С О Д Е Р Ж А Н И Е

ВВЕДЕНИЕ

Настоящее учебное пособие строится на материале, который читается для студентов направлений обучения 162500, 25.03.02 “Техническая эксплуатация авиационных электросистем и пилотажно-навигационных комплексов» (бакалавриат) дневного обучения по дисциплине «Компьютерные сети и интернет-технологии».

За короткий период времени компьютеры по всему миру заняли огромное значение в жизни современного человека. Сегодня большинство компьютеров в мире объединены в сети. Это могут быть сети самые разные, начиная от локальных сетей в офисах, и заканчивая глобальными сетями. Уже сейчас компьютерные сети представляют собой одно из основных средств коммуникации всего человечества. Необходимость объединения компьютеров в сети вызвана целым рядом причин, среди которых усовершенствование путей передачи сообщений, возможность обмена информацией в быстром режиме, иначе говоря, это мгновенные сообщения, а также передача и получение информации, даже не отходя от своего рабочего места.

Учебное пособие состоит из двух частей, в которых рассмотрены принципы организации компьютерных сетей и классификация интернет-технологий.

Первая часть учебного пособия содержит пять разделов, в которых рассматриваются принципы организации компьютерных сетей, приводятся общие сведения о сетевых устройствах, дается понятие «эталонная модель взаимодействия открытых систем OSI/ISO», также излагается физическое, процедурное, логическое сопряжение, функции и задачи уровней данной модели. В данной части также рассматриваются виды и алгоритмы маршрутизации, их достоинства и недостатки, приводится обзор протоколов маршрутизации, их основные характеристики.

Во второй части данного учебного пособия рассмотрена логика работы сетевых протоколов и система адресации в IP-сетях, описаны методы перехода от IPv4 к IPv6, приведена современная классификация интернет/Интернет-технологий, рассматриваются транспортные технологии пакетной коммутации и особенности передачи речевой информации.

Для контроля качества усвоения материала каждый раздел завершается контрольными вопросами, в конце учебного пособия приведен итоговый тест по рассмотренным темам.

Пособие рассчитано на студентов направлений обучения 162500, 25.03.02 (бакалавриат) и слушателей высших учебных заведений, обучающихся по техническим дисциплинам. Может быть использовано как при выполнении выпускной квалификационной работы, так и для изучения вопросов, связанных с использованием компьютерных сетей в своей практической деятельности.

ЧАСТЬ 1. КОМПЬЮТЕРНЫЕ СЕТИ

1.1. Основы компьютерной коммуникации

1.1.1. Понятие «компьютерная сеть». Классификация сетей

Говоря общими словами, компьютерная сеть - это два компьютера, обменивающиеся сообщениями [1]. Разумеется, большинство сетей состоят из большего количества компьютеров. Принципы сетевого общения не зависят от количества составляющих сеть компьютеров. Чтобы понять, как сотни компьютеров общаются между собой, достаточно понять, как это делает пара компьютеров.

Сети бывают локальными или глобальными. Локальные сети (*LAN, Local Area Network*) представляет собой соединение нескольких компьютеров с помощью соответствующего аппаратного и программного обеспечения. Иногда компьютеры могут находиться на расстоянии нескольких миль и все равно принадлежать локальной сети. Компьютеры глобальной сети *(WAN, Wide-area network*) могут находиться в других городах или даже странах. Информация проделывает долгий путь, перемещаясь в глобальной сети. Интернет состоит из тысячи компьютерных сетей, разбросанных по всему миру, однако, пользователь и программист должны рассматривать Интернет как единую глобальную сеть.

Соединяя компьютеры между собой и давая им возможность общаться друг с другом, вы создаете сеть. Соединяя две и более сети, вы создаете межсетевое объединение, называющееся "интернет" (*internet*). На рис. 1.1 показано, как соотносятся сети и межсетевые объединения. *Internet* - самое большое и популярное межсетевое объединение в мире. Оно объединяет более 20000 компьютерных сетей, расположенных более чем в 130 странах. Internet также называют «сетью сетей». Влияние Internet на корпоративные сети способствовало появлению нового понятия – “интранет” (intranet), при котором способы доставки и обработки информации, присущие Internet, переносятся на корпоративную сеть. Главным образом применение технологии интрасетей означает использование стека TCP/IP для транспортировки данных и технологии Web для их представления. Для решения проблем, связанных с передачей корпоративных данных через публичную сеть, была разработана технология виртуальных частных сетей (*VPN – Virtual Private Networks*). Под термином «виртуальные частные сети» понимают достаточно широкий круг технологий, обеспечивающих безопасную и качественную связь в пределах контролируемой группы пользователей по открытой (публичной) глобальной сети. Данный термин используется также для обозначения устойчивых информационных потоков одного предприятия, существующих в публичной сети с коммутацией пакетов и достаточной степени защищенности от влияния потоков данных других пользователей этой публичной сети. В общем случае понятия «интрасеть» и «виртуальная частная сеть» не являются тождественными, т.к. первая не обеспечивает защиту трафика, а VPN можно

создать не только в Internet, но и в любой другой публичной сети с коммутацией пакетов (например, в сети frame relay).

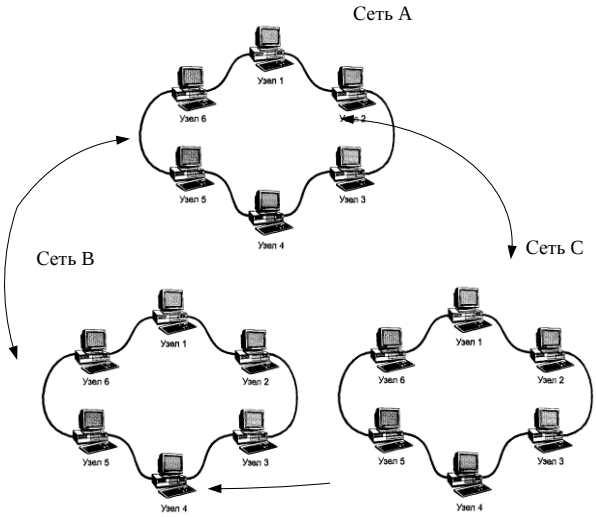


Рис. 1.1. Пример межсетевого объединения

Технология экстранет (*extranet*) расширяет понятие «интрасеть». Организация экстрасети подразумевает взаимодействие через Internet сетей и сотрудников ряда предприятий, которые являются бизнес-партнерами, т.е. экстранет – это сеть «бизнес-бизнес», использующая технологию Internet для взаимодействия бизнес-партнеров через публичные сети IP.

VPN является одной из дополнительных и необходимых функций, которые нужны для превращения Internet в новую публичную сеть NPN (New Public Network). Конечной целью создания NPN является транспортировка необходимой информации с минимальными задержками.

В простейшем случае сеть состоит из 2-х компьютеров. Чтобы узнать, как работает такая сеть, нужно понять, как происходит общение компьютеров. Компьютеры имеют в распоряжении два символа - единицу и ноль. Это так называемые двоичные цифры (символы), в комбинации образующие байты данных (слова по аналогии). Чтобы передать осмысленное сообщение, байты данных собираются в последовательность (предложение). Конечная цель передачи информации по сети - доставить ее человеку. Для компьютеров двоичные данные - понятный язык, в случае человека это не так. Чтобы человек мог прочитать двоичные данные, компьютеры преобразуют их в буквы (символы). Для представления данных в сети используются электрические сигналы. Двоичные числа являются последовательностью из нулей и единиц, и при передаче часто принято считать, что отсутствие электрического сигнала в линии означает ноль, а его наличие - единицу.

1.1.2. Передача данных между компьютерами. Типы соединений

Передача данных между компьютерами и другими устройствами происходит параллельно или последовательно. Большинство персональных

компьютеров пользуется параллельным портом для работы с принтером. Термин «параллельно» означает, что данные передаются одновременно по нескольким проводам. Чтобы послать байт данных по параллельному соединению, компьютер устанавливает одновременно восемь бит на восьми проводах. Схема параллельного соединения представлена на рис. 1.2 [1].

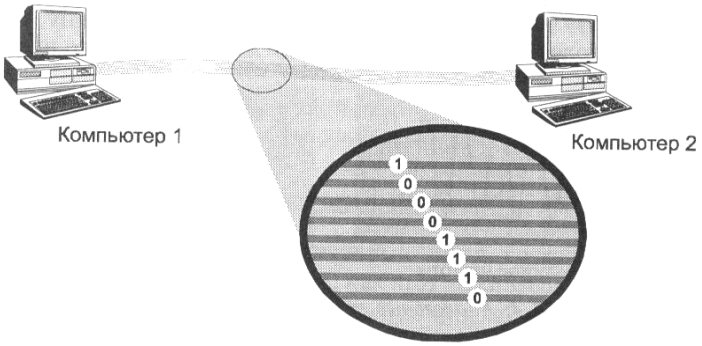


Рис. 1.2. Пример параллельного соединения

Как видно из рис. 1.2, параллельное соединение по восьми проводам позволяет передать байт одновременно. Напротив, последовательное соединение подразумевает передачу данных по очереди, бит за битом. В сетях чаще всего используется именно такой способ работы, когда биты выстраиваются друг за другом и последовательно передаются (и, стало быть, принимаются тоже). На рис. 1.3 показано, как передается двоичное число 10001110.

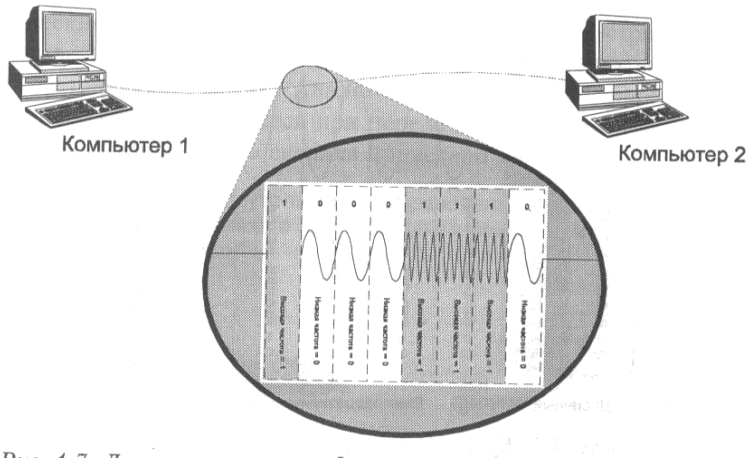


Рис. 1.3. Пример последовательного соединения

При соединении по проводам используются три различных метода, обозначаемые тремя различными терминами. Соединение бывает: симплексное, полудуплексное и дуплексное. О симплексном соединении говорят, когда данные перемещаются лишь в одном направлении. Полудуплексное соединение позволяет данным перемещаться в обоих направлениях, но в разное время. И, наконец, дуплексное соединение, это когда данные следуют в обоих направлениях одновременно. На рис. 1.4 указаны различия в потоках данных в зависимости от применяемого метода.

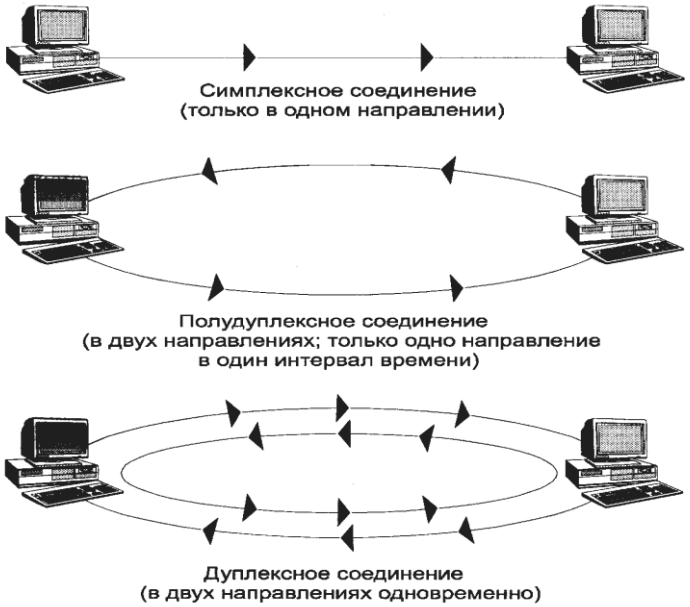


Рис. 1.4. Виды сетевых соединений

Переключение соединения используется сетями для передачи данных. Оно позволяет аппаратным средствам разделять один и тот же физический канал связи между многими устройствами. Рассмотрим, например, телефонные переговоры. Возьмем ситуацию, когда вы не хотите использовать коммутируемые телефонные линии. Для того, чтобы сохранить возможность звонить, например, тысяче абонентов, вы будете должны подсоединить к телефонному аппарату тысячу проводов, соединяющих вас напрямую. Поскольку вышеописанная ситуация чрезвычайно неудобна, большинство людей пользуется коммутируемыми линиям для переговоров. По этой же причине сети используют коммутацию (переключение) соединений. Существует два способа переключения соединения - переключение цепей и переключение пакетов.

Переключение цепей - создает единое непрерывное соединение между двумя сетевыми устройствами (рис. 1.5).

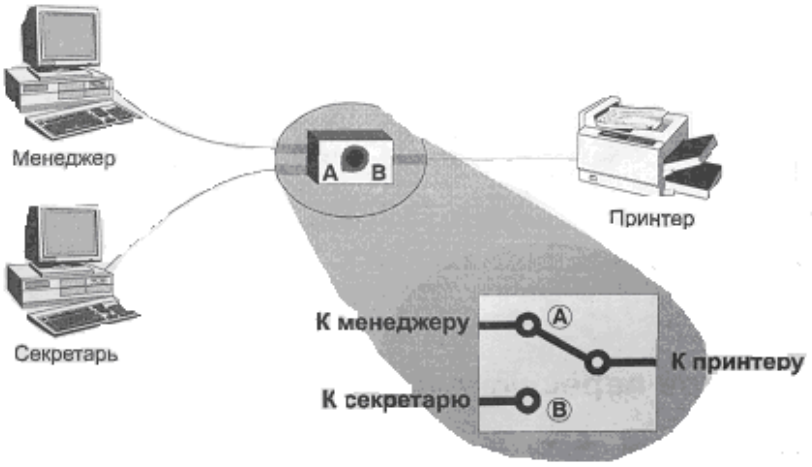


Рис. 1.5. Пример переключения цепей

Переключение цепи позволяет устройствам делить между собой один и тот же коммуникационный канал, однако, каждое должно ждать, пока наступит его очередь передавать или принимать данные. Простой пример переключения цепей - переключатель типа А-В, служащий, чтобы два компьютера соединить с одним принтером. Образуется соединение типа «точка-точка», только один компьютер может печатать в одно и тоже время. Большинство современных сетей, включая Интернет, используют переключение пакетов. Программы передачи данных в таких сетях делят данные на кусочки, называемые пакетами.

Для сравнения двух видов соединений в сети, предположим, что мы прервали канал в каждом из них. Например, отключив принтер от менеджера (рис. 1.5), можно лишить его возможности печатать. Соединение с переключением цепей требует наличия непрерывного канала связи.

Наоборот, данные в сети с переключением пакетов могут двигаться различными путями. Это видно на примере рис. 1.6.

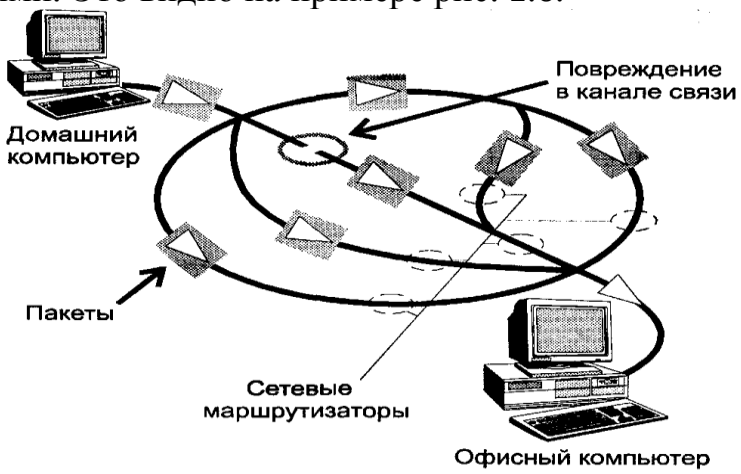


Рис. 1.6. Пример переключения пакетов

Данные необязательно следуют одной дорогой на пути между офисным и домашним компьютерами. Разрыв одного из каналов не приведет к потере соединения - данные просто пойдут другим маршрутом.

На первый взгляд, сети с переключением пакетов кажутся проще, чем какие-либо еще. Достаточно послать пакет, указав ему направление движения (при симплексной связи), и предоставить возможность найти дорогу самому. Однако сети, к сожалению, а может быть к счастью, не так просты и состоят отнюдь не из пары компьютеров. Сети с переключением пакетов имеют множество альтернативных маршрутов для пакетов. Данные перемещаются в обоих направлениях. Следовательно, каждый пакет должен содержать адрес назначения (пакеты часто содержат и адрес отправления). Концепция адресация пакетов - одна из важнейших для сети Интернет.

1.1.3. Топология сетей

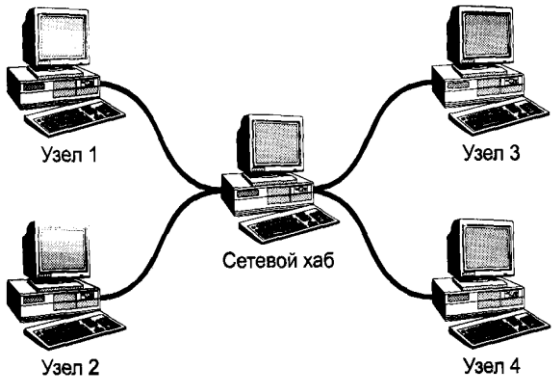
Существует невероятно большое число способов, которыми можно соединить компьютеры. Чем больше разных компьютеров, тем больше способов. Каждое соединение - это новый маршрут для данных. *Топология сети*

- это ее геометрическая форма или физическое расположение компьютеров, кабелей и других компонентов по отношению друг к другу. Топология сети дает способ сравнивать и классифицировать различные сети. Существует несколько основных топологий сети:

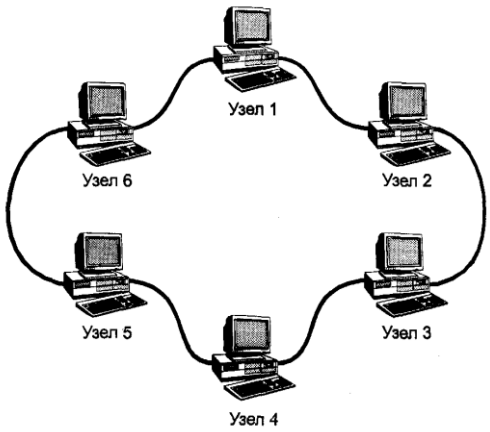
- звезда (star) – компьютеры подключены к сегментам кабеля, исходящим из одной точки, или концентратора (hub) (рис. 1.7а);

- кольцо (ring) - компьютеры подключены к кабелю, замкнутому в кольцо (рис. 1.7б);

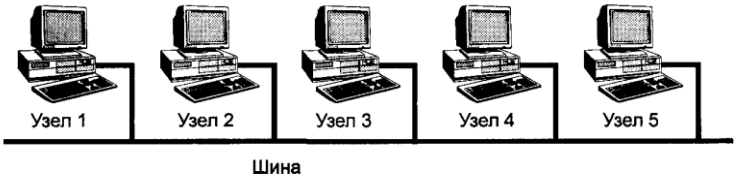
- шина (bus) – компьютеры подключены вдоль одного кабеля (сегмента). В сети с такой топологией данные следуют в обоих направлениях одновременно. На обоих концах кабеля - шины устанавливаются специальные заглушки (терминаторы) (рис. 1.7в).



а) Топология «звезда»



б) Топология «кольцо»



в) Топология «шина»

Рис. 1.7. Типы базовых топологий

В табл. 1.1 приведена сравнительная характеристика базовых топологий.

Таблица 1.1

Сравнительная характеристика базовых топологий

Топология	Преимущества	Недостатки
звезда	Централизованный контроль и управление. Легко модифицировать сеть, добавляя новые компьютеры. Выход из строя одного компьютера	Выход из строя центрального узла выводит из строя всю сеть.

	не влияет на работоспособность сети.	
кольцо	Все компьютеры имеют равный доступ. Количество пользователей не оказывает значительного влияния на производительность.	Выход из строя одного компьютера может вывести из строя всю сеть. Трудно локализовать возникающие проблемы. Изменение конфигурации сети требует остановки всей сети.
шина	Простота построения. Сеть легко расширяется. Экономный расход кабеля. Сравнительно недорогая и несложная в использовании среда передачи данных.	При значительных объемах трафика уменьшается пропускная способность сети. Трудно локализовать проблемы. Выход из строя кабеля останавливает работу многих пользователей.

На первый взгляд при сравнении топологий - лучшая «звезда». Однако ремонтировать неисправный кабель проще, быстрее и дешевле, чем специальный компьютер (центральный хаб). При выходе из строя одного из узлов сети, имеющей топологию шины и, следовательно, построенной с использованием коаксиального кабеля, возникают проблемы в работе всей сети. Поэтому сети, построенные с использованием витой пары (STP или UTP) с топологией "звезды", становятся все более распространенными. Большие затраты на прокладку такой сети оправдываются значительно более высокой степенью надежности.

В настоящее время часто используются комбинированные топологии. Поскольку Интернет является «сетью сетей», в нем встречается любые из перечисленных топологий. Например, вычислительные сети с древовидной структурой (рис. 1.8) применяются там, где невозможно непосредственное применение базовых сетевых структур в чистом виде.

Какую бы топологию не использовали, когда два компьютера начинают одновременно передавать данные, в сети происходят столкновения. *Шинный арбитраж* – процесс, призванный решить эту проблему. Он устанавливает правила, по которым компьютеры узнают, когда линия свободна и можно передавать данные. Существует два метода шинного арбитража: обнаружение столкновений и передача маркера. Используется метод - обнаружение столкновений с прослушиванием несущей (CSCD - carrier sense collision detection). Системы с передачей маркера - для того, чтобы передать данные, компьютер должен сначала получить разрешение - поймать циркулирующий в сети пакет данных специального вида, называемый маркером. Сеть имеет средства для обнаружения пропажи маркера и создания нового. В противном случае пропажа приводила бы к остановке сети. Каждый раз, когда компьютер

должен послать сообщение, он ловит и держит маркер у себя. Как только передача закончилась, он посылает маркер в путешествие дальше по сети.

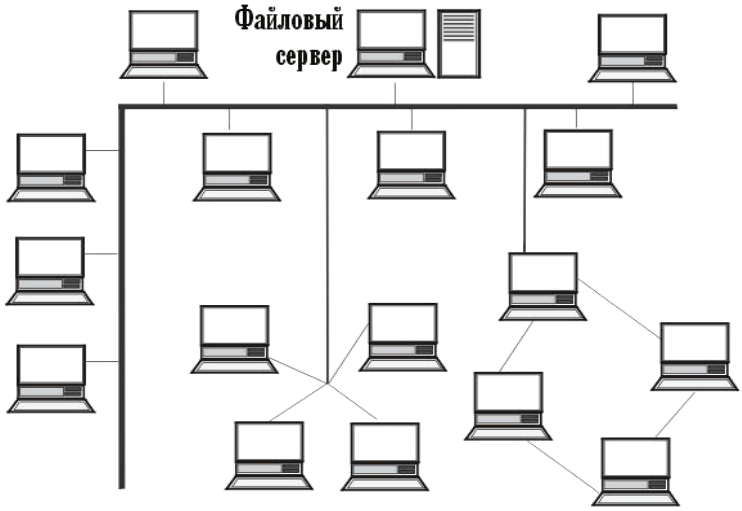


Рис. 1.8. Топология с древовидной структурой

1.1.4. Общие сведения о сетевых взаимодействиях. Классические сервисы в сетях

Процесс разделения (совместного использования) сетевых ресурсов называется сетевым взаимодействием (*networking*). Совместное использование ресурсов может осуществляться разными способами, зависящими от имеющихся в наличии компьютерных средств. Первый способ взаимодействия предполагает полностью централизованную обработку и хранение информации, обеспечивая работу пользователей с терминалов. Часто эту модель взаимодействия называют «терминал-хост» (*terminal-host*) (рис. 1.9).

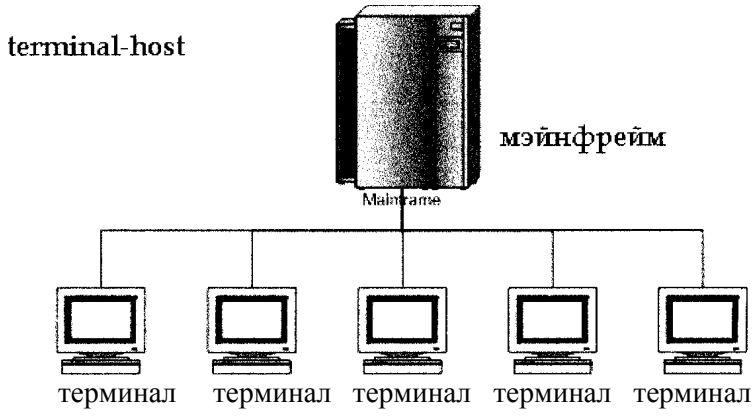


Рис. 1.9. Централизованное взаимодействие

Пользователь взаимодействует с ресурсами центрального компьютера, используя для решения своих задач его процессор, оперативную и дисковую память, а также периферийные устройства. При этом очень часто пользователь работает не один, а совместно с другими пользователями, то есть ресурсы

центрального компьютера используются в режиме разделения. Центральный компьютер должен работать под управлением операционной системы, поддерживающей такое взаимодействие, которое называется централизованным (*centralized computing*). Пользователь работает с центральным компьютером в режиме удаленного доступа. Все ресурсы центрального компьютера одновременно разделяются всеми его пользователями.

Дальнейшее развитие компьютерной индустрии шло различными путями, увеличивались вычислительные мощности компьютеров, предназначенных для работы по взаимодействию «терминал-хост», появились и начали бурно развиваться персональные компьютеры. Персональные компьютеры полностью управляются пользователем, все ресурсы компьютера используются в монопольном режиме для решения задач пользователя. Несмотря на рост вычислительной мощности процессоров, не весь спектр задач может быть решен одним компьютером. Появилась необходимость создания нового взаимодействия, новой структуры, направленной на распределенную обработку информации (*distributed computing*). В этой модели (рис. 1.10) взаимодействия каждый из компьютеров может решать свои задачи, появляется специализация компьютера.

Рис. 1.10. Распределенная обработка информации

Этот тип взаимодействия позволяет на каждом компьютере решать определенный набор задач, каждая задача решается только этим компьютером, и для ее решения используются ресурсы только этого компьютера. Компьютеры объединяются в вычислительную сеть. Задачи распределяются по компьютерам сети, что позволяет расширить функциональные возможности каждого из них путем разделения доступа к другим компьютерам.

Еще одной актуальной является задача объединения распределенных компьютерных ресурсов для выполнения (решения) общей задачи. Такая модель взаимодействия называется совместными, или объединенными вычислениями (*collaborative computing)* (рис. 1.11). При этом задача распределяется по компьютерам, компьютеры обмениваются между собой общими данными, суммарная вычислительная мощность и доступные ресурсы (оперативная и дисковая память) увеличиваются, повышается отказоустойчивость всей системы в целом с точки зрения решения задачи. В этой модели все компьютеры совместно решают одну или более задач. Для решения задачи используются ресурсы всех компьютеров. При отказе одного из них задача продолжает выполняться, часть задачи отказавшего компьютера

перераспределяется между оставшимися. Таким образом достигается высокий уровень устойчивой работы всей системы в целом.

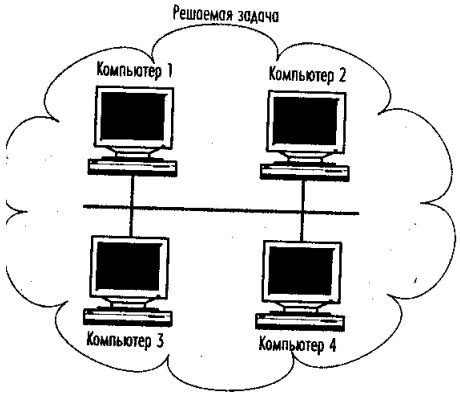


Рис. 1.11. Объединенные вычисления

Сравнительно новой моделью сетевых взаимодействий является организация взаимодействий пользователей сети с сетевыми сервисами. С точки зрения пользователя, его взаимоотношения с множеством компьютеров подпадают под определение «клиент-сеть» (client-network). В компьютерной сети присутствует много различных компонентов, самыми видимыми пользователям сети являются две. Это сервер сети и клиент. Сервер (server - в дословном переводе с английского означает «тот, кто обслуживает») сети предназначен для обслуживания поступающих от клиента (client) сети запросов. Другими словами, клиент всегда запрашивает обслуживание, а сервер всегда обслуживает клиента. В некоторых случаях клиент может выступать и в роли сервера, обеспечивая обработку запросов от других клиентов и запрашивая обслуживание у других серверов. По способу взаимодействия серверов и клиентов определяют два вида сетей "клиент/сервер" (client-server) и "равный с равным" (peer-to-peer). Поскольку клиентом сети является пользователь, работающий на компьютере, то сам компьютер пользователя, подключенный к сети, определяется термином "рабочая станция" (workstation). Этот термин употребляется наравне с термином "компьютер".

В модели «клиент/сервер» (рис. 1.12) рабочие станции формируют запросы на обслуживание и пересылают их серверу (1). Сервер, используя свои вычислительные мощности, обрабатывает запросы (2). Результаты обработки возвращаются рабочим станциям (3). В этой модели максимально используется разделение всех ресурсов сервера, учитывается его специализация. Именно в такой модели работают серверы приложений и клиенты, использующие эти приложения.

Часто модели «клиент/сервер» и "равный с равным" могут одновременно существовать в одной сети (рис. 1.13). Сети, построенные по принципу «равный с равным», называют также одноранговыми сетями, в которых все компьютеры имеют одинаковый статус - ранг.

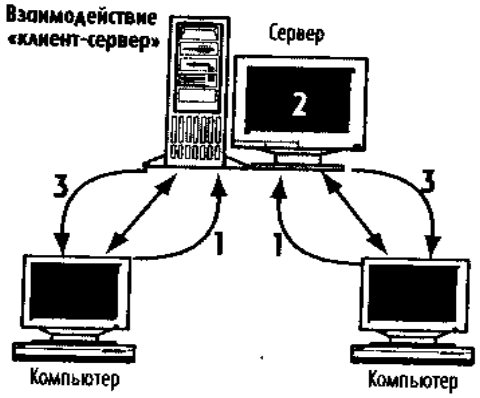
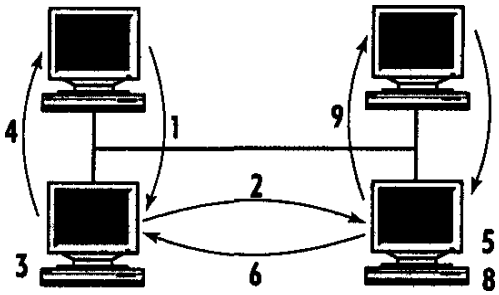


Рис. 1.12. Модель «клиент-сервер»

Компьютер 1 Компьютер 2



Компьютер 3/ Сервер 3 Компьютер 4/ Сервер 4

Рис. 1.13. Пример смешанного использования моделей

Использование дополнительных возможностей по предоставлению сервисов компьютерами (модели «клиент/сервер») в одноранговой сети расширяет функциональные возможности для пользователей. Компьютеры 1 и 2 выступают только как клиенты сети, а компьютеры 3 и 4 обеспечивают разделение своих ресурсов, например, дисков и принтеров, оставаясь при этом клиентами сети. Запросы 1, 2 и 7 поступают от клиентов. Эти запросы обрабатываются серверами (стадии обращений 3, 5 и 8), результаты возвращаются сетевым клиентам (стадии 4, 6, 9).

Сегодня широкое распространение находит *модель облачных вычислений, состоящая* из *внешней* (front end) и *внутренней* (back end) частей. Эти два элемента соединены по сети, в большинстве случаев через Интернет. Посредством внешней части пользователь взаимодействует с системой; внутренняя часть – это собственно само облако. Внешняя часть состоит из клиентского компьютера или сети компьютеров предприятия и приложений, используемых для доступа к облаку. Внутренняя часть предоставляет

приложения, компьютеры, серверы и хранилища данных, создающие облако сервисов. Концепция облака основана на уровнях**,** каждый из которых предоставляет определенную функциональность [2].

Классическими сервисами в сетях принято считать следующие: файловый, печати, сообщений, приложений и баз данных. Наиболее важными из них были и остаются файловый сервис и сервис печати. *Файловый сервис* обеспечивает выполнение задач по организации удаленного доступа, совместного использования, быстрого переноса и тиражирования, резервного копирования файлов. Этот сервис предусматривает наличие централизованных хранилищ файлов, эффективное использование их дисковых систем. *Сервис печати* позволяет пользователям коллективно получать доступ к устройствам печати через ограниченное количество интерфейсов (как правило, устройство печати имеет один, реже два интерфейса), разделять дорогостоящее специализированное оборудование печати, устранять ограничения расстояний между компьютером пользователя и устройством печати, организовывать и обрабатывать очереди запросов на печать. *Сервис сообщений* дает возможность организовать обмен сообщениями между пользователями сети, оперируя текстовой, графической, звуковой и видеоинформацией, позволяя не только передавать, но и сохранять все сообщения. В некоторых случаях этот сервис используется компьютерами (серверами сети) для извещения пользователей о наступлении каких-либо событий. Электронная почта является одной из реализации сервиса сообщений. *Сервис приложений* предоставляет пользователям возможность совместно применять не только данные (как в файловом сервисе), но и вычислительную мощность сервера для выполнения задач. При этом задача пользователя выполняется на процессоре сервера. Сервер приложений имеет специализацию, он оптимизирован для выполнения конкретных задач и должен поддерживать возможности дальнейшего наращивания своей вычислительной мощности. Сервис баз данных предназначен для организации централизованного хранения, поиска и обеспечения защиты данных. Этот сервис реализуется серверами баз данных, программно-аппаратными комплексами, оптимизированными для выполнения перечисленных задач, снижения времени доступа пользователя к информации, управления территориальным местоположением информации в сети.

Контрольные вопросы

1. Что понимается под термином «компьютерная сеть»?
2. Приведите классификацию сетей.
3. Какими способами может осуществляться совместное использование сетевых ресурсов?
4. Что понимается под топологией сети?
5. Перечислите основные топологии сети.
6. Перечислите классические сервисы в сетях.
7. Поясните модель облачных вычислений.
8. На чем базируется концепция облака?

1.2. Общие сведения о сетевых устройствах

Другими компонентами сети являются *средства организации канала передачи данных* между клиентами и серверами сети. В общем случае канал передачи данных строится с использованием следующих компонентов среды передачи данных - проводная (wire) или беспроводная (wireless) и интерфейсных карт (network interface card, NIC), обеспечивающих взаимодействие компьютера со средой передачи данных. Проводная среда передачи данных – данные передаются по кабелям, соединяющим отдельные компьютеры различным образом в зависимости от топологии и вида сети (Ethernet, Arcnet, Token Ring):

- витая пара – это два изолированных медных провода, скрученных между собой. Для Ethernet используется 8-жильный кабель, т.е. состоящий физически из 4-х витых пар. При этом различают неэкранированный (UTP) и экранированный (STP) кабели. Максимальная длина сегмента кабеля – 100 м;

- коаксиальный кабель состоит из центрального проводника (одножильного или многожильного) и внешней экранирующей оплетки. Для Ethernet применяется кабель с волновым сопротивлением 50 Ом. Существует два варианта реализации Ethernet на коаксиальном кабеле: на тонком кабеле и толстом. Для Ethernet на тонком кабеле рекомендуется использовать кабель RG-58. Толстый кабель «Yellow Ethernet» по своим показателям значительно превосходит тонкий. Максимальная длина сегмента кабеля: толстый коаксиальный кабель - 500 м (общая длина кабелей сети при использовании специальных усилителей может составлять 2500 м); тонкий коаксиальный кабель - 185 м (максимальная длина кабелей всей сети при использовании дополнительного оборудования может достигать 925 м);

- оптоволоконный кабель (ВОК), проводящий световые волны, состоит из двух проводов, причем каждый из них может передавать данные только в одном направлении. Этот кабель изготовлен из стекла (или пластика), покрытого материалом, отражающим свет, и оболочкой из различных термопластических материалов. ВОК может быть одномодовым и многомодовым. Лазер или светодиод испускает пульсирующий пучок света в торец стеклянного сердечника, расположенного на одном конце кабеля. Этот пучок распространяется по кабелю в одномодовом или многомодовом режиме, который зависит от физических свойств ВОК. На другом конце кабеля установлен приемник, преобразующий импульсы света в электрический сигнал. Такие кабели обладают следующими достоинствами: они невосприимчивы к электромагнитному и радиочастотному излучениям; позволяют передавать данные с очень высокой скоростью. Однако ВОК значительно дороже медного кабеля, а установка требует участия специалистов очень высокой квалификации.

Компьютеры подключаются к сети с помощью сетевой карты, которая устанавливается в один из свободных слотов материнской платы. Сетевые карты являются посредниками между компьютером и сетью и передают

данные по системе шин CPU и RAM сервера или рабочей станции. Сетевые карты бывают 16- и 32-разрядными и имеют исполнение для различных компьютерных архитектур: ISA, EISA, PCI, MCA. Большинство сетевых карт имеют гнездо для установки микросхемы ПЗУ удаленной загрузки (Remote Boot ROM), что необходимо для бездисковых станций. На внешней стороне карты имеются разъемы для подключения кабелей: BNC - для подключения тонкого коаксиального кабеля Ethernet (RG-58) (сетевая среда 10Base2), AUI - для подключения толстого кабеля Ethernet) (сетевая среда 10Base5), RJ (UTP) – разъем для подключения витой пары (сетевая среда 10BaseT, 10BaseTX), ST – разъем для подключения оптоволоконного кабеля (сетевая среда 10BaseFX, 100BaseFX).

Однако это не единственные средства, используемые для соединения компьютеров и формирования самой вычислительной сети. Объединять компьютеры в сеть и обеспечивать их взаимодействие помогают *сетевые аппаратные и аппаратно-программные средства*. Эти средства можно разделить на группы по их основному функциональному назначению: соединительные разъемы (connectors), повторители (repeaters), преобразователи (adapters), модемы (modems), мосты (bridges), концентратор (hubs), коммутаторы (switches), маршрутизаторы (routers).

Главная цель повторителя - повторять принятый сигнал и усиливать перед передачей. Повторители можно использовать не только на участках объединения сетей, но и в отдельной локальной сети.

Мост (bridge) - это устройство, соединяющее две сети, построенные по одной и той же технологии (например, ETHERNET или ARCNET). Устройство мостов более сложное, чем повторители. В обязанности моста входит анализировать все пакеты данных, проходящих мимо него по обеим сетям. Кроме объединения сетей мост занимается и другими вопросами (безопасность данных, увеличение производительности и надежности сетей). Мост разделяет большую перегруженную сеть на две меньшие и более эффективные сети. Пакеты передаются между этими подсетями только, если адрес назначения пакета принадлежит другой подсети. Мост снижает вдвое нагрузку сети и в идеале может повысить ее пропускную способность на 200%. Отметим, что с обеих сторон моста должен стоять сервер. Соответственно не имеет смысла использовать мост, если в сети только один сервер.

Концентраторы – расширяют радиус действия сети путем ретрансляции сигнала. Их называют также многопортовыми повторителями сети с автосегментацией. Все порты концентратора равноправны. Получив сигнал от одной из подключенных к нему станций, концентратор транслирует его на все свои активные порты. Назначение концентраторов – объединение отдельных рабочих мест в рабочую группу в составе локальной сети. Концентраторы работают на физическом уровне (1 уровень модели OSI), поэтому они не чувствительны к протоколам верхних уровней. Однако даже при наличии повторителей в сети существуют ограничения расстояния передачи пакетов,

например, в сети Ethernet максимально возможны четыре пересылки между любой парой устройств – серверов или рабочих станций – в одном и том же сегменте. Если сеть приходится расширять и дальше, следует использовать коммутатор, мост или маршрутизатор.

Коммутаторы представляют собой высокоскоростные многопортовые мосты, способные пропустить 10Мбит/с при Ethernet или 100Мбит/с при Fast Ethernet – через каждый порт. Подобно мостам, коммутаторы принимают интеллектуальные решения о том, куда направить сетевой трафик, исходя из адреса назначения пакета.

В отличие от мостов ряд коммутаторов не помещает все приходящие пакеты в буфер, а коммутирует пакеты «на лету», т.е. анализирует адрес назначения в заголовке пакета и, сверившись с адресной таблицей, тут же направляет этот пакет в соответствующий порт. Таким образом, коммутаторы микросегментируют сеть, поддерживают параллельный трафик, фильтруют сетевой трафик, могут поддерживать полнодуплексный режим.

Коммутатор - концентратор (Switching Hub) - коммутирующий концентратор похож на мост, но объединяет более двух сегментов сети. В то же время он делит сеть на сегменты, работающие более эффективно. Удобство такого концентратора еще и в том, что он может обрабатывать несколько потоков данных одновременно. Коммутатор-концентратор идеально подходит для сети с двумя серверами. Использование коммутаторов-концентраторов повышает прозрачность доступа к данным для пользователей. Кроме того, сеть с коммутаторами-концентраторами в дальнейшем будет легче реконфигурировать.

Маршрутизатор - это устройство, которое маршрутизирует данные между сетями. Маршрутизатор может соединять сети с одинаковой технологией, но чаще он используется там, где технологии сетей различны, например, между Ethernet и Token Ring. Маршрутизатор - важная составляющая Интернет, поскольку важнейшая функция сети Интернет - соединять различные сети.

Модем для компьютера является периферийным устройством, которое позволяет организовывать соединения с другими компьютерами и обмениваться с ними данными через телефонные линии. MODEM - термин происходит от слов MOdulator (цифр - аналог) DEModulator (аналог- цифр) и обозначает устройство передачи данных, преобразующее цифровой сигнал (последовательность битов) в аналоговый сигнал, который затем можно передавать по телефонным линиям. Первые модемы использовались главным образом для обмена между терминалами данных и хост-компьютерами. Сегодняшние модемы используют различные методы сжатия информации для дополнительного повышения скорости обмена и контроля ошибок, а также их исправления для обеспечения более надежной связи.

Наиболее часто применяемые при построении сетей проводные среды передачи данных создаются с помощью кабельных соединений, в которых

используется либо металлический проводник электрических сигналов, либо волоконно-оптический проводник световых сигналов. Для подключения старых сетевых карт с AUI-разъемами к новым типам соединений, например UTP (витая пара) или FOIRL, используются трансиверы (*transceiver*). Трансивер может использоваться для перехода из среды AUI в какую-либо другую среду.

Беспроводные среды передачи информации предусматривают организацию взаимодействия между компьютерами посредством передачи световых (инфракрасных) и радиочастотных сигналов.

С помощью сред передачи данных и некоторых аппаратно-программных средств обеспечения межкомпьютерного взаимодействия формируется физическая топология сети, осуществляются физические соединения всех компьютеров и других сетевых средств.

Современные сетевые технологии работают на пределе своих возможностей, производительность настольных систем непрерывно увеличивается, приложения требуют все больших сетевых ресурсов, растет потребность в более мощных средствах управления сетями.

Контрольные вопросы

1. Перечислите основные компоненты среды передачи данных.
2. Поясните классификацию сетевых устройств по функциональному назначению.
3. Каковы особенности сетевых устройств?
4. Какое сетевое устройство позволяет повысить прозрачность доступа к данным для пользователей?
5. С какой целью используются трансиверы?

1.3. Эталонная модель взаимодействия открытых систем OSI/ISO

1.3.1. Эталонная модель взаимодействия открытых систем как основа организации информационных процессов

В развитии средств обработки данных переход к распределенным ВС и к информационно-вычислительным сетям вызвал необходимость разработки определенной концепции взаимодействия разнотипных ЭВМ, что оказалось возможным на основе единой системы организации информационных процессов. В основе этого лежит эталонная модель взаимодействия открытых систем (ЭМ ВОС), принятая международной организацией стандартов (МОС).

В период с 1977 по 1984 год профессионалы разработали модель сетевой архитектуры под названием "*рекомендуемая модель взаимодействия открытых систем" (the Reference Model of Open Systems Interconnection, OSI*). Термин *"открытая система"* означает, что свойства и структура данной системы не являются чьей-либо собственностью. Другими словами, вам доступно полное описание данной системы и возможность свободно использовать ее для собственных нужд. Разумеется, для этого вы должны обладать полной документацией на нее, достаточной, чтобы создавать

собственные программы, использующие или расширяющие данную открытую систему.

Модель OSI не возникла на пустом месте. Она базируется на модели, предложенной Международным институтом стандартов (*International Standards Organization, ISO)*. Термин "рекомендуемая модель взаимодействия открытых систем" часто встречается в литературе под названием "*модель ISO/OSI",* отмечая вклад ISO в ее формирование.

Модель ISO/OSI использует деление на уровни, чтобы организовать общее представление о структуре сети в виде четко определенных, взаимосвязанных модулей. Уровни в настоящей сети могут иметь отличающийся набор функций, их может быть другое количество. В результате сети, построенные по одной и той же модели, могут существенно отличаться друг от друга. В сети, поделенной на уровни, каждый уровень служит для исполнения определенной функции или службы сети по отношению к окружающим соседним уровням. Каждый уровень как бы защищает соседний от избыточной информации, способной просочиться от более низкого уровня наверх.

На рис. 1.14 изображены уровни в том виде, в каком они определены в модели ISO/OSI.

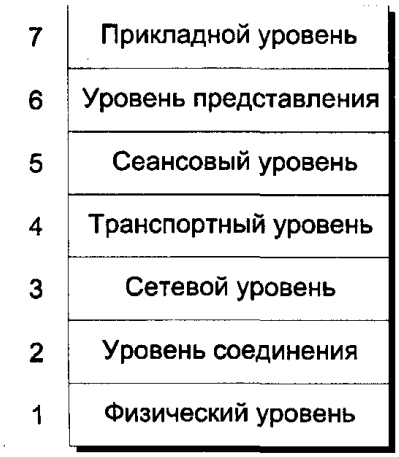


Рис. 1.14. Сетевые уровни модели ISO/OSI

1.3.2. Физическое, процедурное, логическое сопряжение

Взаимодействие информационно-вычислительных сетей как открытых систем реализуется на основе принятых уровней сопряжения. К таким уровням относятся прикладной, представительный, сеансовый, транспортный, сетевой, канальный, физический. Обоснованием целесообразности использования семиуровневой модели взаимодействия открытых систем служит путь, которым прошло совершенствование техники связи и ВТ.

Вполне естественным и необходимым является низший уровень сопряжения, которое обеспечивает физическое (механическое либо электрическое) соединение. Это установление числа проводов, формы и

размеров разъемов, назначение уровня напряжения в проводах и различных характеристик сигналов, передаваемых между объектами.

Следующий уровень должен обеспечить функциональное сопряжение, которое и определяет параметры сигналов, возникающих между сопрягаемыми устройствами.

При обмене информации с целью управления процессом передачи необходимо установить способы представления информации в виде конкретных кадров, для чего предусматривается логическое сопряжение.

Разные форматы сообщений и порядок их использования для управления процессом связи задает процедурное сопряжение. Логическое и процедурное сопряжение определяют протокол обмена информацией.

Базовая сеть состоит из коммуникационных машин, связанных между собой физическими соединениями. Функции базовой сети обмена данными по существу полностью описываются физическим, канальным и сетевым уровнями.

Реализация транспортного уровня осуществляется уже с использованием программного обеспечения ЭВМ.

Верхние три уровня (сеансовый, представительный и прикладной) определяют область обработки данных. На этих уровнях в качестве взаимодействующих объектов выступают процессы. На верхнем (пользовательском) уровне действует протокол "процесс-процесс", базирующийся на понятии логического (виртуального) канала.

Отметим, что эталонная модель взаимодействия открытых систем задает лишь идеологию взаимодействия между процессами, но не определяет стандартов - протоколов взаимодействия для каждого из семи уровней. Ценность эталонной модели в том, что на ее основе дается единая терминология для различных включенных в ИВС объектов.

Каждый сетевой уровень обеспечивает связь для вышележащего уровня.

1.3.3. Функции и задачи уровней

Рассмотрим функциональное содержание уровней эталонной модели и пути их реализации.

*Физический уровень***.** Основная задача физического уровня - реализация интерфейса с канальным уровнем при наличии соответствующих управляющих сигналов. Такой интерфейс реализуют модемы. Реализация интерфейса на физическом уровне, по сути, означает обеспечение стыка между оконечным оборудованием данных и аппаратурой передачи данных.

Следующая функция, выполняемая физическим уровнем, - обеспечение соединения (выделенные линии или коммутируемые каналы связи).

При наличии соединения физический уровень обеспечивает реализацию следующих функций: преобразование дискретных сигналов на входе физического канала в непрерывные сигналы; передачу этих сигналов по

физическому каналу; обратное преобразование в последовательность дискретных элементов.

Физический уровень эталонной модели отображает физическую среду процесса передачи. Процедуры установления и разъединения соединений регламентированы МККТТ Х.60, Х.70.

Канальный уровень. На этом уровне реализуются функции по управлению каналом передачи данных, который, в свою очередь, формируется из совокупности средств физического и канального уровней. На канальном уровне обеспечивается: формирование кадра - определенной последовательности бит заданной длины; введение избыточности в передаваемый код с целью обнаружения и исправления ошибок; управление потоком данных с целью согласования скорости передачи с возможностями канального уровня.

Сетевой уровень - на этом уровне реализуются в основном следующие функции: маршрутизация сообщений в сети (выбор маршрута в соответствии с некоторым критерием при наличии нескольких альтернатив); ограничение нагрузки, что обеспечивается за счет управления информационными потоками (увеличение объема буферной памяти в узлах коммутации/ЭВМ); позволяет реализовать приоритетное обслуживание сообщений. Существует рекомендация МККТТ Х.25, содержащая три уровня протоколов и определяющая условия доступа абонентов к сети.

Транспортный уровень. Основные функции: установление транспортных соединений между прикладными процессами на уровне логического канала; адресация сообщений с определением соответствия между сетевыми адресами и адресами потребителей; контроль ошибок потерь сообщений и их задержек (если на предыдущих уровнях не удается обеспечить требуемый уровень помехоустойчивости); восстановление сообщений при приеме; управление потоками блоков сообщений с преобразованием структуры блоков и представлением приоритета.

Последняя функция является важнейшей. На транспортном уровне решаются принципиально те же задачи по управлению информационным потоком, что и на канальном и сетевом уровнях, однако, транспортный уровень позволяет обеспечить одновременно ряд соединений. Рекомендации протокола Х.25 (кодирование информации).

Сеансовый уровень. Основные функции: установление соединения между парой прикладных процессов, называемого сеансом (решение комплекса задач: направление передачи информации, число сеансов в течение одного соединения идентификация сеансов и т.д.); реализация диалогового взаимодействия процессов в ходе сеанса.

Сеансовые и более высокие уровни эталонной модели в настоящее время практически не регламентируются какими-либо типовыми протоколами.

Представительный уровень. Основные функции: определение и подготовка совокупности форм представления данных, выбираемых

прикладным процессом; преобразование данных в соответствии с потребностями прикладного процесса.

По существу, этот уровень должен обеспечивать унифицированный интерфейс, позволяющий прикладному процессу взаимодействовать с другим конкретным процессом. Функционирование представительного уровня обеспечивается за счет услуг, предоставляемых нижним (сеансовым) уровнем, который реализует соединения между процессами.

Прикладной уровень. К основным функциям прикладного уровня можно отнести следующее: обеспечение доступа через сеть к требуемым прикладным процессам; синхронизация взаимодействующих прикладных процессов; управление данными, необходимыми для реализации прикладных процессов.

Функции прикладного уровня обеспечиваются за счет услуг, предоставляемых представительным уровнем в эталонной модели взаимодействия открытых систем.

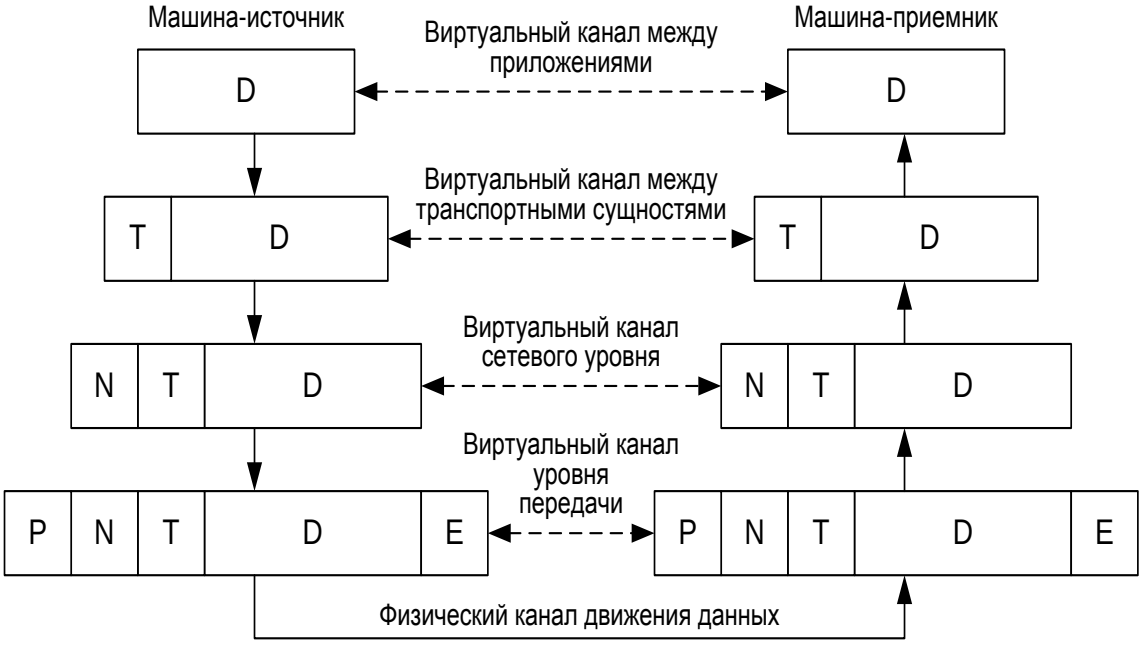
Прикладной уровень является основным пользовательским уровнем в эталонной модели. Именно на этом уровне взаимодействуют современные коммуникационные системы. В таких условиях особую значимость приобретают такие прикладные процессы, как администрирование и управление распределенной обработкой.

Обработка информации на прикладном уровне при наличии квалифицированного пользователя должна приводить к формированию знаний. Модели обработки информации, модели представления знаний, рассмотренные выше, должны быть положены в основу формирования информационного ресурса.

Из семи уровней эталонной модели достаточно подробно раскрыты три нижних уровня, для которых информационные процессы декомпозируются на известные процедуры.

Если два хоста обмениваются информацией по сетевому соединению, с точки зрения пользователя информацией обмениваются приложения, каждое из которых работает на своей машине. То же самое происходит с точки зрения протокола любого другого уровня, т.е. транспортная сущность обменивается информацией с такой же транспортной сущностью на другой машине, сетевой уровень с сетевым уровнем и т.д. Возникают виртуальные каналы связи между протоколами одного уровня (рис. 1.15).

Проходя через стек протоколов, информация упаковывается в пакеты [3]. Пакет состоит из данных и служебной информации, которую добавляет протокол соответствующего уровня. Пакет продвигается сверху вниз по уровням сетевой модели. Протокол каждого уровня добавляет свою служебную информацию. Данными для него является пакет вышележащего уровня.



D – модуль данных приложения; T – заголовок транспортного модуля; N – заголовок сетевого пакета;
P – заголовок кадра уровня передачи; E – концевик кадра уровня передачи

Рис. 1.15. Виртуальные каналы, инкапсуляция данных

Для передачи данных по сети необходимо выполнить три шага (рис. 1.16):

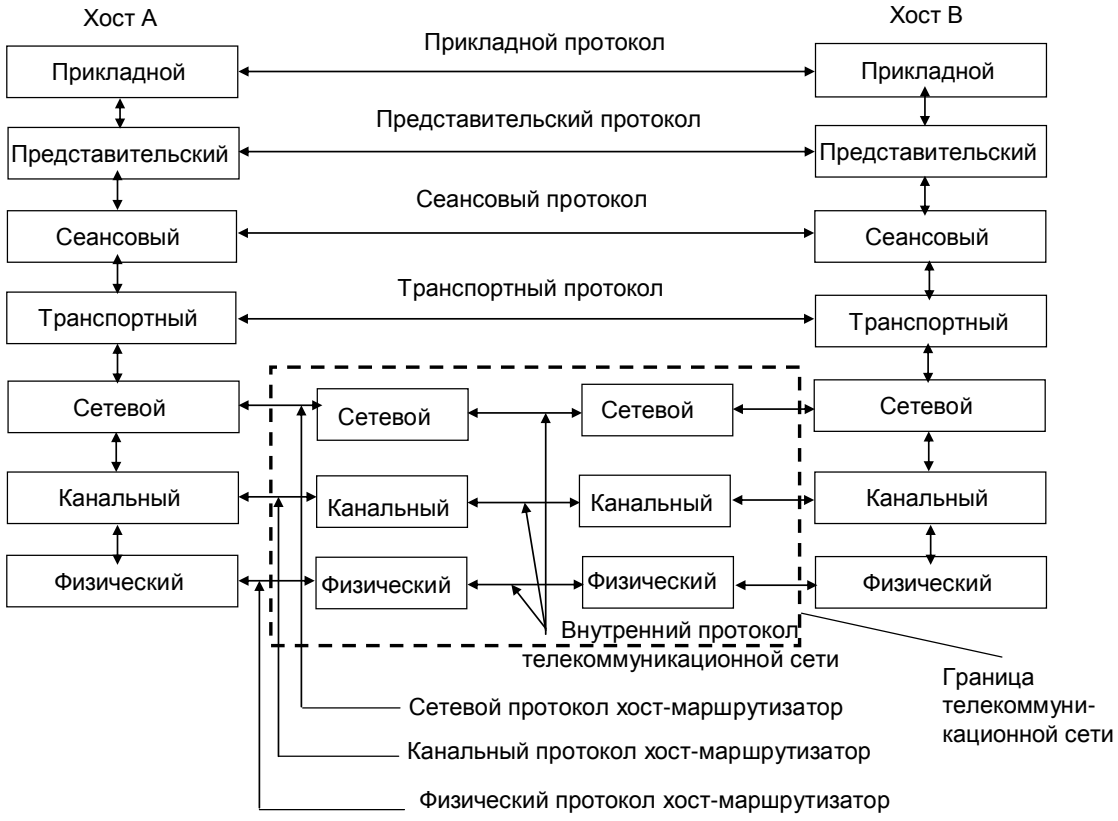


Рис. 1.16. Передача данных через стек протоколов

1) Информация должна пройти между приложением и сетью. Это путь сквозь стек протоколов вниз к физическому уровню.

2) Сеть должна определить место назначения информации или, строго говоря, определить адрес получателя сетевых данных.

3) Сеть должна физически переместить данные к месту назначения, воспользовавшись для этого маршрутизацией. Появившись в месте назначения, данные должны пройти сквозь стек протоколов вверх к сетевому приложению. Стрелками на рис. 1.16 обозначены возможные пути перемещения данных при обмене между различными коммуникационными системами и сетевой аппаратурой.

Контрольные вопросы

1. Назначение и содержание модели ISO/OSI.
2. В чем ценность эталонной модели взаимодействия открытых систем?
3. Определите понятие логического канала.
4. Охарактеризуйте функции физического уровня.
5. Какой уровень модели позволяет реализовать приоритетное обслуживание сообщений?
6. Что такое инкапсуляция?
7. Используя модель ISO/OSI, покажите возможные пути перемещения данных при обмене между различными коммуникационными системами и сетевой аппаратурой.

1.4. Локальная вычислительная сеть

1.4.1. Понятие локальной вычислительной сети (ЛВС)

ЛВС (Local Area Network) - сеть, предназначенная для объединения территориально сгруппированных сетевых устройств. Все сетевые устройства внутри LAN обладают информацией о MAC-адресах соседних сетевых адаптеров и обмениваются данными на втором (канальном) уровне семиуровневой модели OSI. Компьютеры, связанные локальной сетью, объединяются в один виртуальный компьютер, ресурсы которого могут быть доступны всем пользователям, причем этот доступ не менее удобен, чем к ресурсам, входящим непосредственно в каждый отдельный компьютер. Отличительными признаками локальной сети являются: высокая скорость передачи информации, большая пропускная способность сети (не менее 10 Мбит/с); низкий уровень ошибок передачи (допустимая вероятность ошибок передачи данных должна быть порядка 10^{-8} -10^{-12}); эффективный, быстродействующий механизм управления обменом по сети; заранее четко ограниченное количество компьютеров, подключаемых к сети.

При создании локальных сетей чаще всего используется аппаратная архитектура, называемая Ethernet. В простейшем виде сеть Ethernet состоит из одного кабеля, к которому при помощи разъемов, коннекторов и трансиверов подключаются все сетевые узлы.

1.4.2. Технология Ethernet

Локальная сеть Ethernet - стандарт организации локальных вычислительных систем сетей IEEE 802.3, используемых для соединения устройств, находящихся на небольшом удалении друг от друга (в одном здании, группе зданий). Данный стандарт широко распространен, так как отвечает потребностям большинства пользователей локальных сетей и межсетевых соединений.

Сеть может иметь следующие *топологии:* шина (рис. 1.17) или пассивная звезда (пассивное дерево), в качестве *среды передачи данных* можно использовать любые типы кабелей, а также радиочастоты (radioEthernet).

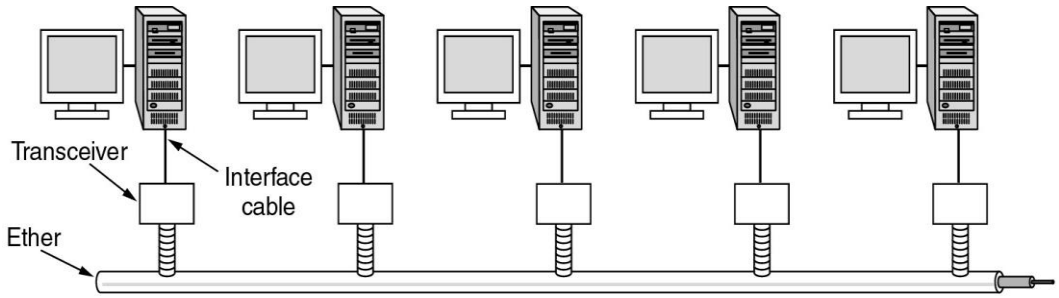


Рис. 1.17. Пример сети Ethernet

Ethernet является сетью множественного доступа с контролем несущей и обнаружением конфликтов CSMA/CD (Carrier Sense Multiple Access with Collision Detection). Каждый сетевой интерфейс может находиться в одном из трех состояний: передачи, конкуренции, простоя (рис. 1.18).

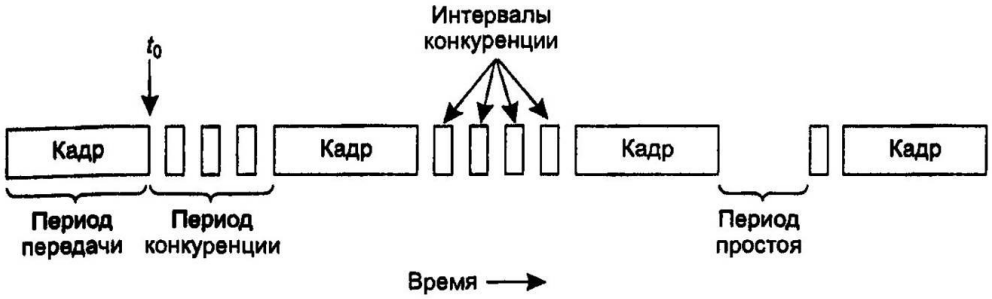


Рис. 1.18. Состояния сетевого интерфейса

В период простоя интерфейс слушает сеть и, обнаружив чужую передачу, не начинает свою, а если он в состоянии передачи, то прекращает ее. Столкновение передач двух интерфейсов называется *коллизией.* После обнаружения коллизии по возрастанию уровня шумового сигнала, интерфейс ждет случайный период времени, после чего снова пытается передать кадр при условии, что кто-то другой не начал передачу. В случае второй, третьей и так далее коллизии время ожидания всякий раз увеличивается на случайный интервал в диапазоне от 0 до 2 в степени номера коллизии. Это называется *экспоненциальным двоичным откатом* и гарантирует, что все интерфейсы рано или поздно передадут свои кадры.

При обнаружении коллизии передатчик обрезает текущий кадр, вследствие чего по сети все время распространяются остатки кадров. Чтобы отличить «мусор» от нормальных кадров, обеспечивают длину кадров не менее 64 байт. Отсюда наличие поля заполнения. Интерфейс услышит столкновение передач через время 2t, где t - время «добегания» кадра до конкурирующей станции. При длинном кабеле и высокой скорости передачи шумовой всплеск не успеет дойти до передатчика за время передачи кадра. Интерфейс зафиксирует успешную передачу, а пакет будет утерян. При длине кабеля 2500 м и скорости передачи 1 Гбит/с минимальный размер кадра должен составить уже 6400 байт. По мере роста скоростей эти проблемы все больше влияют на стандарты.

Для повышения эффективности функционирования сетей, в которых серверы и рабочие станции выполняют критичные к скорости передачи приложения, были разработаны взаимодополняющие технологии коммутации локальных вычислительных сетей и Fast Ethernet.

Устройства Fast Ethernet, базируясь на том же протоколе CSMA/CD (коллективный доступ с опросом канала и обнаружением коллизий), работают со скоростью, в 10 раз превышающую скорость Ethernet. Подобно Ethernet, Fast Ethernet представляет собой технологию коллективного пользования, основанную на конкуренции. Использует такие же инструментальные программные средства для управления и диагностики неисправностей, что дает возможность защитить капиталовложения в оборудование и обучение сотрудников LAN. К различиям данных технологий относятся сетевая кабельная система, число повторителей и ограничения на длину кабеля. В Fast Ethernet используется только кабель - витая пара и волоконно-оптические кабели; коаксиальный кабель не поддерживается. Подобно наличию спецификаций кабелей Ethernet – 10BASE-T для витой пары, 10BASE2 для тонкого коаксиального кабеля, 10BASE5 для толстого коаксиального кабеля, 10BASE-F для волоконно-оптического кабеля - существуют спецификации и для каждого типа кабеля Fast Ethernet (табл.1.2).

Таблица 1.2

Спецификации кабелей Fast Ethernet

Спецификация	Тип кабеля	Категория	Число пар
100BASE-TX	Витая пара 5	5	2
100BASE-T4	Витая пара	3, 4, 5	4
100BASE-TF	Многомодовый волоконнооптический	отсутствует	1

1.4.3. Организация и сервис виртуальных частных сетей (VPN)

VLAN (*Virtual Local Area Network*) - группа устройств, имеющих возможность взаимодействовать между собой напрямую на канальном уровне, хотя физически при этом они могут быть подключены к разным сетевым коммутаторам. И, наоборот, устройства, находящиеся в разных VLAN, невидимы друг для друга на канальном уровне, даже если они подключены к

одному коммутатору, и связь между этими устройствами возможна только на сетевом и более высоких уровнях. В современных сетях VLAN - главный механизм для создания логической топологии сети, не зависящей от её физической топологии. VLAN используются для сокращения широковещательного трафика в сети и имеют большое значение с точки зрения обеспечения безопасности.

Можно выделить следующие способы организации VPN: организация VPN силами потребителя (Customer Provided VPN); организация VPN силами поставщика телекоммуникационных услуг (Provider Provisioned VPN).

Наиболее известными протоколами VPN являются:

- PPTP (Point-to-Point Tunneling Protocol) - туннельный протокол, разработанный Microsoft совместно с другими компаниями, представляет собой расширение протокола PPP (Point-to-Point Protocol) для создания защищенных виртуальных каналов. Предусматривает создание криптозащищенного туннеля на канальном уровне модели OSI. Для передачи данных используются IP-пакеты, содержащие инкапсулированные PPP-пакеты. Инкапсулированные PPP-пакеты содержат, в свою очередь, зашифрованные инкапсулированные исходные пакеты (IP, IPX, NetBEUI);
- L2TP (Layer 2 Tunneling Protocol) - индустриальный стандарт Интернет, туннельный протокол, который обеспечивает инкапсуляцию и пересылку кадров протокола PPP. Протокол L2TP шифрует IP-трафик и пересылает через среду. Реализация протокола Microsoft L2TP использует IPSec шифрование для защиты потоков данных на всем пути от VPN клиента до VPN сервера. L2TP и IPSec обеспечивают более высокую степень защиты данных, чем PPTP, так как использует алгоритм шифрования Triple Data Encryption Standard (3DES). Соединения по протоколу L2TP/IPSec требуют аутентификации, основанной на сертификатах;
- IPsec (IP Security Protocol) - это служба обеспечивающая аутентификацию, доступ и контроль за надежностью. Работает на уровне сети. IPSec позволяет создавать кодированные туннели VPN или кодировать трафик между двумя узлами. В состав службы входят протоколы: AH (Autentication Header) – заголовок аутентификации, ESP (Encapsulating Security Payload – инкапсуляция зашифрованных данных), IKE (Internet Key Exchange – обмен ключами). Для шифрования данных в системе IPsec может быть применен любой симметричный алгоритм шифрования. Более подробно организация VPN описана в [4].

1.4.4. Технические средства построения ЛВС. Концентраторы и маршрутизаторы

Технические средства построения ЛВС можно разделить на пассивные и активные. К пассивным техническим средствам, используемым для объединения отдельных сегментов и расширения ЛВС, относятся повторители и концентраторы (последние практически не выпускаются, но все еще

используются). К активным устройстам, объединяющим отдельные ЛВС, относятся мосты, маршрутизаторы, коммутаторы. Основной функцией пассивных устройств является усиление передаваемого сигнала. Активные устройств управляют трафиком на основе адресов назначения передаваемых данных.

Маршрутизатор - многофункциональное устройство, действует на сетевом уровне и обладает следующими особенностями: учитывает специфику протоколов, используя маршрутную информацию сетевого уровня; может обмениваться с другими маршрутизаторами информацией для сбора данных о топологии и состоянии сети; определяет логические границы между группами сетевых сегментов. Маршрутизаторы отвечают за создание и поддержку для каждого протокола сетевого уровня маршрутных таблиц, которые могут быть статическими или динамическими. Кроме того, они идентифицируют протокол в заголовке каждого пакета, находят адрес получателя сетевого уровня и выбирают путь передачи данных, содержащийся в маршрутной таблице соответствующего протокола. Достоинства маршрутизаторов: создают защитный барьер между подсетями; защищают информацию с помощью фильтров пакетов; могут разбивать длинные сообщения на несколько коротких, что позволяет соединять сети, в которых используются пакеты различной длины. К недостаткам следует отнести: сложность в установке и конфигурировании (в отличие от мостов) и необходимость смены сетевого адреса компьютера при его перемещении из одной подсети в другую.

Коммутаторы представляют собой высокоскоростные многопортовые мосты, способные пропустить 10 Мбит/с при Ethernet или 100 Мбит/с при Fast Ethernet через каждый порт. Коммутаторы ЛВС микросегментируют сеть - делят ее на меньшие сегменты (collision domains), а затем соединяют эти сегменты, давая им возможность связаться друг с другом. Путем сокращения числа узлов в сегменте микросегментация сокращает число коллизий и увеличивает доступную пропускную способность в расчете на один узел. Путем соединения сегментов через коммутаторы формируется единая ЛВС с потенциальной пропускной способностью, во много раз превышающей пропускную способность первоначальной односегментной ЛВС.

Каждый порт коммутатора фактически является входом в отдельный сегмент ЛВС. Этот сегмент можно совместно использовать многими станциями, присоединенными к концентратору, или может быть выделен для одного устройства - сервера или рабочей станции.

Контрольные вопросы

1. Что понимается под ЛВС?
2. Поясните особенности технологии Ethernet.
3. Назовите основное отличие Ethernet и Fast Ethernet.
4. Что такое коллизия и причины ее возникновения.
5. Поясните термин «экспоненциальный двоичный откат».

6. Приведите классификацию технических средств ЛВС.
7. Что является основой VPN?
8. Назовите наиболее известные протоколы VPN.

1.5. Маршрутизация

1.5.1. Определение и концепция маршрутизации

Маршрутизация - процесс определения лучшего пути, по которому пакет может быть доставлен получателю. Возможные пути передачи пакетов называются маршрутами. Лучшие маршруты к известным получателям хранятся в таблице маршрутизации.

Другое определение *маршрутизации* - совокупность процессов сетевого уровня по доставке пакетов от отправителя к получателю. Именно на сетевом уровне система принимает решение о том, что делать с тем или иным пакетом, попавшим на сетевой интерфейс, для того, чтобы данный пакет был доставлен точно по адресу. Более низкие уровни модели (например, уровень передачи) не способны решить эту задачу. Технология Ethernet прекрасно справляется (с помощью протокола ARP) с доставкой кадров в пределах своей сети, но не имеет внутренних возможностей для осуществления этой доставки к хостам других сетей. Так как на этапе перехода пакетов в другие сети вступает в дело сетевой уровень, устройства, объединяющие сети - маршрутизаторы, шлюзы, работают на сетевом уровне. Это могут быть как программно-аппаратные комплексы, устройства типа «черный ящик», так и процессы внутри ядра операционной системы маршрутизатора.

Концептуально маршрутизация сводится к следующей последовательности событий:

- отправитель изучает IP-адрес получателя;
- если адрес не локальный (получатель не находится в той же сети, что и отправитель), он находит для передачи пакета маршрутизатор в нужном направлении;
- приняв пакеты, маршрутизатор повторяет данную операцию. Так продолжается до тех пор, пока пакеты не попадут маршрутизатору, расположенному в той же сети, что и хост-получатель;
- этот конечный маршрутизатор передает пакет получателю - хосту своей сети. Для этого последнего маршрутизатора IP-адрес получателя является локальным.

Вышеописанная последовательность событий носит несколько идеализированный характер. Процесс может осложняться наличием подсетей или повышенных требований к скорости передачи или качеству обслуживания со стороны приложений, но в целом концепция отражает суть маршрутизации.

Конкретная реализация поиска маршрутов и настройки маршрутизации зависит от того, каким именно образом рассматриваемая сеть соединяется с другими сетями. Подключения могут быть разнообразными: простыми, с одним

шлюзом; с подсетью маршрутизаторов, с изменяющейся топологией этой подсети;
с отдельными запрещенными, или, наоборот, предпочтительными маршрутами и т.д. В каждом конкретном случае проблема маршрутизации может быть решена по-своему. *Типичные случаи межсетевых подключений*: сеть соединяется с другой сетью одной линией; сеть с несколькими внешними маршрутами; сеть с несколькими шлюзами; маршрутизация в подсети маршрутизаторов; в маршрутизацию вмешивается «политика». Самый простой случай: сеть соединяется с единственной сетью вовне с помощью единственного маршрутизатора. Система должна иметь представление только лишь о том, что все пакеты с нелокальными адресами получателей направляются этому маршрутизатору. Маршрутизатор должен знать, что все пакеты с внешними адресами, приходящие на его внутренний интерфейс, направляются на его внешний интерфейс. Для этого ядро ОС маршрутизатора содержит информацию о единственном возможном маршруте во все другие сети. При этом «другой сетью» может быть и другая локальная сеть, и Интернет, и даже изолированный хост. Во всех этих случаях маршрутизация будет осуществляться одинаковым образом.

1.5.2. Таблица маршрутов. Виды маршрутизации

Таблица *маршрутизации* – это структура данных ядра, содержащая информацию о маршрутах перенаправляемых пакетов. Управление таблицей маршрутов может выполняться вручную и с помощью специальных алгоритмов. В зависимости от способа заполнения таблицы маршрутизации, различают два вида маршрутизации: статическая маршрутизация и динамическая. *Статическая маршрутизация* - вид маршрутизации, при котором маршруты вручную указываются администратором при настройке маршрутизатора и они находятся в таблице. К достоинствам можно отнести: простоту настройки (в небольших сетях), отсутствие дополнительной нагрузки на сеть (в отличие от динамических протоколов маршрутизации). К недостаткам относится сложность масштабирования, при возникновении каких-либо изменений в сети, как правило, потребуется вмешательство администратора и настройка новых, актуальных статических маршрутов. Если возникают проблемы на канальном уровне, но интерфейс по-прежнему в статусе up, то статический маршрут остается активным, хотя фактически данные передаваться не могут.

В более сложных случаях управление таблицами маршрутов доверяется специальным алгоритмам. Эти алгоритмы обновляют таблицы при изменении в топологии сети связи, включении и выключении маршрутизаторов, перегрузках на маршрутах, при необходимости оптимизировать маршруты в зависимости от их загруженности и т.д. При использовании этих алгоритмов таблицы маршрутов все время изменяются.

Рассмотрим сети с одним внешним маршрутом (рис. 1.19.). В данном случае нет никакой необходимости применять средства динамической маршрутизации, так как существует только один маршрут, о котором и должна существовать запись в таблице маршрутов. Если единственный маршрутизатор сети выйдет из строя, системы будут возвращать ошибку при попытке отправить что-либо в другие сети.

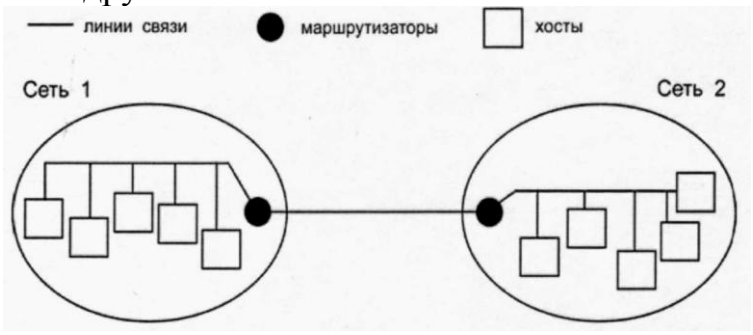


Рис. 1.19. Сеть с одним маршрутом

На рис. 1.20 приведена сеть с несколькими внешними маршрутами. Таблица маршрутов хостов сети содержит одну запись с адресом шлюза по умолчанию, которому и направляются все пакеты с нелокальными IP-адресами. Таблица маршрутов маршрутизатора содержит адреса всех внешних сетей с сопоставленными им сетевыми интерфейсами. Динамическая маршрутизация не требуется, так как к каждой внешней сети ведет только один маршрут.

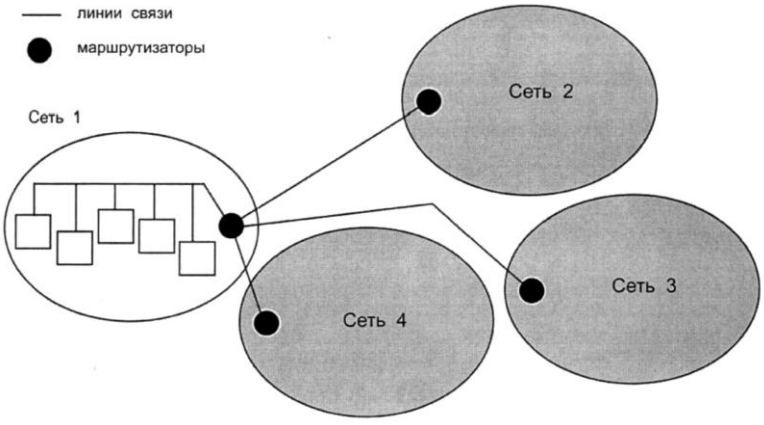


Рис. 1.20. Сеть с несколькими внешними маршрутами

Иногда бывает необходимо иметь несколько внешних подключений. Например, администратор сети, желая разгрузить внешние каналы связи, настраивает три различных подключения к сети Интернет (рис. 1.21). Как в этом случае организовать маршрутизацию в сети?

Рассмотрим вариант при статической маршрутизации. Для различных машин сети можно указать разные шлюзы по умолчанию. Сеть разделится на группы, в каждой из которых будет собственный внешний шлюз. У каждой машины сети в таблице маршрутов содержится единственная запись для

пакетов с IP-адресами внешних сетей. Соединение с внешними сетями у каждого хоста будет происходить всегда через один конкретный шлюз.

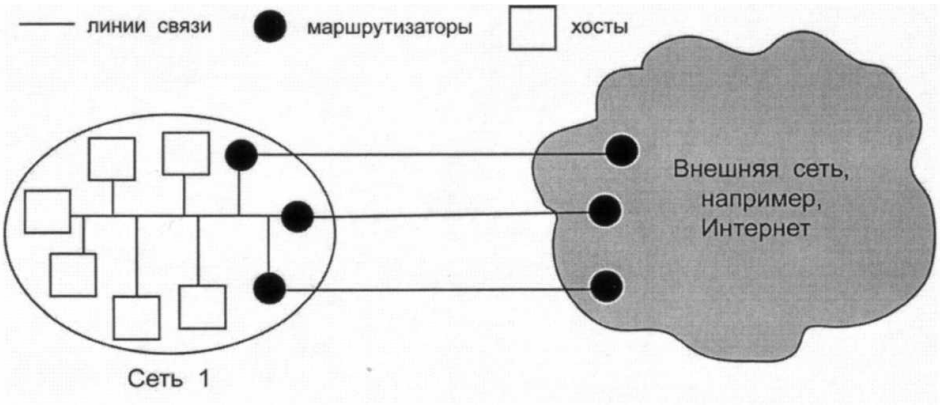


Рис. 1.21. Сеть с несколькими шлюзами

Другой вариант при динамической маршрутизации. Демон маршрутизации, в котором реализован какой-либо алгоритм динамической маршрутизации, сможет выбирать шлюз в зависимости от загруженности того или иного канала, то есть оптимизировать внешние соединения. Каждый шлюз имеет один внутренний интерфейс и один внешний. Получая пакет не с локальным IP-адресом, ядро просто передает его от внутреннего интерфейса шлюза к внешнему.

Основа алгоритма маршрутизации - поиск в таблице маршрутов. Просмотреть таблицы маршрутов можно, введя команды *route* или *netstat -r* . Вывод команд имеет незначительные отличия. Формат вывода таблицы маршрутов приведен ниже:

```
# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref Use Iface
station_aa      *               255.255.255.255 UH 0 0 0 eth0
10.1.0.0        *               255.255.0.0 U 0 0 0 eth0
loopback        *               255.0.0.0 U 0 0 0 lo
default         *               0.0.0.0 UG 0 0 0 eth0
```

Заголовки столбцов и их значение приведены в табл.1.3.

Таблица 1.3

Destination	Адрес получателя, которым может быть маршрутизатор или сеть
Gateway	Шлюз или маршрутизатор, через который проходит маршрут к получателю. Значение * или 0.0.0.0 означает локальную сеть или локальный хост
Genmask	Сетевая маска, маска маршрута
Iface	Интерфейс, через который необходимо посылать пакеты данного маршрута

Flags	Флаги, определяющие состояние маршрута (U - включен, H - определяет соединение между хостами, G - определяет вход на шлюз или маршрутизатор)
Metric	16-битовое значение, используемое некоторыми протоколами динамической маршрутизации. Ядро это поле не использует
Ref	Число активных ссылок на этот маршрут
Use	Число обращений к этой записи

Число в полях Ref и Use отличается от нуля, только если ввести команду: *route -C*.

Смысл записи таблицы маршрутизации - каждый интерфейс (первые четыре записи) имеет собственную запись локального хоста, указывающую на то, что связь с любым локальным интерфейсом производится через локальный же интерфейс. Следующие восемь записей - это записи всех сетей, к которым подключен маршрутизатор. Особенностью таблиц маршрутов семейства ОС UNIX является необходимость иметь по две записи на каждый интерфейс: шлюзовую локальной сети интерфейса. Последняя запись соответствует интерфейсу loopback.

Чтобы система работала в качестве маршрутизатора, ядру необходимо разрешить продвижение пакетов. Для этого достаточно записать 1 в файл */proc/sys/net/ipv4/ip_forward* .

Любой узел, обладающий двумя и более сетевыми соединениями, но не продвигающий пакеты, не является маршрутизатором.

1.5.3. Обзор протоколов маршрутизации

Динамическая маршрутизация - вид маршрутизации, при котором таблица маршрутизации редактируется программно. В случае UNIX-систем - демонами маршрутизации; в других системах - служебными программами, которые называются иначе, но фактически играют ту же роль. Демоны маршрутизации обмениваются между собой информацией, которая позволяет им заполнить таблицу маршрутизации наиболее оптимальными маршрутами. Протоколы, с помощью которых производится обмен информацией между демонами, называются *протоколами динамической маршрутизации.* К протоколам динамической маршрутизации можно отнести протоколы RIP, OSPF, EIGRP, BGP, IS-IS. Демоны динамической маршрутизации: Quagga, GNU Zebra, XORP, Bird. Как правило, демоны динамической маршрутизации поддерживают множество протоколов и используют информацию, полученную по одним протоколам для работы других.

Протоколы динамической маршрутизации можно классифицировать по нескольким критериям:

1) По алгоритмам: *Дистанционно-векторные протоколы* (Distance-vector Routing Protocols) - RIP; *Протоколы состояния каналов связи* (Link-state Routing Protocols) – OSPF, IS-IS. Иногда выделяют третий класс,

усовершенствованные дистанционно-векторные протоколы (advanced distance-vector), для того чтобы подчеркнуть существенные отличия протоколов от классических дистанционно-векторных EIGRP.

2) По области применения: протокол междоменной маршрутизации BGP; протоколы внутридоменной маршрутизации (OSPF, RIP, EIGRP, IS-IS).

Протоколы типа distance vector целесообразно применять в небольших и относительно устойчивых сетях. В больших сетях периодически посылаемые широковещательные пакеты приводят к перегрузке сети. Протоколы типа LSA (алгоритмы обмена о состоянии каналов основаны на динамическом построении маршрутизаторами карты топологии сети за счет сбора информации обо всех объединяющих их каналах связи) используются в больших и быстрорастущих сетях. Самой распространенной реализацией LSA является протокол *OSPF* - открытый стандарт, разработанный для применения в маршрутизаторах сети Интернет и широко используемый в других сетях (XNS, SNA, DECNet). Помимо всех преимуществ алгоритмов LSA, протокол OSPF обеспечивает маршрутизацию пакетов с заказанным типом обслуживания, равномерное распределение нагрузки между альтернативными путями одинаковой стоимости, маршрутизацию пакетов в соответствии с классом обслуживания (эквивалентным классом доставки), аутентификацию маршрутов, создание виртуального канала между маршрутизаторами, соединенными не напрямую, а через некоторую транзитную сеть.

1.5.4. Введение в алгоритмы динамической маршрутизации

К алгоритмам маршрутизации предъявляются следующие требования: корректность (алгоритм маршрутизации должен уметь справляться с аппаратными проблемами, изменениями топологии и трафика); простота; надежность; устойчивость (так как алгоритмы используются в управляющих программах с обратными связями, они должны уметь приходить в состояние равновесия); справедливость; оптимальность. Такие цели, как оптимальность и справедливость могут показаться очевидными, между тем они часто взаимно исключают друг друга. Допустим, что трафик между станциями A и A′, B и B′, C и C′ (рис. 1.22) настолько интенсивный, что горизонтальные линии связи полностью насыщенны. Чтобы максимизировать поток данных, следует станции X и X′ полностью отключить. Однако у них может быть другое мнение на этот счет. Очевидно, необходим компромисс между справедливым выделением трафика и производительностью сети.

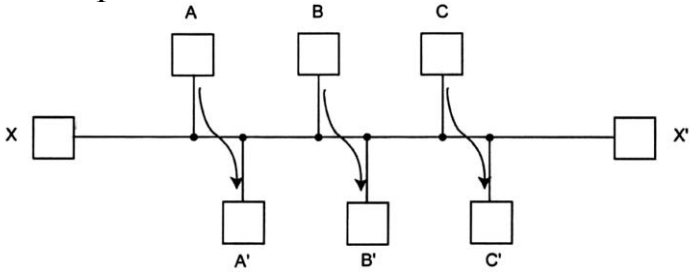


Рис. 1.22. Пример компромисса

Принцип оптимальности является основой для построения алгоритма маршрутизации. Если маршрутизатор B располагается на оптимальном маршруте от маршрутизатора A к маршрутизатору C, то оптимальный маршрут от B к C совпадает с частью этого маршрута. Данное заключение может быть строго доказано [5]. Первым следствием принципа оптимальности является возможность рассмотреть множество оптимальных маршрутов от источников к приемникам в виде графа, который называется *входным деревом* (рис. 1.23).

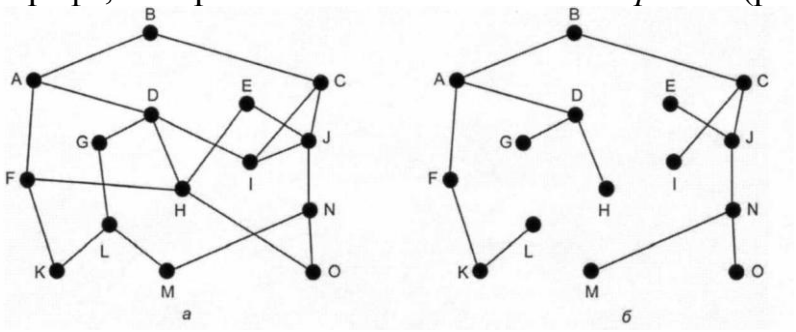


Рис. 1.23. Пример графа «Входное дерево»

Графом подсети связи считается узел, соответствующий маршрутизатору. Дуга соответствует линии связи. Поскольку входное дерево действительно является деревом, оно не содержит петель, поэтому каждый пакет будет доставлен за конечное и ограниченное число пересылок.

При выборе маршрута между двумя узлами алгоритм находит кратчайший путь между ними на графе. Один из способов измерения длины пути состоит в подсчете количества транзитных участков (рис. 1.24).

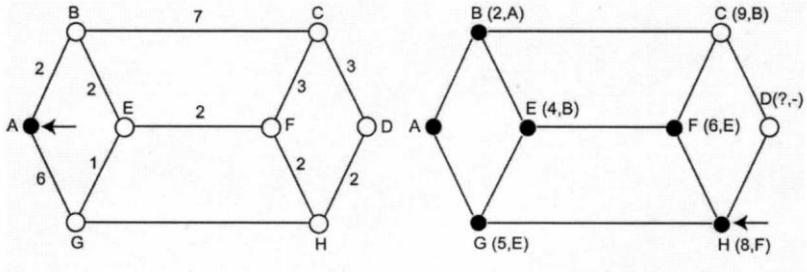


Рис. 1.24. Пример выбора маршрута

В таком случае пути *ABC* и *ABE* имеют одинаковую длину. Если же измерить расстояние в километрах, может оказаться, что путь *ABC* значительно длиннее пути *ABE*.

Помимо учета длины участков и количества переходов возможен и учет других параметров. Например, каждой дуге графа можно сопоставить среднюю длину очереди и время задержки пересылки, определяемую каждый час специальным тестовым пакетом. В таком графе кратчайший путь определяется как самый быстрый, а не как путь с самым коротким кабелем или с наименьшим количеством отрезков кабеля.

Параметры дуг графа являются функциями расстояния, пропускной способности, средней загруженности, стоимости связи, средней длины очереди,

измеренной величины задержки и др. Изменяя весовую функцию, алгоритм может вычислять кратчайший путь с учетом любого количества критериев в различных комбинациях.

Один из ряда известных алгоритмов измерения кратчайшего пути предложен Эдсгером *Дейкстрой* в 1959 г. Каждый узел графа помечается расстоянием от него до узла отправителя по наилучшему пути. Вначале пути неизвестны, поэтому все узлы помечаются символом бесконечности. По мере работы алгоритма и нахождения путей отметки узлов изменяются, показывая оптимальные пути. Отметка может быть постоянной или экспериментальной, вначале все отметки ориентировочны. Когда выясняется, что отметка действительно соответствует кратчайшему пути, она становится постоянной и в дальнейшем не изменяется. Алгоритм работает итерационно, кратчайший путь к соседним узлам находится на каждом узле и совокупность лучших путей на каждом промежуточном узле складывается в наилучший маршрут.

Другой известный алгоритм - *алгоритм заливки*. Каждый приходящий пакет посылается на все исходящие линии кроме той, по которой он пришел. Алгоритм заливки порождает огромное количество дублированных пакетов, даже бесконечное количество, если в сети есть петли. Применение алгоритма требует специальных мер: в заголовок пакета помещается счетчик преодоленных им транзитных участков, уменьшающийся при прохождении каждого маршрутизатора; когда значение счетчика упадет до нуля, пакет уничтожается; альтернативный способ контроля тиражирования пакетов состоит в учете проходящих через маршрутизатор пакетов, что позволяет не посылать их повторно. На практике чаще применяют его вариант - *выборочная заливка*. Маршрутизаторы посылают пакеты не по всем линиям, а только по линиям, примерно соответствующим направлению движения пакета. Например, нет смысла посылать на запад пакет, который двигается на восток. В большинстве случаев алгоритм заливки не практичен, однако, есть ряд случаев, когда его применение оправдано, например, в критичных приложениях, где высокая надежность алгоритма заливки является, наоборот, желательной. Кроме того, с помощью этого алгоритма тестируют другие алгоритмы, так как он находит все возможные пути в сети, а следовательно, и кратчайшие.

Алгоритм маршрутизации по вектору расстояний иногда называют по именам его создателей *распределенным алгоритмом Беллмана-Форда* и *алгоритмом Форда-Фулкерсона*. Изначально данный алгоритм применялся в сети ARPANET и известен в Интернет как протокол маршрутизации RIP (Routing Information Protocol). При работе данный алгоритм опирается на таблицы (векторы), которые поддерживаются всеми маршрутизаторами и содержат наилучшие известные пути к каждому из адресатов. Для обновления данных этих таблиц производится обмен информацией с соседними маршрутизаторами. Таблицы содержат записи о каждом маршрутизаторе подсети связи. Каждая запись состоит из двух частей: предпочитаемого номера линии для данного получателя; предполагаемого расстояния или времени

прохождения пакета до этого получателя. В качестве единиц измерения может использоваться число транзитных участков, миллисекунды, число пакетов, ожидающих в очереди на данном направлении и т.п. Предполагается, что расстояние до каждого из соседей маршрутизаторам известно:

- если в качестве единицы измерения используют число транзитных участков, то расстояние равно одному транзитному участку;
- если используется время задержки, то маршрутизатор измеряет его с помощью специального эхо-пакета, в который получатель помещает время получения и отправляет его назад возможно быстрее.

Пусть в качестве единицы измерения используется время задержки, и этот параметр относительно каждого из соседей известен каждому маршрутизатору. Через каждые t миллисекунд все маршрутизаторы посылают своим соседям список с приблизительными задержками для каждого получателя. Такие же списки они получают и от соседей. Зная время задержки до соседа и получив от него задержки до его соседей, маршрутизатор выполняет расчеты для всех узлов сети и заносит их в новую таблицу наилучших маршрутов. Старая таблица при этом не используется. Если какой-либо узел вышел из строя, все помечают его как имеющего бесконечную задержку, т.е. недоступный.

Алгоритм Беллмана-Форда быстро реагирует на появление новых маршрутизаторов, однако его большой недостаток - очень медленная реакция на выключение маршрутизаторов. Это связано с тем, что маршрутизаторы не могут устанавливать значение расстояний, более чем на единицу превышающее минимальное значение этого расстояния, хранящегося у его соседей. Поэтому при обрыве единственного пути к какому-либо узлу его соседи будут искать ход через другие узлы. Данная проблема получила название проблемы счета до бесконечности. В реальных протоколах это решается установкой предельного значения расстояния (или задержки), после превышения которого узел помечается как недоступный.

С ростом пропускной способности линий передачи и повышения требований к обслуживанию маршрутизация с учетом состояния линий вытесняет маршрутизацию по вектору расстояний [5]. Алгоритм основан на определенных требованиях к маршрутизаторам, которые должны: обнаруживать своих соседей и узнавать их сетевые адреса; измерять задержку или стоимость связи с каждым из своих соседей; создавать пакет, содержащий всю собственную информацию; посылать этот пакет всем маршрутизаторам; вычислять кратчайший путь ко всем маршрутизаторам. В результате выполнения этих требований каждому маршрутизатору высылается полная топология и все измеренные значения задержек. После этого для обнаружения кратчайшего пути к каждому маршрутизатору может локально применяться алгоритм Дейкстры.

Все обычные законы, касающиеся фиксированной топологии, известных соседей, взаимосвязи между IP-адресом и положением, в мобильных

специализированных сетях перестают работать. Маршрутизаторы могут легко появляться в системе и также легко из нее исчезать, появляясь в каком-то другом месте. В обычных сетях путь от маршрутизатора к адресату продолжает оставаться реализуемым до тех пор, пока не произойдет какой-либо сбой системы. В мобильных сетях топология постоянно меняется, а с ней меняется и предпочтительность (и даже реализуемость) путей.

В отличие от классических протоколов маршрутизации Интернета, являющихся превентивными, то есть находящих пути маршрутизации независимо от использования маршрутов, в мобильных сетях используют протокол AODV (Ad hoc On-Demand Distance Vector), который является реактивным протоколом маршрутизации. Он устанавливает маршрут до адресата по требованию. Маршрут вычисляется только в тот момент, когда появляется желающий отправить пакет тому или иному адресату. Данный протокол принимает в расчет ограниченность пропускной способности и срока службы элементов питания (свойства, характерные для мобильных сетей). Как недостаток следует отметить, что в AODV в начале коммутации требуется больше времени на установку маршрута, чем во многих других протоколах.

Контрольные вопросы

1. Что понимается под маршрутизацией?
2. Перечислите типичные случаи межсетевых подключений.
3. Перечислите основные виды и протоколы маршрутизации.
4. В чем отличие статической и динамической маршрутизации?
5. Поясните формат вывода таблицы маршрутов.
6. Назовите алгоритмы измерения кратчайшего пути.
7. Поясните суть алгоритм Беллмана-Форда.
8. Поясните суть алгоритм Дейкстры.
9. Назовите отличие протокола AODV от классических протоколов маршрутизации.

ЧАСТЬ 2. ИНТЕРНЕТ-ТЕХНОЛОГИИ И СТАНДАРТЫ ГЛОБАЛЬНОЙ СЕТИ НА ПРИМЕРЕ ИНТЕРНЕТ

2.1. Стандарты и протоколы

В компьютерных сетях используется два основных понятия, которые постоянно будут встречаться: адрес и протокол. Свой уникальный адрес имеет любой компьютер (точнее, сетевой вход), подключенный к Интернет. Даже при временном соединении компьютеру выделяется уникальный адрес. В любой момент времени все компьютеры, подключенные к Интернет, имеют разные адреса. Стандартные протоколы позволяют разным компьютерам «говорить» на одном языке, таким образом, обеспечивается возможность подключения к Интернет разнообразных компьютеров, работающих под управлением различных операционных систем. Описать в одном протоколе все правила взаимодействия практически невозможно. Поэтому сетевые протоколы строятся по многоуровнему принципу.

2.1.1.Технологии доступа к среде: ISDN, X25, Frame Relay, ATM

ISDN (*Integrated Services Digital Network*) - цифровая сеть с интегрированными услугами. ISDN использует телефонные линии для передачи двоичных цифровых сигналов. Подключение ISDN производится одновременно к 2-м телефонным линиям, по одной из которых идет передача информации, а по другой - ее прием (дуплексная связь). Допустимая скорость передачи данных без сжатия 128 Кбит/с. При использовании сжатия данных может быть достигнута значительно большая скорость. ISDN - мощное средство связи с Интернет, однако пока оно является малодоступным для широкого круга пользователей [6].

Спецификация *X.25* определяет двухточечное взаимодействие между терминальным оборудованием (DTE) и оборудованием завершения действия информационной цепи (DCE). Устройства DTE (терминалы и главные вычислительные машины в аппаратуре пользователя) подключаются к устройствам DCE (модемы, коммутаторы пакетов и другие порты в сеть PDN, обычно расположенные в аппаратуре этой сети), которые соединяются с "коммутаторами переключения пакетов" (*packet switching exchange*) (PSE или просто *switches*) и другими DCE внутри PSN и, наконец, к другому устройству DTE.

Спецификация X.25 составляет схемы Уровней 1-3 эталонной модели OSI. Уровень 3 X.25 описывает форматы пакетов и процедуры обмена пакетами между равноправными об'ектами Уровня 3. Уровень 2 X.25 реализован Протоколом *Link Access Procedure, Balanced* (LAPB). LAPB определяет кадрирование пакетов для звена DTE/DCE. Уровень 1 X.25 определяет электрические и механические процедуры активации и дезактивации физической среды, соединяющей данные DTE и DCE. Необходимо отметить,

что на Уровни 2 и 3 также ссылаются как на стандарты ISO - ISO 7776 (LAPB) и ISO 8208 (пакетный уровень X.25).

Технология *frame relay* изначально рассчитывалась как высокоскоростная технология для территориальных сетей, предназначенная для передачи чувствительного к задержкам трафика [7], (например, видео- и аудиопотоки), что и стало главным фактором высокой востребованности данной технологии. Технология frame relay использует для передачи данных технику виртуальных соединений, аналогичную той, которая применялась в сетях X.25, однако, стек протоколов frame relay передает кадры (при установленном виртуальном соединении) по протоколам только физического и канального уровней, в то время как в сетях X.25 и после установления соединения пользовательские данные передаются протоколом 3-го уровня.

Кроме того, протокол канального уровня LAP-F в сетях frame relay имеет два режима работы: основной (core) и управляющий (control). В основном режиме, который физически практикуется в сегодняшних сетях frame relay, кадры передаются без преобразования и контроля, как и в коммутаторах локальных сетей. За счет данной особенности технология frame relay обладает высокой производительностью, а сеть не передает квитанции подтверждения между коммутаторами на каждый пользовательский кадр, как это происходит в сети X.25. Пульсации трафика передаются достаточно быстро и без больших задержек. При таком подходе уменьшаются накладные расходы при передаче пакетов локальных сетей, так как они вкладываются сразу в кадры канального уровня, а не в пакеты сетевого уровня, как это происходит в сетях, построенных на базе технологии X.25.

Транспортная технология ATM уже несколько лет успешно используется в магистральных сетях общего пользования и в корпоративных сетях, а сейчас ее начинают активно использовать и для высокоскоростного доступа по каналам xDSL (для небольших офисов) и SDH/Sonet (для крупных предприятий). Главные преимущества этой технологии - ее зрелость, надежность и наличие развитых средств эксплуатационного управления сетью. В ней имеются непревзойденные по своей эффективности механизмы управления качеством обслуживания и контроля использования сетевых ресурсов. Однако ограниченная распространенность и высокая стоимость оборудования не позволяют считать ATM лучшим выбором для организации сквозных телефонных соединений от одного конечного узла до другого.

Подобно технологии ретрансляции кадров (frame relay) и X.25, протоколы ATM ориентированы на предварительное установление соединений. Сеансы ATM реализуются в виртуальных сетях связи. Большинство, если не все сегодняшние службы ATM предлагают только постоянные виртуальные каналы связи (permanent virtual circiut, PVC); установление и разрыв виртуальных каналов связи входит в обязанности телефонной компании, если только сеть ATM не является полностью частной. Выделить нужную полосу пропускания по первому требованию будет на практике выполнено только

тогда, когда станут доступны коммутируемые виртуальные каналы связи (switched virtual circuits, SVC). Постоянные виртуальные каналы схожи с выделенными линиями, в то время как коммутируемые виртуальные каналы связи сравнимы с коммутируемой звуковой связью. Однако для установления ATM SVC требуются только доли секунды.

При своей ориентации на установление соединений ATM не очень-то похожа на протоколы, ориентированные на разделяемые среды, такие, как Ethernet и Token Ring, или на протоколы маршрутизации, не требующие предварительного установления соединений: IP или IPX. С развитием стандартов эмуляции локальной сети услуги ATM станут доступны для сетей Ethernet и Token Ring. Объявлено о создании продуктов для транслирования данных из frame relay в ATM. IP и протокол определения адреса для ATM описаны в RFC15776. Таким образом, ATM пригодна для протоколов уровней канала данных и физического, но поскольку для ориентированных на соединение протоколов не нужна маршрутизация, то ATM может непосредственно работать с протоколами верхних уровней.

Верхним уровнем стека протоколов ATM является слой адаптации ATM (AAL). Различные AAL соответствуют различным поддерживаемым ATM типам данных. Так, AAL1 позволяет устройству ATM быть похожим на звуковую линию связи с постоянной скоростью передачи; AAL3/4 и AAL5 используются для типов данных с переменной скоростью передачи, то есть таких, которые обычно встречаются в компьютерных сетях. AAL также предназначен для интегрирования ориентированного на соединение ATM с источниками данных, работающими без установления соединения, что позволяет клиентам ATM эмулировать радиовещание и т.п.

Уровень ATM является ядром технологии. Имеется множество AAL и множество вариантов физических уровней, но протокол, который описывает состав заголовка ячейки и определяет действие коммутаторов над ячейками, для всех один и тот же. Уровень ATM отвечает за маршрутизацию ячейки, мультиплексирование и демультиплексирование. Прежде чем какие-либо данные пользователя отправятся в путь по виртуальному каналу связи ATM, каждый промежуточный коммутатор должен создать строку в локальной таблице маршрутизации, которая задает соответствие между идентификатором входящего виртуального канала и портом выхода.

Для того чтобы снизить накладные расходы на маршрутизацию через промежуточные соединительные устройства, ATM определяет виртуальные пути (VP), которые по сути являются виртуальными каналами, определенными для двух или более физических соединительных устройств, рассматриваемых как единое целое. Виртуальные пути являются полупостоянными соединениями, и для них таблицы маршрутизации могут быть созданы заранее. Нет необходимости заниматься маршрутизацией следующего по VP пакета в каждом из промежуточных узлов виртуального пути.

Определив тип данных, составляющих ячейку, уровень ATM объединяет в единый поток потоки данных в зависимости от приоритета каждого типа. Он также отвечает за обнаружение перегрузок, обработку сбоев и управление трафиком.

Адресация в технологии ATM осложняется тем, что Форумом ATM определено четыре различных формата адресов. Для частных сетей ATM Форум определил три типа адресов конечных систем (ATM End System Addresses, AESA): DCC AESA, ICD AESA и E.164 AESA. В сетях ATM общего пользования выбор состоит между исходным форматом адреса E.164 и тремя адресами AESA, указанными выше. Кроме того, эти форматы могут использоваться совместно.

2.1.2. Основные принципы работы и возможности сети Интернет

Отличительной особенностью Интернет является высокая надежность. При выходе из строя части компьютеров и линий связи сеть будет продолжать функционировать. Такая надежность обеспечивается тем, что в Интернет нет единого центра управления. Как и любая другая компьютерная сеть, Интернет состоит из множества компьютеров, соединенных между собой линиями связи, и установленных на этих компьютерах программ. Но есть и некоторые особенности, присущие только Интернет.

Рассмотрим структуру сети на примере условной схемы соединения компьютеров части Интернет (рис. 2.1). Пользователи Интернет подключаются к сети через компьютеры *специальных организаций,* которые называются поставщиками услуг Интернет - провайдерами (от английского слова *Provider*).

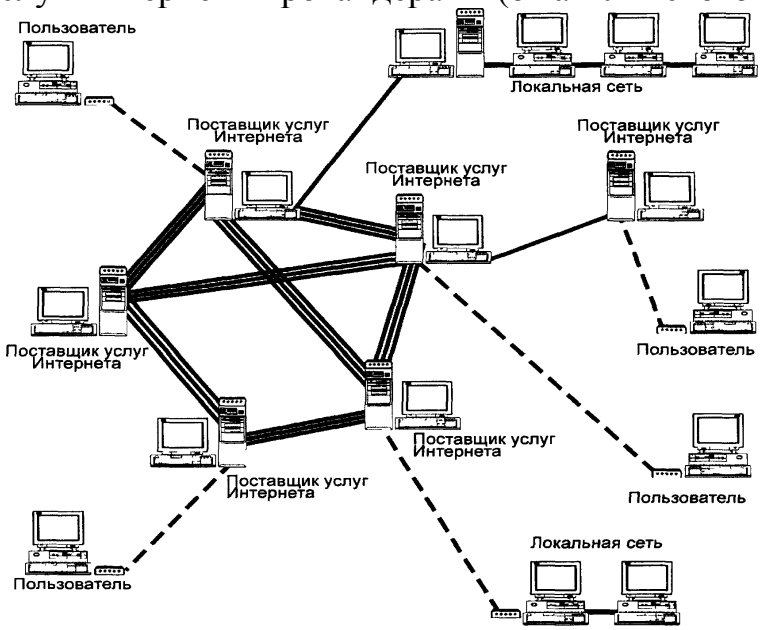


Рис. 2.1. Схема соединения компьютеров в Интернет:

Высокоскоростные выделенные линии связи

Выделенные линии связи

Коммутируемые линии связи

Модемы для подключения компьютеров к линиям связи

К сети могут быть подключены как отдельные компьютеры, так и локальные сети. В последнем случае можно считать, что к Интернет подключены все компьютеры данной локальной сети, хотя линией связи с Интернет соединен только один компьютер. Соединение может быть постоянным или временным. Провайдеры имеют множество линий для подключения пользователей и высокоскоростные линии для связи с остальной частью Интернет. Часто мелкие поставщики подключены к более крупным, которые, в свою очередь, подключены к другим поставщикам. Все организации, соединенные друг с другом высокоскоростными линиями связи, образуют хребет Интернет. Если провайдер подключен непосредственно к хребту (становой магистрали), то скорость передачи информации будет максимальной.

2.1.3. Стек протоколов TCP/IP

Преимуществом сети со стеком протоколов TCP/IP является возможность объединения компьютеров с разной архитектурой и операционными системами. Сеть с TCP/IP это активная, быстроразвивающаяся технология. Постоянно реализуются и появляются новые идеи, а это влечет за собой появление новых возможностей сетей TCP/IP.

Модель ISO/OSI всего лишь руководство к действию, а не готовая конструкция. Так, например, стек TCP/IP состоит всего из пяти, а не семи уровней ISO/OSI. На нижнем уровне используется два основных протокола: IP и TCP. Так как эти два протокола тесно взаимосвязаны, то часто их объединяют и говорят, что в Интернет базовым протоколом является TCP/IP.

Работу протоколов TCP/IP можно объяснить с помощью достаточно условного примера. Предположим, требуется передать информацию с одного компьютера, подключенного к Интернет, на другой компьютер. Протокол TCP разбивает информацию на пакеты и нумерует их так, чтобы при получении можно было правильно собрать информацию.

Далее с помощью протокола IP все части передаются получателю. Так как отдельные пакеты могут путешествовать по Интернет по различным маршрутам, то порядок их прихода может быть нарушен. После получения всех пакетов TCP располагает их в нужном порядке и собирает в единое целое.

Для протокола TCP не имеет значения, какими путями информация перемещается по Интернет. Этим занимается протокол IP. К каждой полученной порции информации протокол IP добавляет служебную информацию, из которой можно узнать адрес отправителя и получателя информации.

Так в самых общих чертах работают протоколы TCP/IP. Они обеспечивают передачу информации между двумя компьютерами. Все остальные протоколы (табл. 2.1) с их помощью реализуют самые разные услуги Интернет, речь о которых пойдет в следующих разделах.

Таблица 2.1

Соответствие уровней модели и протоколов

Номер уровня	**Название**	Примеры протоколов
7	**Прикладной**	HTTP, SMTP, SNMP, FTP, Telnet, scp, NFS, RTSP
6	**Представительный**	XML, XDR, ASN.1, SMB, AFP
5	**Сеансовый**	TLS, SSH, ISO 8327 / CCITT X.225, RPC, NetBIOS, ASP
4	**Транспортный**	TCP, UDP, RTP, SCTP, SPX, ATP, DCCP, BGP, GRE
3	**Сетевой**	IP, ICMP, IGMP, CLNP, ARP, RARP, OSPF, RIP, IPX, DDP
2	**Канальный**	Ethernet, Token ring, PPP, HDLC, X.25, Frame relay, ISDN, ATM, MPLS
1	**Физический**	электричество, радио, лазер

Контрольные вопросы

1. Что понимается под протоколом?
2. Какие технологии гарантированно обеспечивают среднюю скорость передачи данных по виртуальному каналу при допустимых пульсациях трафика?
3. Каково назначение постоянных виртуальных каналов?
4. В чем отличие их от коммутируемых виртуальных каналов?
5. Поясните работу протоколов TCP/IP.

2.2. Адресация в IP-сетях

2.2.1. Адресация в IP-сетях. Типы адресов

Каждый компьютер в сети TCP/IP имеет адреса трех уровней:

1) Локальный адрес узла, определяемый технологией, с помощью которой построена отдельная сеть, в которую входит данный узел. Для узлов, входящих в локальные сети - это MAC-адрес (Media Access Control - контроль доступа к среде) сетевого адаптера или порта маршрутизатора, например, 11-AO-17-3D-BC-01. Эти адреса назначаются производителями оборудования и являются уникальными адресами, так как управляются централизовано. Для всех существующих технологий локальных сетей MAC-адрес имеет формат 6 байтов: старшие 3 байта - идентификатор фирмы производителя, а младшие 3

байта назначаются уникальным образом самим производителем. Для узлов, входящих в глобальные сети, такие как X.25 или frame relay, локальный адрес назначается администратором глобальной сети.

2) IP-адрес, состоящий из 4-х байт, например, 194.226.40.1. Этот адрес используется на сетевом уровне. Он назначается администратором во время конфигурирования компьютеров и маршрутизаторов. IP-адрес состоит из двух частей: номера сети и номера узла. Номер сети может быть выбран администратором произвольно, либо назначен по рекомендации специального подразделения Internet NIC, если сеть должна работать как составная часть Интернет. Обычно провайдеры услуг Интернет получают диапазоны адресов у подразделений NIC, а затем распределяют их между своими абонентами. Номер узла в протоколе IP назначается независимо от локального адреса узла. Узел может входить в несколько IP-сетей. В этом случае узел должен иметь несколько IP-адресов, по числу сетевых связей. Таким образом, *IP-адрес* характеризует не отдельный компьютер или маршрутизатор, а одно *сетевое соединение* (сетевой интерфейс).

3) Символьный идентификатор-имя, например, MSTUCA.RU. Этот адрес назначается администратором и состоит из нескольких частей, например, имени машины, имени организации, имени домена. Такой адрес, называемый также DNS-именем, используется на прикладном уровне, например, в протоколах FTP или telnet.

2.2.2. Проблемы адресации в IP-сетях

Адресация в сетях имеет большое значение. Учитывая ускоряющийся рост сети, ограниченность адресного пространства, использующего 32-х битный IP-адрес, является слабым местом протокола IPv4. Другой проблемой стала недостаточная масштабируемость маршрутизации - процедуры, составляющей суть IP-сетей. Обработка таблицы маршрутизации, содержащей информацию о нескольких десятках тысяч номеров сетей, вызывает перегрузку мартрутизатора. В результате сообщество Интернет выбрало в качестве новых целей модернизации протокола IP: создание масштабируемой схемы адресации; повышение пропускной способности сети за счет сокращения работ, выполняемых маршрутизаторами; обеспечение защиты данных, передаваемых по сети; предоставление гарантий качества транспортных услуг.

Для решения этой проблемы разработан протокол IP нового поколения Next Generation Internet Protocol, IPng - IPv6, в котором для IP- адреса используется 128 бит. Стандарт IPv6 является прямым преемником четвертой версии Интернет-протокола. Разрыв в последовательности номеров объясняется тем, что согласно рекомендациям RFC 1819 версия 5 была зарезервирована для экспериментального потокового протокола (streaming protocol). С начала 1990-х гг., когда стало ясно, что из-за расширения Интернет в обозримом будущем появятся различные проблемы, комитет Internet Engineering Task Force (IETF) начал работу над протоколом-преемником (RFC 2460). Наряду с ликвидацией ограничений по

объему адресного пространства, были предусмотрены дальнейшие усовершенствования и оптимизация IP-протокола [8].

2.2.3. Методы перехода от IPv4 к IPv6

Разработчики IPv6 стремились предусмотреть механизмы плавного перехода с IPv4. Под *переходом* (transition) IETF понимает любые методы перевода оконечных устройств, маршрутизаторов и сетей существующей инфраструктуры, базирующейся на IPv4, на новую версию IPv6. При этом первоочередная цель - обеспечить дальнейшее взаимодействие IPv6 совместимых устройств с традиционным IPv4-оборудованием.

В настоящее время из многообразия подходящих вариантов перехода можно выделить: методы *на основе двойного набора протоколов* (dual stack): сетевые компоненты снабжены *двумя стеками протоколов*, т. е. поддерживают как IPv4, так и IPv6; методы *туннелирования*, предусматривающие упаковку данных IPv6 в заголовки IPv4 (и наоборот): становится возможной передача трафика по сетям, которые сами по себе тот или иной «туннелируемый» протокол не поддерживают; *трансляция IPv4 в IPv6* и обратно, обеспечивающая взаимодействие оборудования IPv6 с системами на основе IPv4.

2.2.4. Особенности адресации IPv6. Форма записи

Адрес IPv6 состоит из 128 бит или 16 байт. Главной целью изменения системы адресации было не механическое увеличение разрядности адреса, а изменение его функциональности за счёт введения новых полей.

Вместо прежних двух уровней иерархии адреса (номер сети и номер узла) IPv6 предлагается использовать четыре уровня, включая трехуровневую идентификацию сетей, и один уровень для идентификации узлов сети. За счет увеличения числа уровней иерархии в адресе новый протокол эффективно поддерживает технологию агрегирования адресов CIDR (Clasless InterDomain Rouning). *Бесклассовая адресация CIDR* - метод IP-адресации, позволяющий гибко управлять пространством IP-адресов, не используя жёсткие рамки классовой адресации. Суть этой технологии состоит в том, что каждому провайдеру выделяется непрерывный диапазон в пространстве IP-адресов (напомним, что в IPv4 существует 5 классов адресов (A, B, C, Д, E)). Бесклассовая адресация основывается на переменной длине маски подсети VLSM (Variable Length Subnet Mask), в то время, как в классовой адресации длина маски строго фиксирована 0,1, 2 или 3 установленными байтами. Благодаря этому, а также усовершенствованной системе групповой адресации и введению адресов типа *anycast,* новая версия IP позволяет уменьшить затраты на маршрутизацию.

Для записи адресов используется компактная нотация. Адреса представлены как 8 шестнадцатеричных чисел, разделенных двоеточиями. Каждое шестнадцатеричное число представляет 16 бит. Например:

32BD:0000:10A12:0006:0001:DDE1:8006:2334

Ведущие нули в шестнадцатеричных полях могут быть опущены (например, 0 вместо 0000 и 6 вместо 0006 и т.д.):

32BD:0:10A12:6:1:DDE1:8006:2334

Формат может быть еще более сжат при замене последовательности смежных нулевых полей на "::". Например, адрес *32BD:0:0:0:1:DDE1:8006:2334* может быть заменен на адрес *32BD::1:DDE1:8006:2334.* Сокращение «::» можно использовать только один раз.

2.2.5. Типы адресов. Соглашения о специальных адресах

Протоколом IPv6 определено три основных типа адресов: unicast; multicast; anycast. Тип адреса задается значением нескольких старших бит адреса, которые названы *префиксом формата.*

Адрес типа *unicast* определяет уникальный идентификатор отдельного *интер*фейса конечного узла или маршрутизатора. Назначение этого типа адреса совпадает с назначением уникальных адресов в версии IPv4 - с их помощью пакеты доставляются определенному интерфейсу узла назначения. В версии IPv6, в отличие от версии IPv4, как говорилось ранее, отсутствует понятие класса сети и связанное с ним фиксированное разбиение адреса на номер сети номер узла по границам байт. Адреса типа unicast делятся на несколько подтипов для отражения специфики некоторых ситуаций, часто встречающихся в современной мультисервисной сети.

Адрес типа *multicast* - групповой адрес, аналогичный по назначению групповому адресу. Он имеет префикс формата 1111 1111 и идентифицирует группу интерфейсов, относящихся, как правило, к разным узлам. Пакет с таким адресом доставляется всем интерфейсам с этим адресом. Адреса типа multicast используются в IPv6 и для замены широковещательных адресов. Для этого вводится особый адрес группы, объединяющей все интерфейсы сети. В версии IPv6 групповой адрес имеет признак обзора (scope), отсутствовавший в групповом адресе версии IPv4. Этот признак позволяет гибко задавать область действия группового адреса. Она может представлять собой, например, только одну подсеть, либо все подсети данного предприятия, либо весь Интернет. Это упрощает работу маршрутизаторов, которым необходимо выявить все узлы, относящиеся к какой-либо группе. Еще один признак задает тип группы - постоянная или временная.

Адрес типа типа *anycast* - это новый тип адреса, который так же, как и multicast, определяет группу интерфейсов. Но пакет с таким адресом доставляется из интерфейсов группы, как правило, "ближайшему" в соответствии с метрикой, используемой протоколами маршрутизации. Синтаксически anycas адрес ничем не отличается от адреса типа unicast, он назначается из того же диапазона адресов, что и unicast- адреса. Следует подчеркнуть, что адрес типа anycast может быть назначен только интерфейсам

маршрутизатора. Интерфейсы маршрутизаторов, входящие в одну anycast-группу, имеют индивидуальные unicast-адреса и, кроме того, общий anycast-адрес. Адреса такого типа ориентированы на применение маршрутизации от источника (Source Routing), когда маршрут прохождения пакету определяется узлом-отправителем путем указания IP-адресов всех промежуточных маршрутизаторов.

128-разрядное пространство IP-адреса версии 6 обеспечивает место для множества различных типов адресов, включая: иерархические глобальные одноадресные рассылки на основе адресов провайдеров; иерархические глобальные одноадресные рассылки по географическому признаку; личные адреса сайтов для использования только в пределах организации; локальные и глобальные многоадресные рассылки.

Для внутрикорпоративного использования Оргкомитет Интернет выделил так называемые *Intranet IP-numbers*:

- одну сеть класса А с номерами 10.*.*.* ;
- 16 сетей класса В с номерами 172.[16…31].*.* ;
- 256 сетей класса С с номерами 192.186.*.* .

В отличие от четвертой версии, в шестой эти адреса имеют специальный формат. Адреса для локального использования представлены в IPv6 двумя разновидностями. Во-первых, это адреса для сетей, не разделенных на подсети (не использующие маршрутизацию). Они называются Link-Local (локальных связей). Во вторых, это адреса локального использования для сетей, разделенных на подсети. Такие адреса называются Site-Local (локальных сайтов).

Адреса Link-Local имеют формат:

1111111010 (10 бит) 00…00 Уникальный адрес технологии связи

Как видно из приведенного формата, адрес Link-Local имеет 10-битный префикс и содержит только 64-разрядное поле идентификатора интерфейса. Остальные разряды, кроме префикса, должны быть нулевыми, поскольку потребность в номерах подсети отсутствует. Адреса локальных связей весьма полезны при инициализации.

В протоколе IPv6 существует несколько заголовков. Между заголовком IPv6 и заголовком верхнего уровня можно вставить несколько дополнительных заголовков для необязательных вариантов, подобных маршрутизации от источника или поддержке безопасности. Фрагментация также может быть перенесена в дополнительные заголовки. Использование дополнительных заголовков (extension header) в IPv6 - прогрессивная идея, позволяющая последовательно добавлять новые функциональные возможности. Некоторые из них содержат информацию, которую следует обрабатывать на каждом узле по пути следования пакета, в то время как другие заголовки обрабатываются только в точке назначения.

2.2.6. Протокол DHCP

Основным назначением протокола DHCP (Dynamic Host Configuration Protocol) является динамическое назначение IP-адресов. Однако DHCP может поддерживать и более простые способы ручного и автоматического статического назначения адресов. В ручной процедуре назначения адресов принимает участие администратор, который предоставляет DHCP-серверу информацию о соответствии IP-адресов физическим адресам. При автоматическом статическом способе DHCP-сервер присваивает IP-адрес (и, возможно, другие параметры конфигурации клиента) из пула наличных IP-адресов без вмешательства оператора. При динамическом распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время, что дает возможность впоследствии повторно использовать IP-адреса другими компьютерами. Это частично решает проблему ограниченного числа IP-адресов. Динамическое распределение адресов позволяет строить IP-сеть, количество узлов в которой намного превышает количество имеющихся в распоряжении администратора IP-адресов. Администратор управляет процессом назначения адресов с помощью параметра "продолжительность аренды" (lease duration), которая определяет, как долго компьютер может использовать назначенный IP-адрес, перед тем как снова запросить его от сервера DHCP в аренду.

Контрольные вопросы

1) Перечислите основные цели модернизации протокола IP.
2) Поясните суть технологии CIDR.
3) Перечислите основные типы адресов IPv6.
4) Поясните взаимодействие сетей IPv6 через промежуточные сети IPv4.
5) Поясните цель использования дополнительных заголовков в IPv6.
6) Какие изменения произошли в DNS?
7) Поясните назначение протокола DHCP/

2.3. Технологии Интернет

В настоящее время существует два созвучных термина - internet и Internet. Под internet понимают технологию обмена данными, основанную на использовании семейства протоколов TCP/IP, а под Internet - глобальное сообщество мировых сетей, которые используют internet для обмена данными. В документе RFC 1180 "A TCP/IP Tutorial " (Учебное пособие по TCP/IP) авторы Theodore Socolofsky и Claudia Kale придерживаются мнения, что термин "internet technology" точно соответствует семейству протоколов TCP/IP и приложениям, использующим его.

2.3.1. Особенности работы в многосистемном сетевом окружении

При работе в многосистемном сетевом окружении, каким является сеть Интернет, возникают проблемы, связанные: с различными правилами

именования файлов и перемещения по каталогам файловой системы; ограничениями на доступ к файлам; с различными способами представления данных в файлах. Для решения этих проблем был создан протокол пересылки файлов FTP (File Transfer Protocol). На эффективность операций пересылки файлов влияют следующие факторы:

- файловая система хоста и производительность его дисков;
- объем обработки по переформатированию данных;
- используемая служба TCP.

В настоящее время существует ряд протоколов данного типа, из которых внимания заслуживают три – вышеупомянутый протокол FTP, SFTP (Simple File Transfer Protocol) и TFTP (Trivial File Transfer Protocol).

2.3.2. Протоколы передачи файлов

Протокол FTP - это совокупность соглашений, выполняемых при пересылке файлов в Интернет. Под протоколом FTP также понимаются средства доступа к удаленному компьютеру, позволяющие просматривать его каталоги и файлы, переходить из одного каталога в другой, копировать, удалять и обновлять файлы. Все эти функции являются частью стандартного стека протоколов TCP/IP. С протоколом FTP связаны следующие понятия: команды и их параметры, передаваемые по управляющему соединению; числовые коды ответа сервера; формат пересылаемых данных.

В FTP так же, как и в SMTP и POP3, команды [9] и ответы на них передаются в формате строк NVT ASCII. При обмене командами используется принцип последовательного выполнения. Однако в отличие от «почтовых» протоколов FTP открывает два TCP-соединения - одно для команд, а другое - для данных. В FTP два TCP-соединения определяются как управляющее соединение и соединение данных. *Управляющее соединение* - типичное соединение клиент-сервер. Сервер FTP обеспечивает пассивное открытие на официальном порту и ждёт запроса на установление соединения от клиента. Клиент FTP, в свою очередь, входит в контакт с FTP-сервером на официальном порту протокола 21 и устанавливает с ним TCP-соединение. Управляющее соединение остаётся активным на протяжении всего FTP-сеанса. Через управляющее соединение клиент и сервер обмениваются строками команд NVT ASCII и кодами ответа. FTP создаёт отдельное *соединение данных* для каждой операции по передаче файла (а также в некоторых других случаях). Основанные на протоколе FTP программы используют соединение данных для трех основных целей: чтобы послать список файлов или каталогов от сервера клиенту; чтобы послать файл от клиента серверу; чтобы послать файл от сервера клиенту.

Рис. 2.2 иллюстрирует типичную конфигурацию для операций по передаче файлов.

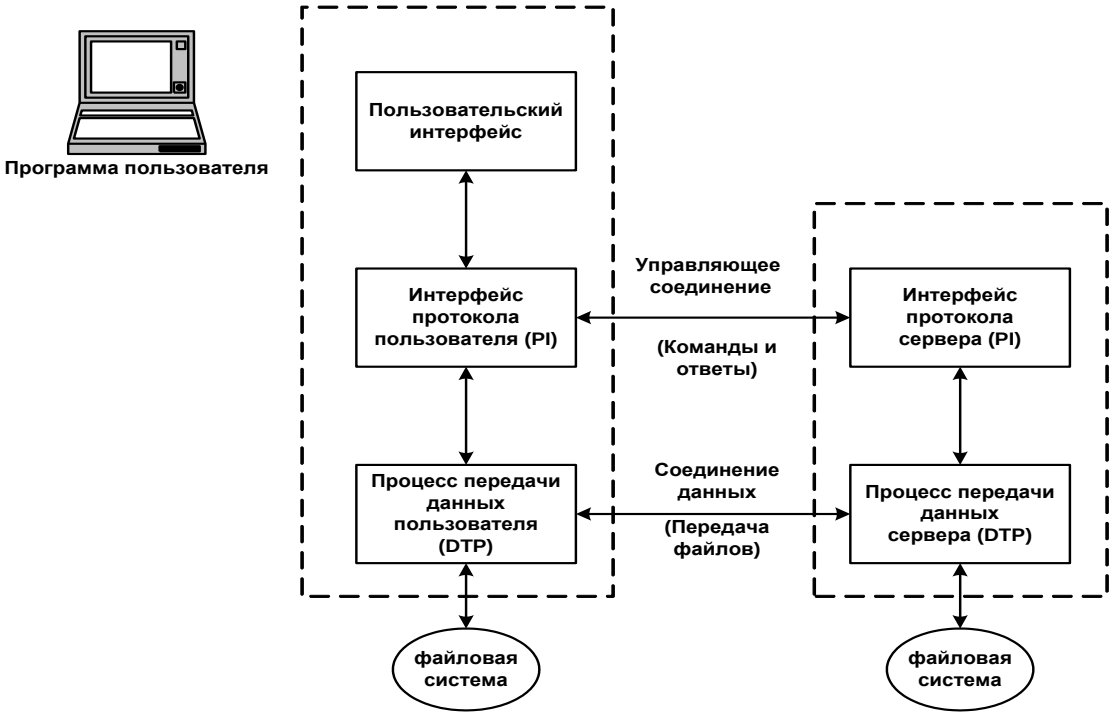


Рис. 2.2. Модель FTP

Основа модели - интерпретаторы протокола (PI) и процессы передачи данных (DTP). Как видим, на стороне клиента и сервера имеются собственные PI и DTP. Процессы передачи данных устанавливают и управляют соединением данных. Интерпретаторы протокола интерпретируют FTP-команды и общаются через управляющее соединение. Интерфейс пользователя ограждает пользователя от непосредственного общения с командами и ответами FTP.

С самого начала протокол FTP разрабатывался для работы с различными компьютерами, использующими различные операционные системы, структуры файлов и наборы символов. В результате FTP требует, чтобы пользователи выбрали соответствующие опций для операций по передаче файла. Опции FTP можно разбить на 4 категории: тип файла, формат файла, структура файла, способы передачи. FTP может управлять четырьмя различными *типами файлов:* локальным - предназначен для передачи файла между хостами, которые используют различные размеры байта; файлами изображений (или двоичными) - передаются как непрерывный поток данных; EBCDIC (Extended Binary Coded Decimal Interchange Code) - используются в некоторых системах, типа универсальных ЭВМ фирмы IBM (мэйнфреймы) и мини-компьютерах; ASCII - принимаемый по умолчанию тип для передачи файлов по FTP. Основная проблема с передачей файла ASCII - маркеры конца строки (end, off-line и т.д.).

В FTP определяет три типа управления форматом: Nonprint - означает, что файл не содержит никакой информации о вертикальном формате типа вертикальной прокрутки страницы; Telnet - использует управление вертикальным форматом для принтеров; FORTRAN - использует специальные,

вложенные в текст символы управления. Большинство реализации FTP (особенно в UNIX-системах) ограничиваются управлением форматом Nonprint.

FTP определяет три режима передачи через TCP- соединение: блочный - файл передается как последовательность блоков, где каждый блок включает один или больше байтов заголовка; со сжатием - последовательные появления одного и того же байта кодируются специальным символом с указанием количества одинаковых символов в последовательности; потоковый - файл передается как непрерывный поток байтов. Если тип структуры FTP - *структура с записями*, FTP использует специальную двухбайтовую последовательность символов, чтобы отметить конец записи и конец файла. Если FTP использует *файловую структуру*, конец файла отмечается закрытием TCP-соединения, т.е. после того как передан последний байт файла, FTP закрывает соединение данных. Более распространена файловая структура, которая используется по умолчанию.

Клиент передает запросы через управляющее соединение, а не непосредственно через соединение данных. Такой принцип функционирования выдвигает на первый план важное различие между FTP-клиентом и другими клиентами. В действительности, клиент и сервер обмениваются сообщениями через сокет, соединенный с сервером через официальный порт протокола. Следует обратить внимание на то, что передача данных FTP не происходит через официальный порт. Она происходит через порт, который выбирает сетевой компьютер клиента. Другими словами, для соединения данных FTP-клиент действует подобно серверу: клиент создает сокет; связывает его с местным адресом; сообщает серверу, какой адрес использовать, чтобы войти в контакт; ожидает входящее соединение. Однако различие между FTP-клиентом и настоящим сервером в том, что первый принимает соединение только от FTP-сервера на другом конце управляющего соединения, а сокет сервера обслуживает запросы на установление соединения, пришедшие от любого удаленного компьютера. FTP-клиент хранит адрес FTP-сервера в сокете, созданном для соединения данных.

За последние годы набор команд FTP, передаваемых по управляющему соединению существенно увеличился, однако хостам необязательно реализовывать все специфированные команды. Команды FTP можно разделить на три категории:

- *команды контроля доступа* - передают серверу информацию, идентифицирующую пользователя или сообщают серверу каталоги, к которым программа-клиент желает получить доступ (например, команды USER, PASS, REIN);

- *команды передачи параметров* - регистрируют тип, формат файла, структуру файла и режим передачи (например, PASV, PORT, STRU, TYPE, MODE);

- *команды обслуживания (сервиса)* - определяют операции по передаче файлов (например, LIST).

Код ответа сервера содержит три цифры. Каждая цифра в коде ответа имеет специальное значение. Смысл первой цифры (диапазон 1-5):

1 – предварительно положительный ответ, сервер начал требуемую операцию;

2 – положительный ответ завершения, сервер успешно закончил требуемое действие;

3 – промежуточный положительный ответ, сервер принял команду, но требуется дополнительная информация;

4 – временный отрицательный ответ завершения, сервер не принял команду, и требуемое действие не выполнялось;

5 – постоянный отрицательный ответ завершения, сервер не принял команду, и требуемое действие не выполнялось.

Вторая цифра в кодах ответа FTP идентифицирует сообщение более подробно (диапазон 0-5):

0 – *Синтаксис*: эти ответы относятся к ошибкам синтаксиса и синтаксически правильным командам, которые не попадают ни в одну из групп данной классификации;

1 – *Информация*: ответы на запрос о дополнительной информации, например, информации о состоянии или помощи;

2 – *Соединение*: ответы относятся к управляющему соединению или соединению данных;

3 – *Авторизация и учет использования ресурсов*: - ответы на регистрацию в системе и процедуру учета;

4 - *Не определен*.

5 - *Файловая система:* ответы информируют о состоянии файловой системы сервера в ответ на запрос файловой системы.

Например, в ответ на команду клиента USER сервер отвечает кодом 331, т.е. первая цифра 3 означает, что сервер принял команду и требуется дополнительная информация. Вторая цифра 3 означает, что эта информация связана с регистрацией. Как правило, после команды USER следует команда PASS.

Другим протоколом, заслуживающим внимание, является протокол TFTP (Trivial File Transfer Protocol) - простой протокол передачи файлов. TFTP узко специализируется на выполнении двух операций передачи файлов: чтение и запись. Протокол TFTP не очень быстр и устойчив, но весьма прост в реализации (эти два критерия его разработки заявлены в спецификации RFC 783). В RFC 906 «Загрузка при помощи TFTP» (*Bootstrap Loading using TFTP*, Finlayson, 1984) TFTP заявлен как стандарт Интернет для выполнения задачи копирования загрузчиков по сети для бездисковых рабочих станций. Особенно полезны для инициализации сетевых устройств маршрутизаторы, мосты, концентраторы.

Протокол TFTP, в отличие от FTP, не содержит возможностей аутентификации (хотя возможна фильтрация по IP-адресу) и основан на

транспортном протоколе UDP. Чтобы гарантировать доставку данных между TFTP-сервером и TFTP-клиентом TFTP использует систему подтверждений.

Действие TFTP начинается UDP-датаграммой - запросом на чтение или на запись файла. Клиент начинает работу после получения порта, соответственно, посылая запрос на чтение или на запись на порт 69. Сервер идентифицирует различные номера портов клиентов и использует их для последующей пересылки данных. Он направляет свои сообщения на порт клиента, посылая требуемый файл блоками длиной 512 байт. Сервер ждет подтверждения принятия клиентом каждого блока данных прежде, чем передать следующий. В качестве сигнала завершения передачи TFTP-сервер посылает UDP-датаграмму с длиной меньшей 512 байтов. Если размер пересылаемого файла кратен 512, то признак EOF должен посылаться дополнительно. Таким образом, пересылка данных производится как обмен блоками данных и сообщениями АСК. Блоки данных нумеруются от единицы. Каждый АСК содержит номер блока данных, получение которого он подтверждает. В улучшенном варианте TFTP разрешается согласование параметров через предварительные запросы чтения и записи. Его основная цель – позволить клиенту и серверу согласовывать между собой размер блока (более 512 байт) для увеличения эффективности пересылки данных.

2.3.3.Технология удаленного доступа к ресурсам сети

2.3.3.1. Понятие, особенности, симметрия взаимодействия

Исторически технология TELNET служила для удалённого доступа к интерфейсу командной строки операционных систем. Впоследствии ее стали использовать для прочих текстовых интерфейсов, вплоть до игр MUD и анимированного ASCII-art. В настоящее время под TELNET понимают триаду, состоящую из telnet-интерфейса пользователя, telnetd-процесса, TELNET-протокола. Эта триада обеспечивает описание и реализацию сетевого терминала для доступа к ресурсам удаленного компьютера. Существует достаточно большое количество программ, которые позволяют работать в режиме удаленного терминала, но ни одна из них не может сравниться с TELNET по степени проработанности деталей и концепции реализации.

TELNET как протокол описан в RFC-854. Его авторы J.Postel и J.Reynolds во введении к документу определяют назначение TELNET следующим образом: "…дать общее описание, насколько это только возможно, двунаправленного, восьмибитового взаимодействия, главной целью которого является обеспечение стандартного метода взаимодействия терминального устройства и терминал-ориентированного процесса. При этом протокол может быть использован и для организации взаимодействий "терминал-терминал" (связь) и "процесс-процесс" (распределенные вычисления)". Теоретически, даже обе стороны протокола могут являться программами, а не человеком. TELNET строится как протокол приложения над транспортным протоколом TCP.

В основу TELNET положены три фундаментальные идеи: концепция сетевого виртуального терминала NVT (Network Virtual Terminal); принцип договорных опций (согласование параметров взаимодействия); симметрия связи "терминал-процесс".

Хотя в сессии Telnet выделяют клиентскую и серверную сторону, протокол на самом деле полностью симметричен. Симметрия взаимодействия позволяет в течении одной сессии программе «user» и программе "server" меняться местами. Это принципиально отличает взаимодействие в рамках telnet от традиционной модели клиент-сервер.

2.3.3.2. Обязательные компоненты. Стандарт NVT

При установке telnet-соединения программа, работающая с реальным терминальным устройством, и процесс обслуживания этой программы используют для обмена информацией спецификацию представления правил функционирования терминального устройства или NVT. *NVT* - это стандартное описание наиболее широко используемых возможностей реальных физических терминальных устройств. NVT позволяет описать и преобразовать в стандартную форму способы отображения и ввода информации. Это позволяет, с одной стороны, унифицировать характеристики физических устройств, а с другой - обеспечить принцип совместимости устройств с разными возможностями. Характеристики диалога диктуются устройством с меньшими возможностями. Если взаимодействие осуществляется по принципу "терминал-терминал" или "процесс-процесс", то "user" - это сторона, инициирующая соединение, а "server" - пассивная сторона.

После установления транспортного соединения (как правило, TCP) оба его конца играют роль сетевых виртуальных терминалов, обменивающихся двумя типами данных: *прикладными данными* (т.е. данными, которые идут от пользователя к текстовому приложению на стороне сервера и обратно); опциями протокола Telnet, служащими для уяснения возможностей и предпочтений сторон.

Прикладные данные проходят через протокол без изменений [10], т.е. на выходе второго виртуального терминала видно именно то, что было введено на вход первого. С точки зрения протокола данные представляют просто последовательность байтов (октетов), по умолчанию принадлежащих набору ASCII, но при включенной опции Binary - любых. Хотя были предложены расширения для идентификации набора символов, но на практике ими не пользуются.

Следует иметь в виду, что сессия Telnet уязвима для любого вида атак, к которым уязвим транспорт, т.е. протокол TCP, если только не осуществляется в полностью контролируемой сети или с применением защиты на сетевом уровне (различные реализации виртуальных частных сетей). Это связано с тем, что в протоколе не предусмотрено использование ни шифрования, ни проверки подлинности данных. Для функциональности удалённого доступа к системе в

настоящее время применяется сетевой протокол SSH, при создании которого значительное внимание было уделено вопросам безопасности.

2.3.4. Использование Telnet для тестирования других протоколов

С помощью программы-клиента Telnet можно войти в систему удаленного компьютера, как если бы ваш компьютер был терминалом того. Необходимо сообщить клиенту имя (адрес) удаленного компьютера, к которому необходимо присоединиться. Клиент, в свою очередь, устанавливает TCP- соединение с портом 23 (официальный порт TELNET). Удаленный компьютер показывает приглашение для входа в систему, в котором можно вести имя пользователя. В зависимости от сервера Telnet, с которым установлено соединение, можно также получить приглашение для ввода пароля. Подобно анонимным FTP-серверам в Интернет существует большое количество общедоступных серверов Telnet. Различие между ними состоит в том, что серверы Telnet не имеют стандартного имени или пароля для анонимного доступа.

Программа работает в двух режимах: в режиме командной строки (command mode) и в режиме удаленного терминала (input mode). При работе в режиме удаленного терминала telnet позволяет работать с буферизацией (line-by-line) или без нее (character-at-a-time). При работе без буферизации каждый введенный символ немедленно отправляется на удаленную машину, откуда приходит "эхо". При буферизованном обмене введенные символы накапливаются в локальном буфере и отправляются на удаленную машину пакетом. "Эхо" в последнем случае также локальное. Для переключения между режимом командной строки и режимом терминала используют последовательность, которая может быть изменена командами TELNET, приведенными в [9].

Клиент Telnet может устанавливать TCP-соединение с любым портом протокола фактически на любом сетевом компьютере Интернет. Эту способность клиента Telnet можно использовать для проверки и тестирования работы других протоколов, Finger (порт 79), POP3 (порт 110), SMTP (порт 25), например: *telnet host.domain.org 25.*

2.3.5. Транспортные технологии пакетной коммутации

В настоящее время стремительными темпами развиваются, наряду с традиционными методами импульсно-кодовой, дельта- и других цифровых видов модуляции, методы речеобразования с существенным сокращением избыточности, методы статистического уплотнения цифровых каналов, пакетов передачи и коммутации с использованием технологий FR, IP, ATM и др.

2.3.5.1. Особенности передачи речевой информации

До недавнего времени сети с коммутацией каналов (телефонные сети) и сети с коммутацией пакетов (IP-сети) существовали практически независимо друг от друга и использовались для различных целей. Телефонные сети

использовались только для передачи голосовой информации, а IP-сети соответственно для передачи данных. Интерес к вопросам передачи речи по сетям передачи данных с пакетной коммутацией (VoIP) возник с тех пор, как стало очевидным, что коммутация каналов более не в состоянии удовлетворять растущие потребности рынка, обеспечивать активное внедрение новых и дополнительных услуг, снижение удельных затрат на расширение сетей.

Проведенный в различных исследовательских группах [11] анализ качества синтезированной речи при передачи речевых данных через сеть Интернет показывает, что основным источником возникновения искажений, снижения качества и разборчивости синтезированной речи является прерывание потока речевых данных, вызванное потерями при передачи по сети либо превышением предельно допустимого времени доставки пакета с речевыми данными. Процесс речевого диалога в системе Интернет с информационной точки зрения имеет следующие три фазы: соединение абонентов; обмен информацией; разъединение абонентов. Во время первой и третьей фаз передаются только управляющие данные, и при этом происходит установление соединения. На протяжении второй фазы абоненты обмениваются как управляющими, так и информационными данными.

Основное требование к передаче командной информации - отсутствие ошибок передачи. Время доставки сообщений также играет немаловажную роль. Источником информационных данных является речевой сигнал, возможной моделью которого является нестационарный случайный процесс. Можно выделить следующие типы сигнальных фрагментов: вокализированные, невокализированные, переходные и паузы. При передаче речи в цифровой форме каждый тип сигнала при одной и той же длительности и одинаковом качестве требует различного числа двоичных единиц (бит) для кодирования и передачи. Следовательно, скорость передачи разных типов сигнала также может быть различной. Поэтому передачу речевых данных в каждом направлении дуплексного канала следует рассматривать как передачу асинхронных логически самостоятельных фрагментов цифровых последовательностей (транзакций) с блочной (дейтаграммной) синхронизацией внутри транзакций, наполненной блоками различной длины.

Асинхронность транзакций позволяет с одной стороны оптимизировать трафик за счет снижения средней скорости передачи и с другой - скомпенсировать неидеальности канала передачи за счет относительной свободы в воспроизведении каждой транзакции.

Особенности функционирования каналов для передачи речевых данных, а также возможные варианты построения систем телефонной связи на базе сети Интернет, предъявляют ряд специфических требований к речевым кодекам (вокодера). Наиболее целесообразным для систем IP-телефонии является применение кодеков с переменной скоростью кодирования речевого сигнала. В основе кодека речи с переменной скоростью лежит классификатор входного сигнала, определяющий степень его информативности и, таким

образом, задающий метод, кодирования и скорость передачи речевых данных. Наиболее простым классификатором речевого сигнала является Voice Activity Detector (VAD), который выделяет во входном речевом сигнале активную речь и паузы. При этом фрагменты сигнала, классифицируемые как активная речь, кодируются каким-либо из известных алгоритмов (как правило, на базе метода Code Excited Linear Prediction - CRLP) с типичной скоростью 4-8 Кбит/с. Фрагменты, классифицированные как паузы, кодируются и передаются с очень низкой скоростью (порядка 0.1-0.2 Кбит/с), либо не передаются вообще. Передача минимальной информации о паузных фрагментах предпочтительна.

Схемы более эффективных классификаторов входного сигнала детальнее осуществляют классификацию фрагментов, соответствующих активной речи. Это позволяет оптимизировать выбор стратегии кодирования (скорости передачи данных), выделяя для особо ответственных за качество речи участков речевого сигнала большее число бит (соответственно большую скорость), для менее ответственных - меньше бит (меньшую скорость). При таком построении кодеков могут быть достигнуты низкие средние скорости (2-4 Кбит/с) при высоком качестве синтезируемой речи.

Основным источником возникновения искажений, снижения качества и разборчивости синтезированной речи является прерывание потока речевых данных, вызванное потерями при передачи по сети, либо превышением предельно допустимого времени доставки пакета. Поэтому одной из важнейших задач при построении вокодеров для IP-телефонии является создание алгоритмов компрессии речи толерантных к потерям пакетов.

Преобразование управляющей информации и данных, поступающих из одной сети (например, PSTN) в пакеты глобальной сети Интернет и обратно в сети IP- телефонии осуществляет шлюз. Причем такое преобразование не должно значительно исказить исходный речевой сигнал, а режим передачи обязан сохранить обмен информацией между абонентами в реальном масштабе времени.

Основные функции, выполняемые шлюзом при соединении типа "точка-точка" состоят в следующем: реализация физического интерфейса с коммуникационной сетью; детектирование и генерация сигналов абонентской сигнализации; преобразование сигналов абонентской сигнализации в пакеты данных и обратно; преобразование речевого сигнала в пакеты данных и обратно; соединение абонентов; передача по сети сигнализационных и речевых пакетов; разъединение связи. Большая часть функций шлюза в рамках архитектуры TCP/IP реализуются в процессах прикладного уровня. Управленческие задачи и связь с сетью осуществляется с помощью универсального процессора, а решения задач сигнальной обработки и телефонного интерфейса выполняются на цифровом процессоре обработки сигналов.

Очень важной является задача обнаружения и детектирования телефонной сигнализации. При использовании двухпроводных абонентских

линий актуальной остаётся задача эхокомпенсации, особенность которой состоит в том, что компенсировать необходимо два различных класса сигналов - речи и телефонной сигнализации. Схема сигнальной обработки в шлюзе при подключении аналогового 2-проводного телефонного канала PSTN показана на рис. 2.3. Телефонный сигнал поступает на дифференциальную систему, которая разделяет приемную передающую часть канала. Далее сигнал передачи вместе с не отфильтрованной частью сигнала приема подается на аналого-цифровой преобразователь и превращается либо в стандартный 12-разрядный сигнал либо в 8-разрядный сигнал, закодированный по μ- либо A- закону.

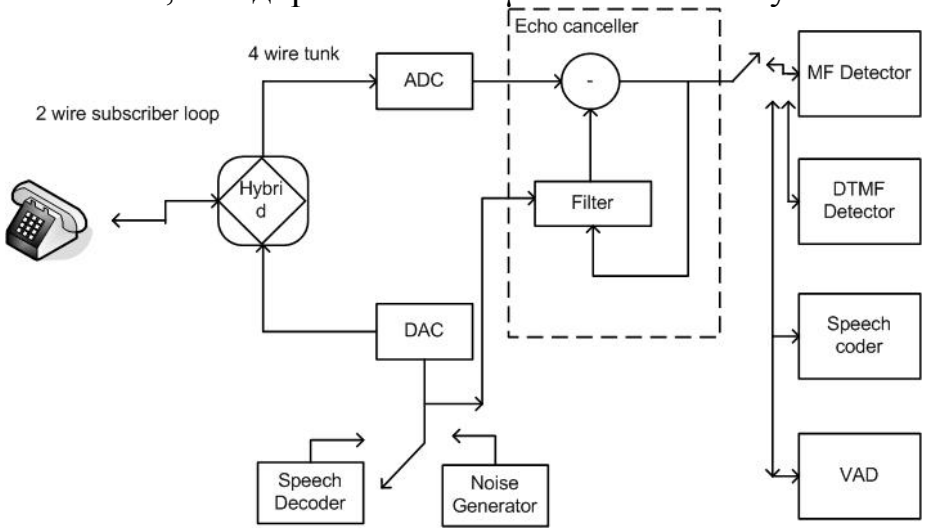


Рис. 2.3. Схема сигнальной обработки в шлюзе

В последнем случае обработка должна также включать соответствующий экспандер. В устройстве эхо-компенсации из сигнала передачи удаляются остатки принимаемого сигнала. Эхо-компенсатор представляет собой адаптивный нерекурсивный фильтр, длина памяти (порядок) которого и механизм адаптации выбираются такими, чтобы удовлетворить требованиям рекомендации МККТТ (ITU-T) G.165.

Для обнаружения и определения сигналов внутриполосной телефонной сигнализации (MF сигналов), сигналов DTMF либо импульсного наборов используются детекторы соответствующих типов. В режиме сессии дальнейшая обработка входного сигнала происходит в речевом кодере. В анализаторе кодера сигнал сегментируется на отдельные фрагменты длительностью 30 мс и каждому входному блоку, состоящему из 240 отсчетов (1920 бит при A- либо μ-коде и 2880 бит при 12-разрядном линейном коде), сопоставляется информационный кадр длиной 137 бит.

Часть параметров, вычисленная в анализаторе, используется в блоке определения голосовой активности (VAD - voice activity detector), который решает, является ли текущий анализируемый фрагмент сигнала речью или паузой. При наличии паузы информационный кадр может не передаваться в службу виртуального канала. Режим передачи паузных кадров следующий. На сеансовый уровень передается лишь каждый пятый кадр такого типа. Кроме

того, при отсутствии речи для кодировки текущих спектральных параметров используется только 27 бит.

На приемной стороне из виртуального канала в логический поступает либо информационный кадр (длиной 137 или 27 бит), либо флаг наличия паузы. На паузных кадрах вместо речевого синтезатора включается генератор комфортного шума, который восстанавливает спектральный состав паузного сигнала.

Параметры генератора обновляются при получении паузного информационного кадра. Наличие информационного кадра длиной 137 бит включает речевой декодер, на выходе которого формируется 12-разрядный речевой сигнал. Для эхо-компенсатора этот сигнал является сигналом дальнего абонента, фильтрация которого дает составляющую электрического эха в передаваемом сигнале. В зависимости от типа цифро-аналогового преобразования сигнал может быть подвергнут дополнительной кодировке по А- либо μ- закону. Сложность состоит в том, что служебные сигналы могут перемешиваться с сигналами речи.

2.3.5.2. Основные компоненты технологии WWW

Под Web-технологиями понимается подмножество Интернет-технологий, упрощающих доступ к мультимедиа информации. Сильные стороны Web-технологий - в первую очередь, это простота применения и независимость от операционной системы. Однако из всего разнообразия решений, доступных сегодня, можно выделить те основные технологии, которые применяются для создания практически любых сайтов или других Интернет-проектов (рис. 2.4).

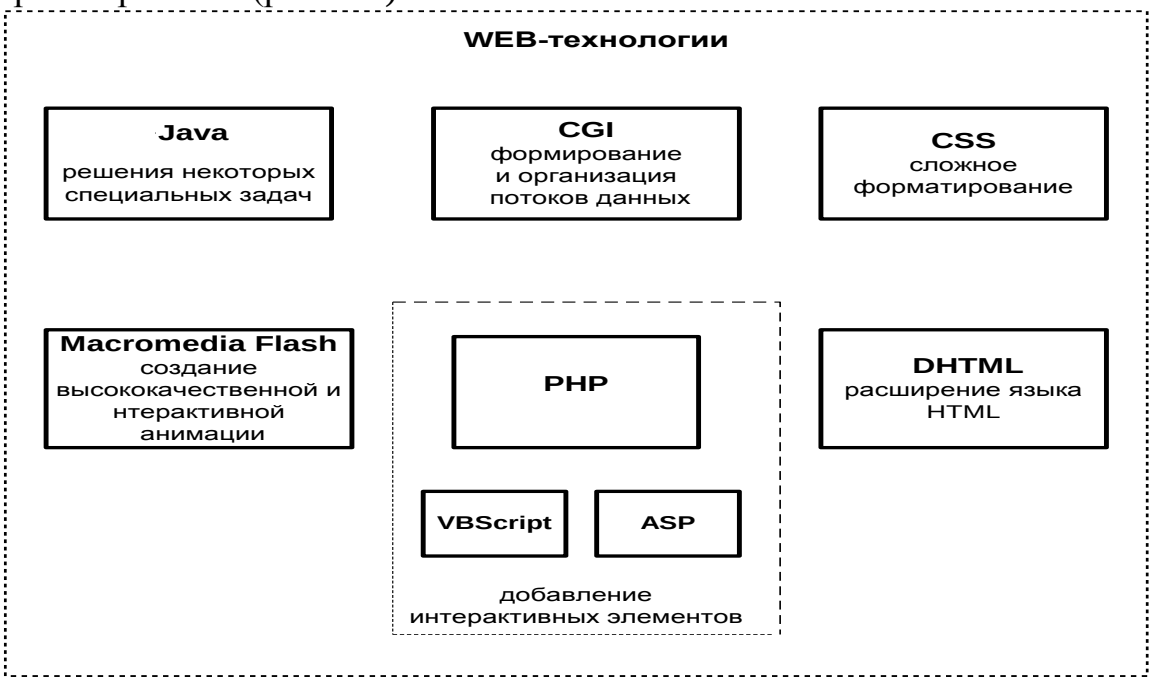


Рис. 2.4. Основные Web-технологии

Применение Web-технологий позволяет значительно упростить задачу обеспечения совместимости при разработке приложений управления и, возможно, может послужить основой для создания интерфейса с хранилищами данных.

За последние годы произошло значительное увеличение пропускной способности сетей благодаря реализации многочисленных высокоскоростных каналов. Впервые появилась возможность организации по-настоящему распределенных систем, обслуживающих миллионы пользователей.

Облачные вычисления (cloud computing) - новая парадигма, предполагающая распределенную и удаленную обработку и хранение данных. В соответствии с определением, предложенным Национальным Институтом Стандартов и Технологий США (NIST) под термином *Cloud Computing* (облачные вычисления) понимается модель предоставления повсеместного и удобного сетевого доступа, по мере необходимости, к общей совокупности конфигурируемых вычислительных ресурсов (например, сетей, серверов, систем хранения, приложений и сервисов), которые могут быть быстро предоставлены и освобождены с минимальными усилиями по управлению и необходимостью взаимодействия с провайдером услуг (сервис- провайдером).

Концепция облака основана на уровнях**,** каждый из которых предоставляет определенную функциональность (рис.2.5). Такое деление компонентов облака позволяет сделать уровни облачных вычислений коммунальным ресурсом, аналогичным электричеству, услугам телефонии или природному газу. Товар "облачные вычисления" - это более дешевые и менее затратные для пользователя вычислительные ресурсы. В перспективе облачные вычисления могут стать еще одним коммунальным ресурсом.

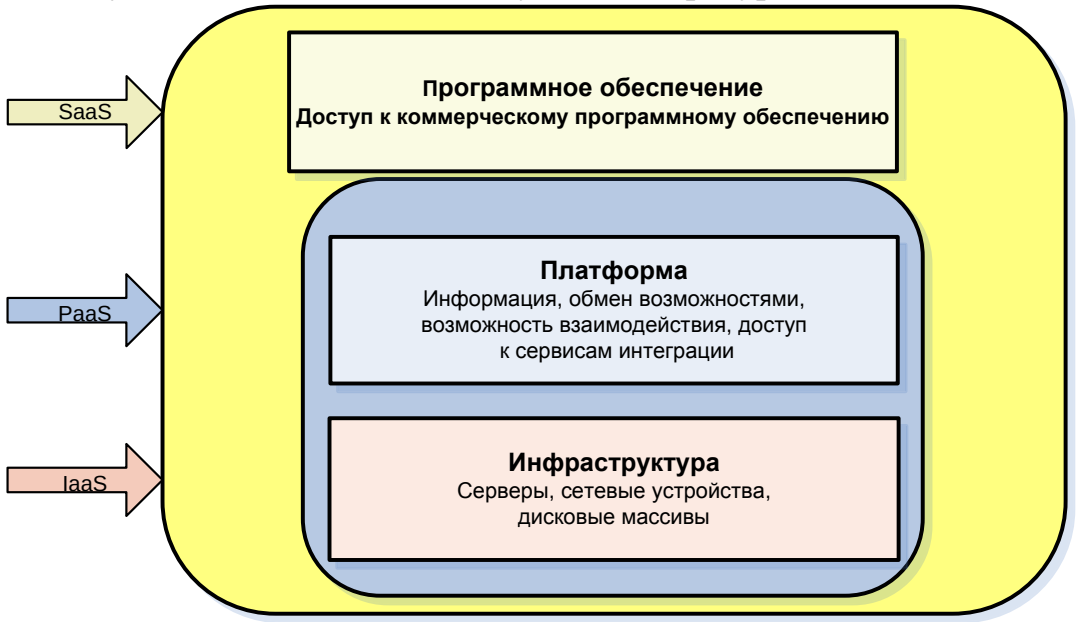


Рис. 2.5. Уровни облачных вычислений

Облачные технологии представляют собой технологии вида «клиент-сервер», которые состоят из виртуального сервера (или группы серверов)

и нескольких клиентов, которые подключаются к нему с помощью сети Интернет. Обозначение «облаков» в данном случае используется как основная ассоциация при обозначении структуры работы данной системы.

Облачная модель поддерживает высокую доступность сервисов и описывается пятью основными характеристиками (essential characteristics), тремя сервисными моделями/моделями предоставления услуг (service models) и четырьмя моделями развертывания (deployment models) [12].

Контрольные вопросы

1. Поясните технологию удаленного доступа к ресурсам сети.
2. В чем особенность передачи речи через каналы Интернет?
3. Перечислите способы голосовой связи через IP-сеть.
4. Назовите основные источники возникновения искажений.
5. Перечислите основные функции, выполняемые шлюзом.
4. Поясните схему сигнальной обработки в шлюзе.
5. Что понимается под web-технологиями?

3. ИТОГОВЫЙ ТЕСТ

Вопрос 1

Укажите правильное соответствие:

1	локальная сеть (LAN)	*1*	набор компьютеров и других сетевых устройств, размещенных в пределах одной физической сети
2	глобальная сеть (WAN)	*2*	всемирный набор связанных в сеть компьютеров
3	виртуальная частная сеть (VPN)	*3*	объединенная сеть, которая соединяет множество узлов и использует сторонние средства коммуникаций для передачи сетевого трафика из одного места в другое
4	Internet	*4*	круг технологий, обеспечивающих безопасную и качественную связь в пределах контролируемой группы пользователей по открытой глобальной сети
5	internet	*5*	способы доставки и обработки информации, присущие Internet, перенесенные на корпоративную сеть
6	intranet	*6*	межсетевое объединение, описывающее логическую сеть, состоящую из 2-х и более физических сетей

1. 1-1, 2-2, 3-3, 4-4, 5-5, 6-6
2. 1-1, 2-6, 3-4, 4-2, 5-5, 6-3
3. 1-1, 2-5, 3-3, 4-4, 5-5, 6-6
4. 1-1, 2-3, 3-4, 4-2, 5-6, 6-5
5. 1-1, 2-3, 3-4, 4-2, 5-5, 6-6

Вопрос 2

Укажите все номера правильных высказываний:

1. Существует два способа переключения соединений: переключение цепей и переключение пакетов.

2. Данные в сети с переключением пакетов имеют множество альтернативных маршрутов

3. При полудуплексном соединении данные следуют в обоих направлениях одновременно.

4. Топология сети - это ее геометрическая форма

5. Топология сети - это физическое расположение компьютеров по отношению друг к другу

Вопрос 3

Укажите номер правильного ответа:

IPCONFIG /all используется для:

1. Проверки конфигурации компьютера, выводит полную информацию.
2. Проверки конфигурации компьютера, выводит только IP-адрес.
3. Проверки конфигурации компьютера, выводит только маску подсети.
4. Проверки IP-связи.

Вопрос 4

Укажите номер правильного ответа:

Команда ARP -a 194.224.40.1 используется для:

1. Проверки конфигурации компьютера, выводит полную информацию.
2. Проверки конфигурации компьютера, выводит только IP-адрес.
3. Проверки конфигурации компьютера, выводит только маску подсети.
4. Поиска записи, относящейся к определенному узлу.

Вопрос 54

Верно ли следующее утверждение:

"Так как компьютеры в сетях соединяются между собой по телефонным линиям, не важно, какой протокол использовать для установления эффективной связи "

1. Да 2. Нет

Вопрос 6

Сколько узлов поддерживает по умолчанию сеть класса B?

1. 65.534 2. 254
2. 2.097.152 4. 16.384

Вопрос 7

Количество октетов, используемых по умолчанию для идентификатора сети адреса класса C, равно:

1. 1 2. 2
2. 3 4. 4

Вопрос 8

Какой из следующих ответов наилучшим образом описывает назначение маски подсети?

1. Маска подсети позволяет TCP/IP отделить адрес сети от адреса узла. Это помогает определять IP-адреса других узлов.
2. Маска подсети позволяет определить местонахождение других TCP/IP-узлов.
3. Маска подсети используется TCP/IP для маскирования части IP-адреса.
4. Маска подсети позволяет TCP/IP отделить адрес сети от адреса узла. Это помогает определить местонахождение других TCP/IP-уэлов.

Вопрос 9

Каждый из приведенных ответов описывает соответствие между уровнем эталонной модели OSI и уровнем модели TCP/IP. Какие соответствия указаны не правильно?

1. *OSI:* уровень представления; *TCP/IP:* уровень приложения.
2. *OSI:* уровень сеанса; *TCP/IP:* уровень транспорта.
3. *OSI:* уровень сети; *TCP/IP:* межсетевой уровень.
4. *OSI:* физический уровень; *TCP/IP*: уровень сетевого интерфейса.

Вопрос 10

Что предпринимает компьютер после проверки кэша при определении адреса в локальной сети?

1. Отправляет запрос на маршрутизатор.
2. Отправляет запрос на сервер ARP.
3. Обращается к файлу HOSTS.
4. Отправляет широковещательный запрос.

Вопрос 11

Чем меньше отношение числа пользователей к числу входных модемов у данного провайдера, тем быстрее и надежнее осуществляется связь с Internet.

1. Данное утверждение верно.
2. Данное утверждение неверно.

Вопрос 12

Какой из следующих протоколов наиболее подходит для разработки приложения, требующего постоянного соединения с другим компьютером (на котором работает соответствующая служба), и не включающего код, проверяющий правильность доставки данных с удаленной машины?

1. ARP
2. ICMP
3. TCP
4. UDP
5. IGMP

Вопрос 13

Какие из следующих файлов не содержат записи ресурсов DNS, используемые для определения имен?

1. 12.122.205.IN-ADDR.ARPA.DNS
2. Boot file
3. CACHE.DNS
4. LANW.COM.DNS

Вопрос 14

Какой из методов определения имен использует один текстовый файл для определения имен Интернет-узлов?

1. LMHOSTS
2. HOSTS
3. Пространство имен доменов
4. Система имен доменов

Вопрос 15

Укажите номер строки, содержащей неверное утверждение:

а) Спецификация класса FEC может содержать несколько компонентов, каждый из которых определяет набор пакетов, соответствующих данному классу.

б) Пакет принадлежит данному классу, если адрес получателя точно совпадает с компонентом адреса узла либо имеет максимальное совпадение с адресным префиксом.

в) Компонентами EFC являются: адрес узла и адресный префикс.

г) MPLS базируется на IP.

д) Архитектура MPLS требует обязательного применения LDP.

Вопрос 16

Укажите неверное высказывание.

а) Шлюз - основная и неотъемлемая часть архитектуры IP-телефонии, непосредственно соединяющая телефонную сеть с IP-сетью.

б) Шлюзы различных производителей отличаются способом подключения к телефонной сети, аппаратной платформой и выполняемыми функциями.

в) Шлюз выполняет следующие функции: ответ на вызов абонента PBX/PSTN; установление соединения с удаленным шлюзом; установление соединения с вызываемым абонентом PBX/PSTN; сжатие, пакетирование и восстановление голоса (или факс-сигнала).

г) Диспетчер (Gatekeeper) - дополнительной устройство, подключенное только к IP-сети и несущее в себе всю логику работы сети IP-телефонии.

д) Функциями Gatekeeper является: аутентификация и авторизация абонента, распределение вызовов между шлюзом, биллинг (интерфейс с профессиональными биллинговыми системами).

Вопрос 17

Укажите последовательность заполнения меток по протоколу LDP, при наличии 3-х LSR в сети MPLS:

1 LDP уведомляет верхний маршрутизатор (N-1) о том, что потоку N присвоена метка M	**5** LSR_{N-1} присваивает собственное значение метки данному потоку
2 LSR устанавливают сеансы LDP с соседними устройствами	**6** Каждое устройство сети MPLS строит базу топологической информации, используя OSPF
3 LSR_{N-1} уведомляет верхний маршрутизатор об этой привязке	**7** Верхний маршрутизатор метку M помещает в поле выходной метки своей таблицы
4 Выходной граничный маршрутизатор ассоциирует класс FEC с пакетами и присваивает случайное значение метки M	**8** Верхний маршрутизатор помещает полученную информацию в поле своей таблицы

а) 2-3-5-7 –8-6-1 б) 1-6-8-7-5-3-2 в) 8-1-5-3-6-4

г) 6-2-4-1-7-5-3-7 д) 4-1-2-7-5-6-3-7 е) 8-3-2-5-7-1-6

Вопрос 18

Базовым протоколом для всех приложений, связанных с интерактивной передачей речевой и видеоинформации по сети с маршрутизацией пакетов является:

1) SIP 2) RTR 3)RTCP 4) MPLS
5) PIM 6) DVMRP 7) H.323

Вопрос 19

DHCP-клиент был перемещен из подсети A в подсеть B, после чего пользователи сообщили, что они больше не могут использовать TCP/IP на этой машине. Чем может быть вызвана эта проблема?

1. DHCP не может поддерживать несколько подсетей.
2. Сервер WINS не видит клиента.
3. Адрес шлюза по умолчанию был установлен вручную до переноса компьютера в другую подсеть.
4. Клиент не прервал DHCP-аренду до его переноса в другую подсеть.

Вопрос 20

Укажите, что означает термин New Public Network:

1. Способы доставки и обработки информации, присущие Internet.
2. Обозначение устойчивых информационных потоков одного предприятия, существующих в публичной сети с коммутацией пакетов и достаточно защищенных от влияния потоков данных других пользователей этой публичной сети.
3. Аббревиатуру для термина сети, конечной целью создания которой является транспортировка необходимой информации с минимальными задержками.

Вопрос 21

Что такое TCP/IP?

4. Набор протоколов, разработанный Microsoft для того, чтобы позволить обычным пользователям получать доступ к ресурсам в Интернет.
5. Набор протоколов, позволяющий взаимодействовать различным приложениям, работающим на различных аппаратных платформах и в различных типах сетей.
6. Протокол, разработанный Microsoft для маршрутизации информации между разнородными сетями.
7. Протокол, разработанный IAB для того, чтобы различные производители программного и аппаратного обеспечения могли получить доступ к Интернету.

Вопрос 22

Укажите номер правильного соответствия:

Топология сети	Преимущества	Недостатки
1 Шина	**1** все компьютеры имеют равный доступ, количество пользователей не оказывает значительного влияния на производительность	**1** трудно локализовать проблемы, выход из строя кабеля останавливает работу многих пользователей, при значительных объемах трафика уменьшается пропускная способность сети
2 Звезда	**2** легко модифицировать сеть, добавляя новые компьютеры, централизованный контроль и управление, выход из строя одного из компьютеров не влияет на работоспособность сети	**2** выход из строя одного компьютера может вывести из строя всю сеть, изменение конфигурации сети требует остановки работы всей сети, трудно локализовать проблемы
3 Кольцо	**3** экономный расход кабеля, простота построения, сеть легко расширяется	**3** выход из строя центрального узла выводит из строя всю сеть

1. 1-1-1 2-2-2 3-3-3
2. 1-3-1 2-2-3 3-1-2
3. 1-3-2 2-2-3 3-1-1
4. 1-2-3 2-1-3 3-3-1
5. 1-2-2 2-3-1 3-1-3

Вопрос 23

Модель облачных вычислений состоит из:

1.SaaS
2. IaaS
3. Слой бизнес-логики.
4. нет правильного ответа.

Литература

1. Джамса К., Коуп К. Программирование для Internet в средеWindows. - СПб.: Питер, 1996. - 688 с.
2. Recommendations of the National Institute of Standards and Technology. http://csrc.nist.gov/publications/nistpubs/800-146/sp800-146.pdf.
3. Таненбаум Э. Компьютерные сети - http://books4study.info/text-book4192.html2
4. Олифер В.Г., Олифер Н.А., Компьютерные сети. Принципы, технология, протоколы: учебник для вузов, 4-е издание. - СПб.: Питер, 2010.- 958 с.
5. http://www.intuit.ru/department/network/pdsi/lit.html - Интернет университет
6. http://kunegin.com/ref3/atm5/3.htm.
7. Ропчан С. Абсолютно все о Frame Relay//Системный администратор, №2(3, 2003) - http://www.net4me.net/docs/pdf/Network/frame-relay.pdf.
8. Сидни Фейт TCP/IP: Архитектура, протоколы, реализация. – М.: ЛОРИ, 2000. – 756 с.
9. Романчева Н.И. Базовые технологии Интернет: учебное пособие. – М.: МГТУ ГА, 2010. – 80 с.
10. Храмцов П.Б., Брик С.А., Русак А.М., Сурин А.И. Основы web-технологий.- Интернет-университет информационных технологий - ИНТУИТ.ру, БИНОМ. Лаборатория знаний, 2007.
11. Романчева Н.И. Современные технологии Интернет: учебное пособие. – М.: МГТУ ГА, 2009 – 80 с.
12. Private Cloud Principles, Concepts, and Patterns. http://social.technet.microsoft.com/wiki/contents/articles/4346.private-cloud-principles-concepts-and-patterns.aspx.