качества обслуживания (QoS) довольно часто называется схемой CoS. Следует отметить, что дифференцированное обслуживание само по себе не предполагает обеспечения гарантий предоставляемых услуг. В соответствии с данной схемой трафик распределяется по классам, каждый из которых имеет свой собственный приоритет. По этой причине дифференцированное обслуживание довольно часто называют мягким QoS (soft QoS). Дифференцированное обслуживание удобно применять в сетях с интенсивным трафиком приложений. В этом случае важно обеспечить отделение административного трафика сети от всего остального трафика и назначить ему приоритет, позволяющий в любой момент времени быть уверенным в связности узлов сети.

– Гарантированное обслуживание (guaranteed service). Гарантированное обслуживание предполагает резервирование сетевых ресурсов с целью удовлетворения специфических требований к обслуживанию со стороны потоков трафика. В соответствии с гарантированным обслуживанием выполняется предварительное резервирование сетевых ресурсов по всей траектории движения трафика. Гарантированное обслуживание довольно часто

называют еще жестким QoS (hard QoS) в связи с предъявлением строгих требований к ресурсам сети. К сожалению, резервирование ресурсов на всем пути следования отдельных потоков трафика невозможно реализовать в масштабах магистрали Internet, обслуживающей в отдельный момент времени тысячи потоков данных. Исправить положение призвано агрегированное резервирование ресурсов, требующее хранения в базовых маршрутизаторах Internet всего лишь небольшого количества информации. Приложения, требующие гарантированного обслуживания, включают в себя мультимедийные приложения, проводящие передачу голосовой информации и видеоизображений. Интерактивные приложения, ориентированные на передачу речи по Internet, могут функционировать нормально (т.е. не вызывая неудобства у пользователей) лишь в том случае, если значение латентности равно или меньше 100 мс. Следует отметить, что аналогичный уровень латентности является приемлемым для большинства мультимедийных приложений. А вот приложениям Intenet-телефонии уже понадобится канал передачи информации с пропускной способностью как минимум 8 Кбит/с и со значением задержки подтверждения приема, равном 100 мс. Для того чтобы удовлетворить подобные требования к гарантированному обслуживанию, сеть должна обладать определенным запасом ресурсов.

Качество обслуживания уровня 2 эталонной модели OSI (Layer 2 QoS) включает в себя все механизмы QoS, предусмотренные различными технологиями канального уровня или технологиями, объектом которых этот уровень является. Качество обслуживания уровня 3 эталонной модели OSI (Layer 3 QoS) включает в себя все механизмы QoS, предусмотренные на сетевом уровне (уровне протокола IP). В табл. 2.1 перечислены три уровня обслуживания и соответствующие им разрешающие функции QoS канального и сетевого уровней эталонной модели OSI[6].

Таблица 2.1. Уровни обслуживания и соответствующие им разрешающие функции QoS

Уровень обслуживания	Разрешающая функция QoS сетевого уровня	Разрешающая функция QoS канального уровня
Негарантированная доставка пакетов	Связность узлов сети	Технология асинхронной передачи данных, обслуживание с неопределенной битовой скоростью
Дифференцированное обслуживание	Механизм согласования скорости доступа CoS, WFQ, WRED	IEEE 802.1p
Гарантированное обслуживание	Протокол резервирования ресурсов (Resource Reservation Protocol)	Диспетчер пропускной способности подсети, CIR

2.2 История возникновения и развития QOS в сетях IP

Качество обслуживания в сетях IP не является чьей-то блестящей идеей, возникшей на протяжении нескольких последних лет. Отцы-основатели Internet предвидели эту потребность и предусмотрели байт типа обслуживания (Type of Service) в заголовке IP-пакета. Следовательно, возможность реализации качества обслуживания была заложена еще в начальной спецификации протокола IP. Ниже приведена выдержка из спецификации протокола IP, в которой описывается предназначение байта ToS.

Байт типа обслуживания (Type of Service) используется для указания абстрактных параметров требуемого качества обслуживания. На основании этих параметров производится выбор реальных характеристик механизмов обслуживания при передаче датаграммы через заданную сеть.

Вплоть до конца 80-х годов Internet пребывала в "зародышевом" состоянии, что характеризовалось низким объемом трафика и малым числом используемых сетевых приложений. Следовательно, поддержкой байта ToS можно было пренебречь, что и сделано практически во всех реализациях протокола IP. IP-приложения не производили установку значения байта ToS, а маршрутизаторы игнорировали его при принятии решения о продвижении IP-пакета.

Важность внедрения механизма QoS в масштабах Internet возросла благодаря увеличению популярности Сети и приобретению ею коммерческих черт. Функционирование Internet базируется на сквозном режиме обслуживания пакетов данных без ориентации на установку соединения, который подразумевает негарантированную доставку информации с использованием для этого связки из двух протоколов — протокола управления передачей (Transmission Control Protocol) и протокола Internet (Internet Protocol), более известную как TCP/IP. Несмотря на то что отсутствие ориентации на установку соединения делает Internet более гибкой и устойчивой к сбоям, динамика передаваемых потоков данных делает ее склонной к перегрузкам, которые чаще всего возникают в местах стыка двух сетей со значительно различающимися пропускными способностями. Проблема снижения работоспособности TCP/IP сетей в моменты перегрузки была рассмотрена Джоном Наглем в середине 80-х годов прошлого столетия.

Первоначально функции качества обслуживания были возложены на узлы Internet. Одна из наиболее серьезных проблем, касающаяся дорогостоящих каналов передачи информации глобальных сетей (Wide-Area Network), заключалась в их чрезмерной перегрузке пакетами протокола TCP, имеющими небольшой объем и сгенерированными такими приложениями, как telnet и rlogin. Алгоритм Нагля, призванный решить эту проблему, реализован сегодня во всем программном обеспечении, установленном в узлах Internet и поддерживающем протокол IP. Можно сказать, что алгоритм Нагля "возвестил" о начале эпохи качества обслуживания в сетях IP.

В 1986 году Ван Якобсон (Van Jacobson) разработал следующий набор функций Internet QoS для конечных систем — механизмы предотвращения перегрузки, являющиеся стандартом де-факто для всех современных реализаций TCP. Следует отметить, что эти механизмы (механизм медленного старта и механизм предотвращения перегрузки) играют огромную роль в предупреждении критического снижения работоспособности сети в сегодняшней Internet. Ответственными за реагирование на сигналы о перегрузке сети (которыми служат отброшенные пакеты) являются потоки трафика TCP. В 1990 году для обеспечения оптимальной производительности сети в моменты потери пакетов были разработаны два дополнительных механизма — механизм быстрой повторной передачи и механизм быстрого восстановления.

Несмотря на то что реализация механизмов QoS в конечных системах является необходимым условием, она не позволяет говорить о сквозном качестве обслуживания до тех пор, пока соответствующие механизмы не будут реализованы в маршрутизаторах— устройствах, ответственных за передачу трафика между конечными системами. Следовательно, с 1990-х годов акцент в разработке механизмов QoS вполне логично переместился на исследование возможности реализации функций качества обслуживания в маршрутизаторах. Маршрутизаторы, поддерживающие только один механизм обслуживания очередей —"первым пришел, первым обслужен" (first-in, first-out), не способны обеспечить дифференцирование потоков трафика на основе их приоритета на уровне алгоритма планирования очередей. Более того, обслуживание очередей по методу FIFO приводит к отбрасыванию пакетов и не способно защитить потоки с предсказуемым поведением от "шалунов". Решить данную проблему на уровне магистрали Internet были призваны алгоритм обслуживания очередей WFQ (Weighted Fair Queuing — взвешенный алгоритм равномерного обслуживания очередей) и алгоритм управления очередями WRED (Weighted Random Early Detection — взвешенный алгоритм произвольного раннего обнаружения).

Разработка механизмов QoS продолжилась попытками стандартизации функций сквозного качества обслуживания в масштабах Internet. Целью рабочей группы IETF по созданию интегрированных услуг (Integrated Services (intserv) Internet Engineering Task Force (IETF) Working Group) является обеспечение приложений средствами формулирования требований к ресурсам при сквозном обслуживании, а также разработка соответствующих механизмов на уровне маршрутизаторов и технологий подсетей. Использующимся в этих целях сигнальным протоколом является протокол RSVP (Resource Reservation Protocol — протокол резервирования ресурсов). В соответствии с моделью intserv обслуживание каждого потока трафика производится на всей траектории соединения, что серьезно затрудняет использование этой модели в масштабах магистрали Internet, обрабатывающей в каждый момент времени тысячи потоков информации. Несмотря на то что байт заголовка IP-пакета ToS до недавнего времени практически не использовался, сейчас это один из самых

распространенных методов определения требований к качеству обслуживания. Байт ToS представляет собой главный механизм обеспечения услуг diffserv в масштабах Internet, что подвигло рабочую группу IETF по созданию дифференцированных услуг (IETF differentiated services (diffserv) Working Group) предложить стандартизировать этот байт в качества байта diffserv[7].

2.3 Характеристики производительности сетевого соединения

Внедрение механизмов QoS предполагает обеспечение со стороны сети соединения с определенными ограничениями по производительности. Основными характеристиками производительности сетевого соединения являются полоса пропускания, задержка, дрожание и уровень потери пакетов.

2.3.1 Полоса пропускания

Термин полоса пропускания (bandwidth) используется для описания номинальной пропускной способности среды передачи информации, протокола или соединения. Этот термин достаточно эффективно определяет "ширину канала", требующуюся приложению для взаимодействия по сети.

Как правило, каждое соединение, нуждающееся в гарантированном качестве обслуживания, требует от сети резервирования минимальной полосы пропускания. К примеру, приложения, ориентированные на передачу оцифрованной речи, создают поток информации интенсивностью 64 Кбит/с. Эффективное использование таких приложений становится практически невозможным вследствие снижения полосы пропускания ниже 64 Кбит/с на каком-либо из участков соединения.

2.3.2 Задержка и дрожание при передаче пакетов

Задержка при передаче пакета (packet delay), или латентность (latency), на каждом переходе состоит из задержки сериализации, задержки распространения и задержки коммутации. Ниже приведены определения каждого из названных выше типов задержки.

– Задержка сериализации (serialization delay). Время, которое требуется устройству на передачу пакета при заданной ширине полосы пропускания. Задержка сериализации зависит как от ширины полосы пропускания канала передачи информации, так и от размера передаваемого пакета. Например, передача пакета размером 64 байт при заданной полосе пропускания 3 Мбит/с занимает всего лишь 171 не. Обратите внимание, что задержка сериализации очень сильно зависит от полосы пропускания: передача того же самого пакета размером 64 байт при заданной полосе пропускания 19.2 Кбит/с занимает уже 26 мс. Довольно часто задержку сериализации называют еще задержкой передачи (transmission delay).

– Задержка распространения (propagation delay). Время, которое требуется переданному биту информации для достижения принимающего устройства на другом конце канала. Эта величина довольно существенна, поскольку в наилучшем случае скорость передачи информации соизмерима со скоростью света. Обратите внимание, что задержка распространения зависит от расстояния и используемой среды передачи информации, а не от полосы пропускания. Для линий связи глобальных сетей задержка распространения измеряется в миллисекундах. Для трансконтинентальных сетей Соединенных Штатов характерна задержка распространения порядка 30 мс.

– Задержка коммутации (switching delay). Время, которое требуется устройству, получившему пакет, для начала его передачи следующему устройству. Как правило, это значение меньше 10 нс.

– Обычно каждый из пакетов, принадлежащий одному и тому же потоку трафика, передается с различным значением задержки. Задержка при передаче пакетов меняется в зависимости от состояния промежуточных сетей.

В том случае, если сеть не испытывает перегрузки, пакеты не ставятся в очередь в маршрутизаторах, а общее время задержки при передаче пакета состоит из суммы задержки сериализации и задержки распространения на каждом промежуточном переходе. В этом случае можно говорить о минимально возможной задержке при передаче пакетов через заданную сеть. Следует отметить, что задержка сериализации становится незначительной по сравнению с задержкой распространения при передаче пакета по каналу с большой пропускной способностью.

Если же сеть перегружена, задержки при организации очередей в маршрутизаторах начинают влиять на общую задержку при передаче пакетов и приводят к возникновению разницы в задержке при передаче различных пакетов одного и того же потока. Колебание задержки при передаче пакетов получило название дрожания при передаче пакетов (packet jitter).

Дрожание пакетов имеет большую важность, поскольку именно оно определяет максимальную задержку при приеме пакетов в конечном пункте назначения. Принимающая сторона, в зависимости от типа используемого приложения, может попытаться компенсировать дрожание пакетов за счет организации приемного буфера для хранения принятых пакетов на время, меньшее или равное верхней границе дрожания. К этой категории относятся приложения, ориентированные на передачу/прием непрерывных потоков данных, например Internet-телефония или приложения, обеспечивающие проведение видеоконференций.

На рисунке 2.1 проиллюстрировано влияние задержки сериализации, задержки распространения и задержки коммутации на общую задержку при передаче пакетов в зависимости от возрастания полосы пропускания канала.

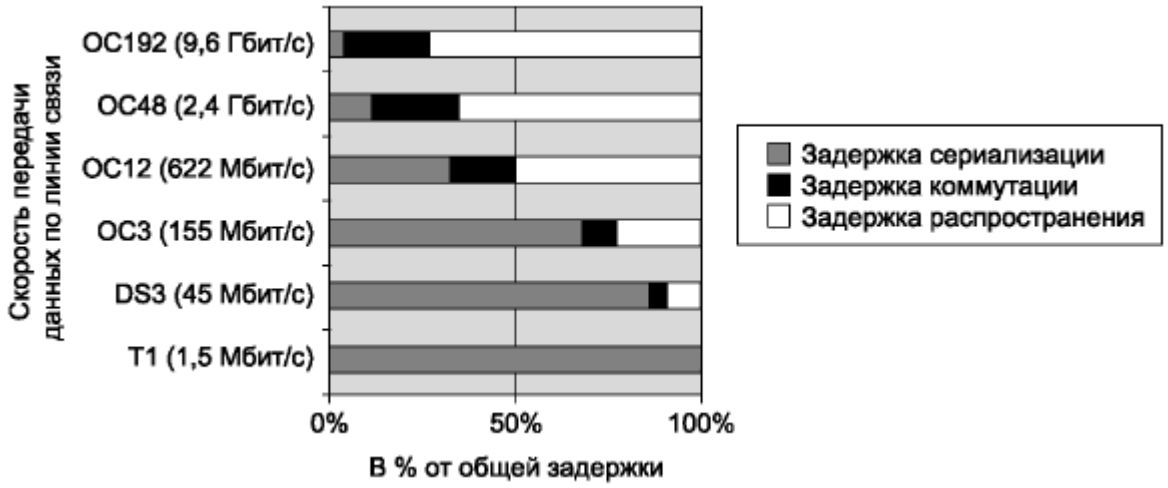


Рисунок 2.1 - Структура общей задержки при передаче пакета размером 1500 байт по каналу в зависимости от возрастания полосы пропускания

Обратите внимание, что задержка сериализации становится незначительной по сравнению с задержкой распространения по мере увеличения полосы пропускания канала. Задержка коммутации пренебрежимо мала в случае отсутствия пакетов в очередях маршрутизаторов, однако склонна к существенному увеличению при росте размеров очередей.

2.3.3 Потеря пакетов

Уровень потери пакетов (packet loss) определяет количество пакетов, отбрасываемых сетью во время передачи. Основными причинами потери пакетов являются перегрузка сети и повреждение пакетов во время передачи по линии связи. Чаще всего отбрасывание пакетов происходите местах перегрузки, где число поступающих пакетов намного превышает верхнюю границу размера выходной очереди. Кроме того, отбрасывание пакетов может быть вызвано недостаточным размером входного буфера. Как правило, уровень потери пакетов выражается как доля отброшенных пакетов за определенный интервал времени.

Некоторые приложения не способны нормально функционировать или же функционируют крайне неэффективно в случае потери пакетов. Подобные приложения требуют от сети гарантии надежной доставки всех пакетов.

Как правило, хорошо спроектированные сети характеризуются очень низким значением потери пакетов. Потеря пакетов также несвойственна приложениям, для которых были заранее зарезервированы требуемые этими приложениями ресурсы. Что касается волоконно-оптических линий связи со значением частоты появления ошибочных битов (Bit Error Rate — BER) 10E-9, то здесь потеря пакетов возможна только в случае их отбрасывания в местах перегрузки сети. Отбрасывание пакетов, к сожалению, является неизбежным явлением при негарантированной доставке трафика, хотя и в этом случае оно

обуславливается крайней необходимостью. Следует отметить, что отброшенные пакеты указывают на неэффективное использование ресурсов сети, часть которых была потрачена на доставку пакетов в точку, где они были потеряны[8].

2.4 Функции качества обслуживания

– Классификация и маркировка пакетов. Маршрутизаторы, расположенные на границе сети, используют функцию классификации для распознавания пакетов, принадлежащих различным классам трафика, в зависимости от значения одного или нескольких полей в заголовке TCP/IP. Функция маркировки пакетов используется для разметки классифицированного трафика путем установки значения поля IP-приоритета или поля кода дифференцированного обслуживания (Differentiated Services Code Point — DSCP).

– Управление интенсивностью трафика. Поставщики услуг используют ограничивающую функцию для приведения параметров поступающего в сеть клиентского трафика в соответствие с его профилем. В то же время, корпорации используют выравнивающую функцию для дозирования поступающего в сеть поставщика услуг трафика и выравнивания его интенсивности в соответствии с заданным профилем. Наиболее распространенной схемой дозирования трафика является так называемая схема корзины маркеров (token bucket).

– Распределение ресурсов. Наиболее распространенным механизмом обслуживания очередей в маршрутизаторах и коммутаторах современной Internet является ставший уже традиционным механизм "первым пришел, первым обслужен" (first-in, first-out). Несмотря на простоту реализации, для механизма FIFO характерно несколько фундаментальных проблем, затрудняющих выполнение функций качества обслуживания. Так, механизм FIFO не предусматривает приоритетной обработки чувствительного к задержке трафика путем его перемещения во главу очереди. Весь трафик обрабатывается одинаково, без учета принадлежности потоков к различным классам с разными требованиями к обслуживанию. Минимальное требование, предъявляемое к поддерживающему функции QoS алгоритму обслуживания очередей, — способность дифференцировать и определять требования к обработке различных пакетов. В соответствии с этими параметрами алгоритм обслуживания должен планировать порядок передачи поставленных в очередь пакетов. Частота обслуживания пакетов одного и того же потока трафика определяет выделенную этому потоку полосу пропускания.

– Предотвращение перегрузки и политика отбрасывания пакетов. Традиционный механизм обслуживания очередей FIFO предусматривает отбрасывание всех входящих пакетов после достижения максимального значения длины очереди. Подобный способ управления очередью получил название "отбрасывание хвоста" (tail drop) и характеризуется тем, что сигнал о

перегрузке поступает лишь в момент фактического переполнения очереди. К сожалению, механизм FIFO не предусматривает проведения каких-либо активных действий по предотвращению перегрузки или по уменьшению размера очереди с целью снижения времени задержки. Активный алгоритм управления очередями позволяет маршрутизатору предвидеть перегрузку еще до переполнения очереди.

– Сигнальный протокол QoS. Сигнальный протокол RSVP является частью разработанной организацией IETF архитектуры intserv, обеспечивающей предоставление сквозных услуг QoS в масштабах Internet. Протокол RSVP позволяет приложениям сообщать о требованиях к обслуживанию отдельных потоков трафика. Для определения количественных показателей качества обслуживания с целью управления доступом протокол RSVP использует служебные параметры.

– Коммутация. Главная функция маршрутизатора заключается в быстрой и эффективной коммутации входящего трафика на соответствующие выходные интерфейсы согласно информации, хранящейся в таблице продвижения пакетов. Традиционный механизм продвижения пакетов, несмотря на его эффективность, обладает низкой масштабируемостью; кроме того, его производительность оставляет желать лучшего в периоды нестабильного функционирования сети, что выражается в резком увеличении расходов на обслуживание кэша и снижении эффективности коммутации пакетов.

– Метод продвижения пакетов, учитывающий топологию сети, обладает бесспорными преимуществами перед методом, базирующимся на кэшировании пакетов, что обусловлено совпадением таблицы продвижения пакетов с таблицей маршрутизации. Механизм продвижения пакетов, учитывающий топологию сети, называется также методом скоростной коммутации пакетов Cisco (Cisco Express Forwarding— CEF).

– Маршрутизация. Традиционная маршрутизация осуществляется на основании адреса назначения пакета и предполагает выбор наименее короткого маршрута, хранящегося в таблице маршрутизации. К сожалению, подобный механизм является недостаточно гибким для некоторых сетевых сценариев. Маршрутизация на основе политики — это функция качества обслуживания, позволяющая заменить традиционный механизм маршрутизации пакетов механизмом, учитывающим всевозможные настраиваемые пользователем параметры. Современные протоколы маршрутизации, работающие по методу выбора наименее короткого пути, позволяют учитывать такие значения метрики, как административное расстояние, вес или число переходов. Маршрутизация пакетов осуществляется на основании хранящейся в таблице маршрутизации информации без учета требований потока трафика к качеству обслуживания или доступности сетевых ресурсов на всем протяжении маршрута. QoS-маршрутизация представляет собой механизм маршрутизации пакетов, учитывающий требования потоков трафика к качеству обслуживания и осуществляющий выбор маршрута в зависимости от наличия сетевых ресурсов.

3 Анализ работы механизма предотвращения перегрузки

Политика отбрасывания пакетов (packet drop policy) представляет собой алгоритм управления очередью, применяющийся для регулирования ее длины. Традиционный алгоритм обслуживания очередей "первым пришел, первым обслужен" (fist-in, first-out) использует достаточно простую политику "отбрасывания хвоста" (tail drop policy), в соответствии с которой любая попытка постановки пакета в полную очередь неминуемо завершится его отбрасыванием.

В настоящий момент основным транспортным протоколом Internet является протокол управления передачей (Transmission Control Protocol — TCP). В этом разделе рассматриваются механизмы предотвращения перегрузки протокола TCP, а также реакция TCP-трафика на применение политики "отбрасывания хвоста". Кроме того, рассмотрен алгоритм активного управления очередью RED (Random Early Detection — алгоритм произвольного раннего обнаружения), позволяющим предотвратить перегрузку сети путем превентивного отбрасывания пакетов с целью уведомления о возможной перегрузке источников TCP-соединения посредством механизма сквозного адаптивного управления с обратной связью.

В этом разделе также рассматривается взвешенный алгоритм произвольного раннего обнаружения (Weighted Random Early Detection — WRED), позволяющий настраивать различные RED-параметры в зависимости от значения поля IP-приоритета или класса трафика. Алгоритм WRED на основе потока (flow WRED) представляет собой расширение алгоритма WRED, предусматривающее возможность назначения штрафа с ненулевой вероятностью тем потокам, которые пытаются завладеть слишком большой долей доступных ресурсов[9].

3.1 Механизм медленного старта и предотвращение перегрузки

С целью поддержки механизма предотвращения заторов в сети источники TCP-соединения используют так называемое окно перегрузки (congestion window — cwnd). Инициализация окна перегрузки осуществляется в момент установки TCP-сеанса. В соответствии с механизмом медленного старта начальное значение окна перегрузки устанавливается равным одному сегменту (максимальный размер сегмента (maximum segment size — MSS) либо сообщается источником на другом конце TCP-соединения, либо устанавливается равным стандартному значению и, как правило, составляет 536 или 512 байт). Значение окна перегрузки представляет собой максимальный размер данных, которые может переслать TCP-отправитель в рамках заданного сеанса без получения подтверждения о доставке.

При получении подтверждения о доставке первого пакета TCP-источник увеличивает размер окна перегрузки до 2, что указывает на возможность отправки уже двух пакетов. Аналогично, при получении подтверждения о

доставке двух пакетов TCP-источник увеличивает размер окна перегрузки до 4. Таким образом, рост размера окна перегрузки является экспоненциальным. Следует отметить, что на самом деле рост размера окна перегрузки может и не быть строго экспоненциальным, поскольку TCP-получатель, как правило, не посылает подтверждение о доставке каждого пакета, а использует так называемые подтверждения с задержкой (подтверждение о получении двух пакетов). Описанное поведение источников TCP-соединения подчиняется алгоритму медленного старта, в соответствии с которым TCP-источник передает в сеть пакеты с интенсивностью, равной интенсивности получения подтверждений о доставке пакетов от TCP-получателя. Это делает протокол TCP самосинхронизирующимся (self-clocking) транспортным протоколом.

В соответствии с протоколом TCP сигналом о перегрузке сети является потеря пакета. TCP-источник обнаруживает перегрузку при отсутствии подтверждения о доставке пакета в течение заданного промежутка времени, называемого оценочным временным лимитом таймера повторной передачи (retransmit timer timeout — RTT). В сложившейся ситуации TCP-источник сбрасывает значение размера окна перегрузки до одного сегмента и перезапускает алгоритм медленного старта. Кроме этого, он также уменьшает пороговое значение алгоритма медленного старта (slow start threshold — ssthresh) до величины, равной половине размера окна перегрузки в момент повторной передачи пакета. Следует отметить, что при установке TCP-сеанса значение параметра ssthresh устанавливается равным либо размеру окна получателя, сообщенного TCP-источи и ком на другом конце соединения, либо стандартному значению в 65535 байт.

После достижения временного лимита таймера повторной передачи (RTT) отправитель следует алгоритму медленного старта до тех пор, пока размер окна перегрузки не достигнет величины ssthresh. Начиная с этого момента размер окна увеличивается линейно (с коэффициентом 1/cwnd) по мере получения подтверждений о доставке пакета. Замедление роста размера окна перегрузки вызвано тем, что значение параметра ssthresh представляет собой оценку доступной полосы пропускания данного TCP-соединения. Пример работы алгоритма медленного старта и алгоритма предотвращения перегрузки схематически представлен на рисунке 3.1.

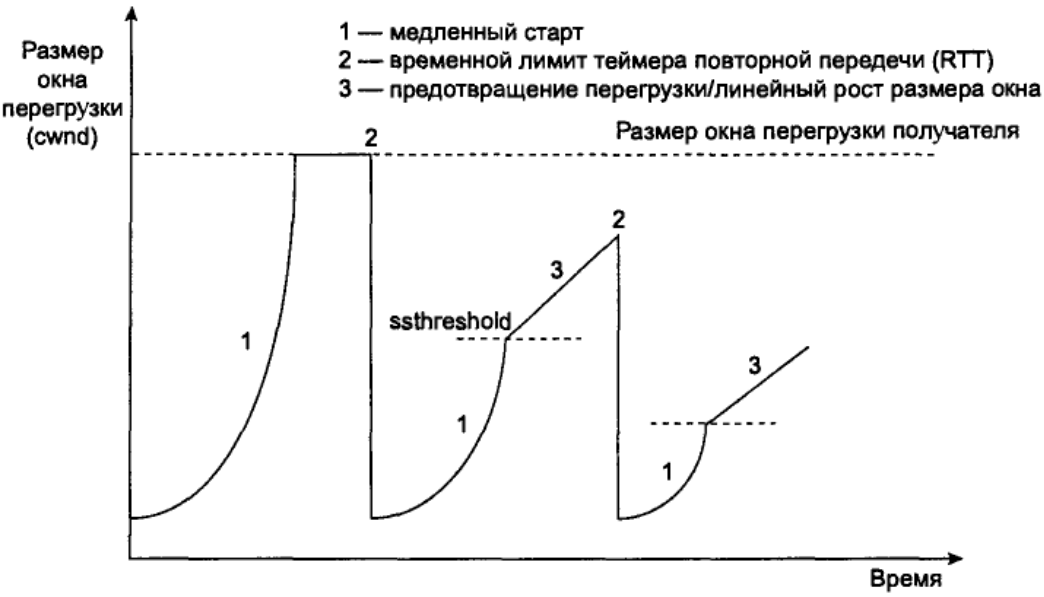


Рисунок 3.1 - Изменение размера TCP - окна в соответствии с алгоритмами медленного старта и предотвращения перегрузки

Когда потеря пакета вызвана не перегрузкой сети, ожидание превышения временного лимита повторной передачи пакета (RTT) может весьма негативным образом сказаться на производительности TCP-соединения, что особенно характерно для высокоскоростных сетей. Для того чтобы избежать подобного развития ситуации, протокол TCP использует алгоритм быстрой повторной передачи и алгоритм быстрого восстановления.

3.2 Реакция TCP-трафика на политику "отбрасывания хвоста"

Традиционная политика обработки пакетов, которые должны быть поставлены в очередь, достигшую своего максимального размера, заключается в их отбрасывании. Подобная "дискриминация" пакетов продолжается до тех пор, пока длина очереди не уменьшится за счет передачи уже находящихся в ней пакетов. Алгоритм управления очередью, в соответствии с которым любая попытка постановки пакета в полную очередь неминуемо завершится его отбрасыванием, получил название алгоритма "отбрасывания хвоста " (tail drop).

Поскольку отбрасывание пакета является сигналом о перегрузке сети для источника TCP-соединения, механизм "отбрасывания хвоста" сообщает о перегрузке сети лишь в момент фактического переполнения очереди. В результате отбрасывания пакета источник TCP-соединения уменьшает размер окна до одного сегмента и перезапускает алгоритм медленного старта, что приводит к резкому уменьшению исходящего TCP-трафика.

Поскольку типичный магистральный маршрутизатор Internet или другой IP-сети большого размера в отдельный момент времени обрабатывает тысячи TCP-потоков, применение алгоритма управления очередью "отбрасывание хвоста" может привести к потере пакетов для очень большого числа TCP-

соединений. Получив уведомления о перегрузке сети, множество TCP-источников практически одновременно уменьшат интенсивность передаваемого ими трафика, что приведет к резкому уменьшению размера очереди маршрутизатора.

Все TCP-источники, перезапустившие механизм медленного старта и сбросившие значение размера окна до одного сегмента, начнут экспоненциальное увеличение размеров окна, ведущее к росту интенсивности обрабатываемого очередью трафика. Монотонное увеличение интенсивности трафика приведет к переполнению очереди и отбрасыванию пакетов. Таким образом, алгоритм "отбрасывания хвоста" и во второй раз приведет к потере пакетов для очень большого числа TCP-соединений, а также к резкому уменьшению размеров очереди. По мере увеличения размеров TCP-окна всех источников, перезапустивших механизм медленного старта, интенсивность обрабатываемого очередью трафика будет расти, что вскорости приведет к уже известному сценарию развития событий.

Периодическое резкое снижение интенсивности трафика и перегрузка сети приводят к волноподобному изменению размера очереди, получившему название эффекта глобальной синхронизации (global synchronization). Эффект глобальной синхронизации схематически представлен на рисунке 3.2.

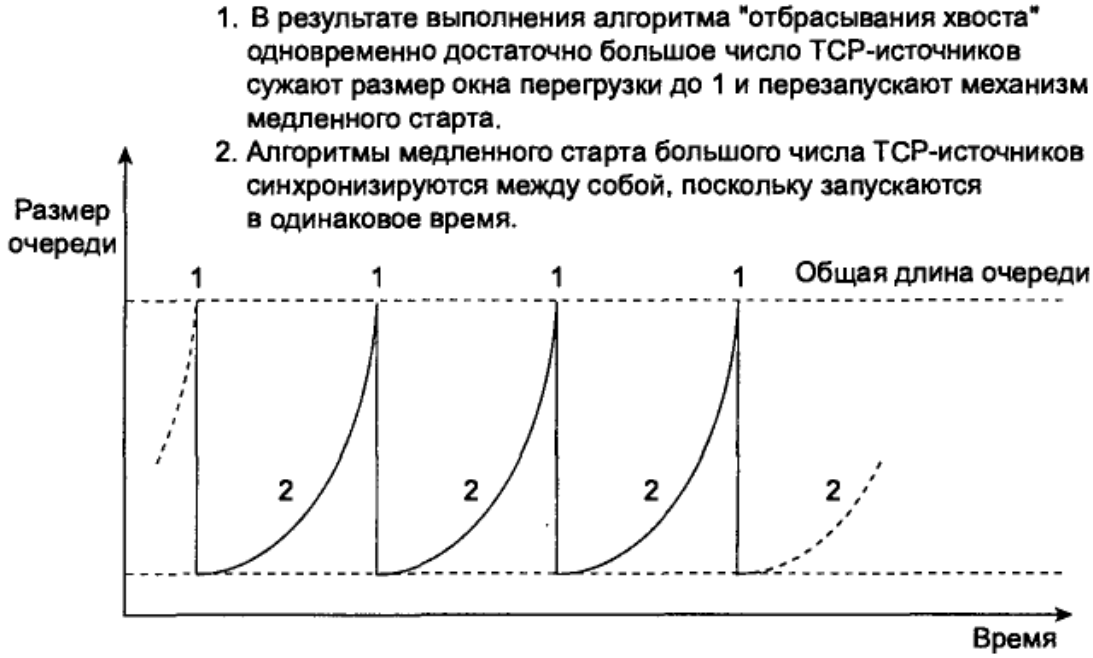


Рисунок 3.2 - Эффект глобальной синхронизации

Эффект глобальной синхронизации обязан своим названием одновременному (синхронному) перезапуску алгоритмов медленного старта множества TCP-источников, вызванного работой алгоритма "отбрасывания хвоста". Помимо нежелательного изменения размеров очереди, этот эффект способен также привести к возрастанию дрожания задержки трафика и снижению пропускной способности всей сети[10].

3.3 Алгоритм превентивного управления очередью

С поведением TCP-источников в моменты работы алгоритма "отбрасывания хвоста" связана необходимость проведения превентивного управления очередью с целью сигнализации о перегрузке сети до фактического переполнения очереди и контроля за размером очереди с целью снижения задержки обработки пакетов. Алгоритм произвольного раннего обнаружения (Random Early Detection) представляет собой алгоритм предотвращения перегрузки, предложенный Салли Флойдом и Ваном Якобсоном. Данный механизм активного управления очередью обладает существенными преимуществами по сравнению с традиционным механизмом "отбрасывания хвоста".

Как уже упоминалось ранее, механизм RED использует превентивный подход к предотвращению перегрузки сети. Вместо ожидания фактического переполнения очереди, RED начинает отбрасывать пакеты с ненулевой вероятностью, когда средний размер очереди превысит определенное минимальное пороговое значение. Вероятностный подход к отбрасыванию пакетов позволяет быть уверенным в том, что механизм RED отбросит пакеты всего лишь нескольких произвольно выбранных потоков, тем самым помогая избежать эффекта глобальной синхронизации. Напомним, что отбрасывание пакета представляет собой сигнал TCP-источнику о необходимости уменьшить интенсивность передаваемого трафика для соответствующего потока, что достигается за счет перезапуска алгоритма медленного старта.

Если средний размер очереди будет продолжать увеличиваться несмотря на отбрасывание произвольных пакетов, то это приведет к линейному росту вероятности отбрасывания. В соответствии с механизмом RED вероятность отбрасывания пакетов растет прямо пропорционально увеличению среднего размера очереди от минимального до максимального порогового значения. Средний размер очереди строго ограничен максимальным пороговым значением, поскольку в этом случае вероятность отбрасывания пакетов достигает своего наибольшего значения (100 процентов). Другими словами, главная цель механизма произвольного раннего обнаружения (RED) заключается в минимизации среднего размера очереди, а значит, и обшей задержки трафика.

Определение вероятности отбрасывания пакета базируется на взвешенном экспоненциальном значении среднего размера очереди. Это позволяет избежать предвзятого отношения механизма RED к характеризующимся кратковременными всплесками потокам трафика в условиях продолжительной перегрузки сети.

Если же средний размер очереди весьма невелик и находится ниже минимального порогового значения, механизм RED не способен обеспечить существенного преимущества по сравнению с традиционными механизмами управления очередью. С другой стороны, при затяжном периоде перегрузки сети поведение механизма RED, несмотря на длинную очередь и высокое

максимальное пороговое значение, аналогично поведению классического механизма "отбрасывания хвоста". Таким образом, основное предназначение механизма RED заключается в сглаживании временных всплесков трафика и предупреждении длительной перегрузки сети посредством уведомления источников трафика о необходимости снижения интенсивности передачи информации. Если источники проявят способность к взаимодействию и одновременно уменьшат интенсивность передаваемого трафика, это поможет предотвратить перегрузку сети. В противном случае средний размер очереди достаточно скоро достигнет максимального порогового значения, что приведет к отбрасыванию всех поступающих пакетов.

Ниже перечислены некоторые из основных целей механизма раннего произвольного обнаружения.

– Минимизация дрожания задержки пакетов путем контроля за средним размером очереди.

– Предотвращение эффекта глобальной синхронизации TCP-трафика.

– Обеспечение непредвзятого обслуживания трафика, характеризующегося кратковременными всплесками.

– Строгое ограничение максимального среднего размера очереди.

Фактически, механизм произвольного раннего обнаружения базируется на двух следующих алгоритмах.

– Алгоритм вычисления среднего размера очереди. Определяет допустимый уровень всплеска трафика в очереди.

– Алгоритм вычисления вероятности отбрасывания пакетов. Определяет вероятность (частоту) отбрасывания пакетов для заданного среднего размера очереди.

3.3.1 Алгоритм вычисления среднего размера очереди

При определении вероятности отбрасывания пакетов механизм RED вычисляет не текущий, а экспоненциально взвешенный средний размер очереди. Текущий средний размер очереди определяется на основании предыдущего среднего и текущего действительного размера. Использование механизмом RED среднего размера очереди обусловлено стремлением реагировать только на продолжительную перегрузку сети и "не замечать" моментальных всплесков трафика.

Экспоненциальный весовой коэффициент является ключевым параметром, который определяет относительный вклад предыдущего среднего и текущего размера очереди в новый средний размер очереди. Практика показала, что наиболее приемлемым значением экспоненциального весового коэффициента n является 9. Увеличение экспоненциального весового коэффициента приведет к доминированию предыдущего среднего размера очереди над ее текущим размером в аспекте вычисления нового среднего размера очереди. Напротив, уменьшение экспоненциального весового

коэффициента приведет к возрастанию значимости текущего размера очереди при вычислении ее нового среднего размера[11].

Большое значение коэффициента n обуславливает математическую близость нового и предыдущего среднего размера очереди, а также позволяет механизму RED более сдержанно реагировать на моментальные изменения в ее текущем размере, что выражается в следующем поведении.

– Значение среднего размера очереди изменяется медленно, для него крайне не характерны резкие скачки. Механизм RED достаточно сдержанно относится к временным всплескам трафика, стараясь выровнять текущий размер очереди.

– Механизм RED не спешит иници ровать процесс отбрасывания пакетов, который, тем не менее, может продолжаться некоторое время после снижения действительного размера очереди ниже минимального порогового значения.

– Если значение коэффициента n слишком велико, механизм RED может и вовсе перестать реагировать на перегрузку сети, поскольку в этом случае текущий размер очереди практически не будет влиять на вычисление ее среднего размера. Пакеты будут передаваться или отбрасываться так, как если бы механизм RED и вовсе не использовался.

Напротив, маленькое значение коэффициента n обуславливает математическую близость нового среднего и текущего размера очереди, что выражается в следующем поведении механизма RED.

– Средний размер очереди изменяется очень быстро, для него характерна сильная зависимость от флуктуации потока трафика.

– Механизм RED немедленно реагирует на длинную очередь, однако как только ее размер оказывается ниже минимального порогового значения, отбрасывание пакетов прекращается.

– Если значение коэффициента n слишком мало, механизм RED начинает очень остро реагировать на временные всплески трафика, что выражается в неоправданном отбрасывании пакетов.

3.3.1 Алгоритм вычисления вероятности отбрасывания пакетов

Вероятность отбрасывания пакетов представляет собой функцию, линейно зависящую от среднего размера очереди. Помимо этого, данная функция зависит также от минимального порогового значения, максимального порогового значения и знаменателя граничной вероятности (mark probability denominator), определяющего часть отбрасываемых пакетов при достижении средним размером очереди максимального порогового значения. Например, если знаменатель граничной вероятности равен 10, то при достижении средним размером очереди максимального порогового значения механизм RED будет отбрасывать 1 из 10 пакетов.

Когда средний размер очереди превышает минимальное пороговое значение, механизм RED начинает отбрасывать пакеты. Интенсивность отбрасывания пакетов возрастает прямо пропорционально возрастанию среднего размера очереди до тех пор, пока он не достигнет максимального порогового значения.

Когда средний размер очереди превышает максимальное пороговое значение, механизм RED отбрасывает все пакеты, предназначенные для постановки в очередь. График вероятности отбрасывания пакетов схематически представлен на рисунок 3.3.

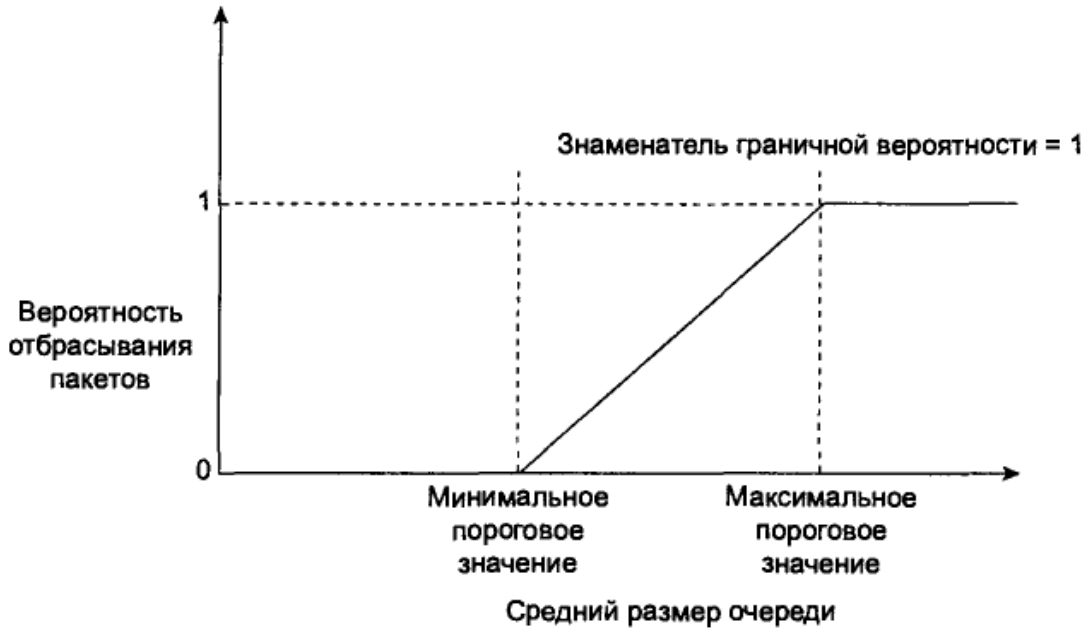


Рисунок 3.3 - График вероятности отбрасывания пакетов механизмом RED

3.4 Взвешенный алгоритм произвольного раннего обнаружения

Взвешенный алгоритм произвольного раннего обнаружения (Weighted Random Early Detection) предоставляет различные уровни обслуживания пакетов в зависимости от вероятности их отбрасывания и обеспечивает избирательную установку параметров механизма RED на основании значения поля IP-приоритета. Другими словами, алгоритм WRED предусматривает возможность более интенсивного отбрасывания пакетов, принадлежащих определенным типам трафика, и менее интенсивного отбрасывания всех остальных пакетов.

3.4.1 Реализация алгоритма WRED

Алгоритм WRED может выполняться как на центрачьном процессоре маршрутизатора, так и в распределенном режиме на маршрутизаторах Cisco серии 7500, построенных на базе интерфейсных плат VIP. По умолчанию все

уровни приоритета трафика характеризуются одинаковым максимальным пороговым значением и различными минимальными пороговыми значениями. Таким образом механизм WRED обеспечивает более интенсивное отбрасывание низкоприоритетных пакетов и менее интенсивное отбрасывание высокоприоритетных. Стандартное минимальное пороговое значение для трафика приоритета 0 составляет половину максимального порогового значения.

Механизм WRED может быть сконфигурирован для обслуживания различных классов трафика с помощью модульного интерфейса командной строки (CLI) QoS, который позволяет определить параметры механизма RED отдельно для каждого класса.

В маршрутизаторах Cisco серии 12000 алгоритм WRED присутствует как в программной, так и в аппаратной реализации в зависимости от версии линейной платы. Маршрутизаторы Cisco серии 12000 характеризуются наличием восьми классов CoS-очередей. Каждой CoS-очереди может быть поставлено в соответствие одно или несколько значений IP-приоритета. После определения CoS-очередей вы можете сконфигурировать параметры механизма RED отдельно для каждой CoS-очереди. Поскольку данная платформа Cisco построена на базе коммутационной архитектуры, механизм RED может быть активизирован как для очередей коммутационной матрицы на принимающей стороне, так и для очередей интерфейсов на передающей стороне.

3.4.2 Применение механизма WRED для предотвращения перегрузки

Поставщик услуг предлагает своим клиентам четыре класса обработки трафика: "платиновый", "золотой", "серебряный" и "бронзовый". Дифференцирование трафика осуществляется в базовой сети поставщика услуг посредством установки значения поля IP-приоритета (4, 3, 2 и 1 в зависимости от оплаченного клиентом уровня обслуживания). Наряду с приоритетным трафиком сеть поставщика услуг поддерживает обработку трафика методом негарантированной доставки (IP-приоритет 0). Каналы поставщика услуг, соединяющие его с другими ISP верхнего уровня, состоящими с ним в отношении соседства, испытывают перегрузку и, следовательно, нуждаются в применении политики отбрасывания пакетов[12].

С целью предотвращения перегрузки сетевой администратор должен сконфигурировать механизм активного управления очередью WRED на всех интерфейсах, соединяющих поставщика услуг с соседними ISP. Конфигурация механизма WRED осуществляется посредством применения команды random-detect.

Обратите внимание, что минимальное пороговое значение механизма WRED, соответствующее высокоприоритетному трафику, должно быть большим, чем минимальное пороговое значение низкоприоритетного трафика. Выполнение этого условия необходимо для обеспечения строгой очередности

отбрасывания пакетов, согласно которой первыми начинают отбрасываться низкоприоритетные пакеты.

Когда алгоритм WRED выполняется на центральном процессоре маршрутизатора, то он применяется к выходной очереди интерфейса, а пороговые значения определяются следующим образом.

Минимальное пороговое значение для трафика с приоритетом i.

– Максимальное пороговое значение. Равняется размеру выходной очереди удержания.

В результате выполнения команды show queueing random-detect на экран выводится следующая статистика обработки пакетов.

– Статистика произвольного отбрасывания пакетов (параметр random и Random drop).

– Количество пакетов, отброшенных механизмом WRED при условии нахождения среднего размера очереди между минимальным и максимальным пороговым значением.

– Статистика отбрасывания пакетов в соответствии с алгоритмом "отбрасывания хвоста" (параметр tail и Tail drop). Количество пакетов, отброшенных механизмом WRED при условии превышения средним размером очереди максимального порогового значения.

– Граничная вероятность (параметр mark-prob и Mark probability). Вероятность отбрасывания пакетов в момент достижения средним размером очереди максимального порогового значения.

Статистика выполнения алгоритма WRED в распределенном режиме на маршрутизаторе Cisco серии 7500, построенном на базе интерфейсных плат VIP

Когда алгоритм WRED выполняется в распределенном режиме на маршрутизаторах Cisco серии 7500, построенных на базе интерфейсных плат VIP, все вычисления производятся с использованием локальных процессоров линейных плат, а не центрального процессора маршрутизатора. Следовательно, распределенный механизм WRED оперирует очередью VIP-платы, а не выходной очередью интерфейса.

Распределенный механизм WRED использует схему межпроцессного взаимодействия на основе метода скоростной коммутации пакетов Cisco (Cisco Express Forwarding) для распространения конфигурационных параметров и статистической информации между процессорами RSP (Route/Switch Processor — процессор маршрутизации и коммутации) и интерфейсными платами VIP. Из этого следует, что выполнение механизма WRED в распределенном режиме требует активизации режима коммутации CEF. Статистика отбрасывания пакетов для заданного интерфейса включает в себя статистику пакетов, отброшенных распределенным механизмом WRED (Distributed WRED).

Механизм DWRED вычисляет стандартное максимальное пороговое значение на основании размера пула (грубо говоря, размера очереди платы VIP), максимального размера единицы передачи информации (MTU) и полосы пропускания заданного интерфейса. Размер пула, в свою очередь, зависит от объема установленной на VIP-плате памяти и ряда других факторов, что

существенно затрудняет его однозначное определение. Следовательно, размер пула может помочь только приблизительно оценить объем допустимого всплеска. При необходимости вы можете установить максимальное пороговое значение механизма WRED, отличающееся от стандартного.

3.4.3 Алгоритм WRED на основе потока

Сигнал о перегрузке сети воспринимается должным образом только адаптивными потоками TCP-трафика, в то время как не обладающий способностью к адаптации UDP-трафик не реагирует на уведомление о перегрузке и не снижает свою интенсивность. Учитывая такое поведение UDP-трафика, несложно представить себе ситуацию, в которой во время перегрузки сети неадаптивные потоки передают данные с намного большей интенсивностью, чем потоки, обладающие способностью к адаптации. Следовательно, на неадаптивные потоки трафика приходится львиная доля ресурсов по сравнению с потоками, снижающими свою интенсивность в ответ на получение сигнала о перегрузке. Алгоритм WRED на основе потока (flow WRED) представляет собой модификацию алгоритма WRED, предусматривающую штрафование потоков, отнимающих чрезмерную долю ресурсов.

С целью обеспечения равномерного обслуживания активных потоков трафика механизм WRED классифицирует все устанавливаемые в очередь пакеты в зависимости от их приоритета и потока трафика, к которому они относятся. Кроме этого, WRED поддерживает информацию о состоянии всех активных потоков (active flows), т.е. потоков, хотя бы один пакет которых поставлен на обработку в какую-либо из очередей.

Информация о состоянии активных потоков используется для определения справедливой доли выделенных потоку ресурсов очереди (размер очереди/количество активных потоков), а также для выявления и штрафования потоков, отнимающих чрезмерно большой объем ресурсов.

Чтобы механизм WRED более адекватно реагировал на всплески потоков трафика, вы можете увеличить справедливую долю ресурсов каждого потока путем применения так называемого коэффициента масштабирования.

Поток, требования которого превышают справедливую долю ресурсов с учетом коэффициента масштабирования, штрафуется путем увеличения ненулевой вероятности отбрасывания для всех вновь пришедших пакетов.

В качестве примера рассмотрим действия, предпринимаемые механизмом WRED на основе потока в отношении только что поставленного в очередь пакета. При определении вероятности отбрасывания пакета механизм WRED на основе потока учитывает как значение поля IP-приоритета пакета, так и информацию о состоянии активных потоков. От IP-приоритета пакета зависят сконфигурированные (либо стандартные) минимальное и максимальное пороговые значения. Если средний размер очереди ниже минимального порогового значения, то вероятность отбрасывания пакета устанавливается

равной нулю (другими словами, этот пакет не будет отброшен). Если же средний размер очереди находится между минимальным и максимальным пороговым значением, то в расчет принимается информация о состоянии активных потоков трафика. Так, если пакет принадлежит потоку, превысившему справедливую долю ресурсов с учетом коэффициента масштабирования, механизм WRED увеличивает вероятность отбрасывания этого пакета путем уменьшения соответствующего максимального порогового значения, как показано ниже.

Ненулевая вероятность отбрасывания пакета рассчитывается на основании минимального и нового максимального порогового значения. Поскольку результатом снижения максимального порогового значения является существенное увеличение угла наклона кривой вероятности отбрасывания, шансы пакета быть отброшенным резко возрастают.

Если же поток трафика не превышает справедливой доли ресурсов с учетом коэффициента масштабирования, то ненулевая вероятность отбрасывания пакета рассчитывается по стандартному методу.

Когда средний размер очереди превышает максимальное пороговое значение, механизм WRED отбрасывает все пакеты, предназначенные для постановки в очередь. Эта ситуация показана на рисунке 3.4.

Следует отметить, что механизм WRED на основе потока увеличивает вероятность отбрасывания пакетов только для тех потоков трафика, чьи требования превысили справедливую долю ресурсов с учетом коэффициента масштабирования. Во всем остальном поведение механизма WRED на основе потока аналогично поведению стандартного механизма WRED[13].

Рисунок 3.4 - График вероятности отбрасывания пакета при использовании механизма WRED

4 Экспериментальное исследование работы механизма WRED

В данной диссертационной работе изучается процесс использования, алгоритма WRED, так как этот механизм реализован практически во всех современных маршрутизаторах, а остальные его модификации лишь бурно обсуждаются и не имеют практической реализации в сетевых устройствах.

Несмотря на то, что теме предотвращения перегрузок уделяется большое внимание различными публикациями, остается проблема настроек параметров ля алгоритма WRED. Многие исследователи WRED согласны с тем, что влияние алгоритма на качество передачи потоков сильно зависит от правильного задания его параметров, но до сих пор нет вразумительной инструкции, как на практике выбирать значения этих параметров [14]. В работе проводится исследование влияния параметров на работу алгоритма WRED, а так же даются рекомендации по выбору оптимальных настроек алгоритма. Исследование проводилось на реальном оборудовании компании Cisco Systems с использованием программного генератора трафика IxChariot.

В работе поведение очередей в случае использования алгоритма WRED. Особенностью этого алгоритма является то, что решение о постановке пакета в очередь принимается по-разному, в зависимости от уровня заполнения буфера (длины очереди). Механизм WRED использует следующие параметры: wq - весовой коэффициент усреднения; TMIN - нижний порог средневзвешенной длины очереди; TMAX - верхний порог средневзвешенной длины очереди; pc - максимальное значение вероятности отбрасывания пакетов для области между TMIN и TMAX. В алгоритме WRED устанавливаются два порога TMIN и TMAX. Пока усреднённая длина очереди ниже TMIN, любой входящий пакет поступает в буфер. В области между TMIN и TMAX вероятность отбрасывания пакета линейно растёт от 0 до значения pc [15]. График вероятности отбрасывания пакетов механизмом WRED представлен на рисунке 4.1.

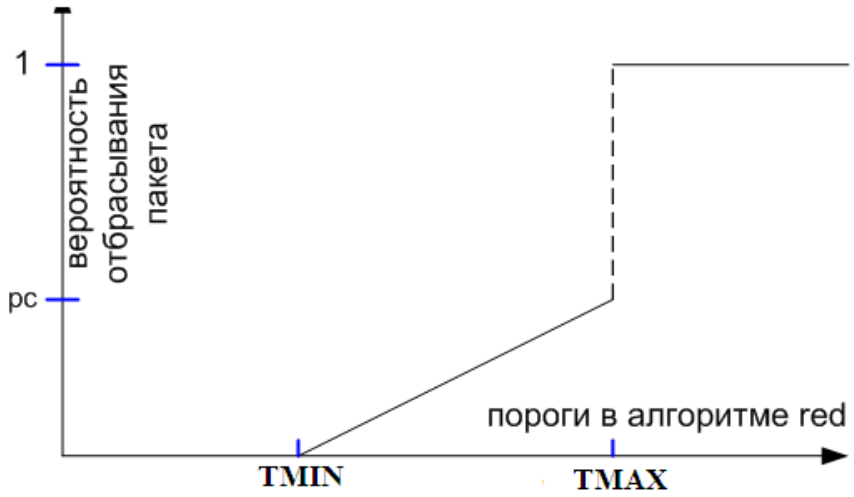


Рисунок 4.1 – График вероятности отбрасывания пакетов механизмом WRED

После достижения порога TMAX все поступающие пакеты отбрасываются. Усреднение длины очереди производится согласно формуле.

$$\bar{q}(k+1) = (1 - w_q)\bar{q}(k) + w_q q(k) \quad (4.1)$$

здесь q(k+1) - усредненная длина очереди в (k+1)–й момент времени, wq -параметр усреднения, q , q - усреднённая и текущая длины очереди в k–й момент времени. При малом значении wq процесс WRED не сразу начинает отбрасывать пакеты при перегрузке, зато продолжит отбрасывание, даже когда перегрузки уже нет (очередь сократилась ниже TMIN). Усреднение длины очереди - важный компонент при управлении процессом буферизации. Без усреднения процесс буферизации был бы подвержен сильному влиянию случайных флуктуаций входящего потока пакетов, но именно усреднение является причиной возникновения осцилляций длины очереди. Зависимость принятия решения об отбрасывании того или иного пакета определяется значением усреднённой длины очереди, которое может существенно отличаться от значения текущей длины очереди[16].

Основной задачей диссертации является экспериментальное исследование влияния параметров алгоритма WRED на его работу и нахождение оптимальных параметров для заданного уровня перегрузки. Для проведения эксперимента была создана модель подключения двух удаленных офисов через сеть сервис провайдера. Исследования проводились на оборудовании компании Cisco Systems: на уровне доступа сети центрального офиса и сети филиала были использованы коммутаторы Catslist 2960 и Catalyst 3560, соответственно, в качестве граничного оборудования для подключения к сети сервис провайдера были задействованы маршрутизаторы Cisco ISR 2811. Схема экспериментальной модели изображена на рисунке 4.2. Связь точка-точка между офисами организована при помощи технологии туннелирования GRE (без настройки протокола IPSec). Пропускная способность канала соединяющего офисы – 2 Мбит/с. Для генерации разнородного трафика был использован программный генератор трафика IxChariot.

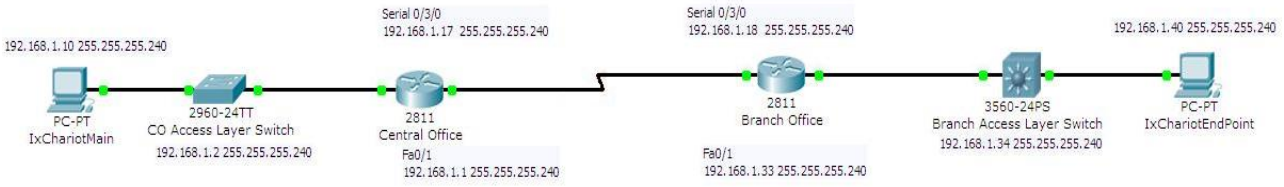


Рисунок 4.2 - Схема экспериментальной модели сети

Генератором трафика была воссоздана реальная ситуация на участках сети, узкий участок сети в экспериментальной модели - это канал между

маршрутизаторами удаленных офисов. Скорость передачи данных внутри локальных сетей 100 Мбит/с, а канал соединяющий локальные сети центрального офиса и филиала имеет пропускную способность 2 Мбит/с. Генерируемый трафик проходит до граничного маршрутизатора без потерь и задержек, а на выходе из маршрутизатора, по направлению к сети сервис провайдера наблюдается перегрузка буфера интерфеса, что приводит к большим потерям пакетов и чувствительным задержкам. Именно на этом участке и было проведено исследование зависимости поведения перегрузки от изменения параметров протокола WRED.

В эксперименте передавались потоки UDP и TCP пакетов через тестовый канал между двумя граничными маршрутизаторами. На маршрутизаторе был настроен механизм WRED с параметрами по умолчанию, затем меняя значения вышеперечисленных параметров алгоритма WRED, было изучено влияние этих изменений на поведение перегрузки.

Результаты экспериментов снимались на конечных точках генератора трафика и с статистических данных работы маршрутизатора. В таблице 4.1 показан трафик генерируемый генератором трафика Ix Chariot.

Таблица 4.1 – Трафик генерируемый программным обеспечением Ix Chariot.

№ пары	Конечная точка 1	Конечная точка 2	Протокол	Приоритет	Имя скрипта/потока
IP Telephony					
1	192.168.1.10	192.168.1.40	RTP	EF/5	G.726
2	192.168.1.10	192.168.1.40	RTP	EF/5	G.726
3	192.168.1.10	192.168.1.40	RTP	EF/5	G.711u
4	192.168.1.10	192.168.1.40	RTP	EF/5	G.726
5	192.168.1.10	192.168.1.40	RTP	EF/5	G.726
6	192.168.1.10	192.168.1.40	RTP	EF/5	G.726
7	192.168.1.10	192.168.1.40	RTP	EF/5	G.726
8	192.168.1.10	192.168.1.40	RTP	EF/5	G.726
9	192.168.1.40	192.168.1.10	RTP	EF/5	G.726
10	192.168.1.40	192.168.1.10	RTP	EF/5	G.726
11	192.168.1.40	192.168.1.10	RTP	EF/5	G.711u
12	192.168.1.40	192.168.1.10	RTP	EF/5	G.726
Internet					
13	192.168.1.10	192.168.1.40	TCP	AF21/2	DNS
14	192.168.1.10	192.168.1.40	TCP	AF21/2	HTTPS
15	192.168.1.10	192.168.1.40	TCP	AF21/2	SMTP
16	192.168.1.10	192.168.1.40	TCP	AF21/2	POP3
17	192.168.1.10	192.168.1.40	TCP	AF21/2	HTTP
18	192.168.1.10	192.168.1.40	TCP	AF21/2	NNTP
19	192.168.1.40	192.168.1.10	TCP	AF21/2	DNS
20	192.168.1.40	192.168.1.10	TCP	AF21/2	HTTPS

21	192.168.1.40	192.168.1.10	TCP	AF21/2	SMTP
22	192.168.1.40	192.168.1.10	TCP	AF21/2	POP3
23	192.168.1.40	192.168.1.10	TCP	AF21/2	HTTP
24	192.168.1.40	192.168.1.10	TCP	AF21/2	NNTP
LotusNotes					
25	192.168.1.10	192.168.1.40	TCP	AF11/1	Notes_Attach
26	192.168.1.10	192.168.1.40	TCP	AF11/1	Notes_Browser.
27	192.168.1.10	192.168.1.40	TCP	AF11/1	Notes_Create_Note
28	192.168.1.10	192.168.1.40	TCP	AF11/1	Notes_Indexed
29	192.168.1.10	192.168.1.40	TCP	AF11/1	Notes_Replicate
30	192.168.1.10	192.168.1.40	TCP	AF11/1	Notes_Send_Mail
31	192.168.1.10	192.168.1.40	TCP	AF11/1	Notessnd.scr
32	192.168.1.40	192.168.1.10	TCP	AF11/1	Notes_Replicate_
33	192.168.1.40	192.168.1.10	TCP	AF11/1	Notes_Send_Mail_
34	192.168.1.40	192.168.1.10	TCP	AF11/1	Notessnd
Management					
35	192.168.1.10	192.168.1.40	TCP	AF31/3	Exchsend
36	192.168.1.10	192.168.1.40	TCP	AF31/3	Telnet
37	192.168.1.40	192.168.1.10	TCP	AF31/3	Exchsend
38	192.168.1.40	192.168.1.10	TCP	AF31/3	Telnet
39	192.168.1.40	192.168.1.10	TCP	AF31/3	Citrix_
40	192.168.1.40	192.168.1.10	TCP	AF31/3	Citrix_
41	192.168.1.40	192.168.1.10	TCP	AF31/3	Citrix_ICA_Open
42	192.168.1.40	192.168.1.10	TCP	AF31/3	Citrix_Server
43	192.168.1.40	192.168.1.10	TCP	AF31/3	Citrix_ICA_Word_
44	192.168.1.10	192.168.1.40	TCP	AF31/3	Citrix_ICA_Excel
45	192.168.1.10	192.168.1.40	TCP	AF31/3	Citrix_ICA_IE_
46	192.168.1.10	192.168.1.40	TCP	AF31/3	Outlook_Open_
47	192.168.1.10	192.168.1.40	TCP	AF31/3	Citrix_ICA_
48	192.168.1.10	192.168.1.40	TCP	AF31/3	Citrix_ICA_Word
SAP					
49	192.168.1.10	192.168.1.40	TCP	AF41/4	SAP_R3_
50	192.168.1.10	192.168.1.40	TCP	AF41/4	SAP_R3
51	192.168.1.10	192.168.1.40	TCP	AF41/4	SAP_R3_Results
52	192.168.1.10	192.168.1.40	TCP	AF41/4	SAPlogin
53	192.168.1.10	192.168.1.40	TCP	AF41/4	SAP_ Sales_Order
54	192.168.1.10	192.168.1.40	TCP	AF41/4	SAPauthp
55	192.168.1.10	192.168.1.40	TCP	AF41/4	SAP_R3_Post
SAP					
56	192.168.1.10	192.168.1.40	TCP	BE/0	FTPget
57	192.168.1.10	192.168.1.40	TCP	BE/0	FTPput
58	192.168.1.10	192.168.1.40	TCP	BE/0	eDonkeyPublicize

59	192.168.1.40	192.168.1.10	TCP	BE/0	eDonkey2000Download
60	192.168.1.40	192.168.1.10	TCP	BE/0	FTPget.scr
61	192.168.1.40	192.168.1.10	TCP	BE/0	FTPput.scr
62	192.168.1.40	192.168.1.10	TCP	BE/0	eDonkey2000Publicize
63	192.168.1.40	192.168.1.10	TCP	BE/0	FTPget.scr
64	192.168.1.40	192.168.1.10	TCP	BE/0	FTPput.scr
65	192.168.1.10	192.168.1.40	TCP	BE/0	FTPget.scr
66	192.168.1.10	192.168.1.40	TCP	BE/0	FTPput.scr
67	192.168.1.10	192.168.1.40	TCP	BE/0	eDonkeyPublicize
68	192.168.1.10	192.168.1.40	TCP	BE/0	eDonkeyDownload
69	192.168.1.10	192.168.1.40	TCP	BE/0	eDonkeyPublicize

4.1 Исследование влияния параметра pc на алгоритм WRED

В первом эксперименте исследовалось влияние параметра p_c на проходящие через алгоритм WRED потоки. В эксперименте можно было плавно менять значение параметра pc на интерфейсе маршрутизатора, но в маршрутизаторах параметр p_c задается как дробь 1/k, где целое значение k лежит в диапазоне <1-65535>. Через тестовый канал передавалось 6 потоков, параметры которых представлены в таблице 4.2.

Таблица 4.2 – Параметры WRED для тестовых потоков.

ip precedence	wq	TMIN	TMAX
0	1/512	20	40
1	1/512	22	40
2	1/512	24	40
3	1/512	26	40
4	1/512	28	40
5	1/512	31	40

Значение p_c варьировалось в интервале [0.001÷1]. Полная полоса тестируемого канала составляла 2 Мбит/с. В ходе эксперимента для каждого значения p_c было проделано по три измерения. Время одного эксперимента 1 минута. Для каждого измерения получена зависимость количество потерянных пакетов разных потоков от параметра p_c. На рисунке 4.3 изображена зависимость процента потерь от параметра p_c для трафика разных приоритетов.

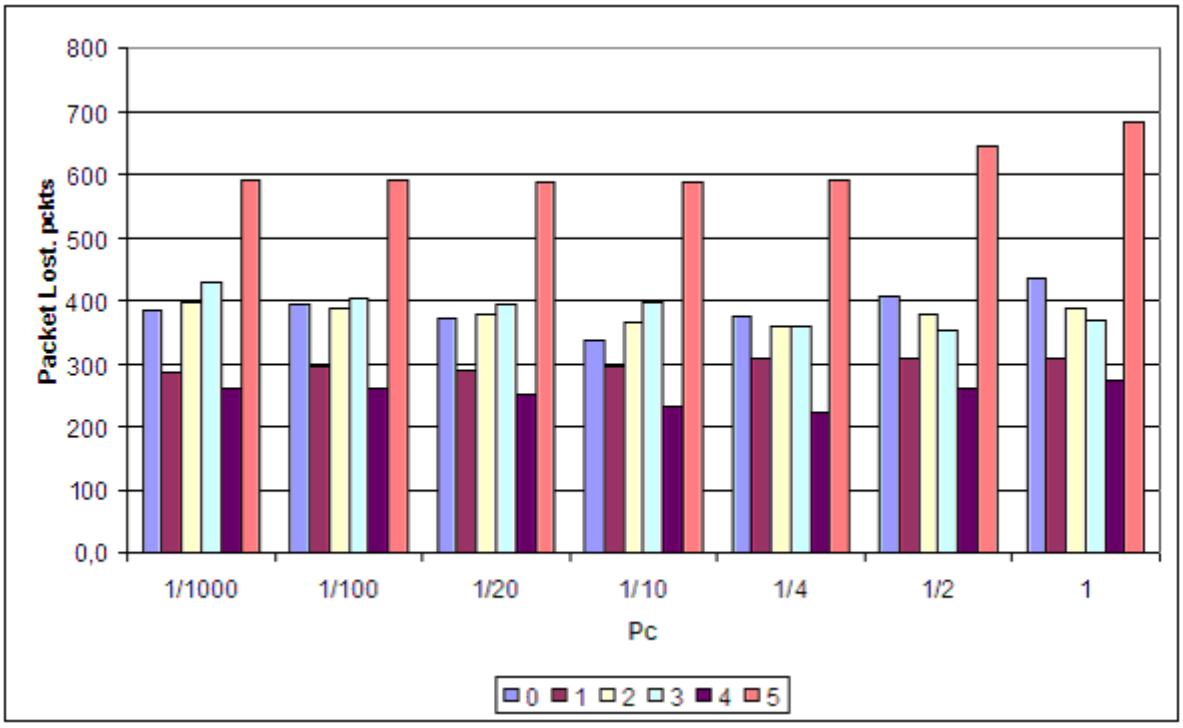


Рисунок 4.2 - Зависимость количества потерянных пакетов от параметра p_c

Из рисунка видно, что значения p_c <0.25 вызывают одинаковый отклик системы. Зависимости процента потерянных пакетов в единицу времени для разных значений p_c практически одинаковы. Параметр p_c влияет лишь на время переходного процесса, который возникает при перегрузке канала. Исходя из полученных результатов, мы убедились, что процент потери пакетов имеет наименьшие значения на отметке 1/10, что является значением p_c по умолчанию на маршрутизаторах. И так, при данном значении w_q разумно использовать значения p_c =1/10. Параметр p_c очень слабо влияет на процент потери пакетов в случае перегрузки канала. Настройка параметра p_c влияет на продолжительность начального переходного процесса, но не влияет на характеристики передачи потока пакетов при продолжительной постоянной перегрузке.

Всплеск задержки доставки пакетов возникает, когда текущая длина очереди WRED растёт, а средневзвешенная длина очереди ещё не достигла порога T_{MAX}. Когда средневзвешенная длина очереди достигает верхнего порога, текущая длина очереди начинает быстро сокращаться, т.к. все пакеты поступающие на вход отбрасываются. Для разных значений p_c длительность перехода будет отличаться. Но конечное стационарное состояние системы одинаково для всех значений параметра p_c Параметр p_c важен в случае кратковременных перегрузок, когда система не успевает выйти на стационарный режим работы.

Также мы выяснили что начения задержки и джиттера не зависят от параметра p_c , среднее значение задержки на время всех экспериментов составила 164 мс, джиттер – 18.3 мс. Среднее значение характеристики MOS

составила 2.13, что явилось следствием больших задержек при передачи голосового трафика.

4.2 Исследование влияния параметра w_q на алгоритм WRED

При определении вероятности отбрасывания пакетов механизм RED вычисляет не текущий, а экспоненциально взвешенный средний размер очереди. Текущий средний размер очереди определяется на основании предыдущего среднего и текущего действительного размера Использование механизмом RED среднего размера очереди обусловлено стремлением реагировать только на продолжительную перегрузку сети и "не замечать" моментальных всплесков трафика. Средний размер очереди вычисляется по формуле:

$$\bar{q}(k+1) = (1 - w_q)\bar{q}(k) + w_q q(k) \qquad (4.2)$$

здесь q(k+1) - усредненная длина очереди в (k+1)–й момент времени, w_q - параметр усреднения, q , q - усреднённая и текущая длины очереди в k–й момент времени. Параметр w_q отвечает за амплитуду колебаний длин очередей, а также за время реакции алгоритма WRED на флуктуации перегрузки. Через канал передавался тот же поток, что и в предыдущих измерениях, потоки промаркированы соответствующим значением приоритета. В эксперименте можно было плавно менять значение параметра w_q на интерфейсе маршрутизатора, но в маршрутизаторах параметр w_q задается как дробь $1/2^n$, где целое значение n - это экспоненциальный весовой коэффициент, лежит в диапазоне <1-16>. Параметры настройки алгоритма WRED для потоков были следующими: T_{MIN}- значения по умолчанию для каждого приоритета, T_{MAX}=40, pc=1/10. Измерения были проведены со следующими значениями параметра w_q : 1/2, 1/4, 1/16, 1/512, 1/4096, 1/65536.Результаты измерений зависимости количества потерянных пакетов от параметра w_q представлены в таблице 4.3.

Таблица 4.3 – . Зависимость количества потерянных пакетов от параметра w_q, для трфика с различным приоритетом.

wq /Pri	0	1	2	3	4	5
1/2	481	195	107	219	24	303
1/4	433	135	100	188	21	296
1/16	419	136	94	169	26	291
1/512	343	131	82	210	25	266
1/4096	431	138	94	244	32	448
1/65536	557	159	136	248	51	604

Экспоненциальный весовой коэффициент является ключевым параметром, который определяет относительный вклад предыдущего среднего и текущего размера очереди в новый средний размер очереди.

Увеличение экспоненциального весового коэффициента приведет к доминированию предыдущего среднего размера очереди над ее текущим размером в аспекте вычисления нового среднего размера очереди. Напротив, уменьшение экспоненциального весового коэффициента приведет к возрастанию значимости текущего размера очереди при вычислении ее нового среднего размера.

Большое значение коэффициента n обуславливает математическую близость нового и предыдущего среднего размера очереди, а также позволяет механизму WRED более сдержанно реагировать на моментальные изменения в ее текущем размере, что выражается в следующем поведении.

– Значение среднего размера очереди изменяется медленно, для него крайне нехарактерны резкие скачки.

– Механизм RED достаточно сдержанно относится к временным всплескам трафика, стараясь выровнять текущий размер очереди.

– Механизм RED не спешит инициировать процесс отбрасывания пакетов, который, тем не менее, может продолжаться некоторое время после снижения действительного размера очереди ниже минимального порогового значения.

Напротив, маленькое значение коэффициента n обуславливает математическую близость нового среднего и текущего размера очереди, что выражается в следующем поведении механизма RED.

– Средний размер очереди изменяется очень быстро, для него характерна сильная зависимость от флуктуации потока трафика.

– Механизм RED немедленно реагирует на длинную очередь, однако как только ее размер оказывается ниже минимального порогового значения, отбрасывание пакетов прекращается.

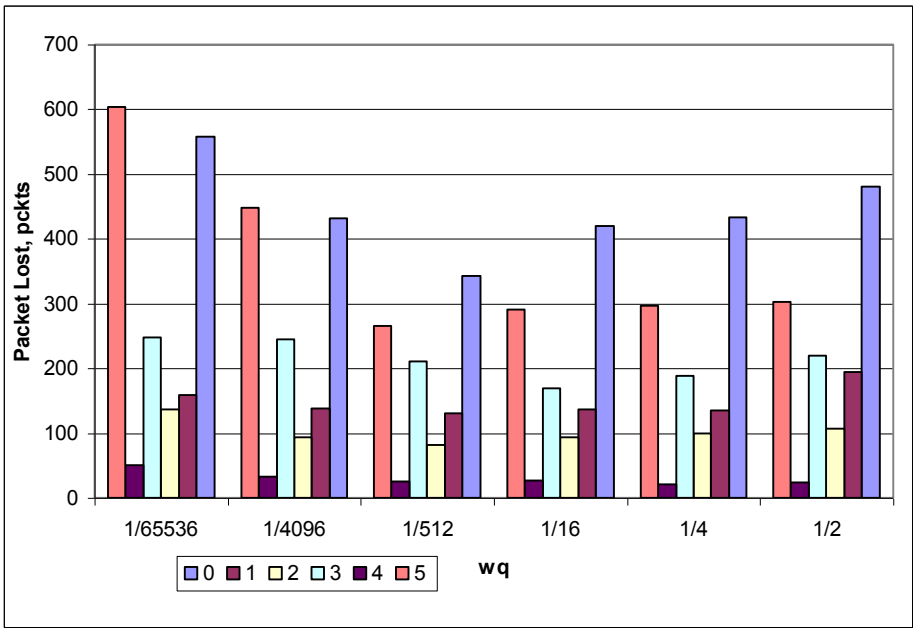


Рисунок. 4.3. Зависимость количества потерянных пакетов от параметра w_q

Из рисунка 4.3 видно, что при малых значениях параметра w_q процент потерянных пакетов максимальный, что можно объяснить тем , что при больших значениях экспоненциального весового коэффициента механизм RED может перестать реагировать на перегрузку сети, поскольку в этом случае текущий размер очереди практически не будет влиять на вычисление ее среднего размера. Пакеты передаются или отбрасываются так, как если бы механизм RED и вовсе не использовался. В этом случае количество отброшенных пакетов механизмом Tail Drop значительно преобладает над количеством отброшенных пакетов алгоритмом WRED.

При больших значениях параметра wq так же имеем большой процент потерянных пакетов, но в этом случае большее количество пакетов из числа потерянных было отброшено механизмом WRED. В такой ситуации механизм WRED начинает очень остро реагировать на временные всплески трафика, что выражается в неоправданном отбрасывании пакетов.

Минимальный процент потерянных пакетов был зафиксирован на значении 1/512 параметра w_q .

Получена зависимость влияния параметра w_q на средние значения задержки передачи пакетов и джиттера. На рисунке 4.4 изображена зависимость средних значений задержки и джиттера и параметра MOS от параметра w_q.

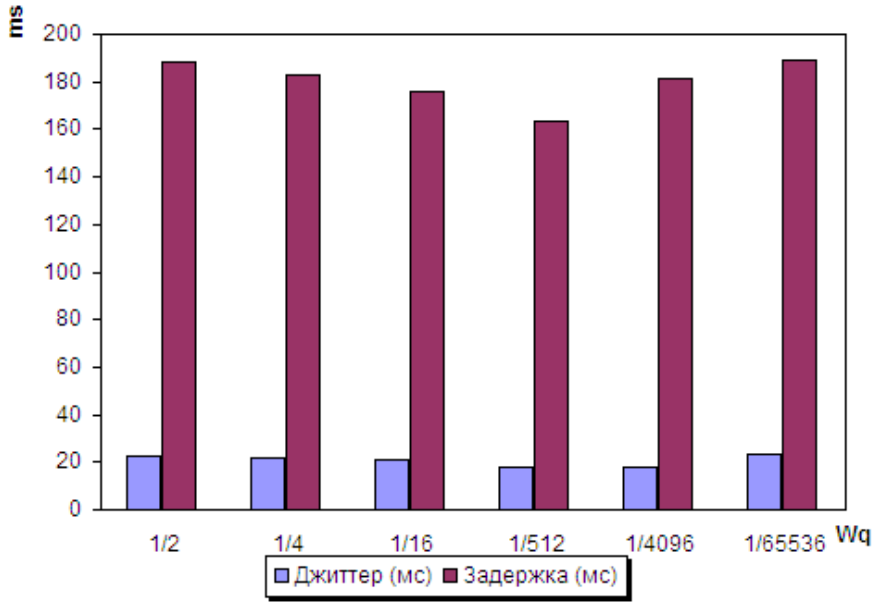


Рисунок 4.4 Зависимость средних значений задержки и джиттера и параметра MOS от параметра w_q

Улучшение усредненной субъективной оценки MOS является следствием уменьшения задержек передачи голосового трафика. Наилучшее качество телефонного разговора зафиксировано на значении w_q=1/16, на этой же отметке были минимальные значения задержки и джиттера голосового трафика.

4.3 Исследование влияния порогов T_{MIN} и T_{MAX} на алгоритм WRED

Пороги алгоритма WRED влияют на минимальное и максимальное время задержки пакетов в случае возникновения перегрузки. Во время эксперимента объем и интенсивность генерируемого трафика оставались постоянными. Нижние пороги алгоритма WRED для разных приоритетов тоже не менялись. Менялось значение верхнего порога T_{MAX} = [35-60] пакетов с шагом 5 пакетов. Не имеет смысл ставить большие значение порога TH_{MAX}, так как при длительной перегрузке происходит переполнение буфера, отведённого на исходящем интерфейсе маршрутизатора под пакетную очередь. Для значения 1/10 параметра pc получены зависимости значения средней задержки и количество потерянных пакетов от значения T_{MAX} для голосовго трафика с приоритетом - 5 , измерена также зависимость джиттера при передаче голосового пакета .

Таблица 4.4 – Параметры WRED для тестовых потоков.

ip precedence	wq	TMIN	pc
0	1/512	20	1/10
1	1/512	22	1/10
2	1/512	24	1/10
3	1/512	26	1/10
4	1/512	28	1/10
5	1/512	31	1/10

Зависимость на рисунке 4.5 демонстрирует потери пакетов при передаче голосового трафика для разных значений порога T_{MAX}.

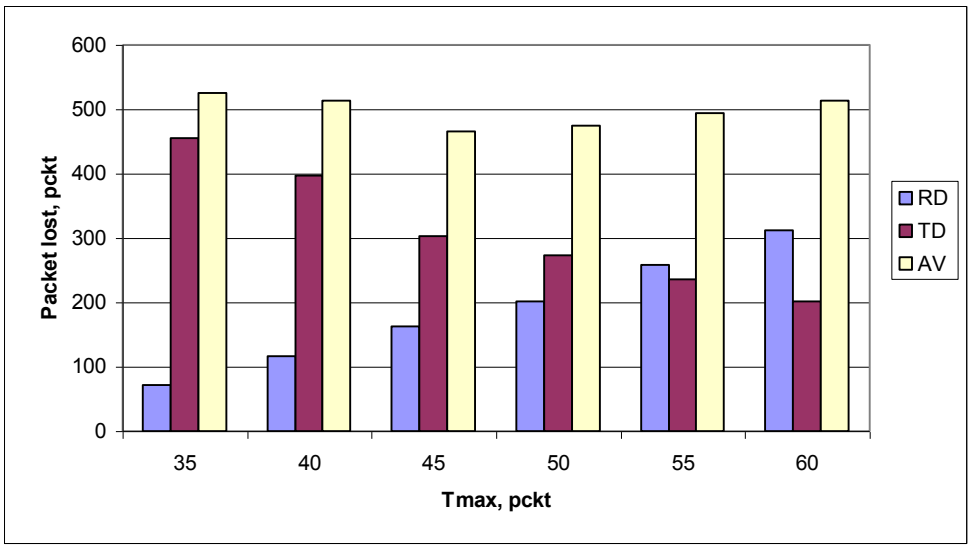


Рисунок 4.5 - Зависимость среднего значения потерянных пакетов, потерь вызванных механизмом Random Drop и Tail Drop, от значения T_{MAX}

Из рисунка 4.5 видно, что при изменении значения максимального порога $T_{MAX,}$ помимо изменения значения потерянных пакетов, изменяется и причина отбрасывания пакетов маршрутизатором. При значении близком к минимальному порогу отбрасывания время достижения очередью максимального порога значительно уменьшается, следовательно работа механизма Tail Drop начинает свою работу намного раньше. Это и является причиной большого количества отброшенных пакетов механизмом Tail Drop. Далее при увеличении T_{MAX}, увеличивается процент отброшенных пакетов механизмом Random Drop, за счет большего времени требуемого очереди для достижения $T_{MAX.}$ Тут начинает преобладать механизм Random Drop, соответственно потери пакетов вызванные механизмом Tail Drop уменьшаются.

Как видно из графика, среднее же значение потерянных пакетов при очень маленьких и очень больших значениях T_{MAX} является максимальным. Оптимальным же значением порога T_{MAX} является значение $T_{MAX} = 45$.

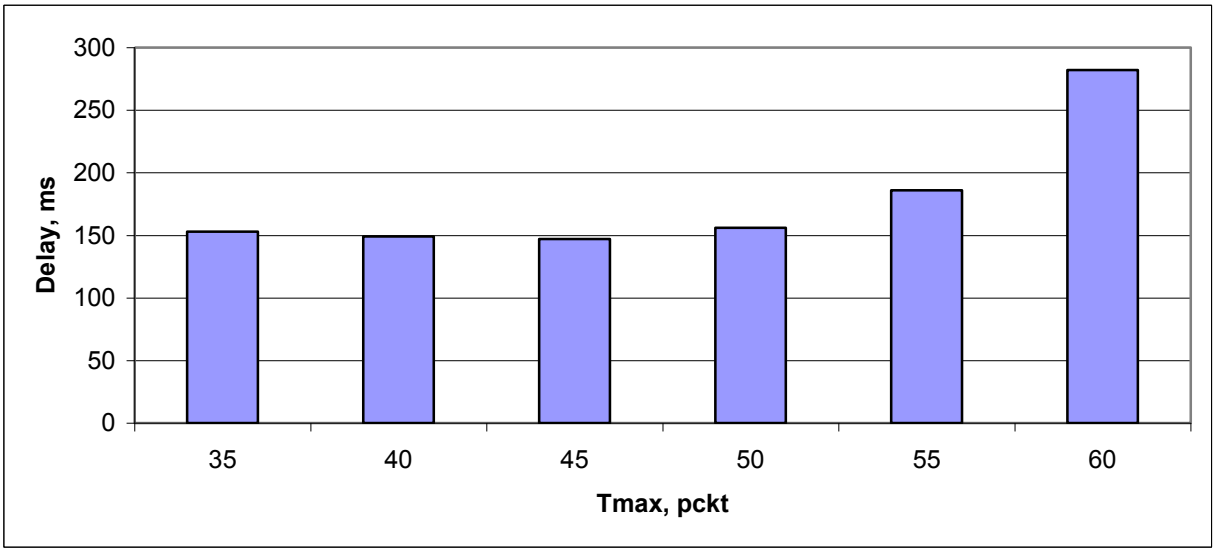


Рисунок 4.6 - Зависимость задержки передачи голосового трафика от значения T_{MAX}

Зависимость на рисунке 4.6 демонстрирует поведение задержки при передаче пакета для разных значений порога T_{MAX}. При высоких значениях T_{MAX} задержка максимальная, причиной является увеличение времени ожидания отправки пакета в буфере интерфейса. Так же было измерено значение коэффициента MOS. Максимальное значение MOS 3,23 наблюдается при значении T_{MAX} равным 45.

Вывод:

По результатам экспиремента можно сделать выводы, что значения p_c <0.25 вызывают одинаковый отклик системы. Зависимости процента потерянных пакетов в единицу времени для разных значений p_c практически одинаковы. Параметр p_c влияет лишь на время переходного процесса, который возникает при перегрузке канала. Исходя из полученных результатов, мы

убедились, что процент потери пакетов имеет наименьшие значения на отметке 1/10, что является значением p_c по умолчанию на маршрутизаторах. И так, при данном значении w_q разумно использовать значения p_c =1/10. Параметр p_c очень слабо влияет на процент потери пакетов в случае перегрузки канала. Настройка параметра p_c влияет на продолжительность начального переходного процесса, но не влияет на характеристики передачи потока пакетов при продолжительной постоянной перегрузке.

Всплеск задержки доставки пакетов возникает, когда текущая длина очереди WRED растёт, а средневзвешенная длина очереди ещё не достигла порога T_{MAX}. Когда средневзвешенная длина очереди достигает верхнего порога, текущая длина очереди начинает быстро сокращаться, т.к. все пакеты поступающие на вход отбрасываются. Для разных значений p_c длительность перехода будет отличаться. Но конечное стационарное состояние системы одинаково для всех значений параметра p_c Параметр p_c важен в случае кратковременных перегрузок, когда система не успевает выйти на стационарный режим работы.

При малых значениях параметра w_q процент потерянных пакетов максимальный, что можно объяснить тем , что при больших значениях экспоненциального весового коэффициента механизм RED может перестать реагировать на перегрузку сети, поскольку в этом случае текущий размер очереди практически не будет влиять на вычисление ее среднего размера. Пакеты передаются или отбрасываются так, как если бы механизм RED и вовсе не использовался. В этом случае количество отброшенных пакетов механизмом Tail Drop значительно преобладает над количеством отброшенных пакетов алгоритмом WRED.

При больших значениях параметра w_q так же имеем большой процент потерянных пакетов, но в этом случае большее количество пакетов из числа потерянных было отброшено механизмом WRED. В такой ситуации механизм WRED начинает очень остро реагировать на временные всплески трафика, что выражается в неоправданном отбрасывании пакетов.

При изменении значения максимального порога $T_{MAX,}$ помимо изменения значения n отерянных пакетов, изменяется и причина отбрасывания пакетов маршрутизатором. При значении близком к минимальному порогу отбрасывания время достижения очередью максимального порога значительно уменьшается, следовательно работа механизма Tail Drop начинает свою работу намного раньше. Это и является причиной большого количества отброшенных пакетов механизмом Tail Drop. Далее при увеличении T_{MAX}, увеличивается процент отброшенных пакетов механизмом Random Drop, за счет большего времени требуемого очереди для достижения $T_{MAX.}$ Тут начинает преобладать механизм Random Drop, соответственно потери пакетов вызванные механизмом Tail Drop уменьшаются.

5 Расчет требований к качеству обслуживания

Основной целью данной работы является исследование механизма WRED для повышения качества работы IP-сети. Известно, что размеры пакетов передаваемых через такую сеть влияют на ее работу [17].

В данной работе проводятся расчеты, показывающие то, как влияют параметры пакета на характеристики сети. Рассмотрим примерную структуру такой сети. В качестве примера рассмотрим типичную операторскую мультисервисную IP-сеть.

5.1 Расчёт производительности узла доступа

Расчёт производительности сети необходимо проводить с учётом всех абонентов, пользующихся услугами. Условно разделим их на 3 группы:

– пользователи телефонии – 60%;
– пользователи телефонии и передачи данных-35%;
– пользователи телефонии, передачи данных и видео-5%.

Схема групп пользователей показана на рисунке 5.1

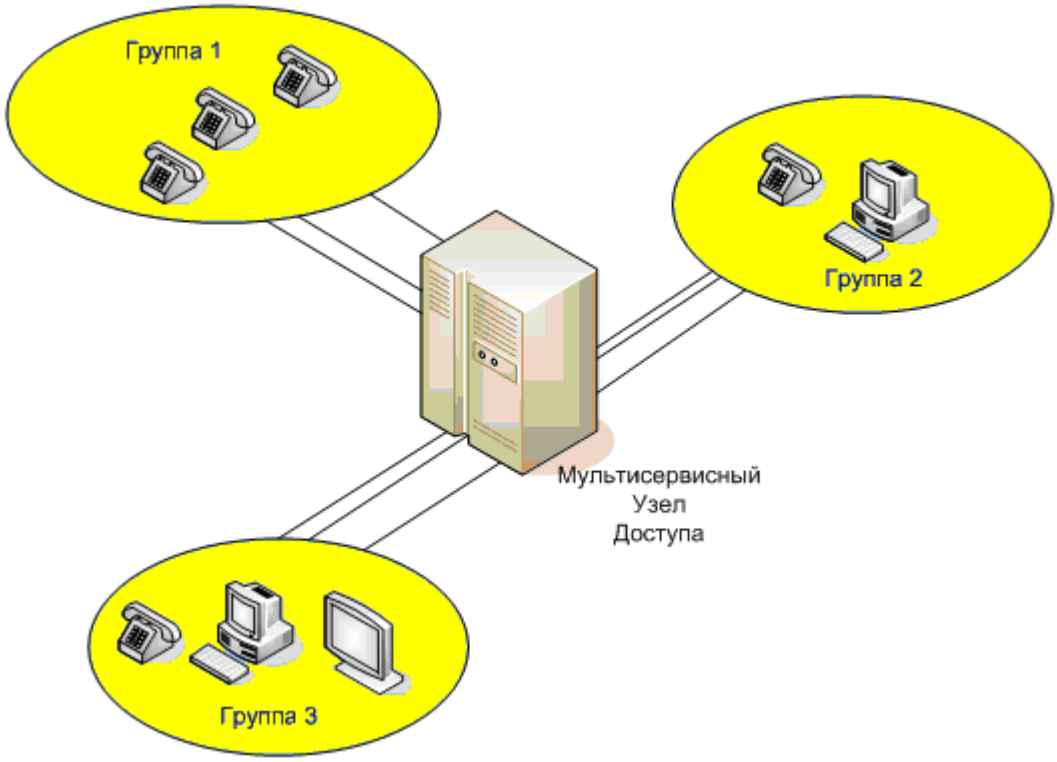


Рисунок 5.1 - Состав абонентов сети доступа

Доля абонентов 1 группы составляет $\pi_1 \approx 60$ %. Это традиционные абоненты, совершающие в среднем $f_1 = 5$ вызовов в час средней длительностью $t_1 = 2$ минуты.

Доля абонентов 2 группы, использующих голосовые сервисы и сервисы передачи информации, составляет $\pi_2 \approx 35$ %. Нагрузка, создаваемая этими абонентами, складывается из двух компонентов: телефония и интернет. Параметры телефонной нагрузки совпадают с аналогичными параметрами для группы 1, $f_2 = f_1 = 5$ вызовов в час, $t_2 = t_1 = 2$ минуты. Объём переданных информации в час наибольшей нагрузки ограничивается 10 Мбайт.

Доля абонентов группы 3, приносящих наибольших удельный доход, составляет $\pi_3 \approx 5$ %. Составляющее трафика для этих пользователей складывается следующим образом: телефония, интернет, видео. Параметры трафика телефонии совпадают с аналогичными параметрами для группы 2, т.е. $f_3 = f_2 = f_1 = 5$ вызовов в час, $t_3 = t_2 = t_1 = 2$ минуты. При расчёте трафика передачи данных необходимо учесть, что пользователи этой группы, как правило, активнее используют ftp и пиринговые сети. Допустим, что они потребляют до 70 Мбайт трафика. Время просмотра видео в час наибольшей нагрузки достигает 50 минут.

Определим число IP-пакетов, генерируемых каждой группой в часы наибольшей нагрузки, при условии, что мультисервисный узел доступа обслуживает N = 3500 абонентов. Расчеты будут произведены для кодеков G.726-32 и G.711u [18].

5.1.1 Расчёт числа пакетов от первой группы (телефония)

Рассчитаем число пакетов создаваемых пользователями телефонии, использующие выбранные ранее кодеки G.726-32 и G.711u. Параметры кодеков представлены в таблице 5.1.

Т а б л и ц а 5.1 - Параметры кодеков

Кодек	Скорость передачи, кбит/с	Длительность датаграм-мы, мс	Задержка пакетизации, мс	Полоса пропускания для двунаправ-ленного соединения, кГц	Задержка в джиттер-буфере	Теоре-тическая максималь-ная оценка MOS
G.711u	64	20	1	174,4	2 датаграммы, 40 мс	4,4
G.711a	64	20	1	174,4	2 датаграммы, 40 мс	4,4
G.726-32	32	20	1	110.4	2 датаграммы, 40 мс	4,22
G.729	8	20	25	62,4	2 датаграммы, 40 мс	4,07

Рассчитаем сначала параметры сети при кодеке G.711 и G.726 соответственно. Длительность дейтаграммы T_{PDU} равна 20 мс, согласно рекомендации RFC 1889. При этом в секунду передаётся

$$n_1 = 1/ T_{PDU,} \quad (5.1)$$

$$n_1 = 1/0,02 = 50, \text{ кадров/с}$$

Размер пакетизированных данных

$$h = v \cdot T_{PDU} \quad (5.2)$$

где v – скорость кодирования, байт/с;
h – размер пакетизированных данных;
T_{PDU} – длительность одной речевой выборки (длительность пакета).

При использовании кодека G.711 скорость кодирования

$$v = 64000/8 = 8000, \text{ байт/с}$$

$$h = 8000 \cdot 0,020 = 160, \text{ байт}$$

При использовании кодека G.726 скорость кодирования

$$v = 32000/8 = 4000, \text{ байт/с}$$

$$h = 4000 \cdot 0,020 = 80, \text{ байт}$$

Что бы определить размер пакета необходимо учесть длину заголовков :

- Ip – 20 байт;
- UDP – 8 байт;
- RTP – 12 байт.

Суммарный размер пакета для (G.711): 160 + 20 + 8 +12 = 200, байт.

Суммарный размер пакета для (G.726): 80 + 20 + 8 +12 = 120, байт.

Для определения числа пакетов, для первой группы абонентов, необходимо учесть их долю в общей структуре пользователей, количество вызовов в час наибольшей нагрузки, среднюю длительность разговора.

$$N_1 = n_1 \cdot t_1 \cdot f_1 \cdot \pi_1 \cdot N \quad (5.3)$$

где N_1 – число пакетов, генерируемое первой группой пользователей в час наибольшей нагрузки;
n_1 – число пакетов, генерируемых в секунду одним абонентом;
t_1 – средняя длительность разговора в секундах для первой группы абонентов;
f_1 – число вызовов в час наибольшей нагрузки для первой группы абонентов;
π_1 – доля пользователей группы 1 в общей структуре абонентов;
N – общее число пользователей.

5.1.2 Расчёт числа пакетов от второй группы (телефония и интернет)

Рассуждения, приведённые для первой группы абонентов, в полной мере применяем и ко второй группе для расчёта числа пакетов, возникающих в результате пользования голосовыми видами сервиса. Разница будет лишь в индексах.

$$N_{2\_т} = n_1 \cdot t_2 \cdot f_2 \cdot \pi_2 \cdot N \qquad (5.4)$$

где $N_{2\_т}$ – число пакетов, генерируемое второй группой пользователей в час наибольшей нагрузки при использовании голосовых сервисов;

n_1 – число пакетов, генерируемых в секунду одним абонентом;

t_2 – средняя длительность разговора в секундах для второй группы абонентов;

f_2 – число вызовов в час наибольшей нагрузки для второй группы абонентов;

π_2 – доля пользователей группы 2 в общей структуре абонентов;

N – общее число пользователей.

Для расчёта числа пакетов, генерируемых второй группой пользователей при использовании сервисов передачи данных, необходимо найти размер пакетов. При построении сети NGN, как правило, на одном или нескольких участках сети на уровне звена данных используется та или иная разновидность технологии Ethernet, поэтому использовать пакеты которые превышают максимальную длину поля данных Ethernet, нету смысла. При передаче длинный пакет рано или поздно будет фрагментирован, что приведёт, во-первых, к излишней нагрузке на свитчи, и, во-вторых, к возможным перезапросам в случае потерь. Кроме того, при использовании пакетов большого размера затрудняется обеспечение качества обслуживания и на магистральной сети, и в сети доступа. Более того, как правило, корпоративные пользователи устанавливают на границе своей сети «firewall», который ограничивает максимальный размер кадра. Поэтому для расчёта выберем одинаковые размеры пакетов и при передаче данных, и при передаче голосового трафика – полезная нагрузка 160 и 80 байт соответственно для кодеков G.711 и G.726. При передаче данных вместо протоколов RTP и UDP используется TCP, вносящий точно такую же избыточность (20 байт).

Для расчёта числа пакетов в часы наибольшей нагрузки необходимо задаться объёмом переданных данных. Предположим, что абоненты второй группы относятся к интернет-сёрферам, т.е. в основном просматривают веб-страницы. Средний объём данных, переданных за час при таком способе подключения, составит около V_2 = 10 Мбайт ≈ 80·1024·1024 бит. Число пакетов, переданных в ЧНН, будет равно

$$N_{2\_д} = \pi_2 \cdot N \cdot V_2/h \qquad (5.5)$$

где $N_{2\_д}$ – количество пакетов, генерируемых в час наибольшей нагрузки

абонентами второй группы при использовании сервисов передачи данных;

π_2 – доля пользователей группы 2 в общей структуре абонентов;

h – размер поля данных пакета;

N – общее число пользователей.

Суммарное число пакетов, генерируемых второй группой пользователей в сеть в час наибольшей нагрузке, будет равно

$$N_2 = N_{2\_т} + N_{2\_д} \quad (5.6)$$

5.1.3 Расчёт числа пакетов от третьей группы абонентов (triple play)

Все рассуждения, проведённые относительно первых двух групп, остаются в силе и для третьей группы, применительно к сервисам передачи голоса, а именно:

$$N_{3\_т} = n_1 \cdot t_{3\_т} \cdot f_3 \cdot \pi_3 \cdot N \quad (5.7)$$

где $N_{3\_т}$ – число пакетов, генерируемое третьей группой пользователей в час наибольшей нагрузки при использовании голосовых сервисов;

n_1 – число пакетов, генерируемых в секунду одним абонентом;

t_3 – средняя длительность разговора в секундах;

f_3 – число вызовов в час наибольшей нагрузки;

π_3 – доля пользователей группы 3 в общей структуре абонентов;

N – общее число пользователей.

Предположим, что абоненты 3 группы относятся к очень «активным» пользователям интернета, т.е., используют не только http, но и ftp, а также прибегают к услугам пиринговых сетей. Объём принятых и переданных данных данных при таком использовании интернета составляет до V_3 = 70 Мбайт = $560 \cdot 10^6$ бит.

Число пакетов, переданных в ЧНН, будет равно

$$N_{3\_д} = \pi_3 \cdot N \cdot V_3/h \quad (5.8)$$

Что бы расчитать число пакетов, генерируемых пользователями видео-услуг, воспользуемся соображениями относительно размера пакета, приведёнными в предыдущем пункте. Размер пакета не должен превосходить 200 (120) байт (вместе с накладными расходами).

Одной из наиболее перспективных и динамически развивающихся услуг в АО «Казактелеком» является IPTV – передача каналов телевещания с помощью протокола IP. При организации данного сервиса для каждого пользователя в транзитной сети доступа не требуется выделения индивидуальной полосы пропускания. До мультисервисного узла доходит определённое количество каналов, которые провайдеры распределяют между заказчиками услуги, причём существует возможность организации широковещательной рассылки. Допустим, что в мультисервисной сети

предоставляется возможность просмотра K_tv = 200 каналов вещания. Для обеспечения удовлетворительного качества скорость кодирования должна быть порядка 2 Мбит/с.

Итак, при скорости передачи v = 2048000 бит/с и размере полезной нагрузки пакета h_1 = 160 байт = 1280 бит (G.711) и h_2 = 80 байт = 640 бит (G.726) число пакетов, возникающих при трансляции одного канала, равно:

$$n_3 = v/h \qquad (5.9)$$

для G.711:

$$n_{31} = 2048000/1280 = 1600 \text{ пакетов/с}$$

для G.726:

$$n_{32} = 2048000/640 = 3200 \text{ пакетов/с}$$

Количество пакетов, передаваемых по каналами в ЧНН, составит

$$N_{3\_B} = \pi_3 \cdot N \cdot n_3 \cdot t_{3\_B} \qquad (5.10)$$

где N_{3_B} – число пакетов, генерируемое третьей группой пользователей в час наибольшей нагрузки при использовании видео-сервисов сервисов;

n_3 – число пакетов, генерируемых в секунду одним абонентом при использовании просмотре видео, сжатого по стандарту MPEG2;

t_{3_B} – среднее время просмотра каналов в ЧНН, сек;

π_3 – доля пользователей группы 3 в общей структуре абонентов;

N – общее число пользователей.

Суммарное число пакетов, генерируемых третьей группой пользователей в сеть в час наибольшей нагрузке, будет равно

$$N_3 = N_{3\_т} + N_{3\_д} + N_{3\_B} \qquad (5.11)$$

5.1.4 Требования к производительности мультисервисного узла доступа

Мультисервисный узел доступа обязан обслуживать трафик от всех трёх групп потребителей. Именно узел доступа должен обеспечить поддержку качества обслуживания путем приоритезации трафика, которая должна осуществляться независимо от используемой технологии транспортной сети доступа.

Суммарное число пакетов, которое должен обработать мультисервисный узел доступа, будет равно:

$$N_\Sigma = N_1 + N_2 + N_3 = n_1 \cdot t_1 \cdot f_1 \cdot \pi_1 \cdot N + (n_1 \cdot t_2 \cdot f_2 \cdot \pi_2 \cdot N + \pi_2 \cdot N \cdot V_2/h) + \\ + (n_1 \cdot t_3 \cdot f3 \cdot \pi_3 \cdot N + \pi_3 \cdot N \cdot V_3/h + \pi_3 \cdot N \cdot n_3 \cdot t_{3\_B}) \qquad (5.12)$$

Учитывая, что:

$t_1 = t_2 = t_3 = t$ – средняя длительность разговора в секундах;

$f_3 = f_2 = f_1 = f$ – число вызовов в ЧНН;
получим

$$N_\Sigma = n_1 \cdot t \cdot f \cdot N \cdot (\pi_1 + \pi_2 + \pi_3) + N/h \cdot (\pi_2 \cdot V_2 + \pi_3 \cdot V_3) + \pi_3 \cdot N \cdot n_3 \cdot t_{3\_B} \quad (5.13)$$

Учитывая, что $\pi_1 + \pi_2 + \pi_3 = 1$, получим

$$N_\Sigma = N \cdot (n_1 \cdot t \cdot f + (\pi_2 \cdot V_2 + \pi_3 \cdot V_3)/h) + \pi_3 \cdot N \cdot n_3 \cdot t_{3\_B} \quad (5.14)$$

При N = 3500 абонентов, n_1 = 50 пакетов в секунду, t =120 секунд, f = 5 вызовов в час, V_2 = 10 Мбайт, V_3 = 70 Мбайт, t_{3_B} = 50 минут, n_{31} = 1600, n_{32} = 3200, π_1 = 60 %, π_2 = 35 %, π_3 = 5 % получим:
для G.711:

$$N_\Sigma = 3500 \cdot (50 \cdot 120 \cdot 5 + (0{,}35 \cdot 10 \cdot 1024 \cdot 1024 + 0{,}05 \cdot 70 \cdot 1024 \cdot 1024)/160) + 0{,}05 \cdot 3500 \cdot 1600 \cdot 3000 = 1{,}1055632 \cdot 10^9, \text{ пакетов/час}$$

Среднее число пакетов в секунду равно

$$N_{\Sigma\_сек} = N_\Sigma/3600, \quad (5.15)$$

$$N_{\Sigma\_сек} = 307101, \text{ пакетов/с.}$$

для G.726:

$$N_\Sigma = 3500 \cdot (50 \cdot 120 \cdot 5 + (0{,}35 \cdot 10 \cdot 1024 \cdot 1024 + 0{,}05 \cdot 7 \cdot 1024 \cdot 1024)/80) + 0{,}05 \cdot 3500 \cdot 3200 \cdot 3000 = 2{,}106126 \cdot 10^9, \text{ пакетов/час}$$

Среднее число пакетов в секунду равно

$$N_{\Sigma\_сек} = N_\Sigma/3600,$$

$$N_{\Sigma\_сек} = 585035, \text{ пакетов/с.}$$

Данные показатели позволяют оценить требования к производительности маршрутизатора, агрегирующего трафик мультисервисной сети доступа NGN. Анализ Приложения А показывает, что выбор такого маршрутизатора осуществляется из весьма ограниченного количества вариантов.

Рассмотрим как и какие группы сети больше всего загружают систему для рассчитываемых длин пакетов (таблица 5.2, рисунок 5.3)

Т а б л и ц а 5 . 2 - количество передаваемых пакетов в сек для трех групп пользователей

	количество передаваемых пакетов в сек	
	G.711	G.726
1 группа (60 %)	17500	17500
2 группа (35 %)	32509	54809
3 группа (5 %)	257092	512726

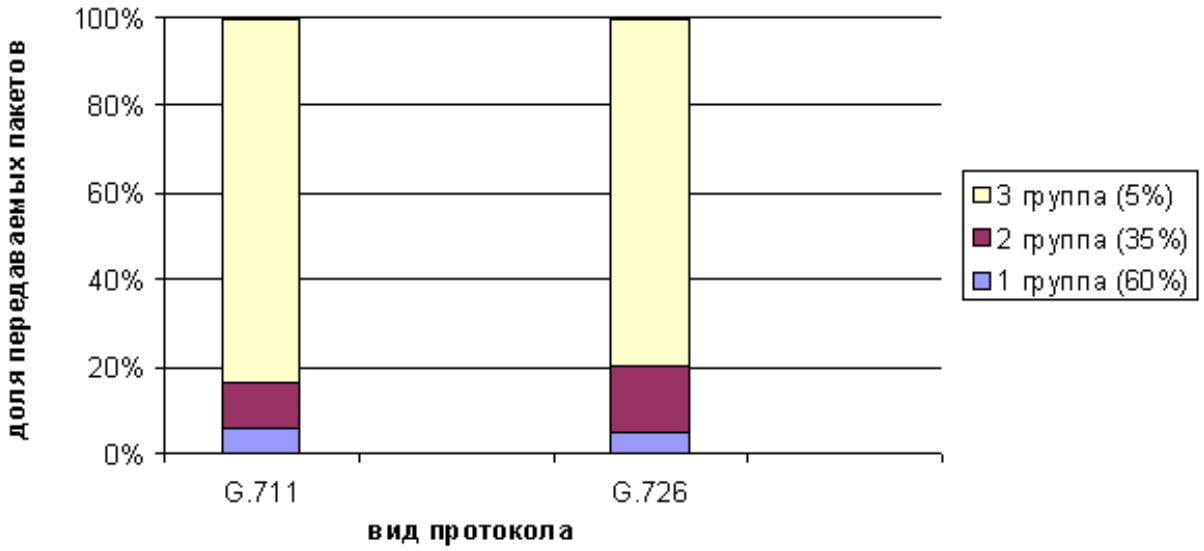


Рисунок 5.3 – Доля передаваемых пакетов тремя группами

Из графика видно, что наибольший передаваемый трафик идет на третью группу при кодеке G.711 и G.726, несмотря на то, что она составляет всего лишь 5 % от общего числа пользователей. Пользователи обычной телефонии, при ее преобладающем количестве, загружают систему меньше всех пользователей.

5.2 Требования к полосе пропускания

Требования к полосе пропускания определяются стандартизированными гарантиями качества обслуживания, предоставляемыми оператором пользователю. Параметры QoS описаны в рекомендации ITU Y.1541. В частности, задержка распространения из конца в конец при передачи речи категорически не должна превышать 100 мс, а вероятность превышения задержки порога в 50 мс не должна превосходить 0,001, т.е.

$$\bar{t}_{\partial} \le 100, \text{ мс}$$

$$p\{t_p > 50 \text{ мс}\} \le 0.001$$

Задержка из конца в конец складывается из следующих составляющих:

$$t_p = t_{пакет} + t_{ад} + t_{core} + t_{буф} \qquad (5.16)$$

где t_p – время передачи пакета из конца в конец;
$t_{пакет}$ – время пакетизации (зависит от типа трафика и кодека);
$t_{ад}$ – время задержки при транспортировке в сети доступа;
t_{core} – время задержки при распространении в транзитной сети;
$t_{буф}$ – время задержки в приёмном буфере.

Из таблицы 5.1 мы видим, что применение низкоскоростных кодеков «съедает» основную часть бюджета задержки. Задержка в буфере приёма также

велика, поэтому на сеть доступа и транспортная сеть должны обеспечивать минимальную задержку связи.

Допустим, что задержка на сети доступа не должна превышать 5 мс. Время обработки заголовков IP-пакета близко к постоянному. Распределение интервалов между поступлениями пакетов соответствует экспоненциальному закону. Поэтому для описания процесса, происходящего на агрегирующем маршрутизаторе, можно воспользоваться моделью M/G/1.

Для данной модели известна формула, определяющая среднее время вызова в системе (формула Полячека – Хинчина) [19].

$$\bar{t}_{ад} = \frac{\tau(1+C_b^2)}{2(1-\lambda\tau)} \qquad (5.17)$$

где τ – средняя длительность обслуживания одного пакета;

C_b^2 – квадрат коэффициента вариации, $C_b^2 \approx 0{,}2$;

λ – параметр потока, λ_1= 307101 (при G.711), λ_2= 585035 (при G.726);

$\bar{t}_{ад}$ – среднее время задержки пакета в сети доступа, $\bar{t}$ = 0,005 с.

Ненулевой коэффициент вариации учитывает возможные отклонения при использовании в заголовках IP полей ToS. Кроме того, время обработки IP-пакета в значительной мере зависит от используемых на маршрутизаторе правил обработки.

Из формулы (5.17) следует зависимость максимальной величины для средней длительности обслуживания одного пакета от среднего времени задержки в сети доступа.

$$\tau = \frac{1}{\lambda + \frac{1+C_b^2}{2\bar{t}_{ад}}} \qquad (5.18)$$

Построим данные зависимости при помощи прикладной программы MathCad. Полученные графики представлены на рисунках 5.5 и 5.6.

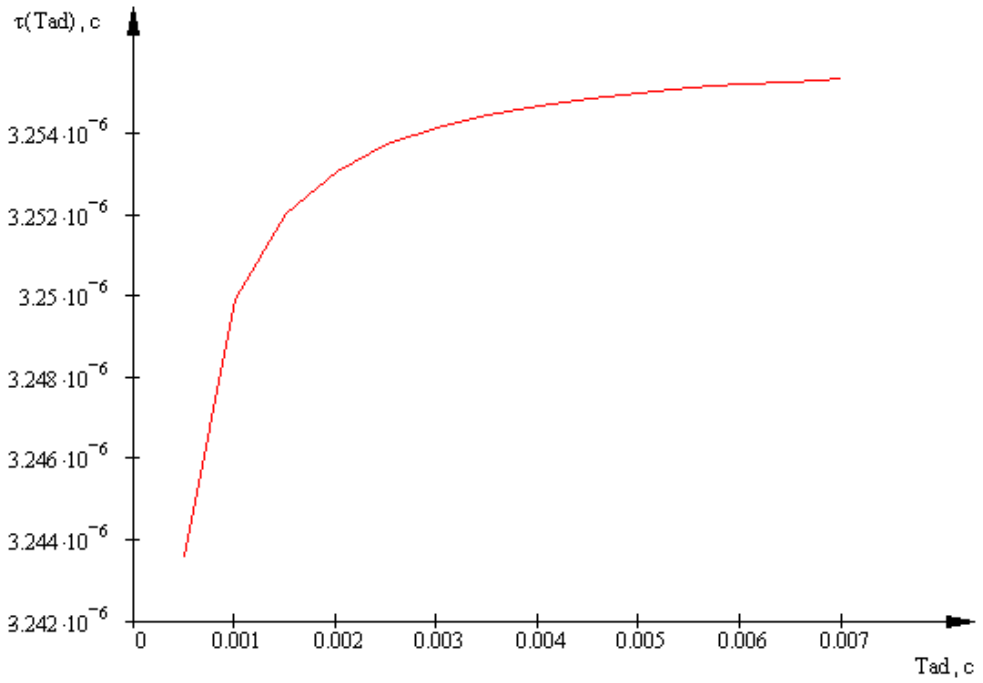


Рисунок 5.5 - Зависимость максимальной величины для средней длительности обслуживания одного пакета от среднего времени задержки в сети доступа для кодека G.711

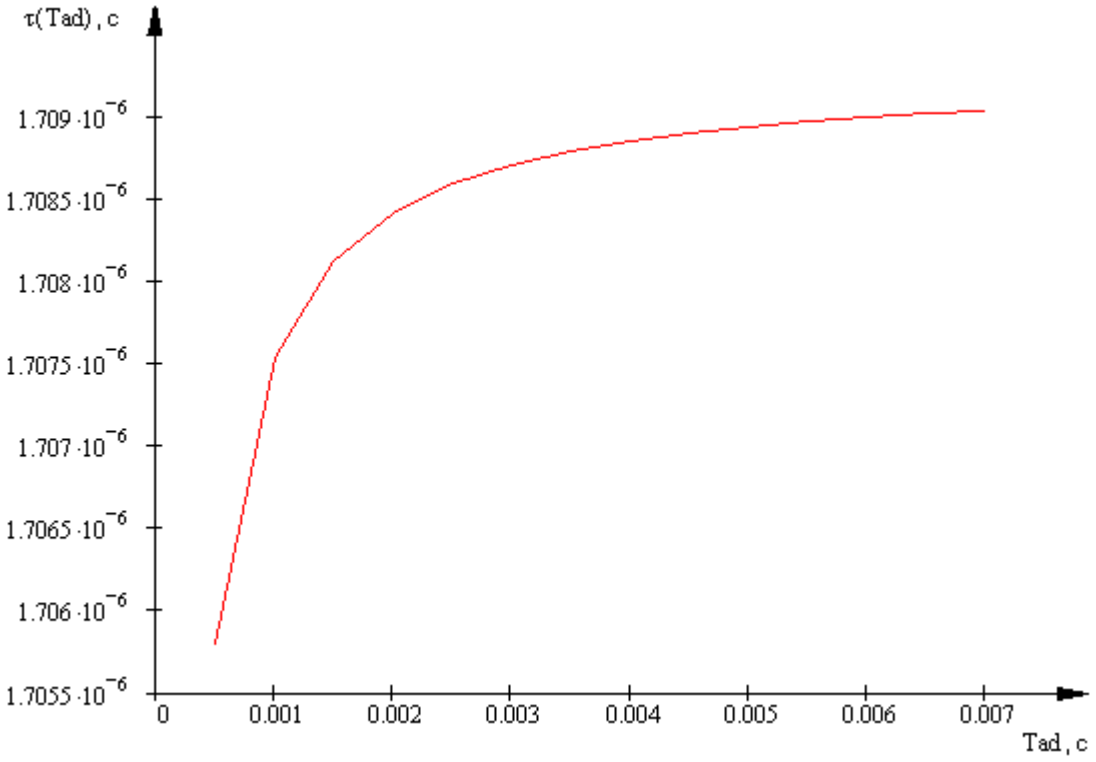


Рисунок 5.6 - Зависимость максимальной величины для средней длительности обслуживания одного пакета от среднего времени задержки в сети доступа для кодека G.726

По графикам 5.5 и 5.6 можно сказать, что время обслуживания одного пакета увеличивается с увеличением времени задержки сети доступа.

Интенсивность обслуживания связана со средним временем задержки пакета в сети доступа обратно пропорционально:

$$\beta = \frac{1}{\tau} \tag{5.19}$$

Графически данные зависимости представлены на рисунках 5.7 и 5.8.

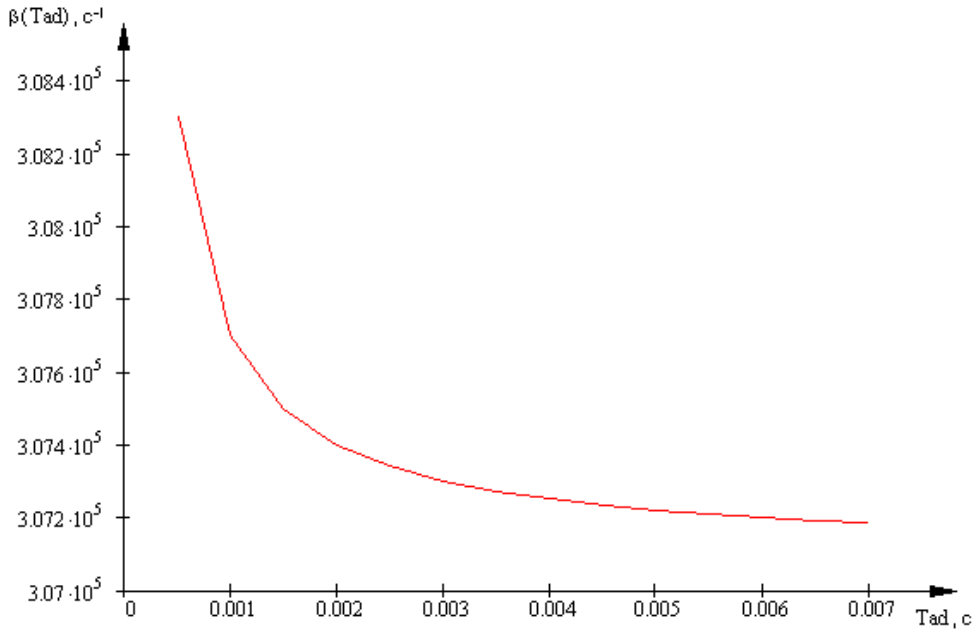


Рисунок 5.7 - Зависимость интенсивности обслуживания от времени задержки в сети доступа для кодека G.711

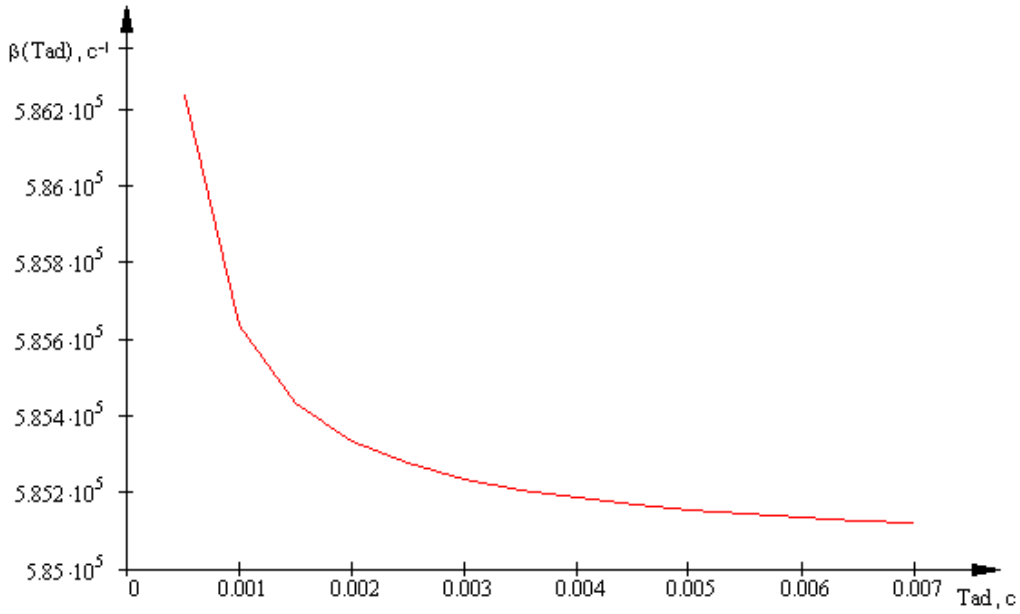
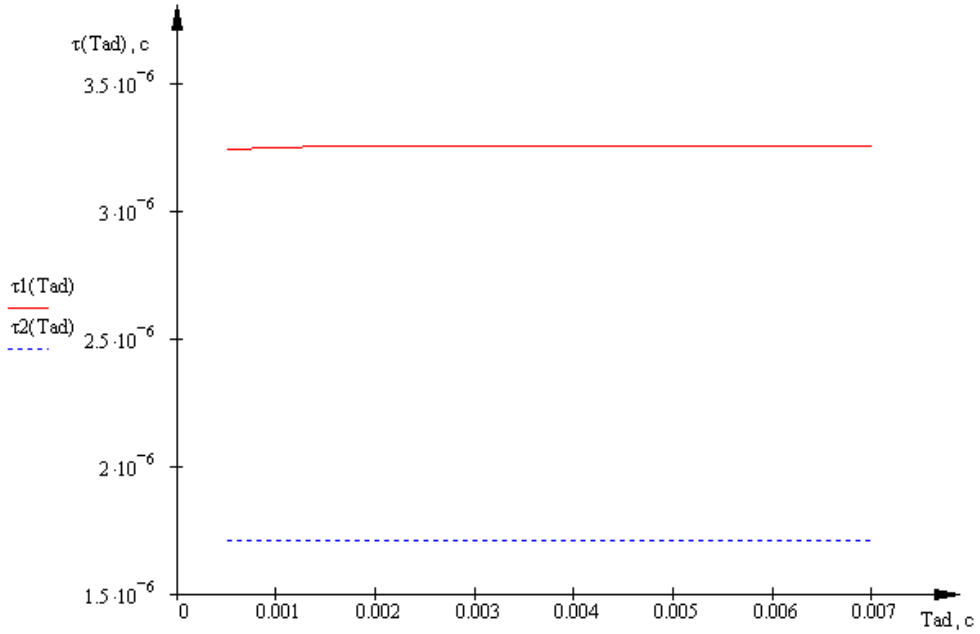


Рисунок 5.8 - Зависимость интенсивности обслуживания от времени задержки в сети доступа для кодека G.726

По графикам 5.7 и 5.8 можно сказать, что интенсивность обслуживания пакетов уменьшается с увеличением времени задержки сети доступа.

Для наглядности можно построить следующие графики (рисунок 5.9 и рисунок 5.10)



τ1(Tad) - Зависимость длительности обслуживания одного пакета от среднего времени задержки в сети доступа для кодека G.711

τ2(Tad) - Зависимость длительности обслуживания одного пакета от среднего времени задержки в сети доступа для кодека G.726

Рисунок 5.9 - Зависимость длительности обслуживания одного пакета от среднего времени задержки в сети доступа

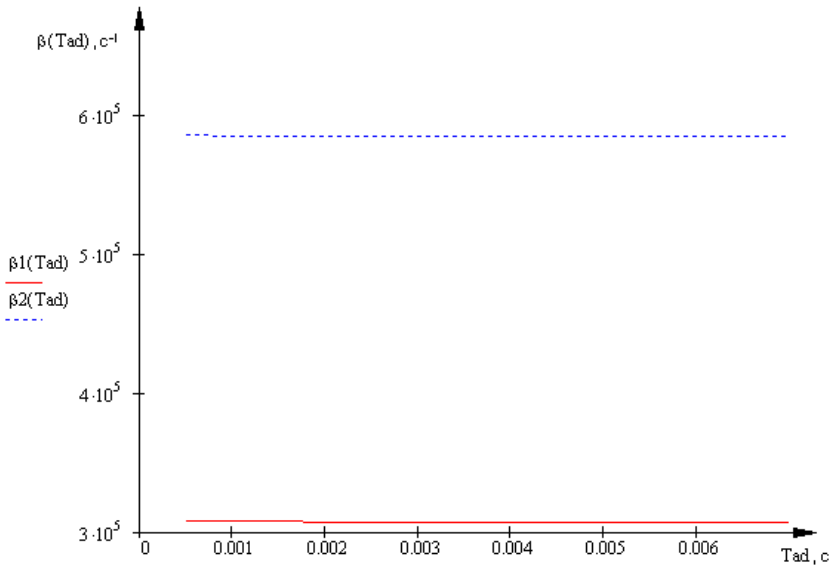


Рисунок 5.10 - Зависимости интенсивности обслуживания от времени задержки в сети доступа

Сделав анализ полученных графиков 5.9 и 5.10 можно сделать следующие выводы:

– чем больше длина пакета, тем больше требуется время для его обслуживания.

– чем меньше длина пакета, тем интенсивность их обслуживания больше.

При норме задержки $\bar{t}_{\text{ад}} = 5$ мс среднее время обслуживания пакета (для рассчитанных выше пропускных способностей) будет равно

для G.711:

$$\tau(0.005) = \frac{1}{307101 + \frac{1+0.2}{2 \cdot 0.005}} = 3.255 \times 10^{-6}, \text{ с}$$

$$\beta = \frac{1}{\tau} = \frac{1}{3.255 \cdot 10^{-6}} = 3.072 \cdot 10^{5}$$

для G.726:

$$\tau(0.005) = \frac{1}{585035 + \frac{1+0.2}{2 \cdot 0.005}} = 1.709 \times 10^{-6}, \text{ с}$$

$$\beta = \frac{1}{\tau} = \frac{1}{1.709 \cdot 10^{-6}} = 5.852 \cdot 10^{5}$$

Время τ должно выбираться как минимальное из двух возможных значений. Первое значение – величина, полученная из последней формулы. Второе значение – та величина, которая определяется из условия ограничения загрузки системы – ρ. Обычно эта величина не должна превышать 0,5.

При среднем значении задержки в сети доступа 5 мс коэффициент использования равен:

$$\rho = \lambda \cdot \tau(0.005) \qquad (5.20)$$

для G.711:

$$\rho = 307101 \cdot 3.255 \times 10^{-6} = 0.9996094$$

для G.726:

$$\rho = 585035 \cdot 1.709 \times 10^{-6} = 0.9997949$$

При таком высоком использовании малейшие флуктуации параметров могут привести к нестабильной работе системы. Определим параметры системы при её использовании на 50 %. Средняя длительность обслуживания будет равна

$$\tau = \frac{\rho}{\lambda} \qquad (5.21)$$

для G.711:

$$\tau = \frac{0.5}{307101} = 1.628 \cdot 10^{-6} \text{ , с}$$

Интенсивность обслуживания при этом

$$\beta = \frac{1}{\tau} = \frac{1}{1.628 \cdot 10^{-6}} = 6.142 \cdot 10^{5} \qquad (5.22)$$

А задержка в сети доступа

$$\bar{t}_{ad} = \frac{\tau(1 + C_b^2)}{2(1 - \lambda\tau)} = \frac{1.628 \cdot 10^{-6} \cdot (1 + 0.2)}{2(1 - 307101 \cdot 1.628 \cdot 10^{-6})} = 1.954 \cdot 10^{-6} \text{ , с}$$

для G.726:

$$\tau = \frac{0.5}{585035} = 0.85 \cdot 10^{-6} \text{ с}$$

Интенсивность обслуживания при этом

$$\beta = \frac{1}{\tau} = \frac{1}{0.85 \cdot 10^{-6}} = 11.7 \cdot 10^{5}$$

А задержка в сети доступа

$$\bar{t}_{ad} = \frac{\tau(1 + C_b^2)}{2(1 - \lambda\tau)} = \frac{0.85 \cdot 10^{-6} \cdot (1 + 0.2)}{2(1 - 585035 \cdot 0.85 \cdot 10^{-6})} = 1.026 \cdot 10^{-6} \text{ , с}$$

Рассчитывать вероятность $s(t)=1-e^{-(\frac{1}{\tau}-\lambda)t}$ при известных λ и τ нецелесообразно, т.к. в Y.1541 вероятность P{t>50 мс} < 0.001 определена для передачи из конца в конец.

При известном среднем размере пакета h получаем требуемую полосу пропускания

для G.711:

$$\varphi = \beta \cdot h = 6.142 \cdot 10^{5} \cdot 200 = 1.228 \cdot 10^{8} = 9.827 \cdot 10^{8} \text{ бит/с} \qquad (5.23)$$

для G.726:

$$\varphi = \beta \cdot h = 11.7 \cdot 10^{5} \cdot 120 = 1.404 \cdot 10^{8} = 1.12 \cdot 10^{9} \text{ бит/с}$$

Сравним полученные результаты (рисунок 5.11)

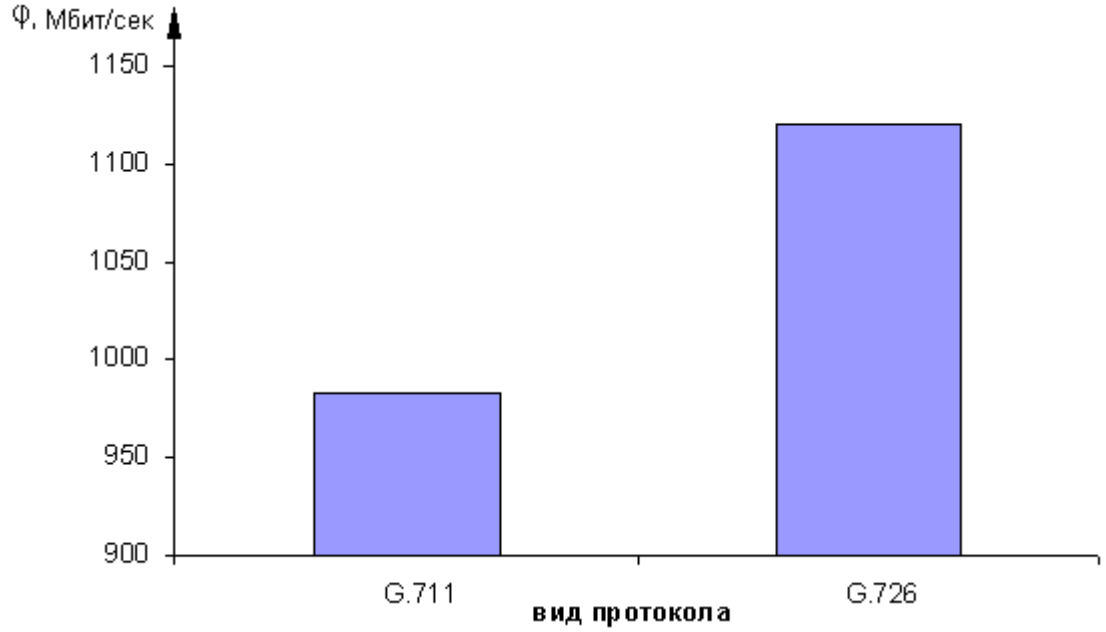


Рисунок 5.11 – Требуемая полоса пропускания

Из графика мы видим что для передачи одной и той же информации, то есть одного объема при использовании услуги Triple Play, необходимо различная полоса пропускания, в нашем случае при использовании кодека G.711 с длиной пакета 200 байт необходимо меньшая полоса пропускания, чем при использовании кодека G.726 с длиной пакета 120 байт, хотя протокол G.726 – есть протокол со сжатием.

Предположим, что в структурном составе абонентов отсутствуют группы пользователей использующие видео услуги, т.е. $\pi_2 \approx 40$ %. При этом в вышеприведённом анализе следует опустить расчёт числа пакетов, возникающих при использовании сервисов высокоскоростной передачи данных и видеоуслуг.

Число генерирующих пакетов, возникающих в ЧНН, будет равно

$$N_{\Sigma} = Ntel + Nint = N \cdot (n \cdot t \cdot f + \frac{\pi_2 \cdot V_2}{h}) \qquad (5.24)$$

где N_{tel} – число пакетов телефонии, генерируемое всеми пользователями в час наибольшей нагрузки;

N_{int} – число пакетов интернета, генерируемое второй группой пользователей в час наибольшей нагрузки

π_2 – доля пользователей группы 2 в общей структуре абонентов

n – число пакетов, генерируемых в секунду одним абонентом при использовании кодека G.711;

t – средняя длительность разговора в секундах;

f – число вызовов в час наибольшей нагрузки;

N_Σ – общее число пользователей.

Число пакетов в секунду:
Для G.711:

$$\frac{N}{3600} = N\cdot(n\cdot t\cdot f + \frac{\pi_2 \cdot V_2}{h}) / 3600 = 3500\cdot(50\cdot 120\cdot \cdot 5 + \frac{0{,}4\cdot 10\cdot 1024\cdot 1024}{160}) / 3600 = 54653 \quad (5.25)$$

Среднее время обслуживания одного пакета при норме задержки 5 мс:

$$\tau(0.005) = \frac{1}{54653 + \frac{1+0.2}{2\cdot 0.005}} = 1.826\times 10^{-5}\ , \text{с}$$

Коэффициент использования:

$$\rho = \lambda \cdot \tau(0.005),$$

$$\rho = 54653\cdot 1.826\times 10^{-5} = 0.997809$$

При использовании системы на 50 %:

$$\tau = \frac{0.5}{54653} = 9.149\cdot 10^{-6}\ , \text{с}$$

$$\beta = \frac{1}{\tau} = \frac{1}{9.149\cdot 10^{-6}} = 1.093\cdot 10^{5}$$

Требуемая пропускная способность:

$$\varphi = \beta\cdot h = 1{,}093\cdot 10^5\cdot 200 = 2{,}186\cdot 10^7 = 1{,}74889\cdot 10^8 \text{ бит/с}$$

Для G.726:

$$\frac{N}{3600} = N\cdot(n\cdot t\cdot f + \frac{\pi_2 \cdot V_2}{h}) / 3600 = 3500\cdot(50\cdot 120\cdot \cdot 5 + \frac{0{,}4\cdot 10\cdot 1024\cdot 1024}{80}) / 3600 = 80139$$

Среднее время обслуживания одного пакета при норме задержки 5 мс:

$$\tau(0.005) = \frac{1}{80139 + \frac{1+0.2}{2\cdot 0.005}} = 1.246\times 10^{-5}\ , \text{с}$$

Коэффициент использования:

$$\rho = \lambda \cdot \tau(0.005)$$

$$\rho = 80139 \cdot 1.246 \times 10^{-5} = 0.99850$$

При использовании системы на 50 %:

$$\tau = \frac{0.5}{80139} = 6.239 \cdot 10^{-6} \text{ , с}$$

$$\beta = \frac{1}{\tau} = \frac{1}{6.239 \cdot 10^{-6}} = 1.603 \cdot 10^{5}$$

Требуемая пропускная способность:

$$\varphi = \beta \cdot h = 1{,}603 \cdot 10^{5} \cdot 120 = 1{,}923 \cdot 10^{7} = 1{,}53867 \cdot 10^{8} \text{ бит/с}$$

Сравним полученные результаты (рисунок 5.12)

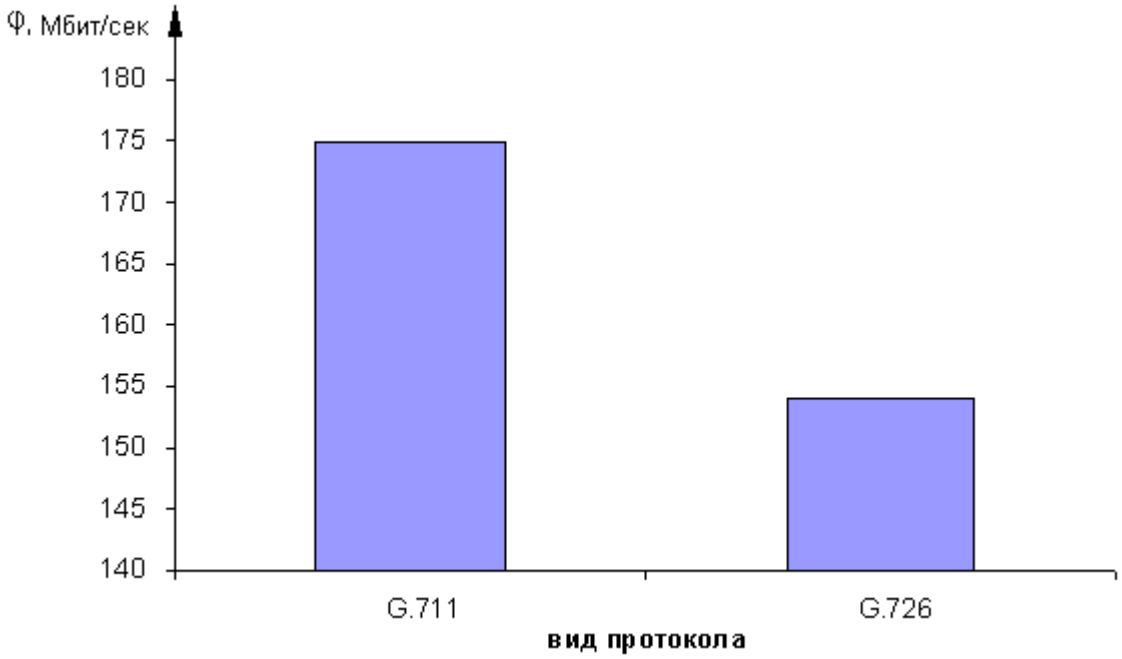


Рисунок 5.12 – Требуемая полоса пропускания

Из графика мы видим, что для передачи информации одного и того же объема, необходима различная полоса пропускания, в данном случае при использовании кодека G.711 с длиной пакета 200 байт необходима большая полоса пропускания, чем при использовании кодека G.726 с длиной пакета 120 байт.

Заключение

В современном мире большое внимание уделяется качеству передачи данных в сетях сервис провайдеров. Решение проблем перегрузки и выбора оптимальных параметров настроек алгоритмов по предотвращению перегрузок является ключевым фактором в данной работе.

Для выполнения задач, поставленных в ходе диссертационной работы, были произведены исследования на базе оборудований Cisco Systems и получены следующие основные результаты:

– подробно исследована работа алгоритма предотвращения перегрузок WRED и поведение механизма в зависимости от настраиваемых параметров ;

– произведён анализ результатов исследования;

– предложены оптимальные параметры настроек алгоритма WRED для повышения качества обслуживания.

Показано, что для предотвращения перегрузок в сети, оптимальными параметрами настройки алгоритма WRED являются следующие: максимальное значение вероятности отбрасывания пакетов равно 1/10, весовой коэффициент усреднения равен 1/16 и верхний порог средневзвешенной длины очереди равен 45.

Экспериментальные результаты подтверждены математическим моделированием, в ходе которого определены требования к производительности маршрутизатора, агрегирующего трафик сети доступа.

Проведённое исследование и полученные результаты несомненно имеют огромную практическую значимость. Поскольку, данный алгоритм предотвращения перегрузок используется большинством сервис провайдеров, а найденные нами параметры подходят для любого из них. При использовании полученных параметров настройки алгоритма WRED можно существенно изменить поведение перегрузки на уровне маршрутизатора. Верно настроенный механизм предотвращения перегрузок может полностью ликвидировать перегрузку в сетях с постоянной интенсивностью трафика.

Список литературы

1 Концептуальные положения по построению мультисервисных сетей // Электронная версия на сайте www.minsvyaz.ru/img

2 Гончаров А.А., Ильин А.Ю., Семенов Ю.А., Исследование возможностей получения гарантированного качества обслуживания при передаче мультимедиа через перегруженные каналы – Информационные процессы, 2006.

3 Гойхман В.Ю., Васильев А.С. Диверсификация городских АТС – «Технологии и средства связи», спецвыпуск АТС 2004.

4 Соколов Н.А.. Семь аспектов развития сети доступа – «Технологии и средства связи», №3, 2005.

5 Соколов Н.А. Сети абонентского доступа. Принципы построения – Научно-техническое издание, 1999.

6 Шринивас Вегешна. Качество обслуживания в сетях – Москва, 2003.

7 Соколов Н. А. Качество обслуживания трафика речи в сети NGN. // Connect! Мир связи, №7, 2006.

8 Гончаров А.А., Семенов Ю.А., Анализ влияния параметров канала и алгоритма подавления перегрузок на ухудшение качества передачи видеоизображений // «Информационно-измерительные и управляющие системы» № 3-4, 2007 .

9 Sally Floyd and Van Jacobson, Random early detection gateway for congestion avoidance, IEEE/ACM Transactions on Networking, vol.1, pp. 397-413, August 1993.

10 Richard Froom, Mike Flannagan, Kevin Turek, Quality of Service in Campus Networks – Cisco Press, August 2003.

11 Santiago Alvarez, QoS for IP Networks – Cisco Press, November 2001.

12 Соколов Н. А. Качество обслуживания трафика речи в сети NGN. // Connect! Мир связи, №7, 2006.

13 Tim Szigeti, Christina Hattingh, End-to-End QoS Network Design – Cisco Press, November 2004.

14 Гончаров А.А., Ильин А.Ю., Семенов Ю.А., Исследование возможностей получения гарантированного качества обслуживания – Информационные процессы, 2006.

15 Сети VoIP - голос в пакетах // Электронная версия на сайте http://www.teleincom.ru/newtech/voip/

16 Система оценки качеста VoIP // Электронная версия на сайте http://www.teleincom.ru/newtech/omni/

17 Требования к системам для модернизации сети связи общего пользования // Электронная версия на сайте http://www.citforum.ru/nets/articles/softswitch/

18 Клейнрок. Л. Теория массового обслуживания. – Москва, 1979.

19 Устройства управления мультисервисными сетями: Softswitch // Электронная версия на сайте http://www.niits.ru/public/2002/200204.pdf

20 Разработка методов оценки параметров трафика мультисервисной сети // Электронная версия на сайте http://www.uran.donetsk.ua/~masters/2004/kita/schitnikova/diss/index.htm

21 Качество голоса в мультисервисных сетях // Электронная версия на сайте http://www.compress.ru/Archive/CP/2001/5/52/

22 Как обеспечить QoS в телефонных сетях с коммутацией пакетов // Электронная версия на сайте

23 Tim Szigeti, Christina Hattingh, End-to-End QoS Network Design – Cisco Press, November 2004

24 Васильев А.Б., Пятаев В.О. Принципы построения мультисервисных сетей – ИнформКурьер-Связь, №8, 2002.

Приложение А

Данные о производительности маршрутизаторов CISCO

Модель	Process Switching (использование классической маршрутизации)		Fast/CEF switching (использование экспресс-маршрутизации)	
	пакеты/с	Мбит/с	пакеты/с	Мбит/с
14xx	600	0,3072	4000	2,05
160x (-R)	600	0,3072	4000	2,05
1701	1700	0,8704	12000	6,14
1710	1300	0,6656	7000	3,58
1711/1712	1700	0,8704	13500	6,91
1720	1400	0,7168	8500	4,35
1721	1700	0,8704	12000	6,14
1750	1400	0,7168	8500	4,35
1751	1500	0,768	12000	6,14
1760	1700	0,8704	16000	8,19
1841	-	-	75000	38,40
2500	800	0,4096	4400	2,25
261x	1500	0,768	15000	7,68
262x	800	0,768	25000	12,80
265x	2000	1,024	37000	18,94
261xXM	1500	0,768	20000	10,24
262xXM	1500	0,769	30000	15,36
265xXM	2000	1,024	40000	20,48
2691	7400	3,788	70000	35,84
2801	-	-	90000	46,08
2811	-	-	120000	61,44
2821	-	-	170000	87,04
2851	-	-	220000	112,64
3620	2000	1,024	20000-40000	10-20
3640/3640A	4000	2,048	50000-70000	25,6-36
3660	12000	6,144	100-120000	51,2-61,4
3631	4000	2,048	50-70000	115,2-128
3725	-	-	100-120000	51,2-61,4
3745	-	-	225-250000	115,2-128
MC3810	2000	1,024	8000	4,10
MC3810-V3	3000	1,536	15000	7,68
3825	-	-	350000	179,20
3845	-	-	500000	256,00
IAD2400	3000	1,536	15000	7,68
4000	1800	0,9216	14000	7,17
4500	3500	1,792	45000	23,04
4700	4600	2,3552	75000	38,40
7120	13000	6,656	175000	89,60
7140	20000	10,24	300000	153,60
7200-NPE100	7000	3,584	100000	51,10
7200-NPE150	10000	5,12	150000	76,80

Продолжение приложения А

Модель	Process Switching (использование классической маршрутизации)		Fast/CEF switching (использование экспресс-маршрутизации)	
	пакеты/с	Мбит/с	пакеты/с	Мбит/с
7200-NPE175	9000	4,608	177848	91,06
7200-NPE200	13000	6,656	200000	102,40
7200-NPE225	13000	6,656	233170	119,38
7200-NPE300	20000	10,24	353000	180,74
7200-NPE400	20000	10,24	420000	215,04
7200-NPE-G1	79000	40,448	1018000	521,22
7200-NSE-1	20000	10,24	300000	153,60
7200-NSE-100	-	-	450000	230,4
7200-NPE-G100	-	-	1099000	562,69
7301	79000	40,448	1018000	521,22
7401	20000	10,24	300000	153,6
7000-RP	2500	1,28	30000	15,36
7500-RSP2	5000	2,56	220000	112,64
7500-RSP4/4+	8000	4,096	345000	176,64
7500-RSP8	22000	11,264	470000	240,64
7500-RSP16	29000	14,848	530000	271,36

Приложение Б

Листинг конфигурации маршрутизаторов экспериментальной модели в ОС Cisco IOS

```
Current configuration : 1342 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router_R1
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$o4jt$fAzXKavU6yWo2YgIeOHDd.
!
no aaa new-model
!
resource policy
!
ip subnet-zero
!
ip cef
!
interface Tunnel0
 ip address 192.168.1.50 255.255.255.240
 tunnel source Serial0/2/0
 tunnel destination 192.168.1.18
!
interface FastEthernet0/0
 ip address 192.168.1.1 255.255.255.240
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial0/2/0
```

```
 bandwidth 2000
 ip address 192.168.1.17 255.255.255.240
 ip nbar protocol-discovery
 random-detect
 random-detect precedence 0 20 40 10
 random-detect precedence 1 22 40 10
 random-detect precedence 2 24 40 10
 random-detect precedence 3 26 40 10
 random-detect precedence 4 28 40 10
 random-detect precedence 5 31 40 10
 random-detect precedence 6 33 40 10
 random-detect precedence 7 35 40 10
 random-detect precedence rsvp 37 40 10
!
router eigrp 100
 network 192.168.1.0
 no auto-summary
!
ip classless
!
ip http server
!
control-plane
!
!
line con 0
 exec-timeout 0 0
 logging synchronous
line aux 0
line vty 0 4
 exec-timeout 0 0
 password cisco
 logging synchronous
 login
!
scheduler allocate 20000 1000
!
end
```