Учреждение образования
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

П. П. Урбанович, Д. М. Романенко

КОМПЬЮТЕРНЫЕ СЕТИ И СЕТЕВЫЕ ТЕХНОЛОГИИ

Допущено
Министерством образования Республики Беларусь
в качестве учебного пособия для студентов
учреждений высшего образования
по техническим специальностям

Минск 2022

УДК 004.7(075.8)
БКК 32.971.35я73
У69

Р е ц е н з е н т ы :
кафедра программного обеспечения информационных технологий учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»;
доцент кафедры многопроцессорных систем и сетей Белорусского государственного университета кандидат физико-математических наук, доцент *Т. В. Соболева*

Урбанович, П. П.

У69 Компьютерные сети и сетевые технологии : учеб. пособие для студентов технических специальностей / П. П. Урбанович, Д. М. Романенко. – Минск : БГТУ, 2022. – 608 с.
ISBN 978-985-530-967-4.

В учебном пособии описаны сетевые компоненты, приведены наиболее распространенные виды топологий, используемые для физического соединения компьютеров в сети, основные методы доступа к каналу связи, рассмотрены применяемые на практике физические среды передачи данных. Определены общие принципы, лежащие в основе построения всех локальных сетей, разъясняющие правила обмена. Описаны типы сетевого оборудования, их назначение и принципы функционирования под управлением современных сетевых операционных систем. Анализируются базовые технологии построения беспроводных сетей, а также структура Bluetooth и сотовых сетей. Рассмотрены методы и средства обеспечения надежного и безопасного функционирования сетей.

Пособие предназначено для студентов технических специальностей. Может быть полезно магистрантам и аспирантам, изучающим вопросы проектирования, эксплуатации и администрирования компьютерных сетей и систем.

УДК 004.7(075.8)
ББК 32.971.35я73

ISBN 978-985-530-967-4

ОГЛАВЛЕНИЕ

ПРЕДИСЛОВИЕ

Термин «инновация» принято ассоциировать с трансформацией созданного научно-технического потенциала в новые технологии и материальные ресурсы. Современные инновационные технологии базируются на все более широком использовании компьютеров и компьютерных сетей. К тому же люди стали хранить гораздо больше данных в сети, работать онлайн и пользоваться корпоративными сервисами с личных устройств.

Стандарты связи и беспроводного интернета, с одной стороны, помогают работать удаленно с передачей данных на высокой скорости, использовать технологические возможности в корпоративном обучении, способствуют развитию интернета вещей и искусственного интеллекта. А с другой стороны, появляются новые угрозы, связанные с безопасностью передачи данных.

Трендом совершенствования образовательных технологий становится использование видеоконференцсвязи и онлайн-курсов, которые реализуются на основе компьютерных сетей и сетевых технологий.

Указанные обстоятельства повышают роль специалистов, разрабатывающих данные технологии, что, в свою очередь, увеличивает значение соответствующих учебных дисциплин, направленных на подготовку таких специалистов.

Дисциплины учебных планов подготовки специалистов на первой и второй ступенях получения высшего образования, объединенные общим названием «Компьютерные сети и сетевые технологии», представляют собой введение в сетевую тематику и дают базовые знания и умения по проектированию, используемым стандартам, организации и функционированию сетей на заданном уровне надежности и информационной безопасности.

Из опыта преподавания данной и других смежных дисциплин известно, что структура одно- или двухсеместрового курса сильно отличается от структуры краткого курса по тому же предмету. В данном пособии основные понятия, общие подходы в проектировании компьютерных сетей, особенности различных видов и топологий

сетей, их программного и аппаратного обеспечения и практического использования авторы пытались изложить в контексте унифицированной структуры. При этом предполагалось, что читатель знаком с основами информационных технологий.

Основная задача пособия – дать обучаемым общие систематизированные сведения об организации и структуре важной отрасли, затрагивающей профессиональные, бытовые, познавательно-развлекательные сферы жизнедеятельности человека, которая интенсивно меняется и развивается.

В целом в учебном пособии в простой и доступной форме даны общие понятия компьютерных сетей, их структуры и сетевых компонентов, а также перспективы развития, виды топологий, используемые для физического соединения компьютеров в сети, методы доступа к каналу связи, физические среды передачи данных. Передача данных в сети рассмотрена на базе эталонной базовой модели, разработанной Международной организацией по стандартам взаимодействия открытых сетей. Описаны правила и процедуры передачи данных между информационными системами. Приведены типы сетевого оборудования, их назначение и принципы работы. Дана характеристика сетевого программного обеспечения, используемого для организации сетей. Изучены наиболее популярные сетевые операционные системы, их достоинства и недостатки. Рассмотрены принципы межсетевого взаимодействия, основные понятия из области надежности сетевой безопасности, а также методы и средства обеспечения надежности сетей и безопасного взаимодействия пользователей посредством компьютерных сетей.

Для лучшего зрительного восприятия материала было использовано различное стилевое оформление **наиболее важных определений понятий**, ***детализации этих понятий***, примеров.

Содержание важных выводов выделяется графически с восклицательным знаком.

Основой данной книги является существенно переработанный и дополненный материал учебного пособия [1]. В настоящем издании он излагается в той последовательности, которая, по нашему мнению, является оптимальной с методической точки зрения при изучении его в полном объеме. Хотя, конечно, такая точка зрения не претендует на бесспорность. В то же время мы старались оформить каждую главу в определенной степени автономной, что может повысить эффективность изучения избранных глав.

После разделов даны вопросы для самостоятельного контроля знаний либо контроля знаний обучаемого преподавателем. На основе сформулированных вопросов и заданий могут быть составлены тесты для компьютерного (дистанционного) контроля знаний. Наш опыт показывает, что такая форма контроля достаточно эффективна для студентов, обучающихся не только по заочной (или дистанционной), но и по дневной форме обучения. Это особенно актуально в свете последних тенденций, обусловленных эпидемиологической обстановкой.

Книга предназначена для студентов технических специальностей учреждений высшего образования, получающих знания в области информационных технологий. Может быть полезна магистрантам и аспирантам, изучающим вопросы проектирования, эксплуатации и администрирования компьютерных сетей и систем, а также для выполнения заданий на занятиях по дисциплинам «Компьютерные сети», «Программирование и безопасность сетевых приложений», «Администрирование информационных систем и web-порталов», «Администрирование и безопасность Интернет-систем», а также учебных дисциплин, схожих по содержанию учебных планов.

Пособие рекомендуется также студентам учреждений среднего специального образования, изучающим компьютерные сети и сетевые технологии. В этом случае авторы рекомендуют ограничиться материалами глав 1–3, 6, 8, 10, 12 и подглав 5.1, 5.2.

Известной особенносью литературы по сетевой тематике является использование большого числа русско- и англоязычных аббревитур. Чтобы читатель не испытывал в связи с этим естественных затруднений, основной материал пособия дополнен тремя приложениями, в которых приведены предметный указатель, содержащий отнесенные к соответствующим страницам книги термины, понятия и определения из предметной области, краткие пояснения к основным русско- и англоязычным терминам и понятиям, а также довольно обширный список пояснений к основным аббревиатурам, используемым в англоязычной литературе из анализирумой предметной области. По мнению авторов, эта информация поможет читателю снять вопросы, которые могут возникнуть из-за довольно часто встречающихся в разных источниках несовпадений в определениях понятий и терминов.

Книга написана и подготовлена к изданию в рамках научно-технического сотрудничества между Белорусским государственным

технологическим университетом и Люблинским Католическим университетом Яна Павла II, Польша (The John Paul II Catholic University of Lublin), а также Люблинским техническим университетом (Lublin University of Technology).

Соавтором раздела 13.5 является доцент кафедры информатики и веб-дизайна Белорусского государственного технологического университета Шутько Н. П.

Авторы выражают признательность и благодарность официальным рецензентам: сотрудникам кафедры программного обеспечения информационных технологий Белорусского государственного университета информатики и радиоэлектроники (заведующий кафедрой кандидат технических наук, доцент Лапицкая Н. В.) и доценту кафедры многопроцессорных систем и сетей Белорусского государственного университета кандидату физико-математических наук, доценту Соболевой Т. В., а также проректору по научной работе Гомельского государственного университета имени Франциска Скорины доктору технических наук, профессору Демиденко О. М. и заведующему кафедрой информационных радиотехнологий Белорусского государственного университета информатики и радиоэлектроники доктору технических наук, профессору Листопаду Н. И., чьи благожелательная критика, замечания и рекомендации способствовали, на наш взгляд, не только исправлению отдельных неточностей, но и, главным образом, повышению методологического и методического уровня пособия.

1

ОСНОВНЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ. КЛАССИФИКАЦИЯ СЕТЕЙ

1.1. Основные термины и определения

Международная организация по стандартизации (International Organization for Standardization, ISO) определила **компьютерную сеть** *как последовательную бит-ориентированную передачу информации между связанными друг с другом независимыми устройствами*.

В общем случае различают два типа сетей: *коммуникационную* и *информационную* (рис. 1.1).

Коммуникационная сеть (КС) предназначена для передачи данных, она также выполняет задачи, связанные с преобразованием данных. Коммуникационные сети различаются по типу используемых физических средств соединения.

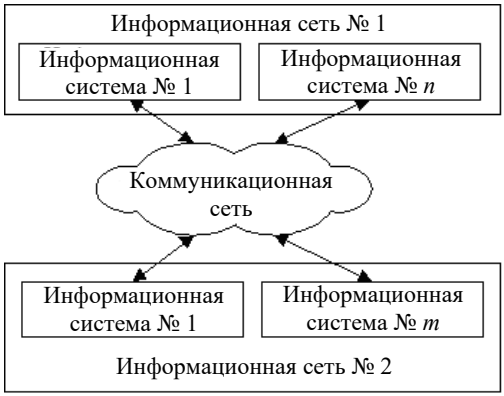


Рис. 1.1. Информационные и коммуникационные сети

Информационная сеть предназначена для хранения информации и состоит из *информационных систем*. На базе коммуникационной сети может быть построена группа информационных сетей.

Под **информационной системой** (ИС) следует понимать систему, которая является поставщиком или потребителем информации.

Вычислительная сеть (ВС) – это одна из разновидностей распределенных систем, предназначенная для распараллеливания вычислений, за счет чего может быть достигнуто повышение производительности и *отказоустойчивости* системы.

Сеть передачи данных (СПД) – совокупности оконечных устройств (терминалов) связи, объединенных каналами передачи данных и коммутирующими устройствами (узлами сети), обеспечивающими обмен сообщениями между всеми оконечными устройствами.

В общем, компьютерная сеть состоит из информационных систем и каналов связи.

Под **информационной системой** в данном случае следует понимать объект, способный осуществлять хранение, обработку или передачу информации. В состав информационной системы входят компьютеры, программы, пользователи и другие составляющие, предназначенные для процесса обработки и передачи данных.

В дальнейшем информационная система, предназначенная для решения задач пользователя, будет называться ***рабочей станцией*** (***client***). Рабочая станция в сети отличается от обычного персонального компьютера (ПК) наличием ***сетевой карты*** (***сетевого адаптера***), канала для передачи данных и сетевого программного обеспечения.

Информационная технология (ИТ) или **информационно-коммуникационная технология** (ИКТ) – это процессы, методы поиска, сбора, хранения, обработки, распространения, предоставления информации и способы осуществления данных процессов и методов с помощью *информационно-вычислительных систем*.

Под **каналом связи** следует понимать путь или средство, по которому передаются сигналы. Средство передачи сигналов называют ***физическим каналом***.

Абонентский канал – это физический канал, соединяющий коммуникационную сеть с абонентской системой. Параметры и характеристики абонентского канала в точке подключения системы определяются абонентским интерфейсом.

Каналы связи создаются по **линиям связи** при помощи сетевого оборудования и физических средств связи.

Физические средства связи построены на основе витых пар, коаксиальных кабелей, оптических каналов или эфира. Между взаимодействующими информационными системами через физические каналы коммуникационной сети и узлы коммутации устанавливаются *логические каналы*.

Ошибка (информационная) – это несоответствие между переданным и принятым символами сообщения или между записанным в память компьютера и считанным символами (по одному и тому же адресу).

Логический канал – это путь для передачи данных от одной системы к другой. Логический канал прокладывается по маршруту в одном или нескольких физических каналах. Логический канал можно охарактеризовать как маршрут, проложенный через физические каналы и узлы коммутации.

Информация в сети передается **блоками данных** по процедурам обмена между объектами. Эти процедуры называют **протоколами передачи данных**.

Протокол – это совокупность правил, устанавливающих формат и процедуры обмена информацией между двумя или несколькими устройствами.

Интерфейс – совокупность средств и методов взаимодействия между элементами или устройствами системы. Интерфейсы являются основой взаимодействия всех современных информационных систем. Если интерфейс какого-либо объекта (рабочей станции, сетевой карты, программы и т. д.) не изменяется (стандартизирован), это дает возможность модифицировать сам объект, не перестраивая принципы его взаимодействия с другими объектами.

Загрузка сети характеризуется параметром, который называется трафиком.

Трафик – это поток сообщений в сети передачи данных. Под ним понимают количественное измерение в выбранных точках сети числа проходящих блоков данных и их длины, выраженное в битах в секунду.

Существенное влияние на характеристику сети оказывает метод доступа.

Метод доступа – это способ определения того, как сеть управляет доступом к каналу связи (кабелю), что существенно влияет на

ее характеристики. В сети все рабочие станции физически соединены между собой каналами связи по определенной структуре, называемой *топологией*.

Топология – это описание соединений в сети, например физических, указывающее, какие рабочие станции могут связываться между собой.

Необходимо отметить, что в настоящее время также используются следующие термины: *физическая топология*, *логическая топология*, *информационная топология* и *топология управления обменом*, которые будут более подробно рассмотрены в пункте 2.2.6.

Тип топологии определяет производительность, работоспособность и надежность эксплуатации рабочих станций, а также время обращения к файловому серверу. В зависимости от топологии сети используется тот или иной метод доступа.

Состав основных элементов в сети зависит от ее *архитектуры*.

Архитектура – это концепция, определяющая взаимосвязь, структуру и функции взаимодействия рабочих станций в сети. Она предусматривает логическую, функциональную и физическую организацию технических и программных средств сети. Архитектура определяет принципы построения и функционирования аппаратного и программного обеспечения элементов сети.

Трехуровневая, или **трехзвенная, архитектура** (Three-Tier) – архитектурная модель комплекса, предполагающая наличие в нем трех компонентов: клиента, сервера приложений (к которому подключено клиентское приложение) и сервера баз данных (с которым работает сервер приложений).

! К важнейшим техническим характеристикам сетей и сетевого оборудования относятся:
– *производительность*;
– *защищенность от несанкционированного доступа* или *безопасность*;
– *надежность*.

Производительность компьютерной сети определяет *скорость* (бит/с) выполнения внутренних операций *сети*, связанных обычно с поиском, сбором, обработкой и передачей информации между узлами *сети* через коммутационные устройства и каналы.

Безопасность информации (в сети) – защищенность информации от нежелательного ее разглашения (нарушения конфиденциальности), искажения (нарушения целостности), утраты или снижения степени доступности информации, а также незаконного ее тиражирования.

Безопасность любого ресурса информационной системы складывается из обеспечения трех его характеристик:
– *конфиденциальности*;
– *целостности*;
– *доступности*.

Иначе: *информационная безопасность* – все аспекты, связанные с определением, достижением и поддержанием конфиденциальности, целостности, доступности информации или средств ее обработки.

Конфиденциальность (Confidentiality) компонента системы заключается в том, что он доступен только тем субъектам доступа (пользователям, программам, процессам), которым предоставлены на то соответствующие полномочия.

Целостность (Integrity) компонента предполагает, что он может быть модифицирован только субъектом, имеющим для этого соответствующие права; целостность является гарантией корректности (неизменности, работоспособности) компонента в любой момент времени.

Доступность (Availability) компонента означает, что имеющий соответствующие полномочия субъект может в любое время без особых проблем получить доступ к необходимому компоненту системы (ресурсу).

Надежность системы – это характеристика способности программного, аппаратного, аппаратно-программного средства (например, компьютерной сети) выполнять при определенных условиях требуемые или заданные в спецификации функции в течение определенного периода времени.

Мы вернемся к данной характеристике предметной области – надежности и безопасности компьютерных сетей и систем – при более углубленном ее изучении в главе 12. С некоторыми специфическими понятиями и терминами будем знакомиться по ходу изучения соответствующих разделов курса.

1.2. Историческая информация о развитии компьютерных и вычислительных сетей и средств

Концепция вычислительных сетей является логическим результатом эволюции *компьютерных технологий* (*КТ*) или *информационных технологий* (*ИТ*). Первые компьютеры 1950-х годов были большими, громоздкими и достаточно дорогими. Такие компьютеры не были предназначены для *интерактивной работы*, а использовались в режиме *пакетной обработки*.

Системы пакетной обработки, как правило, строились на базе *мейнфрейма* (Mainframe) – мощного и надежного компьютера универсального назначения. В Советском Союзе производительные системы этого класса проектировались и создавались в Минске. В 1959 г. начал действовать Минский завод счетных машин имени Орджоникидзе, известный своими мейнфреймами класса «Минск» (Минск-1, 1960 г.; Минск-23, 1966 г.), позднее – Минский завод электронно-вычислительных машин имени Орджоникидзе (ЭВМ третьего поколения ЕС-1020, 1971 г.; ЕС – Единая Серия); Минское производственное объединение вычислительной техники (ЕС-1022, 1975 г.; ЕС-1060, 1976 г.; позже – ЕС-1061, ЕС-1065, ЕС-1066 и др.).

Основным носителем информации, а также кодов программ для перечисленных ЭВМ являлись бумажные и магнитные ленты, а также *перфокарты* (перфорированные карты, 187,325×82,55 мм – стандарт IBM) – небольшие картонные пластинки, на которых можно было разместить примерно 80 символов. Операторы вводили эти карты в компьютер, а распечатанные результаты пользователи получали обычно только на следующий день. Таким образом, одна неверно «набитая» карта означала примерно суточную задержку выполнения программы.

Конечно, *интерактивный режим* (Online Mode) работы, при котором можно с терминала оперативно руководить процессом обработки своих данных, был бы гораздо удобней. Но интересами программистов и пользователей на первых этапах развития вычислительных систем в значительной степени пренебрегали, поскольку *пакетный режим* (Batch Mode) – это самый эффективный режим использования вычислительной мощности, так как он позволяет

выполнить в единицу времени больше задач, чем любые другие режимы. Во главу угла ставилась эффективность работы самого важного и дорогого устройства вычислительной машины – процессора, в ущерб эффективности работы использующих его специалистов.

1.2.1. Многотерминальные системы

По мере удешевления процессоров в начале 1960-х годов появились новые способы организации вычислительного процесса, которые позволили учесть интересы пользователей. Начали развиваться интерактивные *многотерминальные системы* разделения времени (рис. 1.2).

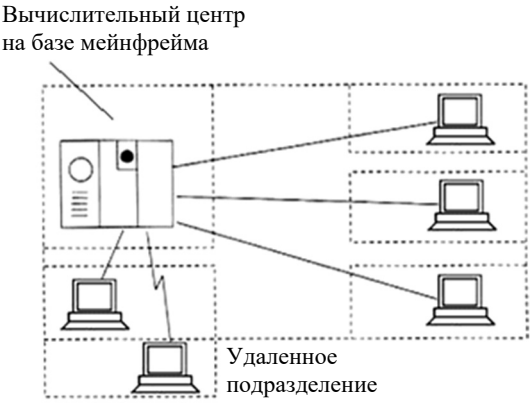


Рис. 1.2. Многотерминальная система – прообраз вычислительной сети

В таких системах компьютер использовали сразу несколько программистов-пользователей. Причем время реакции вычислительной системы было достаточно мало для того, чтобы пользователю была не слишком заметна параллельная работа с компьютером и других программистов-пользователей.

Терминалы, выйдя за пределы вычислительного центра, рассредоточились по всему предприятию. И хотя вычислительная мощность оставалась полностью централизованной, некоторые функции – такие как ввод и вывод данных – стали *распределенными*. Такие многотерминальные централизованные системы внешне уже были очень

похожи на *локальные вычислительные сети* (ЛВС). Действительно, рядовой программист-пользователь работу за терминалом мейнфрейма воспринимал примерно так же, как сейчас он воспринимает работу за подключенным к сети персональным компьютером.

Многотерминальные системы, работающие в режиме разделения времени, стали первым шагом на пути создания локальных вычислительных сетей.

Но до появления локальных сетей нужно было пройти еще большой путь, так как многотерминальные системы, хотя и имели внешние черты распределенных систем, все еще сохраняли централизованный характер обработки данных. К тому же потребность предприятий в создании локальных сетей в это время еще не созрела – в одном здании просто нечего было объединять в сеть, так как из-за высокой стоимости вычислительной техники предприятия не могли себе позволить роскошь приобретения нескольких компьютеров. В этот период был справедлив так называемый закон Гроша, который эмпирически отражал уровень технологии того времени. В соответствии с этим законом производительность компьютера была пропорциональна квадрату его стоимости, отсюда следовало, что за одинаковую сумму было выгоднее купить одну мощную машину, чем две менее мощных – их суммарная мощность оказывалась намного ниже мощности дорогой машины.

1.2.2. Зарождение глобальных сетей

Тем не менее потребность в соединении компьютеров, находящихся на большом расстоянии друг от друга, к этому времени вполне назрела.

В 1958 г. Американское агентство перспективных исследовательских проектов Министерства обороны США (Advanced Research Projects Agency of the U.S. Department of Defense, ARPA) начало реализацию проекта, который позднее получил название ARPANET и из которого позднее вырос современный Интернет (Internet).

В 1962 г. важные исследования были начаты в ряде учебных заведений США и прежде всего в Массачусетском технологическом институте (MIT). Именно в 1962 г. молодой американский ученый из

MIT Дж. С. Ликлидер написал работу, где высказал идею *глобальной сети*, которая обеспечивала бы каждому жителю земли доступ к данным и программам из любой точки земного шара. В это же время другой ученый, Л. Клейнрок, закончил работу над своей докторской диссертацией в области теории коммуникационных сетей. В 1963 г. происходит важное событие: появляется первый универсальный **стандарт ASCII** (American Standard Code for Information Interchange) – схема кодирования, назначающая численные значения-коды буквам, цифрам, знакам пунктуации и некоторым другим символам, в результате чего возникает возможность обмена информацией между компьютерами от различных изготовителей.

В 1964 г. практически одновременно в MIT, RAND Corporation и Great Britain National Physical Laboratory (GBNPL) были развернуты работы по надежной передаче информации. Появилась идея *коммутации пакетов*, суть которой сводилась к тому, что любая информация, передаваемая по сети, разбивается на несколько частей (пакетов), которые затем независимо друг от друга перемещаются различными путями (маршрутами), пока не достигнут адресата. П. Бэран, Д. Дэвис, Л. Клейнрок параллельно вели исследования в этой области. П. Бэран был одним из первых, кто опубликовал свои исследования в статье «Передача данных в сетях».

В 1967 г. произошло событие, которое сыграло важную роль в развитии сетевых технологий: *модем*, изобретенный в начале шестидесятых, был существенно усовершенствован Дж. ван Гином из Станфордского научно-исследовательского института (Stanford Research Institute, SRI). Отметим, что само название прибора (модем) состоит из начальных букв двух слов: **мо**дулятор и **дем**одулятор; в нашем случае первый выполняет операцию преобразования цифрового сигнала в аналоговый (ЦАП), второй – преобразование аналогового в цифровой (АЦП).

Дж. ван Гин предложил приемник, который мог надежно распознавать биты информации на фоне шумовых помех, создаваемых междугородними телефонными линиями. В 1967 г. Л. Робертс организовал научную конференцию в Анн-Арборе (штат Мичиган), на которую он пригласил основных разработчиков сетевого проекта. Конференция имела огромное значение – параллельно проводимые работы начали объединяться. Термин ARPANET впервые упоминался в ходе выступления Л. Робертса именно на этой конференции. На этой же конференции другой выдающийся ученый У. Кларк впервые высказал

идею и предложил термин IMP – Interface Message Processors, обозначающий *устройства для управления трафиком в сети*, которые впоследствии эволюционировали в современные *маршрутизаторы*.

В 1968 г. началась работа по созданию IMP и уже через один год, в 1969 г., заработала сеть ARPANET, охватившая все Западное побережье США.

В 1970 г. наблюдается рост сети – каждый месяц добавляется новый узел. В этом же году произошло еще два важных события. Во-первых, Д. Ритчи и К. Томпсон из BelLabs закончили работу над созданием операционной системы UNIX. Во-вторых, в этом же году рабочая группа NWG (Network Working Group) под руководством С. Крокера завершила работу над протоколом NCP (Network Control Protocol), а еще годом позже закончила работу над *протоколом эмуляции терминала* (*Telnet*) и существенно продвинулась в работе над *протоколом передачи файлов* (*FTP*). В 1971 г. BBN разработала новую платформу – так называемые TIP-устройства (Terminal IMP, Terminal Interface Processor), что обеспечило возможность входить на удаленные хосты, сделав, таким образом, ARPANET доступной большему числу пользователей.

В 1972 г. сеть ARPANET была публично продемонстрирована. Однако в этом же году также произошло еще, по крайней мере, два события, которые оказали огромное влияние на развитие компьютерных технологий: Р. Томильсон написал программу, позволяющую отправлять *электронную почту* по ARPANET, ввел обозначение *user@host* и использовал символ @ (коммерческое *эт*; commercial *at*), который позднее (с 1980 г.) был закреплен в международном стандарте адресов электронной почты. Постепенно сеть ARPANET расширялась, и среди клиентов появились такие частные организации, как BBN, Xerox PARC и MITRE Corporation, а также государственные – NASA's Ames Research Laboratories, National Bureau of Standards и Air Force Research Facilities.

В 1973 г. фирма ARPA переименовывается в DARPA, где буква «D» указывает на Defense (защита, оборона). DARPA (Defense Advanced Research Projects Agency) – агенство перспективного планирования оборонных научно-исследовательских работ, центральная научно-исследовательская организация Министерства обороны США, основной целью которой является выдача рекомендаций по внедрению принципиально новых технологий для военной промышленности. Под руководством Б. Кана начинается весьма сложная работа по объединению сетей, имеющих разные интерфейсы, скорости передачи

данных и размеры пакетов. По сути дела, это была работа по созданию *межсетевого протокола*. В сентябре 1973 г. появилась первая публикация по новому протоколу *TCP* (Transmittion Control Protocol). TCP/IP со временем стал одним из наиболее популярных протоколов сетевого взаимодействия и стандартом для реализации глобальных сетевых соединений в силу открытости, масштабируемости и за счет предоставления одинаковых возможностей глобальным и локальным сетям.

В 1977 г. был анонсирован компьютер Apple II, и появление настольных компьютеров с потенциальной возможностью коммуникаций при помощи модемного подключения дало новый толчок развитию *сетевых технологий* и *модемной* индустрии. К началу 1978 г. эксперимент ARPANET был практически закончен. Годом позже появилась служба USENET4, которая стала одним из первых примеров клиент-серверной организации.

К концу семидесятых годов архитектура и протоколы TCP/IP приобрели современный вид. К этому времени агентство DARPA стало признанным лидером в разработке сетей с коммутацией пакетов. Дальнейшее развитие сетевых технологий, в том числе беспроводных радиосетей и спутниковых каналов связи, стимулировало активность DARPA в исследовании проблем межсетевого взаимодействия и реализации принципов Интернета в ARPANET. DARPA не делало тайны из своей деятельности в области развития технологий Интернета, поэтому различные научные группы проявляли интерес к разработкам технологии глобальной сети.

Свое начало Интернет берет от сети ARPANET, но чаще всего Интернет называют наследником NSFNET – американской сети, которая объединила ученых NSF (National Science Foundation) и сначала сотрудничала с сетью ARPANET, а затем слилась с ней и поглотила.

Многие эксперты называют временем зарождения Интернета начало 1980-х годов. В это время агентства DARPA инициировало перевод машин, подсоединенных к его исследовательским сетям, на использование стека TCP/IP. В 1981 г. IWG (Internet Working Group) в DARPA публикует документ, в котором говорится о полном переходе с протокола NCP (Network Control Protocol) на протокол TCP/IP. С этих

пор ARPANET становится магистральной сетью Интернет и активно используется для многочисленных экспериментов с TCP/IP. Окончательный переход к технологии Интернет произошел в январе 1983 г.: в этом году протокол TCP/IP был принят Министерством обороны США, а сеть ARPANET была разбита на две независимые части. Одна из них (предназначенная для научных целей) сохранила название ARPANET, а вторая, большая по масштабу сеть MILNET, отошла к военному ведомству.

Для того чтобы стимулировать использование новых протоколов в учебных заведениях, агентство DARPA сделало реализацию TCP/IP широко доступной для университетских кругов. В это время многие исследователи использовали версию ОС Unix университета Беркли (штат Калифорния), называемую BSD Unix (от Berkeley Software Distribution). Благодаря тому что DARPA в свое время субсидировало компанию BBN и университет в Беркли с целью реализации протоколов TCP/IP для использования вместе с популярной ОС Unix, более 90% компьютерных факультетов университетов адаптировали новую сетевую технологию, и версия BSD стала фактическим стандартом для реализаций стека протоколов TCP/IP. Было выпущено несколько версий BSD, каждая из которых добавляла в TCP/IP новые возможности, в том числе 4.2BSD (1983 г.), 4.3BSD (1986 г.), 4.3BSD Tahoe (1988 г.), 4.3BSD Reno (1990 г.), 4.4BSD (1993 г.).

С 1985 г. сеть NSF реализовала программу создания сетей вокруг своих суперкомпьютерных центров. И в 1986 г. создание опорной сети (56 Кбит/с) между суперкомпьютерными центрами NSF привело к появлению целого ряда региональных сетей, таких как JVNCNET, NYSERNET, SURANET, SDSCNET, BARRNET и др. Так появилась магистральная сеть NSFNET, которая в конце концов объединила все эти научные центры и связала их с ARPANET. Таким образом, NSFNET связала пять суперкомпьютерных центров и открыла доступ к мощным вычислительным ресурсам для широкого круга исследователей. Для уменьшения платы за использование междугородних линий связи решено было развивать систему *региональных сетей*, которая объединяет компьютеры внутри какого-то региона и имеет выходы на подобные сети поблизости. При такой конфигурации все компьютеры являются равноправными и имеют связь «по цепочке» через соседние компьютеры как друг с другом, так и с суперкомпьютерами NSF. Таким образом, начиная с 1986 г. можно говорить о становлении глобальной компьютерной сети Интернет.

В 1988 г. Интернет становится международной сетью – к нему присоединяются Канада, Дания, Финляндия, Франция, Норвегия и Швеция. Годом позже сеть уже насчитывала 80 000 узлов, а в ноябре еще присоединились Австрия, Германия, Израиль, Италия, Япония, Мексика, Нидерланды, Новая Зеландия и Великобритания – и вскоре количество узлов в сети выросло до 160 000. В том же году появилась технология FDDI (Fiber Distributed Data Interface) – распределенный *интерфейс передачи данных по волоконно-оптическим каналам*.

Если Интернет – изобретение коллективное, то идею гипертекста и WWW связывают с именем конкретного человека.

В 1989 г. Т. Бернерс-Ли высказал идею гипертекста, которая и послужила толчком к созданию *World Wide Web*.

Т. Бернерс-Ли написал программу *Eniquire*, которая стала прообразом будущей WWW. В том же 1989 г. Т. Бернерс-Ли начал работу над глобальным проектом Всемирной паутины, и всего два года спустя (в 1991 г.) первые WWW-объекты были помещены в Интернет.

1.2.3. Первые локальные сети

В начале 1970-х годов произошел технологический прорыв в области производства компьютерных компонентов – появились *большие интегральные схемы* (БИС). Их сравнительно невысокая стоимость и высокие функциональные возможности привели к созданию мини-компьютеров, которые стали реальными конкурентами мейнфреймов. Закон Гроша перестал соответствовать действительности, так как десяток мини-компьютеров выполнял некоторые задачи (как правило, хорошо распараллеливаемые) быстрее одного мейнфрейма, а стоимость такой *мини-компьютерной системы* была меньше.

Даже небольшие подразделения предприятий получили возможность покупать для себя компьютеры. Мини-компьютеры выполняли задачи управления технологическим оборудованием, складом и другие задачи уровня подразделения предприятия. Таким образом, появилась концепция *распределения компьютерных ресурсов* по всему предприятию. Однако при этом все компьютеры одной организации по-прежнему продолжали работать автономно (рис. 1.3).

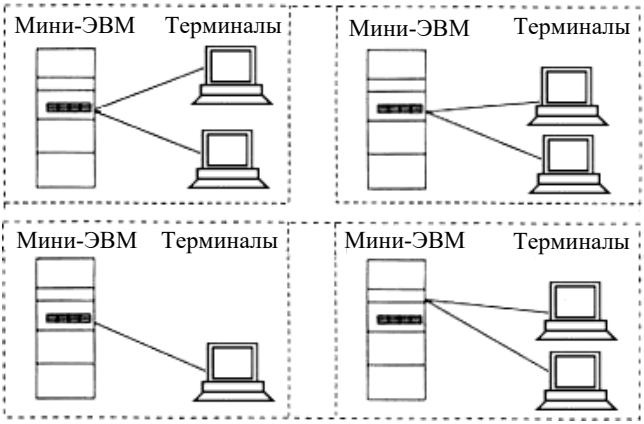


Рис. 1.3. Автономное использование
нескольких мини-компьютеров
на одном предприятии

Позднее предприятия и организации стали соединять свои мини-компьютеры вместе и разрабатывать программное обеспечение, необходимое для их взаимодействия. В результате появились первые *локальные вычислительные сети* (*ЛВС*) (рис. 1.4). Они еще во многом отличались от современных локальных сетей, в первую очередь своими устройствами сопряжения.

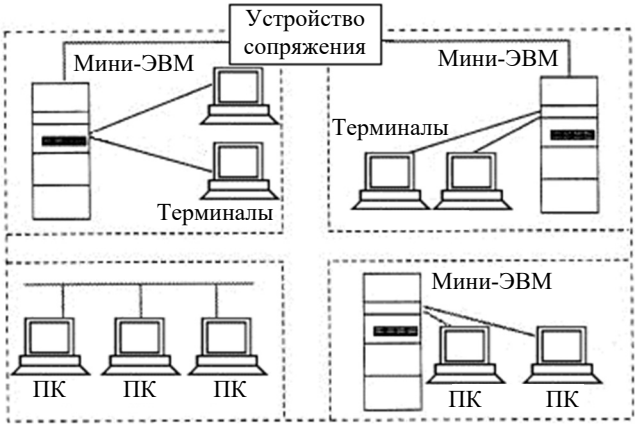


Рис. 1.4. Различные типы связей
в первых локальных сетях

На первых порах для соединения компьютеров друг с другом использовались самые разнообразные нестандартные устройства со своим способом представления данных на линиях связи, своими типами кабелей и т. п.

1.2.4. Стандарты технологий локальных сетей

В середине 1980-х гг. положение дел в локальных сетях стало кардинально меняться. Утвердились стандартные технологии объединения компьютеров в сеть – Ethernet, ARCnet, Token Ring. Мощным стимулом для их развития послужили персональные компьютеры, которые стали преобладать в локальных сетях, причем не только в качестве клиентских компьютеров, но и как центры хранения и обработки данных, т. е. сетевые серверы, потеснившие со своих привычных ролей мини-компьютеры и мейнфреймы.

Стандартные сетевые технологии превратили процесс построения локальной сети из искусства в рутинную работу. Для создания сети достаточно было приобрести *сетевые адаптеры* соответствующего стандарта, например Ethernet, стандартный кабель, присоединить адаптеры к кабелю стандартными разъемами и установить на компьютер одну из популярных сетевых операционных систем, например NetWare.

Локальные сети в сравнении с глобальными внесли много нового в способы организации работы пользователей. Доступ к разделяемым ресурсам стал гораздо удобнее – пользователь мог просто просматривать списки имеющихся ресурсов, а не запоминать их идентификаторы или имена. После соединения с удаленным ресурсом можно было работать с ним с помощью уже знакомых пользователю по работе с локальными ресурсами команд. Реализацию всех нововведений разботчики локальных сетей получили в результате появления качественных кабельных линий связи, на которых даже сетевые адаптеры первого поколения обеспечивали скорость передачи данных до 10 Мбит/с.

Наибольшее распространение на тот момент получили телефонные каналы связи, но они были плохо приспособлены для высокоскоростной передачи *дискретных данных* – скорость в 1200 бит/с была для них хорошим достижением. Поэтому экономное расходование *пропускной способности* каналов связи часто являлось основным критерием эффективности методов передачи данных в глобальных сетях.

1.2.5. Основные тенденции развития компьютерных вычислительных сетей

Компьютерные сети и сетевые технологии стали определяющим фактором и важнейшим инструментом инноваций практически во всех областях человеческой деятельности.

Разрыв между локальными и глобальными сетями сократился во многом из-за появления высокоскоростных территориальных каналов связи, не уступающих по качеству кабельным системам локальных сетей.

Возродился интерес к крупным компьютерам в основном из-за того, что после спада эйфории по поводу легкости работы с персональными компьютерами выяснилось, что системы, состоящие из сотен серверов, обслуживать сложнее, чем несколько больших компьютеров. Поэтому на новом витке эволюционной спирали мейнфреймы стали возвращаться в *корпоративные вычислительные системы*, но уже в качестве полноправных сетевых узлов, поддерживающих Ethernet или Token Ring, а также стек протоколов TCP/IP, ставший благодаря глобальной сети Интернет сетевым стандартом де-факто.

Обработка и передача в сетях различных видов мультимедийной информации (голоса, видеоизображений, рисунков, текста) потребовали внесения изменений в работу протоколов, сетевых операционных систем и коммуникационного оборудования. Сложность передачи такой мультимедийной информации по сети связана с ее чувствительностью к задержкам при передаче пакетов данных. Задержки обычно приводят к искажению такой информации в конечных узлах сети. Так как традиционные службы вычислительных сетей (передача файлов или электронная почта) создают малочувствительный к задержкам трафик и все элементы сетей разрабатываются в расчете на него, то появление трафика реального времени приводит к большим проблемам.

Эти проблемы решаются различными способами, в том числе и с помощью специально рассчитанной на передачу различных типов трафика технологии ATM (Asynchronous Transfer Mode – *асинхронный способ передачи данных*).

В последнее время, особенно в период вынужденного перехода к дистанционной работе сотен миллионов человек по всему миру, роль компьютерных сетей и сетевого оборудования неизмеримо возросла.

От базового сетевого оборудования, в равной мере пригодного для установки дома или на небольшом предприятии, до недавних пор не так уж много требовалось: обеспечить работу локальной сети да предоставить каждому ее узлу доступ к облачным сервисам и Интернету в целом. Теперь ситуация меняется принципиально: живые контакты с клиентами массированно переходят в онлайн, офис становится существенно распределенным, деловые переговоры, совещания проводятся по видеоконференцсвязи.

В последнее время специалисты отмечают 5 ключевых технологий, преобразующих сети:

– *программно-определяемые* (или *программно-реконфигурируемые*) *сети* (Software-Defined Networking, SDN);

– *облачные сервисы* (Cloud Computing Services, CCS);

– *интернет вещей* (Internet of Things, IoT);

– *виртуализация сетевых функций* (Network Functions Virtualization, NFV);

– *качество взаимодействия (восприятия)* (Quality of Experience, QoE).

Важнейшие особенности указанных технологий будут рассмотрены ниже.

1.3. Классификации компьютерных сетей

Чаще всего термин *локальные сети* (Local Area Network, LAN), а также *локальные вычислительные сети* понимают буквально, т. е. это такие сети, которые имеют небольшие, локальные размеры и соединяют близко расположенные компьютеры. Однако некоторые локальные сети легко обеспечивают связь на расстоянии нескольких десятков километров. С другой стороны, по глобальной сети (WAN, Wide Area Network или GAN, Global Area Network) вполне могут связываться компьютеры, находящиеся на соседних столах в одной комнате.

Как правило, локальная сеть связывает от двух до нескольких десятков компьютеров. Но предельные возможности современных локальных сетей гораздо выше: максимальное число абонентов может достигать тысячи. Называть такую сеть малой неправильно.

В пределах одной сети могут использоваться как электрические кабели различных типов (витая пара, коаксиальный кабель), так и оптоволоконные кабели.

По сути, 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. Под удобством в данном случае понимается высокая реальная скорость доступа, скорость обмена информацией между приложениями, практически незаметная для пользователя.

Таким образом, главное отличие локальной сети от любой другой – высокая скорость передачи информации по сети. Но это еще не все, не менее важны и другие факторы. В частности, принципиально необходим низкий уровень *ошибок* передачи, вызванных как внутренними, так и внешними факторами. Ведь даже очень быстро переданная информация, которая искажена ошибками, просто не имеет смысла, ее придется передавать еще раз, что в итоге будет проявляться как снижение интегральной скорости передачи по сети.

Особое значение имеет и такая характеристика сети, как возможность работы с большими нагрузками, т. е. с высокой интенсивностью обмена (или, как еще говорят, с большим *трафиком*). Ведь если механизм управления обменом, используемый в сети, не слишком эффективен, то компьютеры могут подолгу «ждать своей очереди» на передачу.

Механизм управления обменом может гарантированно успешно работать только в том случае, когда заранее известно, сколько компьютеров (или, как еще говорят, абонентов, узлов) допустимо подключить к сети. Иначе всегда можно включить столько абонентов, что вследствие перегрузки замедлится любой механизм управления. Наконец, сетью можно назвать только такую систему передачи данных, которая позволяет объединять до нескольких десятков компьютеров, но никак не два, как в случае связи через стандартные порты.

Таким образом, сформулировать отличительные признаки локальной сети можно следующим образом:

– *высокая скорость передачи* информации, большая пропускная способность сети;

– *низкий уровень ошибок передачи* или *информационных ошибок* (обеспечивается, прежде всего, высококачественными каналами связи).

Считается, что допустимая вероятность ошибок передачи данных не должна превышать 10^{-6} (практика показывает, что обычно уровень ошибок выше; каналы связи, особенно проводные каналы большой протяженности и радиоканалы, обеспечивают вероятность ошибки на уровне 10^{-3}–10^{-4});

– эффективный, быстродействующий механизм управления обменом по сети;

– заранее четко ограниченное количество компьютеров, подключаемых к сети.

При таком определении очевидно отличие глобальных сетей от локальных, которое состоит в том, что они рассчитаны на неограниченное число абонентов. Кроме того, они используют (или могут использовать) не слишком качественные каналы связи и сравнительно низкую скорость передачи. А механизм управления обменом в них не может быть гарантированно быстрым. В глобальных сетях гораздо важнее не качество связи, а сам факт ее существования.

Нередко выделяют еще один класс компьютерных сетей – **городские**, **региональные** или **муниципальные** сети (MAN, Metropolitan Area Network), которые обычно по своим характеристикам ближе к глобальным сетям, хотя иногда все-таки имеют некоторые черты локальных сетей, например высококачественные каналы связи и сравнительно высокие скорости передачи. В принципе городская сеть может быть локальной со всеми ее преимуществами.

Сейчас нельзя провести четкую границу между локальными и глобальными сетями. Большинство локальных сетей имеет выход в глобальную. Но характер передаваемой информации, принципы организации обмена, режимы доступа к ресурсам внутри локальной сети, как правило, сильно отличаются от тех, что приняты в глобальной сети.

По локальной сети может передаваться самая разная цифровая информация: данные, изображения, телефонные разговоры, электронные письма и т. д. Кстати, именно задача передачи изображений, особенно полноцветных динамических, предъявляет самые высокие требования к быстродействию сети.

Чаще всего локальные сети используются для разделения (совместного использования) таких ресурсов, как дисковое пространство, принтеры и выход в глобальную сеть, но это всего лишь незначительная часть тех возможностей, которые предоставляют средства локальных сетей.

1.4. Вычислительные сети – частный случай распределенных систем

Компьютерные сети относятся к **распределенным** (или децентрализованным) вычислительным системам. Поскольку основным признаком распределенной вычислительной системы является наличие нескольких центров обработки данных, то наряду с компьютерными сетями к распределенным системам относят также *мультипроцессорные* компьютеры.

В **мультипроцессорных компьютерах** имеется несколько процессоров, каждый из которых может относительно независимо от остальных выполнять свою программу.

В мультипроцессоре существует общая для всех процессоров операционная система, которая оперативно распределяет вычислительную нагрузку между процессорами. Взаимодействие между отдельными процессорами организуется наиболее простым способом – через общую оперативную память.

Сам по себе процессорный блок не является законченным компьютером и поэтому не может выполнять программы без остальных блоков мультипроцессорного компьютера – памяти и периферийных устройств. Все периферийные устройства являются для всех процессоров мультипроцессорной системы общими. *Территориальную распределенность* мультипроцессор не поддерживает – все его блоки располагаются в одном или нескольких близко расположенных конструктивах, как и у обычного компьютера. Основное достоинство мультипроцессора – его высокая производительность, которая достигается за счет параллельной работы нескольких процессоров.

Еще одним важным свойством мультипроцессорных систем является **отказоустойчивость**, т. е. способность к продолжению работы при отказах некоторых элементов (процессоров или блоков памяти).

Многомашинная система – это вычислительный комплекс, включающий в себя несколько компьютеров (каждый из которых работает под управлением собственной операционной системы), а также программные и аппаратные средства связи компьютеров, которые обеспечивают работу всех компьютеров комплекса как единого целого.

Работа любой многомашинной системы определяется двумя главными компонентами: высокоскоростным механизмом связи процессоров и системным программным обеспечением, которое предоставляет

пользователям и приложениям прозрачный доступ к ресурсам всех компьютеров, входящих в комплекс. В состав средств связи входят программные модули, занимающиеся распределением *вычислительной нагрузки*, *синхронизацией вычислений* и *реконфигурацией системы*. Если происходит отказ одного из компьютеров комплекса, его задачи могут быть автоматически переназначены и выполнены на другом компьютере. Если в состав многомашинной системы входят несколько контроллеров внешних устройств, то в случае отказа одного из них другие контроллеры автоматически подхватывают его работу. Таким образом достигается высокая отказоустойчивость комплекса в целом.

Многомашинные системы позволяют достичь высокой производительности за счет организации *параллельных вычислений*.

По сравнению с мультипроцессорными системами возможности параллельной обработки в многомашинных системах ограничены: эффективность распараллеливания резко снижается, если параллельно выполняемые задачи тесно связаны между собой по данным.

В вычислительных сетях программные и аппаратные связи являются более слабыми, а автономность обрабатывающих блоков проявляется в наибольшей степени – основные элементы сети (стандартные компьютеры) не имеют ни общих блоков памяти, ни общих периферийных устройств. Связь между компьютерами осуществляется с помощью специальных периферийных устройств – *сетевых адаптеров*, соединенных относительно протяженными каналами связи.

Каждый компьютер работает под управлением собственной операционной системы, а какая-либо «общая» операционная система, распределяющая работу между компьютерами сети, отсутствует. Взаимодействие между компьютерами сети происходит за счет передачи сообщений через сетевые адаптеры и каналы связи. С помощью этих сообщений один компьютер обычно запрашивает доступ к локальным ресурсам другого компьютера. Такими ресурсами могут быть как данные, хранящиеся на диске, так и разнообразные периферийные устройства – принтеры, модемы, факс-аппараты и т. д. Разделение локальных ресурсов каждого компьютера между всеми пользователями сети – основная цель создания вычислительной сети.

На тех компьютерах, ресурсы которых должны быть доступны всем пользователям сети, необходимо добавить модули (приложения),

которые постоянно будут находиться в режиме ожидания *запросов*, поступающих по сети от других компьютеров.

Обычно такие модули называются *серверными приложениями* или **программными серверами** (Server), так как их главная задача – обслуживать запросы на доступ к ресурсам своего компьютера.

На компьютерах, пользователи которых хотят получать доступ к ресурсам других компьютеров, также нужно добавить к операционной системе некоторые специальные программные модули, которые должны вырабатывать запросы на доступ к удаленным ресурсам и передавать их по сети на нужный компьютер.

Такие модули обычно называют *клиентскими приложениями* или **программными клиентами** (Client).

Сетевые адаптеры и каналы связи решают в сети достаточно простую задачу – они передают сообщения с запросами и ответами от одного компьютера к другому, а основную работу по организации совместного использования ресурсов выполняют клиентские и серверные части операционных систем.

Пара модулей *клиент-сервер* обеспечивает совместный доступ пользователей к определенному типу ресурсов, например к файлам. В этом случае говорят, что пользователь имеет дело с файловой *службой* (Service). Обычно сетевая операционная система поддерживает несколько видов сетевых служб для своих пользователей – файловую службу, службу печати, службу электронной почты, службу удаленного доступа и т. п.

Термины «клиент» и «сервер» используются для обозначения не только программных модулей, но и компьютеров, подключенных к сети. *Если компьютер предоставляет свои ресурсы другим компьютерам сети, то он называется сервером, а если он их потребляет – клиентом* (более подробно это описано в п. 2.1.3). Иногда (в распределенных системах) один и тот же компьютер может одновременно выполнять роль и сервера, и клиента.

Сетевые службы всегда представляют собой распределенные программы.

Распределенная программа – это программа, которая состоит из нескольких взаимодействующих частей, причем каждая часть, как правило, выполняется на отдельном компьютере сети.

В сети могут выполняться и распределенные пользовательские программы – *приложения*. Распределенное приложение также состоит из нескольких частей, каждая из которых выполняет какую-то

определенную законченную работу по решению прикладной задачи. Например, одна часть приложения, выполняющаяся на компьютере пользователя, может поддерживать специализированный графический интерфейс, вторая – работать на мощном выделенном компьютере и заниматься статистической обработкой введенных пользователем данных, а третья – заносить полученные результаты в базу данных на компьютере с установленной стандартной ***СУБД*** (системой управления базами данных). Распределенные приложения в полной мере используют потенциальные возможности распределенной обработки, предоставляемые вычислительной сетью, и поэтому часто называются сетевыми приложениями.

1.5. Основные программные и аппаратные компоненты сети

Вычислительная сеть – это сложный комплекс взаимосвязанных и согласованно функционирующих программных и аппаратных компонентов.

Изучение сети в целом предполагает знание принципов работы ее отдельных элементов:

– компьютеров;
– коммуникационного оборудования;
– операционных систем;
– сетевых приложений.

Весь комплекс программно-аппаратных средств сети может быть описан *многослойной моделью*. В основе любой сети лежит аппаратный слой стандартизованных компьютерных платформ.

В настоящее время в сетях широко и успешно применяются компьютеры различных классов – от персональных до мейнфреймов и суперЭВМ. Набор компьютеров в сети должен соответствовать набору разнообразных задач, решаемых сетью.

Второй слой – это коммуникационное оборудование. Хотя компьютеры и являются центральными элементами обработки данных в сетях, в последнее время не менее важную роль стали играть коммуникационные устройства. Кабельные системы, повторители, мосты, коммутаторы, маршрутизаторы и модульные концентраторы из

вспомогательных компонентов сети превратились в основные наряду с компьютерами и системным программным обеспечением как по влиянию на характеристики сети, так и по стоимости. Сегодня коммуникационное устройство может представлять собой сложный специализированный мультипроцессор, который нужно конфигурировать, оптимизировать и администрировать. Изучение принципов работы коммуникационного оборудования требует знакомства с большим количеством протоколов, используемых как в локальных, так и глобальных сетях.

Третьим слоем, образующим программную платформу сети, являются операционные системы (ОС). От того, какие концепции управления локальными и распределенными ресурсами положены в основу сетевой ОС, зависит эффективность работы всей сети. При проектировании сети важно учитывать, насколько просто данная операционная система может взаимодействовать с другими ОС сети, насколько она обеспечивает безопасность и защищенность данных, до какой степени она позволяет наращивать число пользователей, можно ли перенести ее на компьютер другого типа и многие другие соображения.

Самым верхним слоем сетевых средств являются различные сетевые приложения, такие как сетевые базы данных, почтовые системы, средства архивирования данных, системы автоматизации коллективной работы и др. Очень важно представлять диапазон возможностей приложений для различных областей применения, а также знать, насколько они совместимы с другими сетевыми приложениями и операционными системами.

1.6. Преимущества и проблемы использования сетей

Основные преимущества сетей вытекают из их принадлежности к распределенным системам.

Концептуальным преимуществом распределенных систем (а значит, и сетей) перед централизованными системами является их способность выполнять параллельные вычисления. За счет этого в системе с несколькими обрабатывающими узлами в принципе может

быть достигнута производительность, превышающая максимально возможную на данный момент производительность любого отдельного, сколь угодно мощного процессора. Распределенные системы потенциально имеют лучшее соотношение производительность – стоимость, чем централизованные системы.

Еще одно очевидное и важное достоинство распределенных систем – это их принципиально более высокая отказоустойчивость или надежность. Основой повышенной надежности распределенных систем является **избыточность**.

Различают *информационную* (на основе помехоустойчивого кодирования данных), *временну́ю* и *аппаратную* (*структурную*) избыточность. Избыточность обрабатывающих узлов (процессоров в многопроцессорных системах или компьютеров в сетях) позволяет при отказе одного узла переназначать приписанные ему задачи на другие узлы.

С этой целью в распределенной системе могут быть предусмотрены процедуры динамической или статической реконфигурации. В вычислительных сетях некоторые наборы данных могут дублироваться на внешних запоминающих устройствах нескольких компьютеров сети, так что при отказе одного их них данные остаются доступными.

Для пользователя, кроме вышеназванных, распределенные системы дают еще и такие преимущества, как возможность совместного использования данных и устройств, а также возможность гибкого распределения работ по всей системе. Такое разделение дорогостоящих периферийных устройств, таких как дисковые массивы большой емкости, цветные принтеры, графопостроители, модемы, оптические диски, во многих случаях является основной причиной развертывания сети на предприятии.

В последнее время стал преобладать другой побудительный мотив развертывания сетей, гораздо более важный в современных условиях, чем экономия средств за счет разделения между сотрудниками корпорации дорогой аппаратуры или программ. Этим мотивом стало стремление обеспечить сотрудникам оперативный доступ к обширной корпоративной информации. В условиях жесткой конкурентной борьбы в любом секторе рынка выигрывает в конечном

счете та компания, сотрудники которой могут быстро и правильно ответить на любой вопрос клиента: о возможностях их продукции, об условиях ее применения, о решении любых возможных проблем и т. п.

Имеющиеся сейчас технологии поддерживают достаточно простой способ представления текстовой и графической информации в виде гипертекстовых страниц, что позволяет быстро поместить самую свежую информацию на WWW-серверы корпорации. Кроме того, она унифицирует просмотр информации с помощью стандартных программ – web-браузеров, работа с которыми несложна даже для неспециалиста.

Корпоративная сеть, которая интегрирует данные и мультимедийную информацию, может использоваться для организации аудио- и видеоконференций. Кроме того, на ее основе может быть создана собственная внутренняя телефонная сеть.

Корпоративная сеть (Corporate Network) – коммуникационная система, принадлежащая и/или управляемая единой организацией в соответствии с правилами этой организации. Корпоративная сеть отличается от сети, например, интернет-провайдера тем, что правила распределения IP-адресов, работы с интернет-ресурсами и т. д. едины для всей корпоративной сети, в то время как провайдер контролирует только ее магистральный сегмент, позволяя своим клиентам самостоятельно управлять сегментами, которые могут являться как частью адресного пространства провайдера, так и быть скрытым механизмом сетевой трансляции адресов за одним или несколькими адресами провайдера. Корпоративную сеть, основанную на компьютерных технологиях, называют *Интранетом*.

Конечно, вычислительные сети имеют (а может быть, создают!) и свои проблемы. Эти проблемы в основном связаны с организацией эффективного взаимодействия отдельных частей распределенной системы.

Во-первых, это сложности, связанные с программным обеспечением – операционными системами и приложениями. Программирование для распределенных систем принципиально отличается от программирования для централизованных систем. Так, сетевая

операционная система, выполняя в общем случае все функции по управлению локальными ресурсами компьютера, сверх того решает многочисленные задачи по предоставлению сетевых служб. Разработка сетевых приложений осложняется из-за необходимости организовать совместную работу их частей, выполняющихся на разных машинах.

Во-вторых, много проблем связано с транспортировкой сообщений по каналам связи между компьютерами. Основные задачи здесь – обеспечение *надежности* (чтобы передаваемые данные не терялись и не искажались) и производительности (чтобы обмен данными происходил с приемлемыми задержками). В структуре общих затрат на вычислительную сеть расходы на решение транспортных вопросов составляют существенную часть, в то время как в централизованных системах эти проблемы полностью отсутствуют.

В-третьих, это вопросы, которые связаны с обеспечением *безопасности*.

Глобализация и виртуализация привели к появлению угроз и вызывов, которые принято связывать с *киберпространством* и *кибербезопасностью*. Об этой проблеме можно получить достаточно полную информацию из книги известного специалиста по компьютерной безопасности Кевина Митника «Искусство быть невидимым».

На рис. 1.5 показана взаимосвязь понятий из анализируемой предметной области.

Рис. 1.5. Взаимосвязь понятий из области кибербезопасности

Выводы

1. Компьютерная сеть – это совокупность компьютеров, соединенных линиями связи. Линии связи образованы кабелями (или построены на основе беспроводных технологий), сетевыми адаптерами и другими коммуникационными устройствами. Все сетевое оборудование работает под управлением системного и прикладного ПО.

2. Основная задача сети – обеспечить пользователям потенциальную возможность совместного использования ресурсов всех компьютеров.

3. Вычислительная сеть – это одна из разновидностей распределенных систем, достоинством которых является возможность распараллеливания вычислений, за счет чего может быть достигнуто повышение производительности и отказоустойчивости системы.

4. Компьютерные сети и сетевые технологии стали определяющим фактором и важнейшим инструментом инноваций практически во всех областях человеческой деятельности.

5. Важнейший этап в развитии сетей – появление стандартных сетевых технологий типа Ethernet, позволяющих быстро и эффективно объединять компьютеры различных типов.

6. Отмечают 5 ключевых технологий, преобразующих сети:

– *программно-определяемые* (или *программно-реконфигурируемые*) *сети* (Software-Defined Networking, SDN);

– *облачные сервисы* (Cloud Computing Services, CCS);

– *интернет вещей* (Internet of Things, IoT);

– *виртуализация сетевых функций* (Network Functions Virtualization, NFV);

– *качество взаимодействия (восприятия)* (Quality of Experience, QoE).

7. Использование сетей дает следующие возможности:

– разделение дорогостоящих ресурсов;

– совершенствование коммуникаций;

– улучшение доступа к информации;

– быстрое и качественное принятие решений;

– свобода в территориальном размещении компьютеров.

8. К важнейшим техническим характеристикам сетей и сетевого оборудования относятся производительность, защищенность от несанкционированного доступа или безопасность и надежность.

9. Безопасность любого ресурса информационной системы складывается из обеспечения трех его характеристик: конфиденциальности, целостности, доступности.

Контрольные вопросы

1. Дайте определение сети.
2. Чем отличается коммуникационная сеть от информационной?
3. Как разделяются сети по территориальному признаку?
4. Что такое информационная система?
5. Что такое каналы связи?
6. Дайте определение физического канала связи.
7. Дайте определение логического канала связи.
8. Как называется совокупность правил обмена информацией между двумя или несколькими устройствами?
9. Что такое метод доступа?
10. Что такое совокупность правил, устанавливающих процедуры и формат обмена информацией?
11. Чем отличается рабочая станция в сети от обычного персонального компьютера?
12. Как называется описание физических соединений в сети?
13. Что такое архитектура сети?
14. Как называется способ определения, какая из рабочих станций будет следующей использовать канал связи?
15. Перечислите преимущества использования сетей.
16. Чем отличается одноранговая архитектура от клиент-серверной архитектуры?
17. Каковы преимущества крупномасштабной сети с выделенным сервером?
18. Какие сервисы предоставляет клиент-серверная архитектура?
19. Как называются рабочие станции, использующие ресурсы сервера?
20. Что такое сервер?
21. Опишите особенности современных сетевых технологий.
22. Дайте определение основных понятий, относящихся к надежности и безопасности компьютерных сетей.
23. Как следует понимать термины «информационная избыточность», «временна́я избыточность», «структурная избыточность»?

2

ОСНОВНЫЕ ПРИНЦИПЫ ПОСТРОЕНИЯ СЕТЕЙ

2.1. Архитектура сетей

Архитектура сети определяет основные элементы сети, характеризует ее общую логическую организацию, техническое и программное обеспечение, описывает методы кодирования.

Архитектура также определяет принципы функционирования и интерфейс пользователя.

В данном пособии будут рассмотрены три вида архитектур:

– архитектура *терминал – главный компьютер*;

– *одноранговая* архитектура;

– архитектура *клиент-сервер*.

2.1.1. Архитектура «терминал – главный компьютер»

Архитектура «терминал – главный компьютер» (Terminal – Host Computer Architecture) – это концепция информационной сети, в которой вся обработка данных осуществляется одним или группой главных компьютеров (рис. 2.1).

Рассматриваемая архитектура предполагает два типа оборудования:

– главный компьютер, где осуществляется управление сетью, хранение и обработка данных;

– терминалы, предназначенные для передачи главному компьютеру команд на организацию сеансов и выполнение заданий, ввода данных и получения результатов.

Главный компьютер взаимодействует с терминалами через *мультиплексоры передачи данных* (МПД), как представлено на рис. 2.1. Классический пример архитектуры сети с главными компьютерами – системная сетевая архитектура (System Network Architecture, SNA).

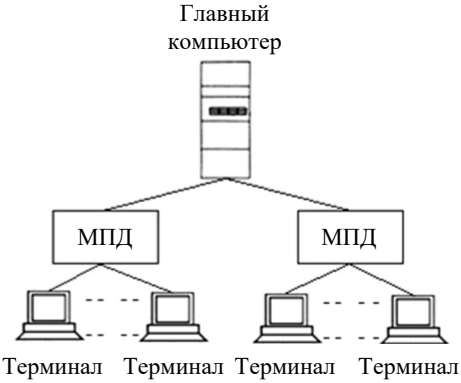


Рис. 2.1. Архитектура «терминал – главный компьютер»

По сути, данная архитектура предлагает максимально закрытую концепцию информационной сети, что и находит применение, например, в банковских системах.

2.1.2. Одноранговая архитектура

Одноранговая архитектура (Peer-to-Peer Architecture) – это концепция информационной сети, в которой ее ресурсы рассредоточены по всем взаимодействующим между собой системам (рис. 2.2). Данная архитектура характеризуется тем, что в ней все системы равноправны.

К одноранговым сетям относятся малые сети, где любая рабочая станция может выполнять одновременно функции файлового сервера и рабочей станции. В *одноранговых ЛВС* дисковое пространство и файлы на любом компьютере могут быть общими.

Чтобы ресурс стал общим, его необходимо отдать в общее пользование, используя службы удаленного доступа сетевых одноранговых операционных систем.

В зависимости от того, как будет установлена защита данных, другие пользователи смогут пользоваться файлами сразу же после

их создания. Одноранговые ЛВС достаточно хороши только для небольших рабочих групп, где не требуется централизованное управление.

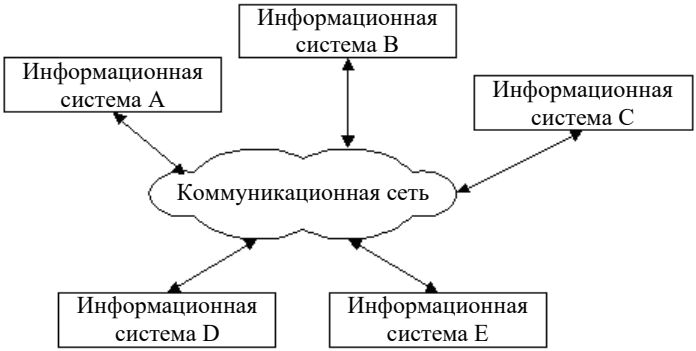


Рис. 2.2. Одноранговая архитектура

Одноранговые ЛВС являются наиболее легким и дешевым типом сетей для установки. Они требуют на компьютере, кроме сетевой карты и сетевого носителя, наличие пользовательской операционной системы. При соединении компьютеров пользователи могут предоставлять ресурсы и информацию в совместное пользование.

Одноранговые сети имеют следующие особенности:

– они легки в установке и настройке;
– отдельные ПК не зависят от выделенного сервера;
– пользователи в состоянии контролировать свои ресурсы;
– малая стоимость и легкая эксплуатация;
– минимум оборудования и программного обеспечения;
– нет необходимости в администраторе;
– хорошо подходят для сетей с небольшим числом пользователей ввиду неприменения средств централизованного управления.

Проблемой одноранговой архитектуры является ситуация, когда компьютеры отключаются от сети. При этом из сети исчезают виды сервиса, которые они предоставляли. Сетевую безопасность одновременно можно применить только к одному ресурсу, и пользователь должен помнить столько паролей, сколько сетевых ресурсов. При получении доступа к разделяемому ресурсу ощущается падение производительности компьютера. Существенным недостатком одноранговых сетей является отсутствие централизованного администрирования.

Использование одноранговой архитектуры не исключает применения в той же сети архитектуры «терминал – главный компьютер» или архитектуры «клиент-сервер».

2.1.3. Архитектура «клиент-сервер»

Архитектура «клиент-сервер» (Client-Server Architecture) – это концепция информационной сети, в которой основная часть ее ресурсов сосредоточена в серверах, обслуживающих своих клиентов (рис. 2.3). Рассматриваемая архитектура определяет два типа компонентов: серверы и клиенты.

Сервер – это объект, предоставляющий сервис другим объектам сети по их запросам.

Сервис – это процесс обслуживания клиентов.

Сервер работает по заданиям клиентов и управляет выполнением их заданий. После выполнения каждого задания сервер отправляет полученные результаты клиенту, пославшему это задание.

Сервисная функция в архитектуре «клиент-сервер» описывается комплексом прикладных программ, в соответствии с которым выполняются разнообразные прикладные процессы.

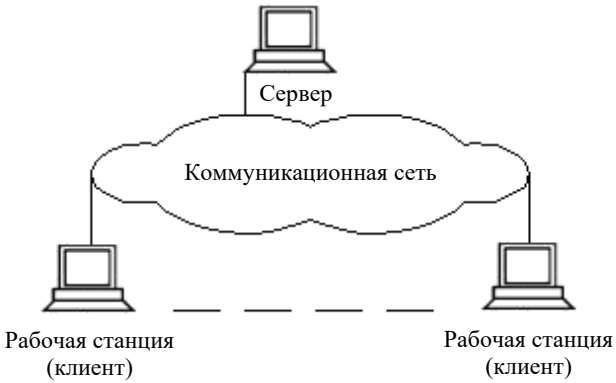


Рис. 2.3. Архитектура «клиент-сервер»

Процесс, который вызывает сервисную функцию с помощью определенных операций, называется **клиентом**. Им может быть программа или пользователь.

На рис. 2.4 приведен перечень сервисов в архитектуре «клиент-сервер».

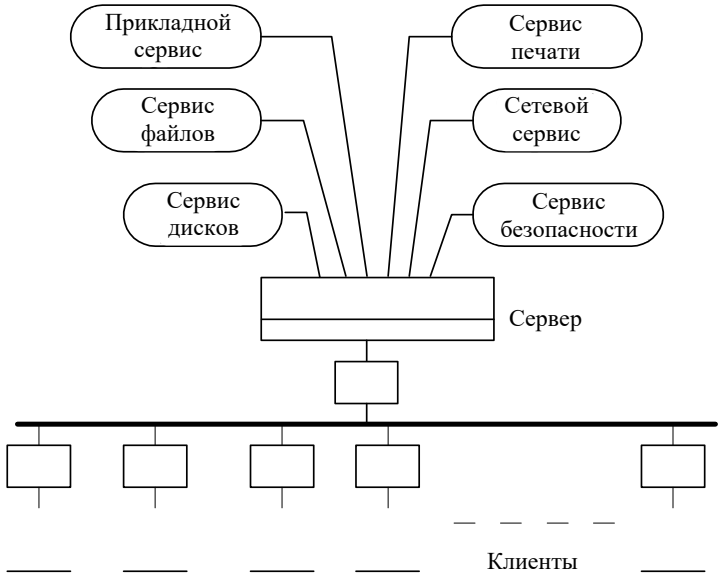


Рис. 2.4. Модель «клиент-сервер»

Клиенты – это рабочие станции, которые используют ресурсы сервера и предоставляют удобные интерфейсы пользователя.

Интерфейсы пользователя – это процедуры взаимодействия пользователя с системой или сетью.

Клиент является инициатором и использует электронную почту или другие сервисы сервера. В этом процессе клиент запрашивает вид обслуживания, устанавливает сеанс, получает нужные ему результаты и сообщает об окончании работы.

! В **сетях с выделенным файловым сервером** на выделенном автономном персональном компьютере (ПК) устанавливается серверная сетевая операционная система. Этот ПК становится сервером. Программное обеспечение (ПО), установленное на рабочей станции, позволяет ей обмениваться данными с сервером.

Наиболее распространенные сетевые операционные системы:
– NetWare фирмы Novel;
– Windows фирмы Microsoft;
– UNIX фирмы AT&T;
– Linux.

Помимо сетевой операционной системы необходимы сетевые прикладные программы, реализующие преимущества, предоставляемые сетью.

Сети на базе серверов имеют лучшие характеристики и повышенную надежность. Сервер владеет главными ресурсами сети, к которым обращаются остальные рабочие станции.

В современной клиент-серверной архитектуре выделяется четыре группы объектов: клиенты, серверы, данные и сетевые службы. Клиенты располагаются в системах на рабочих местах пользователей. Данные в основном хранятся на серверах. Сетевые службы являются совместно используемыми серверами и данными. Кроме того, службы управляют процедурами обработки данных.

Преимущества сети на основе клиент-серверной архитектуры:
– позволяют организовывать сети с большим количеством рабочих станций;
– обеспечивают централизованное управление учетными записями пользователей, безопасностью и доступом, что упрощает сетевое администрирование;
– предоставляют эффективный доступ к сетевым ресурсам;
– пользователю нужен один пароль для входа в сеть и получения доступа ко всем ресурсам, на которые распространяются его права.

Наряду с преимуществами сети на основе клиент-серверной архитектуры имеют и ряд недостатков:
– неисправность сервера может сделать сеть неработоспособной либо как минимум приведет к потере сетевых ресурсов;
– требуют квалифицированного персонала для администрирования;
– имеют более высокую стоимость сетей и сетевого оборудования.

2.1.4. Выбор архитектуры сети

Выбор архитектуры сети зависит от назначения сети, количества рабочих станций и от выполняемых на ней действий.

Следует выбирать одноранговую сеть, если:

– имеют место небольшие финансовые возможности;

– нет необходимости в специализированном сервере, таком как сервер БД, факс-сервер или какой-либо другой;

– нет возможности или необходимости в централизованном администрировании (последнее чаще всего применимо для малого количества пользователей сети).

Следует выбирать клиент-серверную сеть, если:

– количество пользователей достаточно большое, что, как правило, влечет за собой необходимость централизованного управления сетью;

– требуется повышенный уровень надежности и безопасности, управление ресурсами или резервное копирование;

– необходим специализированный сервер;

– требуется разделять ресурсы на уровне пользователей.

Выводы

1. Существует три основные архитектуры сети: терминал – главный компьютер, одноранговая и клиент-серверная.

2. В настоящее время наибольшее распространение получили одноранговая и клиент-серверная архитектуры.

3. Под одноранговой архитектурой подразумевается концепция информационной сети, в которой ее ресурсы рассредоточены по всем взаимодействующим между собой системам. Данная архитектура характеризуется тем, что в ней все системы равноправны.

4. Под клиент-серверной архитектурой подразумевается концепция информационной сети, в которой основная часть ее ресурсов сосредоточена в серверах, обслуживающих своих клиентов. В клиент-серверной архитектуре выделяется четыре группы объектов: клиенты, серверы, данные и сетевые службы. Клиенты располагаются в системах на рабочих местах пользователей. Данные в основном хранятся в серверах. Сетевые службы являются совместно используемыми серверами и данными. Кроме того, службы управляют процедурами обработки данных.

5. Выбор архитектуры сети осуществляется в зависимости от назначения сети, количества узлов и выполняемых на ней действий.

2.2. Топологии компьютерной сети

2.2.1. Виды топологий

Понятие **топологии** широко используется при создании сетей. Одним из подходов в классификации топологий ЛВС является выделение двух основных классов: широковещательные и последовательные топологии.

В **широковещательных топологиях** ПК передает сигналы, которые могут быть восприняты остальными ПК. К таким топологиям относятся общая шина, дерево, звезда и др.

В **последовательных топологиях** информация передается только одному ПК. Примерами таких топологий являются: произвольная (произвольное соединение ПК), кольцо, цепочка.

При выборе топологии преследуются три основные цели:

– обеспечение *альтернативной маршрутизации* и максимальной надежности передачи данных;

– выбор *оптимального маршрута* передачи блоков данных;

– предоставление приемлемого *времени ответа* и нужной *пропускной способности*.

При выборе конкретного типа сети важно учитывать ее топологию. Основными сетевыми топологиями являются: шинная (линейная) топология, звездообразная, кольцевая и древовидная.

Например, в конфигурации сети ARCNet используется одновременно и линейная, и звездообразная топология. Сети Token Ring физически выглядят как звезда, но логически их пакеты передаются по кольцу. Передача данных в сети Ethernet происходит по линейной шине, так что все станции видят сигнал одновременно.

Также возможны комбинации нескольких различных сетевых топологий.

Существуют пять основных топологий компьютерных сетей:
– общая шина (Bus);
– кольцо (Ring);
– звезда (Star);
– древовидная (Tree);
– ячеистая (Mesh).

2.2.2. Топология «общая шина»

Общая шина – это тип сетевой топологии, в которой рабочие станции расположены вдоль одного участка кабеля, называемого *сегментом* (рис. 2.5).

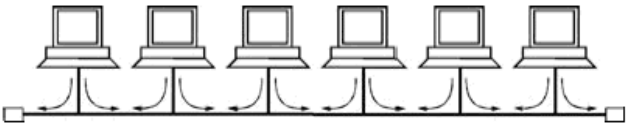


Рис. 2.5. Топология «общая шина»

Топология «общая шина» предполагает использование одного кабеля, к которому подключаются все компьютеры сети. При этом кабель используется всеми станциями по очереди. Для уменьшения зашумленности среды отраженными сигналами, мешающими передаче данных, используют так называемые терминаторы – специальные резисторы на концах кабеля, предотвращающие появление «отраженной волны».

Все сообщения, посылаемые отдельными компьютерами, принимаются и прослушиваются всеми остальными компьютерами, подключенными к сети. Рабочая станция отбирает адресованные ей сообщения, пользуясь *адресной* информацией. Надежность здесь выше, так как выход из строя отдельных компьютеров не нарушит работоспособность сети в целом. Поиск неисправности в сети затруднен. Кроме того, так как используется только один кабель, в случае обрыва нарушается работа всей сети. Шинная топология – это наиболее простая топология сети.

Примерами использования топологии «общая шина» являются сети 10Base-5 (соединение ПК толстым коаксиальным кабелем) и 10Base-2 (соединение ПК тонким коаксиальным кабелем).

2.2.3. Кольцевая топология

Кольцо – это топология ЛВС, в которой каждая рабочая станция соединена с двумя другими рабочими станциями, образуя кольцо (рис. 2.6). Данные передаются от одной рабочей станции к другой в одном направлении (по кольцу).

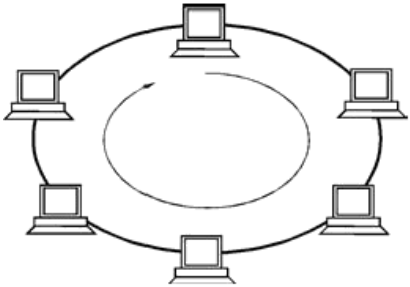


Рис. 2.6. Топология «кольцо»

Каждая рабочая станция выполняет роль ***повторителя***, ретранслируя сообщения к следующей рабочей станции, т. е. данные передаются от одного компьютера к другому по очереди. Если компьютер получает данные, предназначенные для другого компьютера, он передает их дальше по кольцу, в ином случае они дальше не передаются.

Основная проблема при кольцевой топологии заключается в том, что каждая рабочая станция должна активно участвовать в пересылке информации, и в случае выхода из строя хотя бы одной из них, вся сеть парализуется. Подключение новой рабочей станции требует краткосрочного выключения сети, так как во время установки кольцо должно быть разомкнуто. Топология «кольцо» имеет хорошо предсказуемое время отклика, определяемое числом рабочих станций.

Чистая кольцевая топология используется редко. Вместо этого кольцевая топология играет транспортную роль в схеме метода доступа. Кольцо описывает логический маршрут, а пакет передается от одной станции к другой, совершая в итоге полный круг.

В сетях ***Token Ring*** кабельная ветвь из *центрального концентратора* называется MAU (Multiple Access Unit – устройство с множественным доступом). MAU имеет внутреннее кольцо, соединяющее все подключенные к нему станции, и используется как альтернативный путь, когда оборван или отсоединен кабель одной рабочей станции. Когда кабель рабочей станции подсоединен к MAU, он просто образует расширение кольца: сигналы поступают к рабочей станции, а затем возвращаются обратно во внутреннее кольцо.

Также стоит упомянуть в качестве отдельной топологию «цепочка», представляющую «разомкнутое кольцо» (рис. 2.7).

Рис. 2.7. Топология «цепочка»

В данной топологии сохраняются все особенности и правила топологии «кольцо».

2.2.4. Топология «звезда»

Звезда – это топология ЛВС (рис. 2.8), в которой все рабочие станции присоединены к центральному узлу (например, к концентратору), который устанавливает, поддерживает и разрывает связи между рабочими станциями.

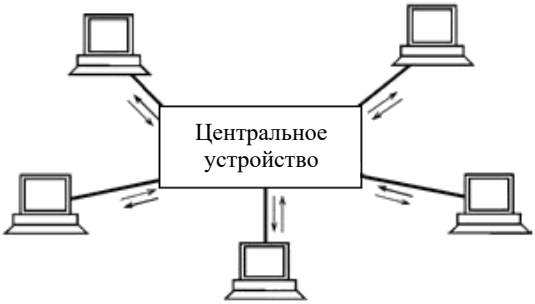


Рис. 2.8. Топология «звезда»

Преимуществом такой топологии является возможность простого исключения неисправного узла. Однако если неисправен центральный узел, вся сеть выходит из строя. В этом случае каждый компьютер через специальный сетевой адаптер подключается отдельным кабелем к объединяющему устройству.

При необходимости можно объединять вместе несколько сетей с топологией «звезда», при этом получаются разветвленные конфигурации сети. В каждой точке ветвления необходимо использовать специальные соединители (распределители, повторители или устройства доступа).

В звездообразных топологиях наибольшее применение нашел кабель **витая пара** (*Twisted Pair*, в соответствии со стандартами передачи 10BASE-T, 100BASE-T и т. д.). *Центром звезды* обычно является *Hub* (**хаб**, **концентратор**) либо *Switch* (**коммутатор**).

Если в качестве центрального устройства в топологии «звезда» выступает коммутатор или компьютер, то ее называют активной звездой, если же используется устройство типа концентратор, то пассивной звездой.

Звездообразная топология обеспечивает защиту от разрыва кабеля. Если кабель рабочей станции будет поврежден, это не приведет к выходу из строя всего сегмента сети. Она позволяет также легко диагностировать проблемы подключения, так как каждая рабочая станция имеет свой собственный кабельный сегмент, подключенный к концентратору. Для диагностики достаточно найти разрыв кабеля, который ведет к неработающей станции. Остальная часть сети продолжает нормально работать.

Однако звездообразная топология имеет и недостатки. Во-первых, она требует для организации сети большого количества кабеля. Во-вторых, концентраторы довольно дороги. В-третьих, кабельные концентраторы при большом количестве кабеля трудно обслуживать. Но в большинстве случаев в такой топологии используется недорогой кабель «витая пара». В некоторых случаях можно даже использовать существующие телефонные кабели. Кроме того, для диагностики и тестирования выгодно собирать все кабельные концы в одном месте.

Еще одним недостатком можно назвать ограниченность расширения сети на основе топологии «звезда», так как и концентраторы, и коммутаторы в определенной степени ограничены по числу портов. Возможно последовательное соединение нескольких «звезд» через центральные устройства, но при большом числе соединенных «звезд» могут возникнуть сложности с передачей между удаленными узлами.

2.2.5. Другие типы топологии

Кроме трех рассмотренных базовых топологий нередко применяется также сетевая топология **дерево** (Tree), которую можно рассматривать как комбинацию нескольких «звезд». Причем как и в случае «звезды», «дерево» может быть активным или истинным (рис. 2.9).

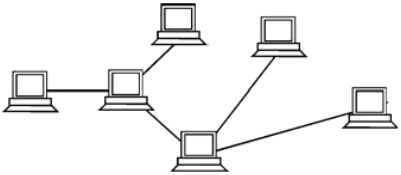


Рис. 2.9. Топология «активное дерево»

Также «дерево» может быть *пассивным* (рис. 2.10).

При *активном дереве* в центрах объединения нескольких линий связи находятся центральные компьютеры, а при *пассивном* – концентраторы (хабы).

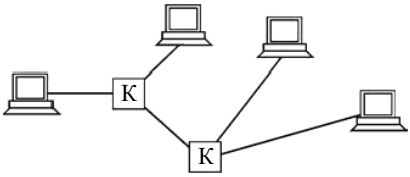


Рис. 2.10. Топология «пассивное дерево» (К – концентратор)

Однако в отличие от варианта сетевой топологии с последовательным объединением нескольких «звезд», все древовидные топологии предполагают иерархическую структуру со строго выделенной вершиной «дерева». При этом надо отметить, что использовать древовидную топологию, как активную, так и пассивную, целесообразно при большом числе узлов (при малом числе узлов эффективна будет и топология «звезда»). Так, например, активное и пассивное дерево в таком случае может выглядеть, как показано на рис. 2.11 и 2.12 соответственно.

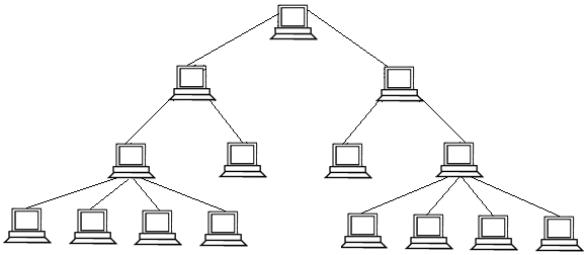


Рис. 2.11. Топология «активное дерево» с большим числом узлов

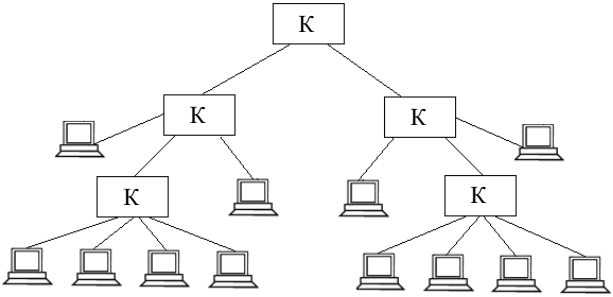


Рис. 2.12. Топология «пассивное дерево»
с большим числом узлов

Также отметим, что на практике концентратор не может эффективно исполнять роль вершины дерева в пассивном варианте топологии, целесообразно использовать коммутатор.

Сетевая топология **Fat Tree** (**утолщенное дерево**), изобретенная Ч. Лейзорсоном, является дешевой и эффективной для суперкомпьютеров (рис. 2.13).

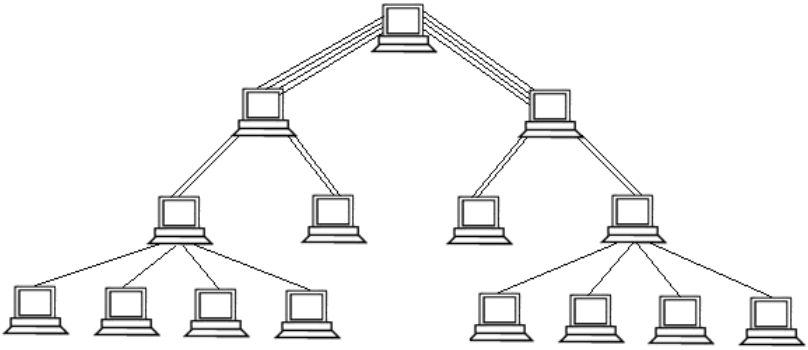
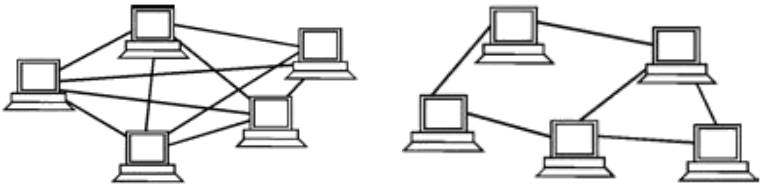


Рис. 2.13. Топология Fat Tree

В отличие от классической топологии «дерево», в которой все связи между узлами одинаковы, связи в «утолщенном дереве» становятся более широкими (производительными по пропускной способности) с каждым уровнем по мере приближения к корню дерева. Часто используют удвоение пропускной способности на каждом уровне. Сети с топологией Fat Tree являются предпочтительными для построения кластерных межсоединений.

В **сеточной** (**ячеистой**) (Mesh) топологии компьютеры связываются между собой не одной, а многими линиями связи, образующими сетку (рис. 2.14).



а *б*

Рис. 2.14. Сеточная топология: полная (*а*) и частичная (*б*)

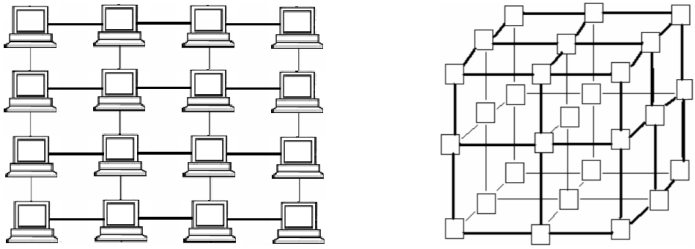
В ***полной сеточной топологии*** каждый компьютер напрямую связан со всеми остальными компьютерами. В этом случае при увеличении числа компьютеров резко возрастает количество линий связи. Кроме того, любое изменение в конфигурации сети требует внесения изменений в сетевую аппаратуру всех компьютеров, поэтому полная сеточная топология не получила широкого распространения.

Частичная сеточная топология предполагает прямые связи только для самых активных компьютеров, передающих максимальные объемы информации. Остальные компьютеры соединяются через промежуточные узлы.

Сеточная топология позволяет выбирать маршрут для доставки информации от абонента к абоненту, обходя неисправные участки. С одной стороны, это увеличивает надежность сети, с другой же – требует существенного усложнения сетевой аппаратуры, которая должна выбирать маршрут.

В заключение несколько слов о решетчатой топологии, в которой узлы образуют регулярную многомерную решетку. При этом каждое ребро решетки параллельно ее оси и соединяет два смежных узла вдоль этой оси.

Одномерная решетка – это цепь, соединяющая два внешних узла (имеющие лишь одного соседа) через некоторое количество внутренних (у которых по два соседа – слева и справа). При соединении обоих внешних узлов получается топология *кольцо*. Двух- и трехмерные решетки используются в архитектуре суперкомпьютеров (рис. 2.15).



а *б*

Рис. 2.15. Пример сетевых топологий:
а – двухмерная решетка; *б* – трехмерная решетка

Многомерная решетка, соединенная циклически в более чем одном измерении, называется **тор**.

! Основным достоинством топологии «решетка» является высокая надежность, а недостатком – сложность реализации.

Довольно часто применяются комбинированные топологии, среди которых наиболее распространены **звездно-шинная** (Star-Bus) (рис. 2.16) и **звездно-кольцевая** (Star-Ring) (рис. 2.17).

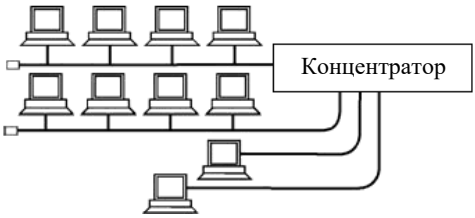


Рис. 2.16. Пример звездно-шинной топологии

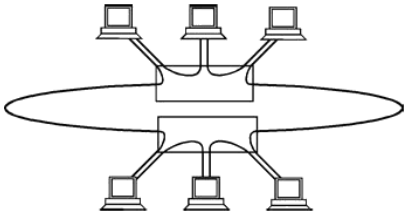


Рис. 2.17. Пример звездно-кольцевой топологии

В звездно-шинной топологии используется комбинация шины и пассивной звезды. К концентратору подключаются как отдельные компьютеры, так и целые шинные сегменты. На самом деле реализуется физическая топология «шина», включающая все компьютеры сети. В данной топологии может использоваться и несколько концентраторов, соединенных между собой и образующих так называемую магистральную, опорную шину. К каждому из концентраторов при этом подключаются отдельные компьютеры или шинные сегменты. В результате получается «звездно-шинное дерево». Таким образом, пользователь может гибко комбинировать преимущества шинной и звездной топологий, а также легко изменять количество компьютеров, подключенных к сети. С точки зрения распространения информации данная топология равноценна классической шине.

В случае звездно-кольцевой топологии в кольцо объединяются не сами компьютеры, а специальные концентраторы (прямоугольник на рис. 2.17), к которым в свою очередь подключаются компьютеры с помощью звездообразных двойных линий связи. В действительности все компьютеры сети включаются в замкнутое кольцо, так как внутри концентраторов линии связи образуют замкнутый контур. Данная топология дает возможность комбинировать преимущества звездной и кольцевой топологий. Например, концентраторы позволяют собрать в одно место все точки подключения кабелей сети. Если говорить о распространении информации, данная топология равноценна классическому кольцу.

2.2.6. Многозначность понятия топологии

Топология сети указывает не только на физическое расположение компьютеров, как часто считают, но, что гораздо важнее, на характер связей между ними, особенности распространения информации, сигналов по сети.

Именно характер связей определяет степень отказоустойчивости сети, требуемую сложность сетевой аппаратуры, наиболее подходящий метод управления обменом, возможные типы сред передачи (каналов связи), допустимый размер сети (длина линий связи и количество абонентов) необходимость электрического согласования и многое другое.

Более того, физическое расположение компьютеров, соединяемых сетью, почти не влияет на выбор топологии. Как бы ни были расположены компьютеры, их можно соединить с помощью любой заранее выбранной топологии (рис. 2.18).

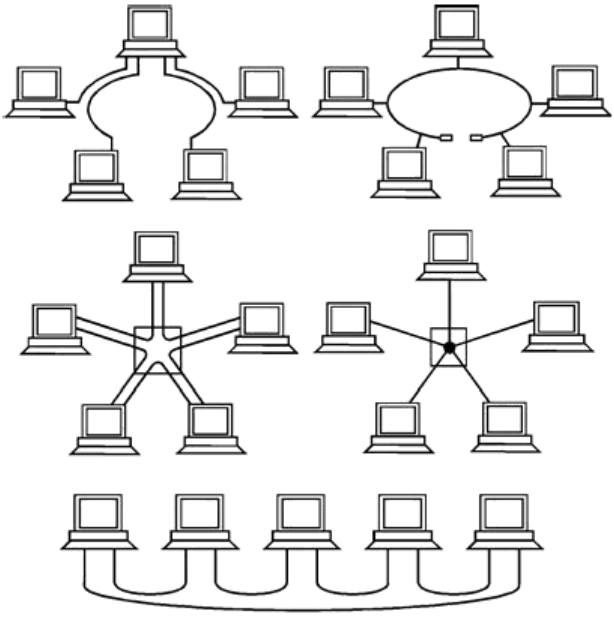


Рис. 2.18. Примеры использования разных топологий для соединения компьютеров

В том случае если соединяемые компьютеры расположены по контуру круга, они могут соединяться, как звезда или шина. Когда компьютеры расположены вокруг некоего центра, их допустимо соединить с помощью топологий «шина» или «кольцо».

Наконец, когда компьютеры расположены в одну линию, они могут соединяться звездой или кольцом. Другое дело, какова будет требуемая длина кабеля.

Строго говоря, при упоминании о топологии сети могут подразумеваться четыре совершенно разных понятия, относящихся к различным уровням сетевой архитектуры.

Физическая топология – географическая схема расположения компьютеров и прокладки кабелей. При этом, например, пассивная

звезда ничем не отличается от активной, поэтому ее нередко называют просто звездой.

Логическая топология – структура связей, характер распространения сигналов по сети. Это наиболее правильное определение топологии.

Топология управления обменом, или **метод доступа**, – это принцип и последовательность передачи права на использование сети для передачи данных между отдельными компьютерами.

Информационная топология – направление потоков информации, передаваемой по сети.

Например, сеть с физической и логической топологией «шина» может в качестве метода управления использовать эстафетную передачу права захвата сети (быть в этом смысле кольцом) и одновременно передавать всю информацию через выделенный компьютер (т. е. быть звездой). Или сеть с логической топологией «шина» может иметь физическую топологию «звезда» (пассивная) или «дерево» (пассивное).

Сеть с любой физической топологией, логической топологией, топологией управления обменом может считаться звездой в смысле информационной топологии, если она построена на основе одного сервера и нескольких клиентов, общающихся только с этим сервером.

В данном случае справедливы все рассуждения о низкой отказоустойчивости сети к неполадкам центра (сервера). Точно так же любая сеть может быть названа шиной в информационном смысле, если она построена из компьютеров, являющихся одновременно и серверами, и клиентами. Такая сеть будет малочувствительна к отказам отдельных компьютеров.

Выводы

1. Топология сети определяет как физическое расположение компьютеров, так и характер связей между ними, особенности распространения информации, сигналов по сети. Именно характер связей определяет степень отказоустойчивости сети, требуемую сложность

сетевой аппаратуры, наиболее подходящий метод управления обменом, возможные типы сред передачи (каналов связи), допустимый размер сети (длина линий связи и количество абонентов) необходимость электрического согласования и многое другое.

2. Выделяют два основных класса топологий: широковещательные и последовательные.

3. В широковещательных топологиях передаемые сигналы могут быть восприняты всеми узлами сети. В последовательных топологиях информация передается только одному сетевому узлу.

4. Выделяются пять базовых топологий: общая шина, дерево, звезда, сеточная (ячеистая), кольцо.

5. Довольно часто на практике применяются комбинированные топологии, среди которых наибольшее распространение получили звездно-шинная и звездно-кольцевая.

2.3. Основные параметры и характеристики сетей

При организации и эксплуатации сети учитываются важные характеристики и параметры (некоторые были описаны ранее):

– производительность;
– прозрачность;
– поддержка разных видов трафика;
– управляемость;
– совместимость;
– надежность и безопасность;
– расширяемость и масштабируемость.

Наиболее важными являются производительность, надежность и безопасность. Далее проанализируем некоторые из перечисленных характеристик.

2.3.1. Производительность сети

Производительность сети (Network Performance) – это характеристика сети, позволяющая оценить, насколько быстро информация передающей рабочей станции достигнет до приемной рабочей станции.

На производительность влияют следующие характеристики сети:
– конфигурация;
– скорость передачи данных (пропускная способность);
– метод доступа к каналу;
– топология сети;
– технология.

Если производительность сети перестает отвечать предъявляемым к ней требованиям, то администратор сети может прибегнуть к различным приемам:
– изменить конфигурацию сети таким образом, чтобы ее структура более соответствовала структуре информационных потоков;
– перейти к другой модели построения распределенных приложений, которая позволила бы уменьшить сетевой трафик;
– заменить коммутирующие устройства на более скоростные.

Но самым радикальным решением в такой ситуации является переход на более скоростную технологию, например с Fast Ethernet на Gigabit Ethernet или даже на 10-Gigabit Ethernet, что позволит увеличить *пропускную способность* каналов передачи данных.

Пропускная способность характеризует объем данных, переданных сетью в единицу времени. Измеряется либо в битах в секунду (бит/с), либо в пакетах в секунду (пакет/с). Пропускная способность может быть *мгновенной*, *максимальной* и *средней*.

Средняя пропускная способность – результат деления общего объема переданных данных на время их передачи (промежуток времени – часы или дни, недели и т. д.).

Мгновенная пропускная способность отличается от средней тем, что промежуток времени соответствует 1 с или ее части.

Максимальная пропускная способность – это наибольшая мгновенная пропускная способность, зафиксированная в течение периода наблюдения.

Иногда полезно оперировать ***общей пропускной способностью*** сети, которая определяется как среднее количество информации, переданной между всеми узлами сети в единицу времени. Этот показатель характеризует *качество сети* в целом, не дифференцируя его по отдельным сегментам или устройствам.

С ростом масштаба сетей возникла необходимость в повышении их производительности. Одним из способов достижения этого стала ***микросегментация***. Она позволяет уменьшить число пользователей на один сегмент, снизить объем широковещательного трафика, а значит, повысить производительность сети.

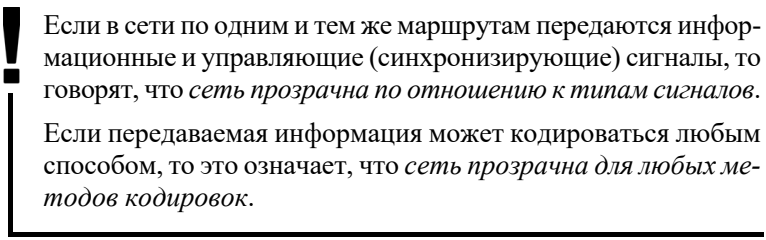
Первоначально для микросегментации использовались маршрутизаторы, не очень приспособленные для этой цели. Решения на их основе были достаточно дорогостоящими и отличались большой временной задержкой и невысокой пропускной способностью. Более подходящими устройствами для микросегментации сетей стали коммутаторы. Благодаря относительно низкой стоимости, высокой производительности и простоте в использовании, они быстро завоевали популярность.

Таким образом, сети стали строить на базе коммутаторов и маршрутизаторов. Первые обеспечивали высокоскоростную пересылку трафика между сегментами, входящими в одну подсеть, а вторые передавали данные между подсетями, ограничивали распространение широковещательного трафика, решали задачи безопасности и т. д.

2.3.2. Прозрачность сети

Прозрачность сети (Network Transparency) – это такое состояние сети, при котором пользователь воспринимает сеть как отдельный (персональный) компьютер.

Коммуникационная сеть является прозрачной относительно проходящей сквозь нее информации, если выходной поток битов в точности повторяет входной поток. Но сеть может быть непрозрачной во времени, если из-за меняющихся размеров очередей блоков данных изменяется и время прохождения различных блоков через узлы коммутации. Прозрачность сети по скорости передачи данных указывает, что данные можно передавать с любой нужной скоростью.



Если в сети по одним и тем же маршрутам передаются информационные и управляющие (синхронизирующие) сигналы, то говорят, что *сеть прозрачна по отношению к типам сигналов*.

Если передаваемая информация может кодироваться любым способом, то это означает, что *сеть прозрачна для любых методов кодировок*.

Прозрачная сеть является простым решением, в котором для взаимодействия локальных сетей, расположенных на значительном расстоянии друг от друга, используется принцип ***Plug-and-Play*** (подключись и работай).

Прозрачное соединение. Служба прозрачных локальных сетей обеспечивает сквозное (End-to-End) соединение, связывающее между собой удаленные локальные сети. Привлекательность данного решения состоит в том, что эта служба объединяет удаленные друг от друга на значительное расстояние узлы как части локальной сети. Поэтому не нужно вкладывать средства в изучение новых технологий и создание территориально распределенных сетей (Wide-Area Network – WAN). Пользователям требуется только поддерживать локальное соединение, а провайдер службы прозрачных сетей обеспечит беспрепятственное взаимодействие узлов через сеть масштаба города (Metropolitan Area Network – MAN) или сеть WAN. У служб прозрачной локальной сети много преимуществ. Например, пользователь может быстро и безопасно передавать большие объемы данных на значительные расстояния, не обременяя себя сложностями, связанными с работой в сетях WAN.

2.3.3. Поддержка разных видов трафика

Трафик в сети складывается случайным образом, однако в нем отражены и некоторые закономерности. Как правило, некоторые пользователи, работающие над общей задачей (например, сотрудники одного отдела), чаще всего обращаются с запросами либо друг к другу, либо к общему серверу, и только иногда они испытывают необходимость доступа к ресурсам компьютеров другого отдела. Желательно, чтобы структура сети соответствовала структуре информационных потоков.

В зависимости от сетевого трафика компьютеры в сети могут быть разделены на группы (**сегменты сети**; Segment of the Network).

Компьютеры объединяются в группу, если большая часть порождаемых ими сообщений адресована компьютерам этой же группы.

Для разделения сети на сегменты используются ***мосты*** и ***коммутаторы***. Они экранируют локальный трафик внутри сегмента, не передавая за его пределы никаких кадров, кроме тех, которые адресованы компьютерам, находящимся в других сегментах.

Таким образом, сеть распадается на отдельные подсети. Это позволяет более рационально выбирать пропускную способность имеющихся линий связи, учитывая интенсивность трафика внутри каждой группы, а также активность обмена данными между группами.

Однако локализация трафика средствами мостов и коммутаторов имеет существенные ограничения. С другой стороны, использование механизма виртуальных сегментов, реализованного в коммутаторах локальных сетей, приводит к полной локализации трафика; такие сегменты полностью изолированы друг от друга, даже в отношении широковещательных кадров. Поэтому в сетях, построенных только на мостах и коммутаторах, компьютеры, принадлежащие разным виртуальным сегментам, не образуют единой сети.

Нетривиальным является совмещение в одной сети традиционного компьютерного и мультимедийного трафика. Передача исключительно мультимедийного трафика компьютерной сетью вызывает меньшие трудности. Наиболее близки к этой цели сети на основе технологии ATM.

Для того чтобы эффективно консолидировать различные виды трафика в сети ATM, требуется специальная предварительная подготовка (адаптация) данных, имеющих различный характер: кадры – для цифровых данных, сигналы импульсно-кодовой модуляции – для голоса, потоки битов – для видео. Эффективная консолидация трафика требует также учета и использования статистических вариаций интенсивности различных типов трафика.

2.3.4. Управляемость сети

Управляемость сети (Network Manageability) подразумевает возможность централизованно контролировать состояние основных элементов сети, выявлять и разрешать проблемы, возникающие при работе сети, выполнять анализ производительности и планировать выделение, внедрение, координацию и мониторинг ресурсов компьютерной сети.

Модель управления сетью является основным средством для понимания главных функций системы управления сетью.

Система управления элементами сети (Element Management System, EMS) – программное обеспечение, предназначенное для управления и контроля отдельного сетевого элемента группы однотипных элементов. Состоит из систем и приложений, которые связаны

с управлением на сетевом уровне *модели управления телекоммуникационной сетью*.

Эта модель состоит из пяти концептуальных компонент:

– управление эффективностью (Performance Management);

– управление конфигурацией (Configuration Management);

– управление учетом использования ресурсов (Accounting Management);

– управление неисправностями (Fault Management);

– управление защитой данных или безопасностью (Security Management).

Цель **управления эффективностью** – измерение, обеспечение и поддержание различных аспектов эффективности сети на приемлемом уровне.

Примерами переменных для определения эффективности являются *пропускная способность* сети, *время реакции* пользователей и *коэффициент использования линии*.

Управление эффективностью включает несколько этапов:

– сбор информации об эффективности по тем переменным, которые представляют интерес для *администраторов сети*;

– анализ информации для определения нормальных (базовая строка) уровней;

– определение соответствующих порогов эффективности для каждой важной переменной таким образом, что превышение этих порогов указывает на наличие проблемы в сети, достойной внимания.

Цель **управления конфигурацией** – контролирование информации о сетевой и системной конфигурации для отслеживания происходящего и управления воздействием на работу различных аппаратных и программных элементов.

Так как все аппаратные и программные элементы имеют эксплуатационные отклонения, погрешности (или то и другое вместе), которые могут влиять на работу сети, такая информация важна для поддержания гладкой работы сети.

Каждое устройство сети располагает разнообразной информацией о версиях, ассоциируемых с ним. Чтобы обеспечить легкий доступ,

подсистемы управления конфигурацией хранят эту информацию в базе данных. Когда возникает какая-нибудь проблема, в этой базе данных может быть проведен поиск ключей, которые могли бы помочь решить эту проблему.

Цель **управления учетом использования ресурсов** – измерение параметров использования сети, чтобы можно было соответствующим образом регулировать ее использование индивидуальными или групповыми пользователями.

Такое регулирование минимизирует число проблем в сети (так как ресурсы сети могут быть поделены исходя из возможностей источника) и максимизирует равнодоступность к сети для всех пользователей.

Цель **управления неисправностями** – выявить, зафиксировать, уведомить пользователей и (в пределах возможного) автоматически устранить проблемы в сети для того, чтобы эффективно поддерживать работу сети.

Так как неисправности могут привести к простоям или недопустимой деградации сети, управление неисправностями, по всей вероятности, является наиболее широко используемым элементом модели управления сети ISO.

Управление неисправностями включает в себя несколько шагов:

– определение симптомов проблемы;

– изолирование проблемы;

– устранение проблемы;

– проверка устранения неисправности на всех важных подсистемах;

– регистрация обнаружения проблемы и ее решения.

Цель **управления защитой данных** – контроль доступа к сетевым ресурсам в соответствии с установленными руководящими принципами, чтобы сделать невозможными *саботаж сети* и доступ к *конфиденциальной* информации лицам, не имеющим соответствующего разрешения, т. е. *несанкционированный доступ*.

Например, одна из подсистем управления защитой данных может контролировать регистрацию пользователей ресурса сети, отказывая в доступе тем, кто вводит коды доступа, не соответствующие установленным.

Подсистемы управления защитой данных работают путем разделения источников на *санкционированные* и *несанкционированные* области. Для некоторых пользователей доступ к любому источнику сети является несоответствующим.

Подсистемы управления защитой данных выполняют следующие функции:

– идентифицируют конфиденциальные ресурсы сети (включая системы, файлы и другие объекты);

– определяют отображения в виде карт между конфиденциальными источниками сети и набором пользователей;

– контролируют точки доступа к конфиденциальным ресурсам сети;

– регистрируют несанкционированный доступ к конфиденциальным ресурсам сети.

2.3.5. Совместимость сети

Совместимость, или ***интегрируемость***, означает, что сеть способна включать в себя самое разнообразное программное и аппаратное обеспечение, т. е. в ней могут использоваться различные операционные системы, поддерживающие разные стеки коммуникационных протоколов, и работать аппаратные средства и приложения от разных производителей.

Сеть, состоящая из разнотипных элементов, называется ***неоднородной*** или ***гетерогенной***, а если гетерогенная сеть работает без проблем, то она является ***интегрированной***.

Концепция совместимости сети (Network Compatibility) впервые в широких масштабах была применена разработчиками системы IBM/360. Основная задача при проектировании всего ряда моделей этой системы заключалась в создании такой архитектуры, которая была бы одинаковой с точки зрения пользователя для всех моделей системы независимо от цены и производительности каждой из них.

Огромные преимущества такого подхода, позволяющего сохранять существующий задел программного обеспечения при переходе

на новые (как правило, более производительные) модели, были быстро оценены как производителями компьютеров, так и пользователями. Начиная с этого времени практически все фирмы-поставщики компьютерного оборудования взяли на вооружение данные принципы, поставляя серии совместимых компьютеров. Следует заметить однако, что со временем даже самая передовая архитектура неизбежно устаревает и возникает потребность внесения радикальных изменений в архитектуру и способы организации вычислительных систем.

В настоящее время одним из наиболее важных факторов, определяющих тенденции в развитии информационных технологий, является ориентация компаний-поставщиков компьютерного оборудования на рынок прикладных программных средств.

Во-первых, вычислительная среда должна позволять гибко менять количество и состав аппаратных средств и программного обеспечения в соответствии с меняющимися требованиями решаемых задач. Во-вторых, она должна обеспечивать возможность запуска одних и тех же программных систем на различных аппаратных платформах, т. е. обеспечивать мобильность программного обеспечения. В-третьих, эта среда должна гарантировать возможность применения одних и тех же человеко-машинных интерфейсов на всех компьютерах, входящих в неоднородную сеть.

Основной путь построения интегрированных сетей – использование модулей, выполненных в соответствии с открытыми стандартами и спецификациями.

В условиях жесткой конкуренции производителей аппаратных платформ и программного обеспечения сформировалась концепция открытых систем (КОС).

Концепция открытых систем представляет собой совокупность стандартов на различные компоненты вычислительной среды, предназначенных для обеспечения мобильности программных средств в рамках неоднородной распределенной вычислительной системы.

Открытая система предназначена для взаимодействия с другими приложениями на локальной и удаленной платформах, а также для взаимодействия с пользователями в режиме, который облегчает переход от системы к системе.

Открытая спецификация, по определению POSIX (Portable Operating System Interface), – общедоступная спецификация, которая поддерживается открытым, гласным согласительным процессом, направленным на приспособление новой технологии к ее применению, и которая согласуется со стандартами.

2.3.6. Надежность и безопасность. Введение в проблематику

Чтобы систему можно было отнести к высоконадежным, она должна обеспечить сохранность данных и защиту их от искажений. Кроме того, должна поддерживаться согласованность (непротиворечивость) данных. Например, если для повышения надежности (Reliability) на нескольких файловых серверах хранится несколько копий данных, то нужно постоянно обеспечивать их идентичность.

Как уже отмечалось, цифровые трансформации привели к тому, что информация стала одним из самых дорогих продуктов в сфере межличностных отношений. При этом стоимость информации часто превосходит в сотни и тысячи раз стоимость компьютерной системы, в которой она обрабатывается. Это обстоятельство приводит к тому, что именно информация и информационные системы (в том числе компьютерные сети) все чаще становятся объектами *атак* (несанкционированного доступа).

В общем случае атака (или сетевая атака) – попытка несанкционированного преодоления защиты информационной системы или сети.

На рис. 2.19 схематически показаны наиболее уязвимые места локальной сети.

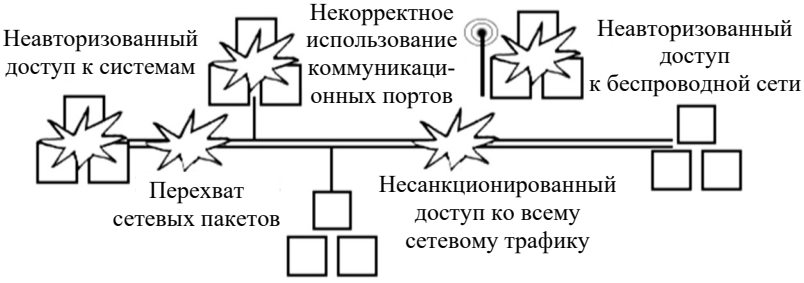


Рис. 2.19. Уязвимые места локальной сети

Все многообразие дестабилизирующих работу сети факторов можно разделить на два класса: внутренние и внешние.

Внутренние дестабилизирующие факторы влияют:

1) на программные средства (ПС):

– некорректный исходный алгоритм;

– неправильно запрограммированный исходный алгоритм (первичные ошибки);

2) на аппаратные средства (АС):

– системные ошибки при постановке задачи проектирования;

– отклонения от технологии изготовления комплектующих изделий и АС в целом;

– нарушение режима эксплуатации, вызванное внутренним состоянием АС.

Внешние дестабилизирующие факторы оказывают влияние:

1) на программные средства:

– неквалифицированные пользователи;

– несанкционированный доступ к ПС с целью модификации кода;

2) на аппаратные средства:

– внешние климатические условия;

– электромагнитные и ионизирующие помехи;

– перебои в электроснабжении;

– недостаточная квалификация обслуживающего персонала;

– несанкционированный (в том числе – удаленный) доступ с целью нарушения работоспособности АС – сетевая атака.

Некоторые из перечисленных выше дестабилизирующих факторов влияют на безопасность системы, а некоторые – на ее надежность. Понятно, что в силу достаточно тонкой грани между надежностью и безопасностью системы многие из факторов влияют на оба параметра.

В результате атак злоумышленники могут нарушать работу сети, изменять права аккаунта, получать персональные данные пользователей и реализовывать другие цели.

Виды сетевых атак и их последствия имеют значительные отличия друг от друга. Современная классификация угроз производится в основном по следующим параметрам:

– характер воздействия, оказываемого на сеть;

– цель оказываемого воздействия;

– наличие обратной связи с сетью, подвергнутой атаке;

– условие начала атаки;

– расположение субъекта по отношению к объекту атаки;

– уровень эталонной модели ISO.

Поэтому вполне естественно возникает необходимость в защите информации. Однако по сравнению с другими информационно-вычислительными системами проблема защиты информации в компьютерных сетях значительно усложняется и происходит это по ряду следующих причин:

– наличие большого числа пользователей в компьютерной сети и их переменный состав; защита на уровне имени и пароля пользователя недостаточна для предотвращения входа в сеть посторонних лиц;

– значительная протяженность сети и наличие многих потенциальных каналов проникновения в сеть;

– уже отмеченные недостатки в аппаратном и программном обеспечении, которые зачастую обнаруживаются не на предпродажном этапе, называемом бета-тестированием, а в процессе эксплуатации.

И еще один один важный момент.

В условиях все большего вовлечения смартфонов и планшетов в бизнес-процессы организаций важнейшей задачей становится *управление мобильными устройствами*.

Возможность мобильного доступа к корпоративным информационным ресурсам организации порождает ряд проблем с точки зрения информационной безопасности:

– нарушение конфиденциальности информации в результате кражи или утери устройства;

– нарушение конфиденциальности информации в результате доступа посторонних лиц к устройству, оставленному без присмотра;

– доступ к конфиденциальной информации внешних нарушителей посредством использования вредоносного программного кода;

– хищение информации работником (*инсайдером*), имеющим легитимный доступ к информации и хранящим эту информацию на своем устройстве (путем отправки через личную почту, выкладывания в сервисы облачного хранения данных и пр.).

Кроме того, любые дополнительные соединения сегментов компьютерной сети с другими сегментами или подключение к сети Интернет порождают новые проблемы в компьютерной сети.

Более подробно вопросы безопасности и надежности сетей будут рассмотрены в главе 12.

2.3.7. Расширяемость и масштабируемость сети

Расширяемость (Extensibility) – возможность сравнительно легкого добавления отдельных элементов сети (пользователей, компьютеров, приложений, служб), наращивания длины сегментов сети и замены существующей аппаратуры более мощной.

Масштабируемость (Scalability) означает, что сеть позволяет наращивать количество узлов и протяженность связей в широких пределах, при этом производительность сети не снижается. Для обеспечения масштабируемости сети нужно применять дополнительное коммуникационное оборудование и специальным образом структурировать сеть. Классическим примером хорошо масштабируемой сети является Интернет. Также хорошей масштабируемостью обладает многосегментная сеть, построенная с использованием коммутаторов и маршрутизаторов и имеющая иерархическую структуру связей. Такая сеть может включать тысячи компьютеров и при этом обеспечивать каждому пользователю сети нужное качество обслуживания.

Выводы

1. Информация стала одним из самых дорогих продуктов в сфере межличностных отношений.

2. Качество работы сети определяется следующими свойствами: производительность, надежность, совместимость, управляемость, защищенность, расширяемость и масштабируемость.

3. Существует два основных подхода к обеспечению качества работы сети. Первый состоит в том, что сеть гарантирует пользователю соблюдение некоторой числовой величины показателя качества обслуживания. Например, сети Frame Relay и ATM могут гарантировать пользователю заданный уровень пропускной способности. При втором подходе сеть «старается» по возможности более качественно обслужить пользователя, но ничего при этом не гарантирует.

4. К основным характеристикам производительности сети относятся: время реакции, которое определяется как время между возникновением запроса к какому-либо сетевому сервису и получением

ответа на него; пропускная способность, которая отражает объем данных, переданных сетью в единицу времени; задержка передачи, которая равна интервалу между моментом поступления пакета на вход какого-либо сетевого устройства и моментом его появления на выходе этого устройства.

5. Для оценки надежности сети используются различные характеристики, в том числе: коэффициент готовности, означающий долю времени, в течение которого система может быть использована; безопасность, т. е. способность системы защитить данные от несанкционированного доступа; отказоустойчивость – способность системы работать в условиях отказа некоторых ее элементов.

6. Расширяемость означает возможность сравнительно легкого добавления отдельных элементов сети (пользователей, компьютеров, приложений, сервисов), наращивания длины сегментов сети и замены существующей аппаратуры более мощной.

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

8. Прозрачность – свойство сети скрывать от пользователя детали своего внутреннего устройства, упрощая тем самым его работу в сети.

9. Управляемость сети подразумевает возможность централизованно контролировать состояние основных элементов сети, выявлять и разрешать проблемы, возникающие при работе сети, выполнять анализ производительности и планировать развитие сети.

10. Совместимость означает, что сеть способна включать в себя самое разнообразное программное и аппаратное обеспечение.

11. В условиях все большего вовлечения смартфонов и планшетов в бизнес-процессы организаций важнейшей задачей становится управление мобильными устройствами.

Контрольные вопросы

1. Поясните понятие архитектуры сети.
2. В каком случае используется одноранговая архитектура?
3. Что характерно для сетей с выделенным сервером?

4. Что такое топология?

5. Основные достоинства и недостатки топологии «общая шина».

6. Основные достоинства и недостатки топологии «кольцо».

7. Основные достоинства и недостатки топологии «звезда».

8. Основные достоинства и недостатки сеточной топологии.

9. В чем заключаются основные различия между активным и пассивным деревом?

10. Приведите основные различия между полной и частичной сеточными топологиями.

11. Поясните понятия «надежность» и «безопасность» компьютерных сетей.

12. Назовите основные технические и эксплуатационные свойства сетей.

13. Перечислите и охарактеризуйте внешние и внутренние дестабилизирующие факторы, влияющие на надежность и безопасность функционирования компьютерной сети.

14. Опишите особенности проблемы безопасности мобильных сегментов компьютерных сетей.

15. Что такое пропускная способность сети? Каковы ее виды?

16. Какие характеристики влияют на пропускную способность сети?

17. Что понимают под прозрачностью сети?

18. В чем состоит разница между расширяемостью и масштабируемостью сети?

19. Приведите примеры масштабируемых сетей.

3 ОСНОВЫ ПЕРЕДАЧИ ДАННЫХ ПО СЕТИ

3.1. Пакеты и их структура

3.1.1. Назначение пакетов

Информация в локальных сетях, как правило, передается отдельными порциями, кусками, называемыми в различных источниках **пакетами** (Packets), **кадрами** (Frames) или **блоками** (Blocks).

Причем предельная длина этих пакетов строго ограничена (обычно величиной в несколько килобайтов). Ограничена длина пакета и снизу (как правило, несколькими десятками байтов). Выбор пакетной передачи связан с несколькими важными соображениями.

Локальная сеть, как уже отмечалось, должна обеспечивать качественную, прозрачную связь всем абонентам (компьютерам) сети.

> **!** Важнейшим параметром является так называемое **время доступа к сети** (Access Time), которое определяется как временной интервал между моментом готовности абонента к передаче (когда ему есть, что передавать) и моментом начала этой передачи. Это время ожидания абонентом начала своей передачи.

Естественно, оно не должно быть слишком большим, иначе величина реальной, интегральной скорости передачи информации между приложениями сильно уменьшится даже при высокоскоростной связи.

Ожидание начала передачи связано с тем, что в сети не может происходить несколько передач одновременно (во всяком случае, при топологиях «шина» и «кольцо»). Всегда есть только один передатчик и один приемник (реже – несколько приемников). В противном случае информация от разных передатчиков смешивается и искажается. В связи с этим абоненты передают свою информацию по

очереди. И каждому абоненту, прежде чем начать передачу, надо дождаться своей очереди. Вот это время ожидания своей очереди и есть *время доступа*.

Если бы вся требуемая информация передавалась каким-то абонентом сразу, непрерывно, без разделения на пакеты, то это привело бы к монопольному захвату сети этим абонентом на довольно продолжительное время. Все остальные абоненты вынуждены были бы ждать окончания передачи всей информации, что в ряде случаев могло бы потребовать десятков секунд и даже минут (например, при копировании содержимого целого жесткого диска). С тем чтобы уравнять в правах всех абонентов, а также сделать примерно одинаковой величину времени доступа к сети и интегральную скорость передачи информации, как раз и применяются пакеты (кадры) ограниченной длины.

Важно также и то, что при передаче больших массивов информации *вероятность ошибки* (передана «1» – принимается «0» или наоборот) из-за помех и сбоев довольно высока. Например, при характерной для локальных сетей величине вероятности одиночной ошибки в 10^{-8} (в среднем одна ошибка приходится на 100 Мбайт переданных двоичных символов) пакет длиной 10 Кбит будет искажен с вероятностью 10^{-4}, а массив длиной 10 Мбит – уже с вероятностью 10^{-1}. К тому же выявить ошибку в массиве из нескольких мегабайтов намного сложнее, чем в пакете из нескольких килобайтов, а при обнаружении ошибки придется повторить передачу всего большого массива. Но и при повторной передаче большого массива снова высока вероятность ошибки, и процесс этот при слишком большом массиве может повторяться до бесконечности.

С другой стороны, сравнительно большие пакеты имеют преимущества перед очень маленькими пакетами, например перед побайтовой (8 битов) или пословной (16 битов или 32 бита) передачей информации.

Дело в том, что каждый пакет помимо собственно данных, которые требуется передать, должен содержать некоторое количество *служебной информации*. Прежде всего, это адресная информация, которая определяет, от кого и кому передается данный пакет (как на почтовом конверте – адреса получателя и отправителя). Если порция передаваемых данных будет очень маленькой (например, несколько байтов), то доля служебной информации станет непозволительно высокой, что резко снизит интегральную скорость обмена информацией по сети.

Существует некоторая оптимальная длина пакета (или оптимальный диапазон длин пакетов), при которой *средняя скорость обмена информацией* по сети будет максимальна. Эта длина не является неизменной величиной, она зависит от уровня помех, метода управления обменом, количества абонентов сети, характера передаваемой информации и от многих других факторов. Имеется диапазон длин, который близок к оптимуму.

Процесс ***информационного обмена в сети*** представляет собой чередование пакетов, каждый из которых содержит информацию, передаваемую от абонента к абоненту.

В частном случае (рис. 3.1) все эти пакеты могут передаваться одним абонентом (когда другие абоненты «не хотят» передавать). Но обычно в сети чередуются пакеты, посланные разными абонентами (рис. 3.2).

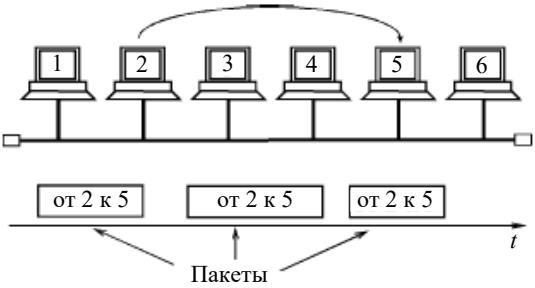


Рис. 3.1. Передача пакетов в сети между двумя абонентами

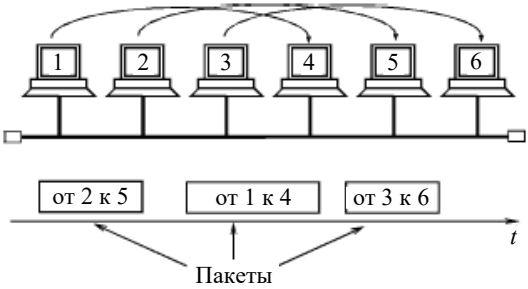


Рис. 3.2. Передача пакетов в сети между несколькими абонентами

3.1.2. Структура пакетов

Структура и размеры пакета в каждой сети жестко определены стандартом на данную сеть и связаны, прежде всего, с аппаратными особенностями (аппаратной платформой) данной сети, выбранной топологией и типом среды передачи информации. Кроме того, эти параметры зависят от используемого протокола (порядка обмена информацией).

Но существуют некоторые общие принципы формирования структуры пакета, которые учитывают характерные особенности обмена информацией по любым локальным сетям.

Чаще всего пакет содержит в себе следующие основные *поля* или части, представленные на рис. 3.3.

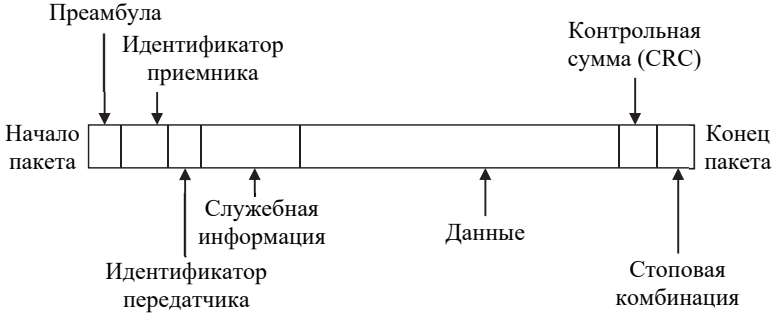


Рис. 3.3. Типичная структура пакета

Преамбула, или **стартовая комбинация битов**, обеспечивает предварительную настройку аппаратуры адаптера или другого сетевого устройства на прием и обработку пакета. Это поле может полностью отсутствовать или же сводиться к единственному стартовому биту.

Идентификатор приемника, или **сетевой адрес принимающего абонента**, – индивидуальный или групповой номер, присвоенный каждому принимающему абоненту (компьютеру) в сети.

Этот адрес (или *IP-адрес*) позволяет приемнику распознать пакет, адресованный ему лично, группе, в которую он входит, или всем абонентам сети одновременно (при широком вещании).

Идентификатор передатчика, или **сетевой адрес передающего абонента**, – индивидуальный номер, присвоенный каждому передающему абоненту.

Этот адрес информирует принимающего абонента, откуда пришел данный пакет. Включение в пакет адреса передатчика необходимо в том случае, когда одному приемнику могут попеременно приходить пакеты от разных передатчиков.

Служебная (управляющая) информация – информация, которая указывает на тип пакета, его номер, размер, формат, маршрут его доставки, на то, что с ним надо делать приемнику, и т. д.

Данные (**поле данных**) – это та информация, ради передачи которой используется пакет.

В отличие от всех остальных полей пакета поле данных имеет переменную длину, которая, собственно, и определяет полную длину пакета. Существуют специальные управляющие пакеты, которые не имеют поля данных. Их можно рассматривать как сетевые команды. Пакеты, включающие поле данных, называются информационными пакетами. Управляющие пакеты могут выполнять функцию начала и конца сеанса связи, подтверждения приема информационного пакета, запроса информационного пакета и т. д.

Контрольная сумма пакета – это числовой код, формируемый передатчиком по определенным правилам и содержащий в свернутом виде информацию обо всем пакете.

Приемник, повторяя вычисления, сделанные передатчиком, с принятым пакетом, сравнивает их результат с контрольной суммой и делает вывод о правильности или ошибочности передачи пакета. Если пакет ошибочен, то приемник запрашивает его повторную передачу. Обычно используется *циклическая контрольная сумма* (CRC, Cyclic Redundancy Check).

Стоповая комбинация служит для информирования принимающего абонента об окончании пакета, обеспечивает выход аппаратуры приемника из состояния приема. Это поле может отсутствовать, если используется самосинхронизирующийся код, позволяющий определять момент окончания передачи пакета.

Нередко в структуре пакета выделяют всего три поля:

– начальное управляющее поле пакета (или заголовок пакета), т. е. поле, включающее в себя стартовую комбинацию, сетевые адреса приемника и передатчика, а также служебную информацию;

– поле данных пакета;

– конечное управляющее поле пакета (заключение, трейлер), куда входят контрольная сумма и стоповая комбинация, а также, возможно, служебная информация.

Структура и вид отдельных полей зависят от применяемой технологии (Ethernet, Token Ring, ARCNet, FDDI и т. д.) и будут рассмотрены в главе 9.

Как уже упоминалось, помимо термина «пакет» (Packet) в литературе также нередко встречается термин «кадр» (Frame). Иногда под этими терминами имеется в виду одно и то же. Но иногда подразумевается, что они различаются: кадр вложен в пакет. В этом случае все перечисленные поля пакета, кроме преамбулы и стоповой комбинации, относятся к кадру (рис. 3.4). Например, в описаниях сети Ethernet говорится, что в конце преамбулы передается признак начала кадра. В других, напротив, поддерживается мнение о том, что пакет вложен в кадр. И тогда под пакетом подразумевается только информация, содержащаяся в кадре, который передается по сети и снабжен служебными полями.

Во избежание путаницы в данной книге термин «пакет» будет использоваться как более понятный и универсальный.

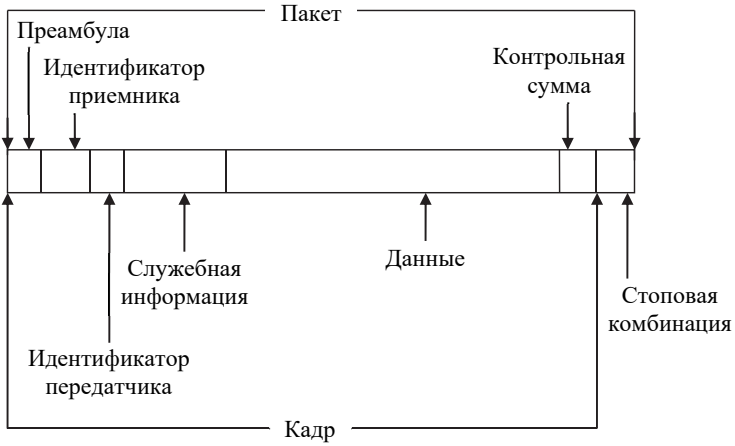


Рис. 3.4. Вложение кадра в пакет

3.1.3. Правила обмена и управления пакетами

В процессе сеанса обмена информацией по сети между передающим и принимающим абонентами происходит обмен информационными и управляющими пакетами по установленным правилам,

которые называются протоколом обмена. Это позволяет обеспечить надежную передачу информации при любой интенсивности обмена по сети.

Пример простейшего протокола показан на рис. 3.5.

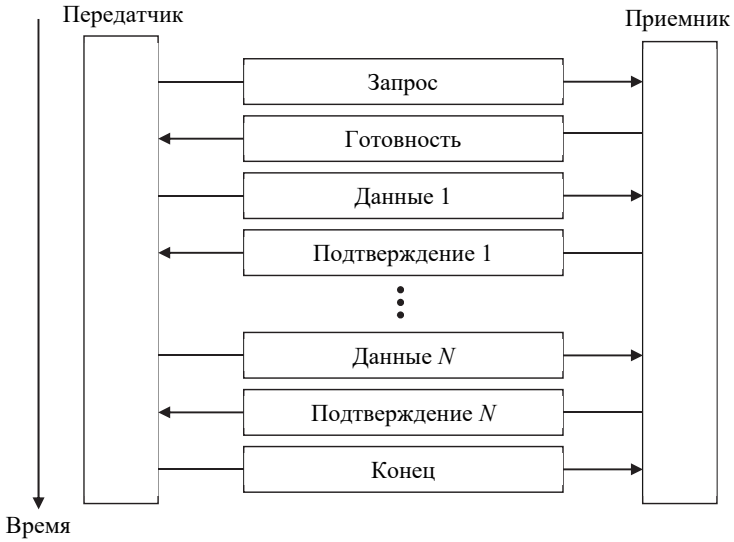


Рис. 3.5. Пример обмена пакетами при сеансе связи

Сеанс обмена начинается с запроса передатчиком готовности приемника принять данные. Для этого используется управляющий пакет «Запрос». Если приемник не готов, он отказывается от сеанса специальным управляющим пакетом. В случае когда приемник готов, он посылает в ответ управляющий пакет «Готовность». Затем начинается собственно передача данных. При этом на каждый полученный информационный пакет приемник отвечает управляющим пакетом «Подтверждение».

В случае когда пакет данных передан с ошибками, в ответ на него приемник запрашивает повторную передачу. Заканчивается сеанс управляющим пакетом «Конец», которым передатчик сообщает о разрыве связи.

Существует множество стандартных протоколов, которые используют как передачу с подтверждением (с гарантированной доставкой пакета), так и передачу без подтверждения (без гарантии

доставки пакета). Подробнее о протоколах обмена будет рассказано в следующей главе.

> **!** При реальном обмене по сети применяются ***многоуровневые протоколы***, каждый из уровней которых предполагает свою структуру пакета (адресацию, управляющую информацию, формат данных и т. д.).

Ведь протоколы высоких уровней имеют дело с такими понятиями, как *файл-сервер* или приложение, которые запрашивают данные у другого приложения и вполне могут не иметь представления ни о типе аппаратуры сети, ни о методе управления обменом. Все пакеты более высоких уровней последовательно вкладываются в передаваемый пакет, точнее говоря, в поле данных передаваемого пакета (рис. 3.6).

Процесс последовательной упаковки данных для передачи называется также **инкапсуляцией пакетов** (Pakcets Encapsulation).

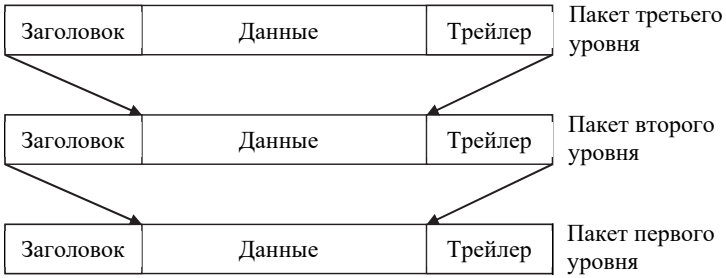


Рис. 3.6. Многоуровневая система вложения пакетов

Каждый следующий вкладываемый пакет может содержать собственную служебную информацию, располагающуюся как до данных (заголовок), так и после них (трейлер), причем ее назначение может быть различным.

Безусловно, доля вспомогательной информации в пакетах при этом возрастает с каждым следующим уровнем, что снижает эффективную скорость передачи данных. Для увеличения этой скорости предпочтительнее, чтобы протоколы обмена были проще и уровней этих протоколов было меньше. Иначе никакая скорость передачи

битов не поможет, и быстрая сеть будет передавать файл дольше, чем медленная сеть, которая пользуется более простым протоколом.

Обратный процесс последовательной распаковки данных приемником называется **декапсуляцией пакетов** (Pakcets Decapsulation).

Выводы

1. Информация в локальных сетях передается отдельными порциями, называемыми пакетами. Причем предельная длина этих пакетов ограничена и в основном зависит от сетевой технологии (Ethernet, Token Ring, FDDI и т. д.).

2. Процесс информационного обмена в сети представляет собой чередование пакетов, каждый из которых содержит информацию, передаваемую от абонента к абоненту.

3. Локальная сеть характеризуется таким важнейшим параметром, как время доступа, которое определяется как временной интервал между моментом готовности абонента к передаче (когда ему есть, что передавать) и моментом начала этой передачи.

4. Пакет содержит в себе следующие основные поля: преамбула, идентификатор передатчика, идентификатор приемника, служебная информация, данные, контрольная сумма, стоповая комбинация.

3.2. Методы доступа в сетях

В современных сетях в основном используются следующие методы доступа:

– *множественный доступ с прослушиванием несущей* (Carrier Sense Multiple Access), который будет рассмотрен в двух вариантах: *множественный доступ с прослушиванием несущей и разрешением коллизий* (Carrier Sense Multiple Access with Collision Detection – CSMA/CD) и *множественный доступ с прослушиванием несущей и предотвращением коллизий* (Carrier Sense Multiple Access with With Collision Avoidance – CSMA/CA);

– *централизованный метод доступа* (Demand Priority, DP);

– *множественный доступ с передачей полномочия* (Token Passing Multiple Access – TPMA), или метод с передачей *маркера*;

– *множественный доступ с разделением во времени* (Time Division Multiple Access – TDMA);

– *множественный доступ с разделением частоты* (Frequency Division Multiple Access – FDMA), или *множественный доступ с разделением длины волны* (Wavelength Division Multiple Access – WDMA).

3.2.1. Множественный доступ с прослушиванием несущей

В компьютерных сетях могут быть использованы различные варианты реализации метода множественного доступа с прослушиванием несущей (CSMA). Первый из них – CSMA/CD.

Метод множественного доступа с прослушиванием несущей и разрешением коллизий (***CSMA/CD***) основан на следующих правилах определения права на передачу: если рабочая станция «хочет» воспользоваться сетью для передачи данных, она сначала должна проверить состояние канала, начинать передачу рабочая станция может, если канал свободен.

В целом метод доступа CSMA/CD описывается алгоритмом, представленным на рис. 3.7.

В процессе передачи рабочая станция продолжает прослушивание сети для обнаружения возможных конфликтов (коллизий). Если возникает конфликт из-за того, что два узла попытаются занять канал, то обнаружившая конфликт интерфейсная плата соответствующего компьютера выдает в сеть специальный сигнал и обе станции одновременно прекращают передачу. Принимающая рабочая станция отбрасывает частично принятое сообщение, а все рабочие станции, желающие передать сообщение, в течение некоторого случайно выбранного промежутка времени выжидают, прежде чем начать сообщение.

Все сетевые интерфейсные платы запрограммированы на разные псевдослучайные промежутки времени ожидания. Если конфликт возникнет во время повторной передачи сообщения, этот промежуток времени будет увеличен.

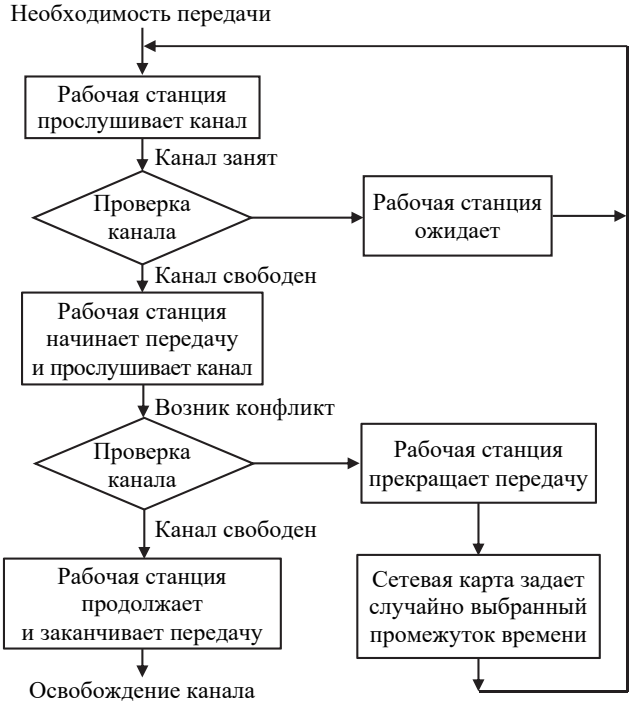


Рис. 3.7. Алгоритм CSMA/CD

Стандарт типа ***Ethernet*** определяет сеть с конкуренцией, в которой несколько рабочих станций должны конкурировать друг с другом за право доступа к сети.

Метод множественного доступа с прослушиванием несущей* и *предотвращением коллизий (CSMA/CA) в отличии от CSMA/CD характеризуется следующими особенностями:
– станция, которая собирается начать передачу, посылает jam signal (сигнал затора);
– после продолжительного ожидания всех станций, которые могут послать jam signal, станция начинает передачу;
– если во время передачи станция обнаруживает jam signal от другой станции, она останавливает передачу на отрезок времени случайной длины и затем повторяет попытку.

В сущности CSMA/CA отличается от CSMA/CD тем, что узлы сообщают о намерении передать данные по сети до фактической их передачи. Узлы постоянно «прослушивают» объявления других узлов и при обнаружении объявления отменяют передачу своих данных.

В таком случае при использовании CSMA/CA коллизиям подвержены не пакеты данных, а только jam-сигналы. Избегание коллизий используется для того, чтобы улучшить производительность CSMA. Улучшение производительности достигается за счет снижения вероятности коллизий и повторных попыток передачи. Но ожидание jam signal создает дополнительные задержки, поэтому другие методы доступа позволяют достичь лучших результатов. Избегание коллизий полезно на практике в тех ситуациях, когда своевременное обнаружение коллизии невозможно.

3.2.2. Централизованный метод доступа

Развитием метода CSMA/CD является *централизованный метод доступа* (Demand Priority, DP), в котором концентратор выступает в роли «арбитра» – проблема доступа к разделяемой среде решается через передачу запросов концентратору, циклически прослушивающему всех абонентов по очереди и дающему право передачи абоненту, следующему по порядку за тем, который закончил передачу. Величина времени доступа в таком случае в отличии от обычного CSMA/CD гарантирована. В данном методе доступа реализованы два уровня приоритетов: низкий – для обычных приложений и высокий – для мультимедийных. Запросы с высоким уровнем приоритета (высокоприоритетные) обслуживаются раньше, чем запросы с нормальным приоритетом (низкоприоритетные). Если приходит запрос высокого приоритета, то нормальный порядок обслуживания прерывается, и после окончания приема текущего пакета обслуживается запрос высокого приоритета. Если таких высокоприоритетных запросов несколько, то возврат к нормальной процедуре обслуживания происходит только после полной обработки всех этих запросов. Можно сказать, что высокоприоритетные запросы обслуживаются вне очереди, но они образуют свою очередь. При этом концентратор следит за тем, чтобы не была превышена установленная величина гарантированного времени доступа для низкоприоритетных запросов. Если высокоприоритетных запросов слишком много, то запросы с

нормальным приоритетом автоматически переводятся им в ранг высокоприоритетных. Типичная величина времени повышения приоритета равна 200–300 мс (устанавливается при конфигурировании сети). Таким образом, даже низкоприоритетные запросы не будут ждать своей очереди слишком долго.

Данный метод доступа разрабатывался специально для сетей 100VG-AnyLAN, ориентированных на передачу мультимедийной информации. Он обеспечивает более справедливое распределение пропускной способности сети.

3.2.3. Множественный доступ с передачей полномочия

Алгоритм *множественного доступа с передачей полномочия* (TPMA), или *маркера* (Token), приведен на рис. 3.8.

Метод с передачей маркера – это метод доступа к среде, в которой от рабочей станции к рабочей станции передается маркер, дающий разрешение на передачу сообщения.

При получении маркера рабочая станция может передавать сообщение, присоединяя его к маркеру, который переносит это сообщение по сети. Каждая рабочая станция между передающей и принимающей станциями видит это сообщение, но только станция-адресат принимает его. При этом она создает новый маркер.

Маркер, или **полномочие**, – уникальная комбинация битов, позволяющая начать передачу данных.

Каждый узел принимает пакет от предыдущего, восстанавливает уровни сигналов до номинального (требуемого) уровня и передает дальше. Передаваемый пакет может содержать данные или являться маркером. Когда рабочей станции необходимо передать пакет, ее адаптер дожидается поступления маркера, а затем преобразует его в пакет, содержащий данные, отформатированные по протоколу соответствующего уровня, и передает результат далее по ЛВС. Пакет распространяется по ЛВС от адаптера к адаптеру, пока не найдет своего адресата, который установит в нем определенные биты для подтверждения того, что данные достигли адресата, и ретранслирует его вновь в ЛВС. После чего пакет возвращается в узел, из которого

был отправлен. Здесь после проверки безошибочной передачи пакета узел освобождает ЛВС, генерируя новый маркер. Таким образом, в ЛВС с передачей маркера невозможны *коллизии* (конфликты).

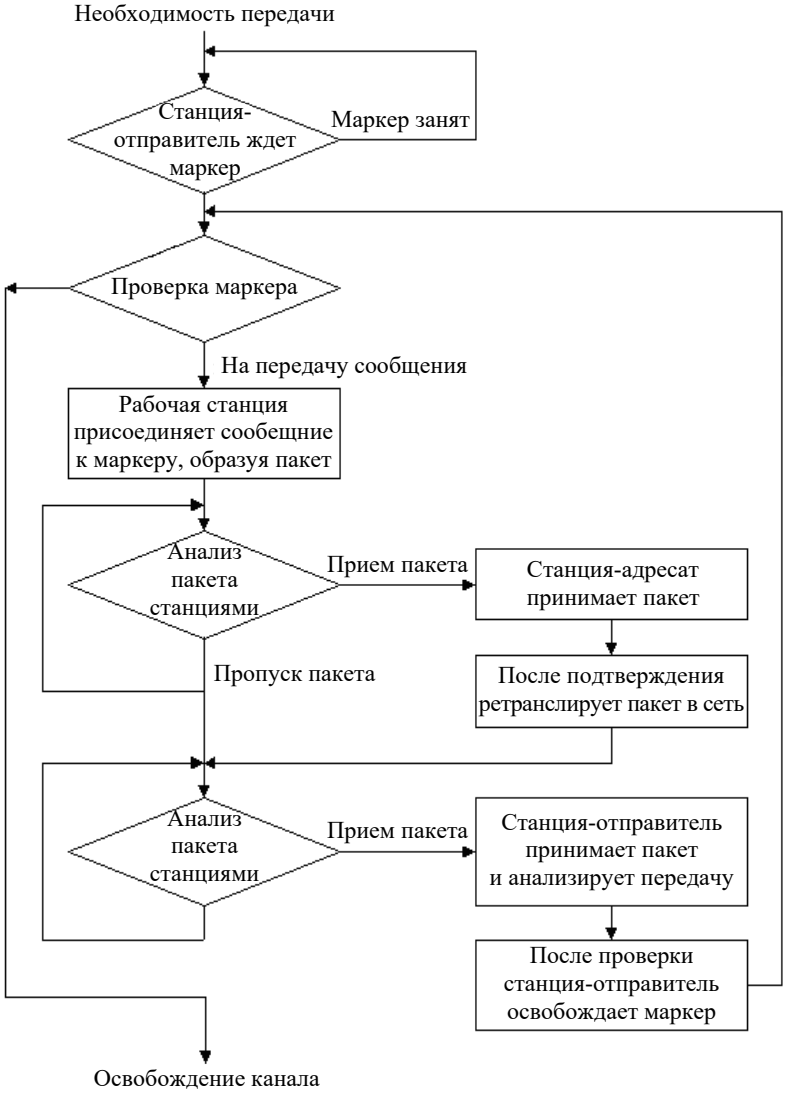


Рис. 3.8. Алгоритм TPMA

Метод с передачей маркера в основном используется в кольцевой топологии.

Данный метод характеризуется следующими достоинствами:

– гарантирует время доставки блоков данных в сети;

– дает возможность предоставления различных *приоритетов передачи данных*.

Вместе с тем метод с передачей маркера имеет существенные недостатки:

– в сети возможна потеря маркера, а также появление нескольких маркеров, при этом сеть прекращает работу;

– включение новой рабочей станции и отключение связаны с изменением адресов всей системы.

3.2.4. Множественный доступ с разделением во времени

Множественный доступ с разделением во времени (TDMA) основан на распределении времени работы канала между системами (рис. 3.9).

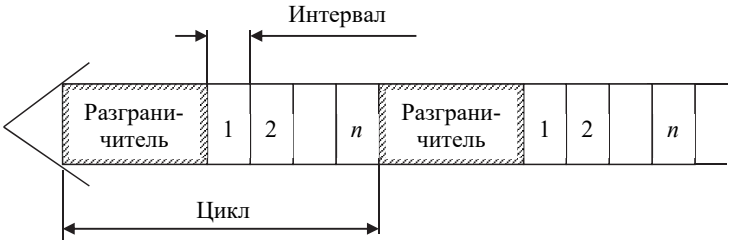


Рис. 3.9. Структура множественного доступа с разделением во времени

Доступ TDMA основан на использовании специального устройства, называемого *тактовым генератором*. Этот генератор делит время канала на повторяющиеся циклы. Каждый из циклов начинается *сигналом-разграничителем*. Цикл включает *n* пронумерованных временных интервалов, называемых *ячейками*. Интервалы предоставляются для загрузки в них блоков данных.

Этот способ позволяет организовать передачу данных с *коммутацией пакетов* и *коммутацией каналов*.

Первый (простейший) вариант использования интервалов заключается в том, что их число (n) делается равным количеству абонентских систем, подключенных к рассматриваемому каналу. Тогда во время цикла каждой системе предоставляется один интервал, в течение которого она может передавать данные. При использовании рассмотренного метода доступа часто оказывается, что в одном и том же цикле одним системам нечего передавать, а другим не хватает выделенного времени. В результате наблюдается неэффективное использование пропускной способности канала.

Второй более сложный, но высокоэкономичный вариант заключается в том, что система получает интервал только тогда, когда у нее возникает необходимость в передаче данных, например при асинхронном способе передачи. Для передачи данных система может в каждом цикле получать интервал с одним и тем же номером. В этом случае передаваемые системой блоки данных появляются через одинаковые промежутки времени и приходят с одним и тем же временем запаздывания. Такой режим называется передачей данных с имитацией коммутации каналов. Способ особенно удобен при передаче речи.

3.2.5. Множественный доступ с разделением частоты

Множественный доступ с разделением частоты (FDMA) заключается в разделении *полосы пропускания* канала на группу полос частот, представленных на рис. 3.10, которые образуют *логические каналы*.

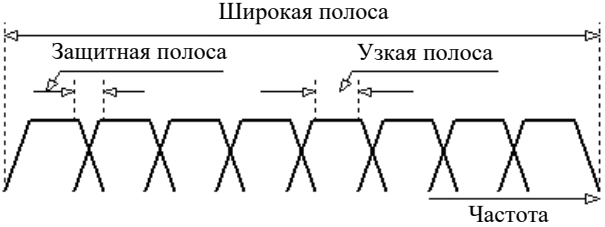


Рис. 3.10. Схема выделения логических каналов

Широкая полоса пропускания канала делится на ряд узких полос, разделенных защитными полосами. Размеры узких полос могут быть различными.

При использовании метода доступа FDMA, а также схожего по принципам организации ***множественного доступа с разделением длины волны*** (WDMA), широкая полоса пропускания канала делится на ряд узких полос, которые разделены защитными полосами. В каждой узкой полосе создается логический канал. Размеры узких полос могут быть различными.

Передаваемые по логическим каналам сигналы накладываются на разные несущие и поэтому в частотной области не должны пересекаться. Вместе с этим, несмотря на наличие защитных полос, спектральные составляющие сигнала могут выходить за границы логического канала и вызывать *шум* в соседнем логическом канале.

В *оптических* каналах разделение частоты осуществляется направлением в каждый из них лучей света с различными частотами. Благодаря этому пропускная способность физического канала увеличивается в несколько раз. При осуществлении этого мультиплексирования в один световод большое число лазеров излучает свет (на различных частотах). Через световод излучение каждого из них проходит независимо от другого. На приемном конце разделение частот сигналов, прошедших физический канал, осуществляется путем фильтрации выходных сигналов.

Метод доступа FDMA относительно прост, но для его реализации необходимы передатчики и приемники, работающие на различных частотах.

Выводы

1. В современных сетях в основном используются следующие методы доступа: множественный доступ с прослушиванием несущей и разрешением коллизий; множественный доступ с передачей полномочия или метод с передачей маркера; множественный доступ с

разделением во времени; множественный доступ с разделением частоты; множественный доступ с разделением длины волны.

2. Множественный доступ с прослушиванием несущей является самым простым с точки зрения реализации и применяется в технологии Ethernet в качестве базового.

3. Множественный доступ с передачей полномочия гарантирует определенное время доставки блоков данных в сети, а также дает возможность предоставления различных приоритетов передачи данных, но включение новой рабочей станции и (или) ее отключение приводит к изменениям адресов всей системы. Применяется в различных сетевых технологиях, например Token Ring.

4. Множественный доступ с разделением во времени позволяет организовать передачу данных с коммутацией пакетов и коммутацией каналов. Данный метод доступа особенно удобен при передаче речи или мультимедийной информации в режиме реального времени.

5. Множественный доступ с разделением частоты нашел применение в беспроводных системах связи, в то же время множественный доступ с разделением длины волны активно применяется в оптоволоконных системах.

3.3. Семиуровневая модель OSI

Для единого представления данных в сетях с неоднородными устройствами и программным обеспечением международная организация по стандартам ISO (International Standardization Organization) разработала ***базовую модель связи открытых систем OSI*** (Open System Interconnection).

Модель OSI описывает правила и процедуры передачи данных в различных сетевых средах при организации сеанса связи. Основными элементами модели являются уровни, прикладные процессы и физические средства соединения.

На рис. 3.11 представлена структура базовой модели. Каждый уровень модели OSI выполняет определенную задачу в процессе

передачи данных по сети (см. п. 2.3.5). Базовая модель является основой для разработки сетевых протоколов. OSI разделяет коммуникационные функции в сети на семь уровней, каждый из которых обслуживает различные части процесса взаимодействия открытых систем.

Рис. 3.11. Структура модели OSI

Модель OSI описывает только системные средства взаимодействия, не касаясь приложений конечных пользователей. Приложения реализуют свои собственные протоколы взаимодействия, обращаясь к системным средствам. Если приложение может взять на себя функции некоторых верхних уровней модели OSI, то для обмена данными оно обращается напрямую к системным средствам, выполняющим функции оставшихся нижних уровней модели OSI.

3.3.1. Взаимодействие уровней модели OSI

Модель OSI можно разделить на две различных модели (рис. 3.12):

– ***горизонтальную модель*** на базе протоколов, обеспечивающую механизм взаимодействия программ и процессов на различных машинах;

– ***вертикальную модель*** на основе услуг, обеспечиваемых соседними уровнями друг другу на одной машине.

Каждый уровень компьютера-отправителя взаимодействует с таким же уровнем компьютера-получателя, как будто он связан напрямую. Такая связь называется **логической**, или **виртуальной**, **связью**. В действительности взаимодействие осуществляется между смежными уровнями одного компьютера.

Рис. 3.12. Схема взаимодействия компьютеров в базовой эталонной модели OSI

Итак, информация на компьютере-отправителе должна пройти через все уровни. Затем она передается по физической среде до компьютера-получателя и опять проходит сквозь все слои, пока не доходит до того же уровня, с которого она была послана на компьютере-отправителе.

В горизонтальной модели двум программам требуется общий протокол для обмена данными. В вертикальной модели соседние

уровни обмениваются данными с использованием интерфейсов прикладных программ *API* (Application Programming Interface).

Перед подачей в сеть данные разбиваются на *пакеты*. При отправке данных пакет проходит последовательно через все уровни программного обеспечения. На каждом уровне к пакету добавляется управляющая информация данного уровня (заголовок), которая необходима для успешной передачи данных по сети, как это показано на рис. 3.13, где Заг – заголовок пакета, Кон – конец пакета.

На принимающей стороне пакет проходит через все уровни в обратном порядке. На каждом уровне протокол этого уровня читает информацию пакета, затем удаляет информацию, добавленную к пакету на этом же уровне отправляющей стороной, и передает пакет следующему уровню. Когда пакет дойдет до ***Прикладного*** уровня, вся управляющая информация будет удалена из него и данные примут свой первоначальный вид.

Каждый уровень модели выполняет свою функцию. Чем выше уровень, тем более сложную задачу он решает. Отдельные уровни модели OSI удобно рассматривать как группы программ, предназначенных для выполнения конкретных функций. Один уровень, к примеру, отвечает за обеспечение преобразования данных из ASCII в EBCDIC и содержит программы, необходимые для выполнения этой задачи.

Каждый уровень обеспечивает сервис для вышестоящего уровня, запрашивая, в свою очередь, сервис у нижестоящего уровня. Верхние уровни запрашивают сервис почти одинаково: как правило, это требование маршрутизации каких-то данных из одной сети в другую. Практическая реализация принципов адресации данных возложена на нижние уровни.

Модель OSI определяет *взаимодействие открытых систем* разных производителей в одной сети.

Поэтому она выполняет для них координирующие действия по следующим аспектам:

– взаимодействие прикладных процессов;
– формы представления данных;
– единообразное хранение данных;
– управление сетевыми ресурсами;
– безопасность данных и защита информации;
– диагностика программ и технических средств.

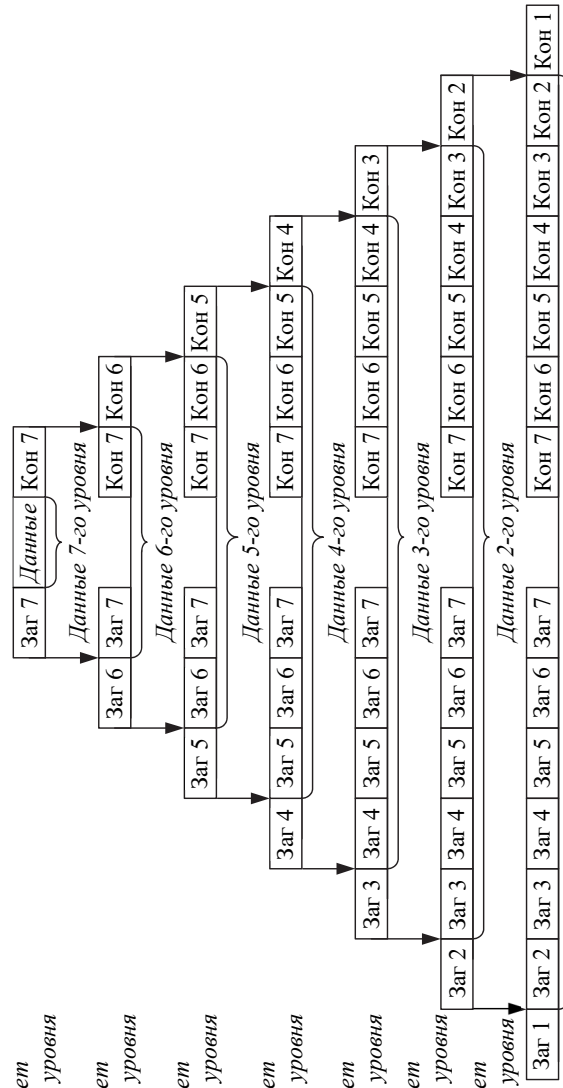


Рис. 3.13. Формирование пакета каждого уровня семиуровневой модели

В следующей таблице приведено краткое описание функций всех уровней.

Краткое описание функций уровней модели OSI

Наименование уровня	Функция	Тип данных (PDU, protocol data units)
Прикладной	Представляет набор интерфейсов, позволяющих получить доступ к сетевым службам, согласует требования к процессу передачи и т. д.	Сообщение (message)
Представления	Преобразует данные, например, в общий формат, засекречивает и т. д.	Сообщение (message)
Сеансовый	Поддерживает сеанс связи, т. е. взаимодействие между процессами	Сообщение (message)
Транспортный	Управляет передачей данных по сети, обеспечивает требуемый уровень надежности (исправление ошибок, подтверждение передачи и т. д.)	Блоки / Дейтаграммы. Разбиение сообщения на блоки фактически является началом процесса формирования пакета
Сетевой	Обеспечивает выполнение задач маршрутизации, управляет потоками данных, адресацией сообщений для доставки, преобразованием логических сетевых адресов и имен в соответствующие им физические	Пакет (packet)
Канальный	Управляет формированием кадров (LLC) и доступом к среде (MAC)	Кадр (frame)
Физический	Реализует битовые протоколы передачи данных, управляет передачей и приемом потока байтов через физическое устройство, выполняет контроль (физический, технический) за процессом передачи	На «входе» – кадр в виде набора битов данных, на «выходе» – физический сигнал

Отметим, что дальше рассматривать модель OSI будем от седьмого, т. е. прикладного, и до первого, т. е. физического, уровня, так как именно в данной последовательности «движется» информация от пользовательского программного обеспечения к процессу передачи по сети.

3.3.2. Прикладной уровень

Прикладной уровень (Application Layer) обеспечивает прикладные процессы средствами доступа к области взаимодействия, является верхним (седьмым) уровнем и непосредственно примыкает к прикладным процессам. В действительности прикладной уровень – это набор разнообразных протоколов, с помощью которых пользователи сети получают доступ к разделяемым ресурсам, таким как файлы, принтеры или гипертекстовые web-страницы, а также организуют свою совместную работу, например, с помощью протокола электронной почты. Специальные элементы прикладного сервиса обеспечивают сервис для конкретных прикладных программ, таких как программы пересылки файлов и эмуляции терминалов. Если, например, программе необходимо переслать файлы, то обязательно будет использован ***протокол передачи, доступа и управления файлами***, FTAM (File Transfer, Access, and Management). В модели OSI прикладная программа, которой нужно выполнить конкретную задачу (например, обновить базу данных на компьютере), посылает конкретные данные в виде ***дейтаграммы*** (Datagram) на прикладной уровень. Одна из основных задач этого уровня – определить, как следует обрабатывать запрос прикладной программы. Другими словами, какой вид должен принять данный запрос.

Единица данных, которой оперирует прикладной уровень, обычно называется **сообщением** (Message).

Прикладной уровень выполняет следующие функции:

– описание форм и методов взаимодействия прикладных процессов;

– согласование требований к различным видам работ (например, передача файлов, управление заданиями, управление системой и т. д.) и управление ими;

– идентификация пользователей по их паролям, адресам, электронным подписям;

– выявление функционирующих абонентов и возможности доступа к новым прикладным процессам;

– определение достаточности имеющихся ресурсов;

– организация запросов на соединение с другими прикладными процессами;

– передача заявок представительскому уровню на необходимые методы описания информации;

– выбор процедур планируемого диалога процессов;

– управление данными, которыми обмениваются прикладные процессы, и синхронизация взаимодействия прикладных процессов;

– определение качества обслуживания (время доставки блоков данных, допустимой частоты ошибок);

– получение соглашения об исправлении ошибок и определении достоверности данных;

– согласование ограничений, накладываемых на синтаксис (наборы символов, структура данных).

Указанные функции определяют виды сервиса, которые прикладной уровень предоставляет прикладным процессам. Кроме того, прикладной уровень передает прикладным процессам сервис, предоставляемый физическим, канальным, сетевым, транспортным, сеансовым и представительским уровнями.

На *прикладном уровне* необходимо предоставить в распоряжение пользователей уже переработанную информацию. С этим может справиться системное и пользовательское ПО.

Прикладной уровень отвечает за доступ приложений в сеть. Задачами этого уровня является перенос файлов, обмен почтовыми сообщениями и управление сетью.

К числу наиболее распространенных протоколов верхних трех уровней относятся:

– HTTP (HyperText Transfer Protocol) – протокол прикладного уровня передачи данных, изначально в виде гипертекстовых документов в формате HTML, в настоящее время используется для передачи произвольных данных;

– FTP (File Transfer Protocol) – протокол передачи файлов по сети;

– TFTP (Trivial File Transfer Protocol) – простейший протокол пересылки файлов;

– X.400 – электронная почта;

– Telnet – работа с удаленным терминалом;

– SMTP (Simple Mail Transfer Protocol) – простой протокол почтового обмена;

– POP3 (Post Office Protocol, Version 3) – стандартный протокол, используемый клиентами электронной почты для получения почты с удаленного сервера по TCP-соединению;

– IMAP4 (Internet Message Access Protocol, Version 4) – протокол прикладного уровня для доступа к электронной почте;

– CMIP (Common Management Information Protocol) – общий протокол управления информацией;

– SNMP (Simple Network Management Protocol) – простой протокол сетевого управления;

– FTAM (File Transfer, Access, and Management) – протокол передачи, доступа и управления файлами.

3.3.3. Уровень представления данных

Уровень представления данных, или представительский уровень (Presentation Layer), представляет данные, передаваемые между прикладными процессами, в нужной форме.

Этот уровень обеспечивает то, что информация, передаваемая прикладным уровнем, будет «понятна» прикладному уровню в другой системе или транспортному уровню той же системы. В случаях необходимости уровень представления в момент передачи информации выполняет преобразование форматов данных в некоторый общий формат представления, а в момент приема, соответственно, выполняет обратное преобразование.

Таким образом, прикладные уровни могут преодолеть, например, синтаксические различия в представлении данных. Подобная ситуация может возникнуть в ЛВС с неоднотипными компьютерами (IBM PC и Macintosh), которым необходимо обмениваться данными. Так, в полях баз данных информация должна быть представлена в виде букв и цифр, а зачастую и в виде графического изображения. Обрабатывать же эти данные нужно, например, как числа с плавающей запятой.

В основу общего представления данных положена единая для всех уровней модели система ASN.1. Эта система служит для описания структуры файлов и позволяет решить проблему шифрования данных.

На *уровне представления данных* может выполняться их шифрование и дешифрование, благодаря которым секретность обмена данными обеспечивается сразу для всех прикладных сервисов. Примером такого протокола является протокол ***Secure Socket Layer*** (SSL), обеспечивающий секретный обмен сообщениями для протоколов прикладного уровня стека TCP/IP.

Представительный уровень выполняет следующие функции:

– генерация запросов на установление сеансов взаимодействия прикладных процессов;

– согласование представления данных между прикладными процессами;

– реализация форм представления данных;

– преобразование данных (кодирование, компрессия и т. д.);

– засекречивание данных (шифрование);

– передача запросов при необходимости на прекращение сеансов.

Протоколы уровня представления данных обычно являются составной частью протоколов трех верхних уровней модели. Отдельно можно выделить только протокол SSL, который зачастую используется в паре с HTTP (получается HTTPs) для обеспечения безопасности.

3.3.4. Сеансовый уровень

Сеансовый уровень (Session Layer) – это уровень, определяющий процедуру проведения сеансов между пользователями или прикладными процессами.

Сеансовый уровень обеспечивает управление диалогом для того, чтобы фиксировать, какая из сторон является активной в настоящий момент, а также предоставляет средства синхронизации. Последние позволяют вставлять контрольные точки в длинные передачи, чтобы в случае отказа можно было вернуться назад к последней контрольной точке, а не начинать все сначала. На практике немногие приложения используют сеансовый уровень, и он редко реализуется.

Сеансовый уровень управляет передачей информации между прикладными процессами, координирует прием, передачу и выдачу одного сеанса связи. Кроме того, этот уровень содержит дополнительно функции управления паролями и диалогом, синхронизации и отмены связи в сеансе передачи после сбоя вследствие ошибок в нижерасположенных уровнях.

Функции этого уровня состоят в координации связи между двумя прикладными программами, работающими на разных рабочих станциях. Это происходит в виде хорошо структурированного диалога.

В число этих функций входит создание сеанса, управление передачей и приемом пакетов сообщений во время сеанса и завершение сеанса.

На сеансовом уровне определяется, какой будет передача между двумя прикладными процессами:

– *полудуплексной* (Half Duplex; процессы или средства будут передавать и принимать данные по очереди);

– *дуплексной* (Duplex или Full Duplex; процессы или средства будут передавать и принимать данные одновременно).

В полудуплексном режиме сеансовый уровень выдает маркер данных тому процессу, который начинает передачу. Когда второму процессу приходит время отвечать, маркер данных передается ему. Сеансовый уровень разрешает передачу только той стороне, которая обладает маркером данных.

Сеансовый уровень обеспечивает выполнение следующих функций:

– установление и завершение на сеансовом уровне соединения между взаимодействующими системами;

– управление выполнением нормального и срочного обмена данными между прикладными процессами;

– управление взаимодействием прикладных процессов;

– синхронизация сеансовых соединений;

– извещение прикладных процессов об исключительных ситуациях, касающихся сеанса связи;

– установление в прикладном процессе меток, позволяющих после отказа либо ошибки восстановить его выполнение от ближайшей метки;

– прерывание в нужных случаях прикладного процесса и его корректное возобновление;

– прекращение сеанса без потери данных;

– передача особых сообщений о ходе проведения сеанса.

Сеансовый уровень отвечает за организацию сеансов обмена данными между оконечными машинами. Протоколы сеансового уровня обычно являются составной частью протоколов трех верхних уровней модели. Но есть и отдельные протоколы, относящиеся прежде всего к сеансовому уровню:

– ADSP (AppleTalk Data Stream Protocol);

– ASP (AppleTalk Session Protocol);

– RPC (Remote Procedure Call);

– PAP (Password Authentication Protocol).

3.3.5. Транспортный уровень

Транспортный уровень (Transport Layer) предназначен для управления передачей пакетов через коммуникационную сеть. На транспортном уровне сообщение (Message), приходящее с вышележащих уровней, разбивается на блоки. Фактически это является началом формирования пакета.

На пути от отправителя к получателю пакеты могут быть искажены или утеряны. Хотя некоторые приложения имеют собственные средства обработки ошибок, существуют и такие, которые предпочитают сразу иметь дело с надежным соединением.

Работа *транспортного уровня* заключается в том, чтобы обеспечить приложениям или верхним уровням модели (прикладному и сеансовому) передачу данных с той степенью надежности, которая им требуется.

Модель OSI определяет пять классов сервиса, предоставляемых транспортным уровнем. Эти виды сервиса отличаются качеством предоставляемых услуг: срочностью, возможностью восстановления прерванной связи, наличием средств мультиплексирования нескольких соединений между различными прикладными протоколами через общий транспортный протокол, а главное способностью к обнаружению и исправлению ошибок передачи, таких как искажение, потеря и дублирование пакетов.

Этот уровень гарантирует доставку блоков информации адресатам и управляет этой доставкой. Его главной задачей является обеспечение эффективных, удобных и надежных форм передачи информации между системами. Когда в процессе обработки находится более одного пакета, транспортный уровень контролирует очередность прохождения пакетов. Если проходит дубликат принятого ранее сообщения, то данный уровень опознает это и игнорирует сообщение.

В функции транспортного уровня входят:

– управление передачей по сети и обеспечение целостности пакетов данных;

– восстановление передачи после отказов и неисправностей;

– укрупнение или разделение пакетов данных;

– обнаружение ошибок, частичная их ликвидация и сообщение о неисправленных ошибках;

– предоставление приоритетов при передаче пакетов (нормальная или срочная);

– подтверждение передачи;

– ликвидация пакетов при тупиковых ситуациях в сети.

Начиная с транспортного уровня, все вышележащие протоколы реализуются программными средствами, обычно включаемыми в состав сетевой операционной системы.

Наиболее распространенные протоколы транспортного уровня включают в себя:

– TCP (Transmission Control Protocol) – протокол управления передачей стека TCP/IP;

– UDP (User Datagram Protocol) – пользовательский протокол дейтаграмм стека TCP/IP;

– NCP (NetWare Core Protocol) – базовый протокол сетей NetWare;

– SPX (Sequenced Packet eXchange) – упорядоченный обмен пакетами стека Novell;

– SCTP (Stream Control Transmission Protocol) – является более новым протоколом, поэтому SCTP имеет несколько нововведений, таких как многопоточность, защита от DDoS-атак, синхронное соединение между двумя хостами по двум и более независимым физическим каналам (multi-homing).

3.3.6. Сетевой уровень

Сетевой уровень (Network Layer) служит для образования единой транспортной системы, объединяющей несколько сетей, причем эти сети могут использовать совершенно различные принципы передачи сообщений между конечными узлами и обладать произвольной структурой связей.

Сетевой уровень обеспечивает прокладку логических каналов, соединяющих абонентские и административные системы через коммуникационную сеть, а также выбор наиболее быстрого и надежного пути.

Виртуальный, или **логический**, **канал** – это такое функционирование компонентов сети, которое создает взаимодействующим компонентам иллюзию прокладки между ними нужного тракта. Кроме того, сетевой уровень сообщает транспортному уровню о появляющихся ошибках.

В целом для регулирования доставки данных внутри сети достаточно наличия канального уровня и его протоколов (передавать данные можно по физическому (MAC) адресу), а вот доставкой данных между сетями занимается именно сетевой уровень, так как данный процесс требует определения оптимального маршрута передачи – при этом используется именно сетевой, а не физический адрес, который состоит из номера сети и номера компьютера в этой сети. Именно номер сети используется для определения маршрута передачи данных. Соответственно, сетевой уровень должен уметь выполнять преобразования MAC-адресов в сетевые адреса и обратно.

Сети соединяются между собой специальными устройствами, называемыми маршрутизаторами.

Маршрутизатор (Router) – это устройство, которое собирает информацию о топологии межсетевых соединений и на ее основании пересылает пакеты сетевого уровня в сеть назначения.

Для того чтобы передать сообщение от отправителя, находящегося в одной сети, получателю, находящемуся в другой сети, нужно совершить некоторое количество транзитных передач (hops) между сетями, каждый раз выбирая подходящий маршрут. Таким образом, маршрут представляет собой последовательность маршрутизаторов, по которым проходит пакет.

На рис. 3.14 показаны четыре сети, связанные маршрутизаторами. Между узлами *A* и *B* данной сети пролегают два маршрута: первый – через маршрутизаторы 1 и 3, а второй – через маршрутизаторы 1, 2 и 3.

Прокладка наилучшего пути для передачи данных называется **маршрутизацией** (Routing), и это является главной задачей сетевого уровня.

Эта проблема осложняется тем, что самый короткий путь не всегда самый лучший. Часто критерием при выборе маршрута является время передачи данных по этому маршруту; оно зависит от пропускной способности каналов связи и интенсивности трафика, которая может изменяться с течением времени.

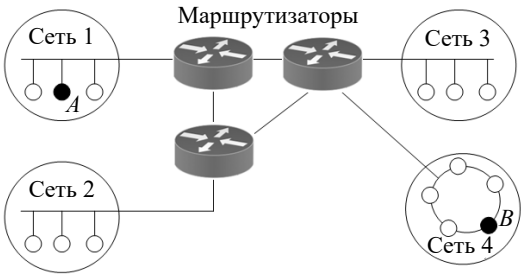


Рис. 3.14. Пример составной сети

Некоторые алгоритмы маршрутизации пытаются приспособиться к изменению нагрузки, в то время как другие принимают решения на основе средних показателей за длительное время. Выбор маршрута может осуществляться и по другим критериям, например надежности передачи.

> ! *Сетевой уровень* также обеспечивает прозрачную передачу пакетов на транспортный уровень.

В целом сетевой уровень выполняет следующие функции:
– создание сетевых соединений и идентификация их портов;
– управление потоками пакетов;
– организация (упорядочение) последовательностей пакетов;
– маршрутизация и коммутация;
– сегментирование и объединение пакетов.

На сетевом уровне определяется два вида протоколов. Первый вид относится к определению правил передачи пакетов с данными конечных узлов от узла к маршрутизатору и между маршрутизаторами. Именно эти протоколы обычно имеют в виду, когда говорят о протоколах сетевого уровня. Однако часто к сетевому уровню относят и другой вид протоколов, называемых протоколами обмена маршрутной информацией. С помощью этих протоколов маршрутизаторы собирают информацию о топологии межсетевых соединений.

> ! *Протоколы сетевого уровня* реализуются программными модулями операционной системы, а также программными и аппаратными средствами маршрутизаторов.

Наиболее часто на сетевом уровне используются протоколы:

– IP (Internet Protocol) – протокол Internet, сетевой протокол стека TCP/IP, который предоставляет адресную и маршрутную информацию;

– IPX (Internetwork Packet Exchange) – протокол межсетевого обмена пакетами, предназначенный для адресации и маршрутизации пакетов в сетях Novell;

– X.25 – международный стандарт для глобальных коммуникаций с коммутацией пакетов (частично этот протокол реализован на уровне 2);

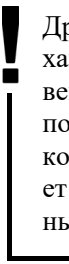
– CLNP (Connection Less Network Protocol) – сетевой протокол без организации соединений.

3.3.7. Канальный уровень

Главная задача *канального уровня* (Data Link) – брать пакеты, поступающие с сетевого уровня и готовить их к передаче, укладывая в кадр соответствующего размера, который далее передается физическому уровню.

Этот уровень обязан определить, где начинается и где заканчивается блок, а также обнаруживать ошибки передачи. Поэтому единицей информации канального уровня являются *кадры* (*Frame*). Фактически **кадры** – это логически организованная структура, в которую можно помещать данные.

На физическом уровне просто пересылаются биты в виде физических сигналов. При этом не учитывается, что в некоторых сетях, в которых линии связи используются попеременно несколькими парами взаимодействующих компьютеров, физическая среда передачи может быть занята. Поэтому одной из задач канального уровня является определение доступа к среде и управление передачей посредством процедуры передачи данных по каналу. При больших размерах передаваемых блоков данных канальный уровень делит их на кадры и передает кадры в виде последовательностей. При получении кадров уровень формирует из них переданные блоки данных. Размер блока данных зависит от способа передачи, качества канала, по которому он передается.



Другой задачей *канального уровня* является реализация механизмов обнаружения и коррекции ошибок. Канальный уровень обеспечивает корректность передачи каждого кадра, помещая специальную последовательность битов в начало и конец каждого кадра, чтобы отметить его, а также вычисляет контрольную сумму, суммируя все байты кадра определенным способом и добавляя контрольную сумму к кадру.

Когда кадр приходит, получатель снова вычисляет *контрольную сумму* (CRC) полученных данных и сравнивает результат с контрольной суммой из кадра. Если они совпадают, кадр считается правильным и принимается. Если же контрольные суммы не совпадают, то фиксируется ошибка. Вместо простой контрольной суммы могут применяться *корректирующие коды*, которые не только обнаруживают факт наличия ошибки, но и могут ее исправить.

На этом же уровне определяются правила использования физического уровня узлами сети. Физическое (электрическое) представление данных в сетях (биты данных, методы кодирования данных и маркеры) распознаются на этом и только на этом уровне.

Итак, канальный уровень обеспечивает создание, передачу и прием кадров данных. Этот уровень обслуживает запросы сетевого уровня и использует сервис физического уровня для приема и передачи пакетов. Спецификации IEEE 802.X делят канальный уровень на два подуровня:

– *LLC* (*Logical Link Control, управление логическим каналом*), который осуществляет логический контроль связи. Подуровень LLC обеспечивает обслуживание сетевого уровня и связан с передачей и приемом пользовательских сообщений;

– *MAC* (*Media Assess Control, контроль доступа к среде*), который регулирует доступ к разделяемой физической среде (передача маркера, обнаружение коллизий или столкновений) и управляет доступом к каналу связи. Подуровень *LLC* находится выше подуровня *MAC.*

Канальный уровень может выполнять следующие виды функций:

– организация (установление, управление, расторжение) канальных соединений и идентификация их портов;

– организация и передача кадров;

– обнаружение и исправление ошибок;

– управление потоками данных;

– обеспечение прозрачности логических каналов (передачи по ним данных, закодированных любым способом).

Наиболее часто используемые протоколы (или реализуемые технологии) на канальном уровне включают:

– HDLC (High Level Data Link Control) – протокол управления каналом передачи данных высокого уровня, для последовательных соединений;

– IEEE 802.2 LLC (типы I, II) – обеспечивают MAC для сред 802.x;

– Ethernet – сетевая технология по стандарту IEEE 802.3 для сетей, использующая шинную топологию и коллективный доступ с прослушиванием несущей и обнаружением конфликтов;

– Token Ring – сетевая технология по стандарту IEEE 802.5, использующая кольцевую топологию и метод доступа к кольцу с передачей маркера;

– FDDI (Fiber Distributed Date Interface Station) – сетевая технология по стандарту IEEE 802.6, использующая оптоволоконный носитель;

– X.25 – международный стандарт для глобальных коммуникаций с коммутацией пакетов;

– Frame Relay – сеть, организованная из технологий X.25 и ISDN.

3.3.8. Физический уровень

Физический уровень (Physical Layer) предназначен для сопряжения с физическими средствами соединения.

Физические средства соединения (Physical Interconnection Facility) – это совокупность физической среды, аппаратных и программных средств, обеспечивающая передачу сигналов между системами.

Физическая среда (Physical Environment) – это материальная субстанция, через которую осуществляется передача сигналов. Физическая среда является основой, на которой строятся физические средства соединения. В качестве физической среды широко используются эфир, металлы, оптическое стекло и кварц.

Физический уровень состоит из *подуровня стыковки* со средой и *подуровня преобразования передачи*. Первый из них обеспечивает сопряжение потока данных с используемым физическим каналом связи. Второй осуществляет преобразования, связанные с применяемыми протоколами.

Физический уровень обеспечивает физический интерфейс с каналом передачи данных, а также описывает процедуры передачи сигналов в канал и получения их из канала. На этом уровне определяются электрические, механические, функциональные и процедурные параметры для физической связи в системах. Физический уровень получает пакеты данных от вышележащего канального уровня и преобразует их в оптические или электрические сигналы, соответствующие 0 и 1 бинарного потока.

Эти сигналы посылаются через среду передачи на приемный узел. Механические и электрические/оптические свойства среды передачи определяются на физическом уровне и включают:

– тип кабелей и разъемов;
– разводку контактов в разъемах;
– схему кодирования сигналов для значений 0 и 1.

Физический уровень выполняет следующие функции:

– установление и разъединение физических соединений;
– передачу и прием сигналов в последовательном коде;
– прослушивание каналов в необходимых случаях;
– идентификацию каналов;
– оповещение о появлении неисправностей и отказов.

Оповещение о появлении неисправностей и отказов связано с тем, что на физическом уровне происходит обнаружение определенного класса событий, которые мешают нормальной работе сети. К ним относятся: столкновение кадров, посланных сразу несколькими системами, обрыв канала, отключение питания, потеря механического контакта и т. д. Виды сервиса, предоставляемого канальному уровню, определяются протоколами физического уровня. Прослушивание канала необходимо в тех случаях, когда к одному каналу подключается группа систем, но одновременно передавать сигналы разрешается только одной из них. Поэтому прослушивание канала позволяет определить, свободен ли он для передачи.

В ряде случаев для более четкого определения структуры физический уровень разбивается на несколько подуровней. Например, физический уровень беспроводной сети делится на три подуровня (рис. 3.15).

1в	Подуровень, не зависимый от физических средств соединений
1б	Переходной подуровень
1а	Подуровень, зависимый от физических средств соединений

Рис. 3.15. Физический уровень беспроводной локальной сети

Функции физического уровня реализуются во всех устройствах, подключенных к сети. Со стороны компьютера функции физического уровня выполняются сетевым адаптером. Повторители являются единственным типом оборудования, которое работает только на физическом уровне.

Выполняется преобразование данных, поступающих от более высокого уровня, в сигналы, передаваемые по кабелю. В глобальных сетях на этом уровне могут использоваться модемы и интерфейс ***RS-232C***. В локальных сетях для преобразования данных применяют сетевые адаптеры, обеспечивающие скоростную передачу данных в цифровой форме. Пример протокола физического уровня – это известный интерфейс ***RS-232C/CCITT V.2***, который является наиболее широко распространенной стандартной последовательной связью между компьютерами и периферийными устройствами.

Можно считать этот уровень отвечающим за аппаратное обеспечение. Физический уровень может обеспечивать как *асинхронную* (последовательную), так и *синхронную* (параллельную) передачу, которая применяется для некоторых мейнфреймов и мини-компьютеров. На физическом уровне должна быть определена схема кодирования для представления двоичных значений с целью их передачи по каналу связи. Во многих локальных сетях используется манчестерское кодирование.

Примером протокола физического уровня может служить спецификация 100Base-T технологии Ethernet, которая определяет в качестве используемого кабеля *неэкранированную витую пару* категории 5 с волновым сопротивлением 100 Ом, разъем RJ-45, максимальную длину физического сегмента 100 м, манчестерский код для представления данных на кабеле и другие характеристики среды и электрических сигналов.

К числу наиболее распространенных спецификаций физического уровня относятся:

– EIA-RS-232-C, CCITT V.24/V.28 – механические/электрические характеристики несбалансированного последовательного интерфейса;

– EIA-RS-422/449, CCITT V.10 – механические, электрические и оптические характеристики сбалансированного последовательного интерфейса;

– Ethernet – сетевая технология по стандарту IEEE 802.3 для сетей, использующая шинную топологию и коллективный доступ с прослушиванием несущей и обнаружением конфликтов;

– Token Ring – сетевая технология по стандарту IEEE 802.5, использующая кольцевую топологию и метод доступа к кольцу с передачей маркера.

Модель OSI представляет собой хотя и очень важную, но одну из многих моделей коммуникаций. Эти модели и связанные с ними стеки протоколов могут отличаться количеством уровней, их функциями, форматами сообщений, сервисами, предоставляемыми на верхних уровнях, и прочими параметрами.

Иерархически организованная совокупность протоколов, решающих задачу взаимодействия узлов сети, называется **стеком коммуникационных протоколов** (Communication Protocol Stack).

Протоколы соседних уровней, находящихся в одном узле, взаимодействуют друг с другом также в соответствии с четко определенными правилами и с помощью стандартизованных форматов сообщений. Эти правила принято называть **интерфейсом**. Интерфейс определяет набор услуг, которые нижележащий уровень предоставляет вышележащему уровню.

Выводы

1. В компьютерных сетях идеологической основой стандартизации является многоуровневый подход к разработке средств сетевого взаимодействия.

2. Формализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются протоколом.

3. Формализованные правила, определяющие взаимодействие сетевых компонентов соседних уровней одного узла, называются интерфейсом. Интерфейс определяет набор сервисов, предоставляемый данным уровнем соседнему уровню.

4. Иерархически организованный набор протоколов, достаточный для организации взаимодействия узлов в сети, называется стеком коммуникационных протоколов.

5. Открытой системой может быть названа любая система, которая построена в соответствии с общедоступными спецификациями, соответствующими стандартам и принятыми в результате публичного обсуждения всеми заинтересованными сторонами.

6. Модель OSI стандартизирует взаимодействие открытых систем. Она определяет семь уровней взаимодействия: прикладной, представительный, сеансовый, транспортный, сетевой, канальный и физический.

7. Три нижних уровня (физический, канальный и сетевой) являются сетезависимыми – протоколы этих уровней тесно связаны с технической реализацией сети, с используемым коммуникационным оборудованием.

8. Важнейшим направлением стандартизации в области вычислительных сетей является стандартизация коммуникационных протоколов. Наиболее популярными являются стеки: TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA и OSI.

Контрольные вопросы

1. Дайте определение пакета.
2. В чем заключаются преимущества использования пакетов?
3. Дайте определение времени доступа.
4. Опишите типичную стуктуру пакета.
5. Для чего предназначена преамбула в пакете?
6. Какие функции выполняет служебная информация в пакете?

7. Что такое инкапсуляция пакетов?

8. Что означает понятие «метод доступа» и как он влияет на передачу данных в сети?

9. Какие существуют методы доступа?

10. Охарактеризуйте метод доступа с прослушиванием несущей и разрешением коллизий.

11. При каком методе доступа обе станции могут одновременно начать передачу и войти в конфликт?

12. В каких сетевых технологиях используется метод CSMA/CD?

13. Дайте характеристику метода доступа с разделением во времени и перечислите, в каких случаях используется данный метод.

14. Что такое маркер?

15. В каком случае рабочая станция может начать передачу данных при использовании метода доступа с передачей полномочия?

16. Охарактеризуйте метод доступа с передачей полномочия.

17. Охарактеризуйте метод множественного доступа с разделением частоты.

18. Какие существуют варианты использования множественного доступа с разделением во времени?

19. Что такое OSI?

20. Каково назначение базовой модели взаимодействия открытых систем?

21. На какие уровни разбивается базовая модель OSI?

22. Что обеспечивает горизонтальная составляющая модели взаимодействия открытых систем?

23. Какие элементы являются основными элементами для базовой модели взаимодействия открытых систем?

24. Какие функции выполняются на физическом уровне?

25. Какие вопросы решаются на физическом уровне?

26. Какой уровень модели OSI преобразует данные в общий формат для передачи по сети?

27. Какое оборудование используется на физическом уровне?

28. Какие известны спецификации физического уровня?

29. Перечислите функции канального уровня.

30. Каковы функции канального уровня?

31. На какие подуровни разделяется канальный уровень? Опишите их функции.

32. Какие протоколы используются на канальном уровне?

33. Какое оборудование используется на канальном уровне?

34. Какие функции выполняются, какие протоколы используются на сетевом уровне?
35. Какое оборудование используется на сетевом уровне?
36. Перечислите функции транспортного уровня.
37. Какие протоколы используются на транспортном уровне?
38. Перечислите оборудование транспортного уровня.
39. Дайте определение сеансового уровня.
40. Какой уровень отвечает за доступ приложений в сеть?
41. Перечислите задачи уровня представления данных.
42. Назовите функции прикладного уровня.
43. Перечислите протоколы верхних уровней.

4

ПОНЯТИЕ ПРОТОКОЛА. СТЕК ПРОТОКОЛОВ TCP/IP

4.1. Спецификации стандартов канального и физического уровней

Спецификации института инженеров-электриков и инженеров-электронщиков (Institute of Electrical and Electronics Engineers, IEEE) IEEE 802 определяют стандарты для физических компонентов сети – ***сетевой карты*** (Network Interface Card, NIC) и ***сетевого носителя*** (Network Media). Они относятся к физическому и канальному уровням модели OSI. Спецификации IEEE 802 определяют механизм доступа адаптера к каналу связи и механизм передачи данных. Стандарты IEEE 802 подразделяют канальный уровень на подуровни:

– Logical Link Control (LLC) – *подуровень управления логической связью*;

– Media Access Control (MAC) – *подуровень управления доступом к устройствам*.

Существует более двадцати спецификаций IEEE 802.

Стандарт IEEE 802.1 (Internetworking – *межсетевое взаимодействие*) задает механизмы управления сетью на MAC-уровне. В разделе 802.1 приводятся основные понятия и определения, общие характеристики и требования к локальным сетям, а также поведение маршрутизации на канальном уровне, где логические адреса должны быть преобразованы в их физические адреса и наоборот.

Стандарт IEEE 802.2 (Logical Link Control – *управление логической связью*) определяет функционирование подуровня LLC на канальном уровне модели OSI. LLC обеспечивает интерфейс между методами доступа к среде и сетевым уровнем.

Стандарт IEEE 802.3 (Ethernet Carrier Sense Multiple Access with Collision Detection, CSMA/CD LANs Ethernet, *множественный доступ к сетям Ethernet с проверкой несущей и обнаружением конфликтов*)

описывает физический уровень и подуровень MAC для сетей, использующих шинную топологию и коллективный доступ с прослушиванием несущей и обнаружением конфликтов. Прототип этого метода – метод доступа стандарта Ethernet (10BaseT, 10Base2, 10Base5).

Метод доступа CSMA/CD также включает технологии Fast Ethernet (100BaseTX, 100BaseFX, 100BaseFX):

– 100Base-TX – *двухпарная витая пара*; использует метод MLT-3 (Multi Level Transmission 3 – один из способов линейного кодирования: физического кодирования, канального кодирования, импульсно-кодовой модуляции) для передачи сигналов 5-битовых порций кода 4B/5B по витой паре, а также имеется функция автопереговоров (Autonegotiation) для выбора режима работы порта;

– 100Base-T4 – *четырехпарная витая пара*; вместо кодирования 4B/5B в этом методе используется кодирование 8B/6T.

– 100BaseFX – *многомодовое оптоволокно*; определяет работу протокола Fast Ethernet по многомодовому оптоволокну в *полудуплексном* и *полнодуплексном* режимах на основе хорошо проверенной схемы кодирования и передачи оптических сигналов, использующейся уже на протяжении ряда лет в стандарте FDDI (Fiber Distributed Data Interface – оптоволоконный интерфейс распределенных данных). Как и в стандарте FDDI, каждый узел соединяется с сетью двумя оптическими волокнами, идущими от приемника и передатчика.

Этот метод доступа используется в сетях с общей шиной (к которым относятся и радиосети, породившие этот метод). Все компьютеры такой сети имеют непосредственный доступ к общей шине, поэтому она может быть использована для передачи данных между любыми двумя узлами сети.

Простота схемы подключения – это один из факторов, определивших успех стандарта Ethernet. Говорят, что кабель, к которому подключены все станции, работает в режиме *коллективного доступа (Multiply Access*, *MA).*

Метод доступа CSMA/CD определяет основные временные и логические соотношения, гарантирующие корректную работу всех станций в сети.

Передаваемые по сети данные помещаются в *кадры* определенной структуры и снабжаются уникальным адресом станции назначения.

Кадр передается по кабелю. Все станции, подключенные к кабелю, могут распознать факт передачи кадра, и та станция, которая узнает собственный адрес в заголовках кадра, записывает его содержимое в свой *внутренний буфер*, обрабатывает полученные данные и посылает по кабелю *кадр-ответ*. Адрес *станции-источника* включен в исходный кадр, поэтому *станция-получатель* знает, куда послать ответ.

Стандарт IEEE 802.4 (Token Bus LAN – *локальные сети Token Bus*) определяет метод доступа к шине с передачей маркера, прототип – ARCNet.

При подключении устройств в ARCNet применяют топологию «шина» или «звезда». Адаптеры ARCNet поддерживают метод доступа Token Bus (маркерная шина) и обеспечивают производительность 2,5 Мбит/с. Этот метод предусматривает следующие правила:

– все устройства, подключенные к сети, могут передавать данные, только получив разрешение на передачу (маркер);

– в любой момент времени только одна станция в сети обладает таким правом;

– кадр, передаваемый одной станцией, одновременно анализируется всеми остальными станциями сети.

В сетях *ARCNet* используется асинхронный метод передачи данных (в сетях Ethernet и Token Ring применяется синхронный метод), т. е. передача каждого байта в ARCNet выполняется посылкой ISU (Information Symbol Unit – единица передачи информации), состоящей из трех служебных стартовых или стоповых битов и восьми битов данных.

Стандарт IEEE 802.5 (Token Ring LAN – *локальные сети Token Ring*) описывает метод доступа к кольцу с передачей маркера, прототип – Token Ring.

Сети стандарта Token Ring, как и сети Ethernet, используют *разделяемую среду передачи данных*, которая состоит из отрезков кабеля, соединяющих все станции сети в кольцо. Кольцо рассматривается как общий разделяемый ресурс, и для доступа к нему используется алгоритм, основанный на передаче станциями права на использование кольца в определенном порядке. Это право передается с помощью маркера (токена).

Стандарт IEEE 802.6 (Metropolitan Area Network – *городские* или *муниципальные сети*) описывает рекомендации для региональных сетей.

Стандарт IEEE 802.7 (Broadband Technical Advisory Group – *техническая консультационная группа по широковещательной передаче*) описывает рекомендации по широкополосным сетевым технологиям, носителям, интерфейсу и оборудованию.

Стандарт IEEE 802.8 (Fiber Technical Advisory Group – *техническая консультационная группа по оптоволоконным сетям*) содержит обсуждение использования оптических кабелей в сетях со стандартом 802.3–802.6, а также рекомендации по оптоволоконным сетевым технологиям, носителям, интерфейсу и оборудованию; прототип – сеть *FDDI* (Fiber Distributed Data Interface).

Стандарт FDDI использует оптоволоконный кабель и доступ с применением *маркера*. Сеть FDDI строится на основе двух оптоволоконных колец, которые образуют основной и резервный пути передачи данных между узлами сети.

Использование двух колец – это основной способ повышения отказоустойчивости в сети FDDI, и узлы, которые хотят им воспользоваться, должны быть подключены к обоим кольцам. Скорость сети – до 100 Мбит/с. Данная технология позволяет включать до 500 узлов на расстоянии 100 км.

Стандарт IEEE 802.9 (Integrated Voice and Data Network – *интегрированные сети передачи голоса и данных*) задает архитектуру и интерфейсы устройств одновременной передачи данных и голоса по одной линии, а также содержит рекомендации по гибридным сетям, в которых объединяют голосовой трафик и трафик данных в одной и той же сетевой среде.

В ***стандарте IEEE 802.10*** (Network Security – *сетевая безопасность*) рассмотрены вопросы обмена данными, *шифрования* (на основе криптографического преобразования информации), управления сетями и безопасности в сетевых архитектурах, совместимых с моделью OSI.

Стандарт IEEE 802.11 (Wireless Network – *беспроводные сети*) описывает рекомендации по использованию беспроводных сетей.

Стандарт IEEE 802.12 описывает *рекомендации по использованию сетей 100VG*-AnyLAN со скоростью100 Мб/с и методом доступа

по очереди запросов и по приоритету (Demand Priority Queuing, DPQ; Demand Priority Access, DPA).

Технология *100VG* – это комбинация Ethernet и Token Ring со скоростью передачи 100 Мбит/с, работающая на *неэкранированных витых парах*. В проекте 100Base-VG усовершенствован метод доступа с учетом потребности мультимедийных приложений. В спецификации *100VG* предусматривается поддержка волоконно-оптических кабельных систем. Технология 100VG использует *централизованный метод доступа* (Demand Priority). В этом случае узлам сети предоставляется право равного доступа. Концентратор опрашивает каждый порт и проверяет наличие запроса на передачу, а затем разрешает этот запрос в соответствии с приоритетом. Имеется два уровня приоритетов – высокий и низкий.

Стандарт IEEE 802.14 определяет *функционирование кабельных модемов*.

Стандарт IEEE 802.15 (Personal Area Network, PAN – *персональные сети*) рассматривает вопросы организации персональных сетей. В настоящее время уже существует несколько спецификаций данного стандарта.

1. ***Стандарт IEEE 802.15.1*** базируется на спецификациях Bluetooth v1.x и предназначен для построения так называемых персональных беспроводных сетей (Wireless Personal Area Network, WPAN). Для работы радиоинтерфейса *Bluetooth* используется нижний (2,45 ГГц) диапазон ISM (industrial, scientific, medical), предназначенный для работы промышленных, научных и медицинских приборов.

2. ***Стандарт IEEE 802.15.3*** предназначен для *беспроводных частных сетей* и является прямым наследником Bluetooth (частота 2,4 ГГц). IEEE 802.15.3 обеспечивает скорость передачи данных до 55 Мбит/с на расстоянии до 100 м, одновременно работать в такой сети могут до 245 пользователей. Шифрование данных в сетях IEEE 802.15.3 может осуществляться по стандарту AES 128.

3. ***Стандарт IEEE 802.15.4*** (ZigBee) ориентирован, главным образом, на использование в качестве средства связи между автономными приборами и оборудованием.

4. ***Стандарт IEEE 802.15.4a*** (Ultra Wideband, UWB) базируется на технологии сверхширокополосной связи (Ultra Wideband, UWB) основана на передаче множества закодированных импульсов негармонической формы очень малой мощности и малой длительности в широком диапазоне частот.

Стандарт IEEE 802.16 предназначен для реализации широкополосных каналов в городских сетях (MAN). В отличии от 802.11 он ориентирован для соединения стационарных, а не мобильных объектов. Его задачей является обеспечение сетевого уровня между локальными (IEEE 802.11) и региональными (WAN) сетями, где планируется применение разрабатываемого стандарта IEEE 802.20. Эти стандарты совместно со стандартом IEEE 802.15 и 802.17 образуют взаимосогласованную иерархию протоколов беспроводной связи.

Стандарт IEEE 802.17 называется RPR (Resilient Packet Ring – *адаптивное кольцо для пакетов*), и в отличие от FDDI (а также Token Ring или DQDB) пакеты удаляются из кольца узлом-адресатом, что позволяет осуществлять несколько обменов одновременно.

Стандарт IEEE 802.18 представляет собой требования и рекомендации технической консультативной группы по радиочастотному регулированию – RTAG (*Radio Regulatory Technical Advisory Group*).

Стандарт IEEE 802.19 представляет собой требования и рекомендации технической консультативной группы по сосуществованию – CTAG (*Coexistence Technical Advisory Group*).

Стандарт IEEE 802.20 описывает правила беспроводного мобильного широкополосного доступа MBWA (*Mobile Broadband Wireless Access*) для пакетного интерфейса в беспроводных городских сетях – WMAN. Этот стандарт должен поддерживать услуги по передаче данных с IP в качестве транспортного протокола и дополнять стандарт IEEE 802.16 в масштабе WiMAX. Стандарт обеспечит скорость передачи данных более 1 Мбит/с и позволит получить мобильный доступ к данным из движущихся транспортных средств (если скорость их не превышает 250 км/ч). Для беспроводного интерфейса HPI (Highspeed Portable Internet) устанавливаются уровни скорости передачи и безопасности. Быстродействие HPI выше, чем универсальной системы мобильной связи UMTS, которая ориентирована на передачу голоса. Стандарт обеспечивает подключение ПК в небольших и домашних офисах (SOHO) как альтернативу сетей «последней мили» по медным или оптическим кабелям, использующим технологии DSL.

Стандарт IEEE 802.21 – стандарт независимой от среды эстафетной передачи соединений – MIHS (*Media Independent Handover Services*).

Стандарт IEEE 802.22 определяет функционирование беспроводных региональных сетей WRAN (*Wireless Regional Area Network*), использующих для передачи данных телевизионные частотные диапазоны.

Стандарт IEEE 802.23 определяет независимую от среды структуру в рамках IEEE 802 для обеспечения согласованного доступа к данным. Сюда входит интерфейс уровня канала передачи данных для согласованного просмотра сетей IEEE 802 с помощью возможностей служб экстренной помощи на основе протокола IP.

Стандарт IEEE 802.24 – технологии IEEE 802 применяются для поддержки вертикальных приложений. В данном контексте стандарт IEEE 802.24 определяет, что делают горизонтальные технологии в поддержке приложений. Примерами потенциальных категорий вертикальных приложений могут выступать умные сети, интеллектуальные транспортные системы (ITS), умные дома, умные города, электронное здравоохранение и т. д.

Стандарт IEEE 802.25 (пока не ратифицирован) – затрагивает вопросы организации Omni-Range Area Network.

4.2. Протоколы и стеки протоколов

Правила взаимодействия двух машин могут быть описаны в виде набора процедур для каждого из уровней, которые называются **протоколами**.

Согласованный набор протоколов разных уровней, достаточный для организации межсетевого взаимодействия, называется **стеком протоколов**.

Для каждого уровня определяется набор функций-запросов для взаимодействия с вышележащим уровнем, который называется **интерфейсом**.

Существует достаточно много стеков протоколов, которые широко применяются в сетях. Это и стеки, являющиеся международными и национальными стандартами, и фирменные стеки, которые стали общеизвестными благодаря распространенности оборудования той или иной фирмы. Примерами популярных стеков протоколов могут служить стек IPX/SPX фирмы Novell, стек TCP/IP, используемый в сети Internet и во многих сетях на основе операционной системы UNIX, стек OSI международной организации по стандартизации, стек DECnet корпорации Digital Equipment и некоторые другие.

В общем случае можно выделить три укрупненных уровня протоколов, характерных для любых стеков:
– сетевые;
– транспортные;
– прикладные.

4.2.1. Протоколы сетевого уровня

Сетевые протоколы предоставляют следующие услуги: адресацию и маршрутизацию информации, проверку на наличие ошибок, запрос повторной передачи и установление правил взаимодействия в конкретной сетевой среде. Ниже приведены наиболее популярные сетевые протоколы:

– DDP (Datagram Delivery Protocol – протокол доставки дейтаграмм). *Протокол передачи данных Apple*, используемый в Apple Talk;

– IP (Internet Protocol – протокол Internet). *Протокол стека TCP/IP*, обеспечивающий адресную информацию и информацию о маршрутизации;

– IPX (Internetwork Packet eXchange – межсетевой обмен пакетами) в NWLink. *Протокол Novel NetWare*, используемый для маршрутизации и направления пакетов;

– NetBEUI (NetBIOS Extended User Interface – расширенный пользовательский интерфейс базовой сетевой системы ввода-вывода). Разработан совместно IBM и Microsoft, обеспечивает транспортные услуги для NetBIOS.

4.2.2. Протоколы транспортного уровня

Транспортные протоколы предоставляют услуги надежной транспортировки данных между компьютерами. Ниже приведены наиболее популярные транспортные протоколы:

– ATP (Apple Talk Protocol – транзакционный протокол Apple Talk) и NBP (Name Binding Protocol – *протокол связывания имен*). Сеансовый и транспортный протоколы Apple Talk;

– NetBIOS (Network Basis Input/Output System – *базовая сетевая система ввода-вывода*). NetBIOS устанавливает соединение между

компьютерами, а NetBEUI предоставляет услуги передачи данных для этого соединения;

– SPX (Sequenced Packet eXchange – последовательный обмен пакетами) в NWLink. Протокол Novel NetWare, используемый для обеспечения доставки данных;

– TCP (Transmission Control Protocol – протокол управления передачей). Протокол стека TCP/IP отвечает за надежную доставку данных.

4.2.3. Протоколы прикладного уровня

Прикладные протоколы отвечают за взаимодействие приложений. Ниже приведены наиболее популярные прикладные протоколы:

– AFP (Apple Talk File Protocol, файловый протокол Apple Talk) – *протокол удаленного управления файлами Macintosh*;

– FTP (File Transfer Protocol, протокол передачи файлов) – *протокол стека TCP/IP*, используемый для обеспечения услуг по передаче файлов;

– NCP (NetWare Core Protocol – *базовый протокол NetWare*) – оболочка и редиректоры клиента Novel NetWare;

– SNMP (Simple Network Management Protocol, *простой протокол управления сетью* – это протокол стека TCP/IP, используемый для управления и наблюдения за сетевыми устройствами;

– HTTP (Hyper Text Transfer Protocol, протокол *передачи гипертекста*) и другие протоколы.

4.3. Стек OSI

Следует различать стек протоколов OSI и модель OSI (рис. 4.1).

Стек OSI – это набор вполне конкретных спецификаций протоколов, образующих согласованный стек протоколов.

Этот стек протоколов поддерживает правительство США в своей программе GOSIP. Стек OSI, в отличие от других стандартных стеков, полностью соответствует модели взаимодействия OSI и включает спецификации для всех семи уровней модели взаимодействия открытых систем.

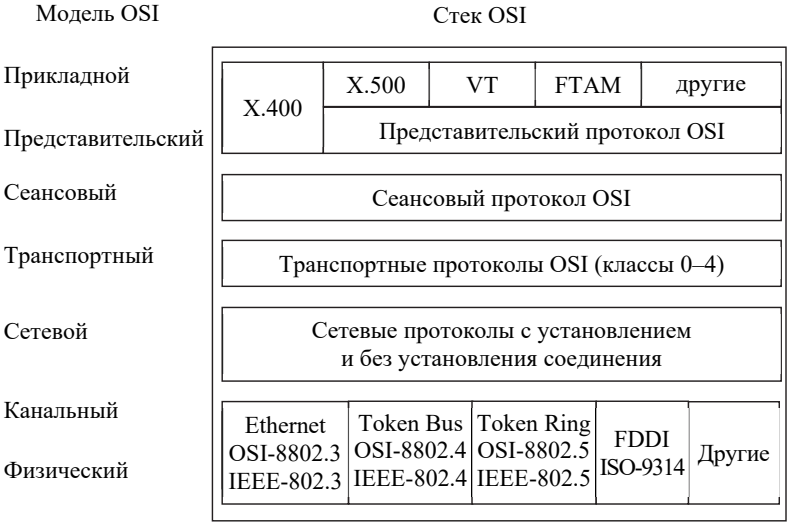


Рис. 4.1. Стек OSI

На физическом и канальном уровнях стек OSI поддерживает спецификации Ethernet, Token Ring, FDDI, протоколы LLC, X.25 и ISDN.

На сетевом уровне реализованы протоколы как без установления, так и с установлением соединений. Транспортный протокол стека OSI скрывает различия между сетевыми сервисами с установлением соединения и без установления соединения, так что пользователи получают нужное качество обслуживания независимо от нижележащего сетевого уровня. Для обеспечения этого транспортный уровень требует, чтобы пользователь задал нужное качество обслуживания. Определено 5 классов транспортного сервиса: от низшего класса (0) до высшего класса (4), которые отличаются степенью устойчивости к *ошибкам* и требованиями к восстановлению данных после ошибок.

Сервисы прикладного уровня включают передачу файлов, эмуляцию терминала, службу каталогов и почту. Из них наиболее перспективными являются служба каталогов (стандарт X.500), электронная почта (X.400), протокол виртуального терминала (Virtual Terminal, VT), протокол передачи, доступа и управления файлами (File Transfer Access and Management, FTAM), протокол пересылки и управления работами (JTM). В последнее время ISO сконцентрировала свои усилия именно на сервисах верхнего уровня.

4.4. Архитектура стека протоколов TCP/IP

Стек TCP/IP, представляющий собой набор многоуровневых протоколов, предназначен для использования в различных вариантах сетевого окружения.

> *Стек TCP/IP* с точки зрения системной архитектуры соответствует эталонной модели OSI (Open Systems Interconnection – взаимодействие открытых систем) и позволяет обмениваться данными по сети приложениям и службам, работающим практически на любой платформе, включая Unix, Windows, Macintosh и др.

Стандартная реализация TCP/IP (например, фирмы Microsoft) соответствует *четырехуровневой модели* вместо *семиуровневой*, как показано на рис. 4.2.

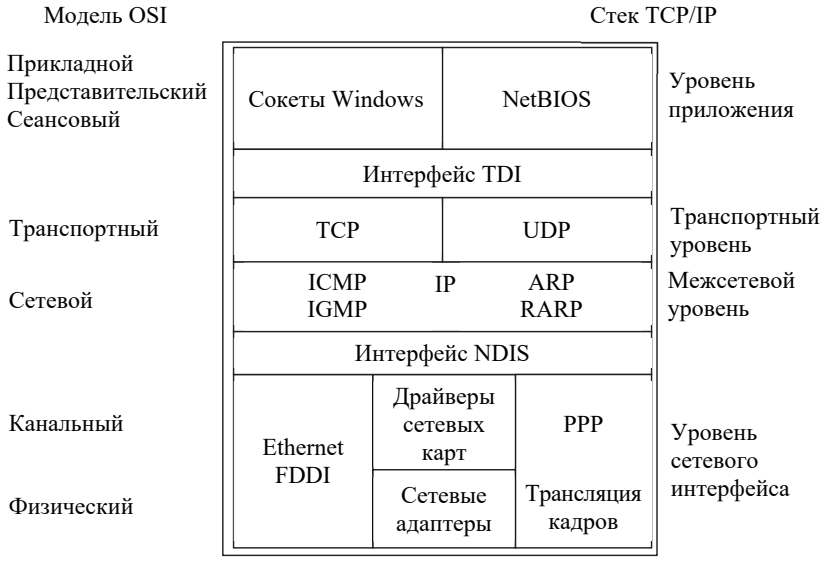


Рис. 4.2. Соответствие семиуровневой модели OSI и четырехуровневой модели TCP/IP

Модель TCP/IP включает большее число функций на один уровень, что приводит к уменьшению числа уровней. В модели используются следующие уровни:

– уровень *приложения* модели TCP/IP соответствует *прикладному*, *представительскому* и *сеансовому* уровням модели OSI;

– *транспортный* уровень модели TCP/IP соответствует аналогичному уровню модели OSI;

– *межсетевой* уровень модели TCP/IP выполняет те же функции, что и *сетевой* уровень модели OSI;

– уровень *сетевого интерфейса* модели TCP/IP соответствует *канальному* и *физическому* уровням модели OSI.

4.4.1. Уровень приложения

Через уровень *приложения* модели TCP/IP приложения и службы получают доступ к сети. Доступ к протоколам TCP/IP осуществляется посредством двух программных интерфейсов API: сокеты Windows и NetBIOS.

Интерфейс сокетов Windows, или как его называют *WinSock*, является сетевым программным интерфейсом, предназначенным для облегчения взаимодействия между различными TCP/IP – приложениями и семействами протоколов.

Интерфейс NetBIOS используется для связи между процессами (IPC – Interposes Communications) служб и приложений ОС Windows. NetBIOS выполняет три основные функции:

– определение имен NetBIOS;

– служба дейтаграмм NetBIOS;

– служба сеанса NetBIOS.

В табл. 4.1 приведено семейство протоколов TCP/IP.

Таблица 4.1

Назначение протоколов TCP/IP

Название протокола	Описание протокола
WinSock	Сетевой программный интерфейс
NetBIOS	Связь с приложениями ОС Windows
TDI	Интерфейс транспортного драйвера (Transport Driver Interface); позволяет создавать компоненты сеансового уровня

Окончание табл. 4.1

Название протокола	Описание протокола
TCP	Протокол управления передачей (Transmission Control Protocol)
UDP	Протокол пользовательских дейтаграмм (User Datagram Protocol)
ARP	Протокол разрешения адресов (Address Resolution Protocol)
RARP	Протокол обратного разрешения адресов (Reverse Address Resolution Protocol)
IP	Протокол Internet (Internet Protocol)
ICMP	Протокол управляющих сообщений Internet (Internet Control Message Protocol)
IGMP	Протокол управления группами Интернета (Internet Group Management Protocol)
NDIS	Интерфейс взаимодействия между драйверами транспортных протоколов
FTP	Протокол пересылки файлов (File Transfer Protocol)
TFTP	Простой протокол пересылки файлов (Trivial File Transfer Protocol)

4.4.2. Транспортный уровень

Транспортный уровень TCP/IP отвечает за установление и поддержание соединения между двумя узлами. Основные функции уровня:

– подтверждение получения информации;

– управление потоком данных;

– упорядочение и ретрансляция пакетов.

В зависимости от типа службы используются два протокола:

– TCP (Transmission Control Protocol – *протокол управления передачей*);

– UDP (User Datagram Protocol – *пользовательский протокол дейтаграмм*).

TCP используют в тех случаях, когда приложению требуется передать большой объем информации и убедиться, что данные своевременно получены адресатом. Приложения и службы, отправляющие небольшие объемы данных и не нуждающиеся в получении подтверждения, используют протокол ***UDP***, который является протоколом без установления соединения.

Протокол управления передачей (TCP) отвечает за надежную передачу данных от одного узла сети к другому. Он создает сеанс с установлением соединения, иначе говоря, виртуальный канал между машинами. Установление соединения происходит в три шага.

1. Клиент, запрашивающий соединение, отправляет серверу пакет, указывающий номер порта, который клиент желает использовать, а также код (определенное число) ISN (Initial Sequence Number).

2. Сервер отвечает пакетом, содержащим ISN сервера, а также ISN клиента, увеличенный на 1.

3. Клиент должен подтвердить установление соединения, вернув ISN сервера, увеличенный на 1.

Отметим, что для адресации на транспортном уровне используются так называемые порты. На практике это фактически число от 1 до 65 535, определяющее номер процесса, связанного с обработкой пакета. Соответсвтенно, на одном хосте номера портов не должны повторяться, иначе будет не понятно, к какому именно процессу должен быть отправлен для выполнения последующих действий полученный пакет.

Итак, трехступенчатое открытие соединения устанавливает номер порта, а также ISN клиента и сервера. Каждый отправляемый TCP-пакет содержит номера TCP-портов отправителя и получателя, номер фрагмента для сообщений, разбитых на меньшие части, и контрольную сумму, позволяющую убедиться, что при передаче не произошло ошибок.

В отличие от TCP *пользовательский протокол дейтаграмм UDP* не устанавливает соединения. Протокол UDP предназначен для отправки небольших объемов данных без установки соединения и используется приложениями, которые не нуждаются в подтверждении адресатом их получения. UDP также использует номера портов для определения конкретного процесса по указанному IP-адресу. Однако UDP-порты отличаются от TCP-портов и, следовательно, могут использовать те же номера портов, что и TCP, без конфликта между службами.

4.4.3. Межсетевой уровень

Межсетевой уровень отвечает за маршрутизацию данных внутри сети и между различными сетями. На нем работают маршрутизаторы,

которые зависят от используемого протокола и применяются для отправки пакетов из одной сети (или ее сегмента) в другую (или другой сегмент). В стеке TCP/IP на этом уровне используется протокол IP.

> ***Протокол Интернета IP*** обеспечивает обмен дейтаграммами между узлами сети и является протоколом, не устанавливающим соединения и использующим дейтаграммы для отправки данных из одной сети в другую.

Данный протокол не ожидает получение подтверждения (ASK, Acknowledgment) отправленных пакетов от узла адресата. Подтверждения, а также повторные отправки пакетов осуществляются протоколами и процессами, работающими на верхних уровнях модели.

К функциям протокола относится фрагментация дейтаграмм и межсетевая адресация. Протокол IP предоставляет управляющую информацию для сборки фрагментированных дейтаграмм. Главной функцией протокола является межсетевая и глобальная адресация. В зависимости от размера сети, по которой будет маршрутизироваться дейтаграмма или пакет, применяется одна из трех схем адресации.

> *Протокол IP* действует на сетевом уровне модели OSI, поэтому ***IP-адреса называются сетевыми***.

Они предназначены для передачи сообщений в составных сетях, связывающих подсети, построенные на различных локальных или глобальных сетевых технологиях, например Ethernet или ATM. Но для непосредственной передачи сообщения в рамках одной подсети вместо IP-адреса нужно использовать локальный (аппаратный) адрес технологии канального уровня, чаще всего MAC-адрес. При этом к IP-пакету добавляются заголовок (в нем указываются MAC-адреса источника и приемника кадра) и концевик кадра канального уровня (рис. 4.3).

> При формировании кадра канального уровня возникает проблема: каким образом по известному IP-адресу определить соответствующий MAC-адрес. Указанная проблема решается при помощи протокола ARP (Address Resolution Protocol – протокол разрешения адресов). Это обстоятельство имеет важное значение для ***защиты персональных данных***.

Протокол Internet (IP) версии 6 (IPv6) – это следующая версия протокола IP, которая является значительным шагом вперед по сравнению с IP версии 4 (IPv4).

Протокол сопоставления адреса ARP определяет MAC-адреса следующим образом. Осуществляется рассылка всем узлам сети специального кадра, который называется **ARP-запрос** (***ARP Request***).

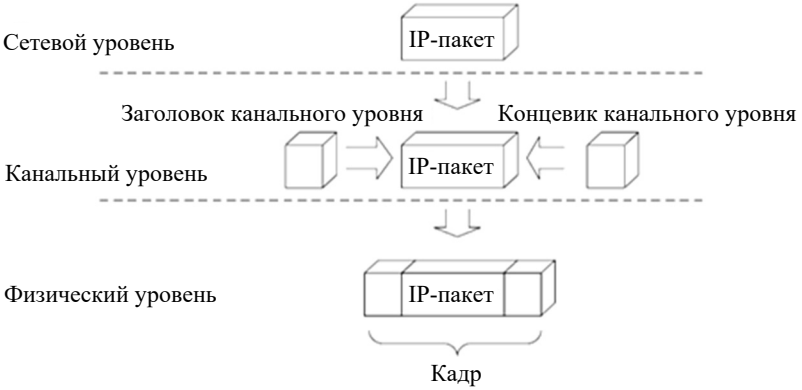


Рис. 4.3. Формирование кадра
на канальном уровне

В кадре содержится IP-адрес компьютера, у которого требуется узнать MAC-адрес. Каждый узел сети принимает ARP-запрос и сравнивает IP-адрес из запроса со своим IP-адресом. Если адреса совпадают, узел высылает *ARP-ответ* (***ARP Reply***), содержащий требуемый MAC-адрес.

Результаты своей работы протокол ARP сохраняет в специальной таблице, хранящейся в *оперативной памяти*, которая называется **ARP-кэш**.

При необходимости разрешения IP-адреса, протокол ARP сначала ищет IP-адрес в ARP-кэше и только в случае отсутствия нужной записи производит рассылку ARP-запроса.

Записи в ARP-кэше могут быть двух типов: статические и динамические. Статические записи заносятся в кэш администратором при помощи утилиты arp с ключом /s. Динамические записи помещаются в кэш после полученного ARP-ответа и по истечении двух минут удаляются. ARP-кэш имеет структуру, представленную в табл. 4.2.

Таблица 4.2

Внешний вид таблицы ARP-кэш

IP-адрес	MAC-адрес	Тип записи
192.168.1.1	03-E8-48-A1-57-7B	Статический
192.168.1.2	03-E8-48-A1-43-88	Динамический
192.168.1.3	03-E8-48-A1-F8-D9	Динамический

Процесс получения по известному IP-адресу MAC-адреса называется **разрешением IP-адреса**.

Удаление происходит для того, чтобы при перемещении в другую подсеть компьютера с MAC-адресом, занесенным в таблицу, кадры не отправлялись бесполезно в сеть.

Иногда требуется по известному MAC-адресу найти IP-адрес (например, в начале работы компьютеров без жесткого диска, у которых есть MAC-адрес сетевого адаптера и им нужно определить свой IP-адрес). В этом случае используется реверсивный протокол RARP (Reverse ARP). Это обстоятельство имеет важное значение для ***защиты персональных данных***.

Протокол управления сообщениями Интернета (Internet Control Message Protocol, ICMP) используется IP и другими протоколами высокого уровня для отправки и получения отчетов о состоянии переданной информации. Этот протокол используется для контроля скорости передачи информации между двумя системами. Если маршрутизатор, соединяющий две системы, перегружен трафиком, он может отправить специальное сообщение ICMP – ошибку для уменьшения скорости отправления сообщений.

Узлы локальной сети используют *протокол управления группами Интернета* (Internet Group Management Protocol, IGMP), чтобы зарегистрировать себя в группе. Информация о группах содержится на маршрутизаторах локальной сети. Маршрутизаторы используют эту информацию для передачи групповых сообщений.

Групповое сообщение, как и широковещательное, используется для отправки данных сразу нескольким узлам.

Также можно упомянуть протоколы обмена маршрутной информацией (RIP, OSPF), которые будут более подробно рассмотрены в

подглаве 5.5. Отметим, что данные протоколы не являются протоколами исключительно стека TCP/IP, также существуют их реализации под другие стеки протоколов.

NDIS (Network Device Interface Specification) – спецификация интерфейса сетевого устройства, программный интерфейс, обеспечивающий взаимодействие между драйверами транспортных протоколов и соответствующими драйверами сетевых интерфейсов. Позволяет использовать несколько протоколов, даже если установлена только одна сетевая карта.

4.4.4. Уровень сетевого интерфейса

Этот уровень модели TCP/IP отвечает за распределение IP-дейтаграмм. Он работает с ARP для определения информации, которая должна быть помещена в заголовок каждого кадра. Затем на этом уровне создается кадр, подходящий для используемого типа сети, такого как Ethernet, Token Ring или ATM, после чего IP-дейтаграмма помещается в область данных этого кадра, и он отправляется в сеть.

4.4.5. Недостатки модели TCP/IP

В условиях быстрого развития сетевых технологий в модели TCP/IP стали проявляться существенные недостатки, которые сделали модель не эффективной для использования в современных сетях. Все недостатки при передаче простых данных не были существенными, но при развитии сетей и увеличении объема передаваемых данных они дали о себе знать.

Основные недостатки используемой модели:

1. В модели нет четкого разграничения концепций служб, интерфейса и протокола. При разработке программного обеспечения желательно провести четкое разделение между спецификацией и реализацией, чего не делает TCP/IP. В результате модель TCP/IP не эффективна при разработке сетей, которые используют новые технологии.

2. Модель TCP/IP не является универсальной и довольно плохо описывает любой стек протоколов, кроме TCP/IP. Так, например, описать технологию Bluetooth с помощью модели TCP/IP очень сложно.

3. Хост-сетевой уровень в действительности не является тем уровнем, который обычно используется в контексте уровневых протоколов. Это, скорее, интерфейс между сетью и уровнями передачи данных.

4. В модели TCP/IP не различаются физический уровень и уровень передачи данных. Между тем они абсолютно разные. Физический уровень должен иметь дело с характеристиками передачи информации по медному кабелю, оптическому волокну и по радио, тогда задачей уровня передачи данных является определение начала и конца кадров и передача их с одной стороны на другую с требуемой степенью надежности. Корректная модель должна содержать их как два различных уровня. В модели TCP/IP этого нет.

5. Хотя протоколы IP и TCP были тщательно продуманы и неплохо реализованы, многие другие протоколы были созданы несколькими студентами, работавшими над ними, пока это занятие им не наскучило. Реализации этих протоколов свободно распространялись, в результате чего они получили широкое признание, глубоко укоренились, и теперь их трудно заменить на что-либо другое. Некоторые из них в настоящее время оказались серьезным препятствием на пути трансформаций.

Выводы

1. Спецификации IEEE 802 определяют стандарты для физических компонентов сети: сетевая карта и сетевой носитель, относящиеся к физическому и канальному уровням модели OSI; механизм доступа адаптера к каналу связи и механизм передачи данных. Стандарты IEEE802 подразделяют канальный уровень на подуровни управления логической связью и подуровень управления доступом к устройствам.

2. Согласованный набор протоколов разных уровней, достаточный для организации межсетевого взаимодействия, называется стеком протоколов. Каждому уровню соответствует набор функций-запросов для взаимодействия с вышележащим уровнем, который называется интерфейсом. Правила взаимодействия двух машин могут быть описаны в виде набора процедур для каждого из уровней, которые называются протоколами.

3. Наибольшее распространение для построения составных сетей в последнее время получил стек TCP/IP. Он имеет 4 уровня: прикладной, основной, уровень межсетевого взаимодействия и уровень

сетевых интерфейсов. Соответствие уровней стека TCP/IP уровням модели OSI достаточно условно.

4. Прикладной уровень объединяет все службы, предоставляемые системой пользовательским приложениям: традиционные сетевые службы типа Telnet, FTP, TFTP, DNS, SNMP, протокол передачи гипертекстовой информации HTTP и т. д.

5. На основном уровне стека TCP/IP, называемом также транспортным, функционируют протоколы TCP и UDP. Протокол управления передачей TCP решает задачу обеспечения надежной информационной связи между двумя конечными узлами. Дейтаграммный протокол UDP используется как экономичное средство связи уровня межсетевого взаимодействия с прикладным уровнем.

6. Уровень межсетевого взаимодействия реализует концепцию коммутации пакетов в режиме без установления соединений. Основными протоколами этого уровня являются дейтаграммный протокол IP и протоколы маршрутизации (RIP, OSPF, BGP и др.). Вспомогательную роль выполняет протокол межсетевых управляющих сообщений ICMP, протокол группового управления IGMP и протокол разрешения адресов ARP.

7. Протоколы уровня сетевых интерфейсов обеспечивают интеграцию в составную сеть других сетей. Этот уровень не регламентируется, но поддерживает все популярные стандарты физического и канального уровней: для локальных сетей – Ethernet, Token Ring, FDDI и т. д., для глобальных сетей – X.25, Frame Relay, PPP, ISDN и др.

8. В стеке TCP/IP для именования единиц передаваемых данных на разных уровнях используют разные названия: поток, сегмент, дейтаграмма, пакет, кадр.

9. В условиях быстрого развития сетевых технологий в модели TCP/IP стали проявляться существенные недостатки, которые сделали модель не эффективной для использования в современных сетях.

Контрольные вопросы

1. Назначение спецификации стандартов IEEE 802.

2. Какой стандарт описывает сетевую технологию Ethernet?

3. Каким стандартом устанавливаются задачи управления логической связью?

4. Какой стандарт задает механизмы управления сетью?

5. Какой из стандартов описывает сетевую технологию ARCNet?

6. Каким стандартом описывается сетевая технология Token Ring?

7. В каком стандарте содержатся рекомендации по оптоволоконным сетевым технологиям?

8. Что такое интерфейс уровня базовой модели OSI?

9. Что собой представляет протокол уровня базовой модели OSI?

10. Дайте определение стека протоколов.

11. На какие уровни разбиваются стеки протоколов?

12. Назовите наиболее популярные сетевые протоколы.

13. Перечислите самые популярные транспортные протоколы.

14. Какие наиболее популярные прикладные протоколы известны?

15. Перечислите наиболее популярные стеки протоколов.

16. Назначение программных интерфейсов сокетов Windows и NetBIOS.

17. Чем отличается протокол TCP от UDP?

18. Функции протокола IP.

19. Какие существуют виды адресации в IP-сетях?

20. Какой протокол используется для определения локального адреса по IP-адресу?

21. Какой протокол используется для определения IP-адреса по локальному адресу?

22. Какой протокол используется для управления сообщениями Интернета?

23. Назначение уровня сетевого интерфейса стека TCP/IP.

5

АДРЕСАЦИЯ И МАРШРУТИЗАЦИЯ В TCP/IP-СЕТЯХ

Каждый компьютер в сетях TCP/IP имеет адреса трех уровней: *физический* (MAC-адрес), *сетевой* (IP-адрес) и *символьный* (DNS-имя).

5.1. Физический адрес

Физический, или ***локальный***, ***адрес узла*** определяется технологией, с помощью которой построена сеть, в которую входит узел. Для узлов, входящих в локальные сети, это MAC-адрес сетевого адаптера или порта маршрутизатора.

В качестве стандартного выбран 48-битный формат адреса, что соответствует примерно 280 трлн различных адресов. Понятно, что столько сетевых адаптеров никогда не будет выпущено.

С тем чтобы распределить возможные диапазоны адресов между многочисленными изготовителями сетевых адаптеров, была предложена следующая структура адреса (рис. 5.1).

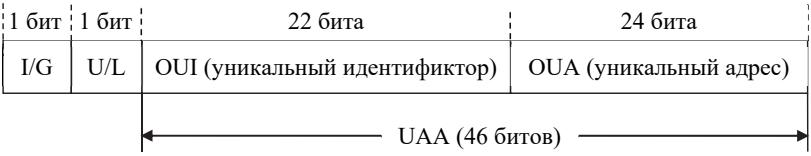


Рис. 5.1. Структура 48-битного стандартного MAC-адреса

Младшие 24 разряда двоичного кода адреса называются OUA (Organizationally Unique Address) – ***уникальный адрес***. Именно их присваивает каждый из зарегистрированных производителей сетевых

адаптеров. Всего возможно свыше 16 млн комбинаций, т. е. каждый изготовитель может выпустить 16 млн сетевых адаптеров.

Следующие 22 разряда кода называются OUI (Organizationally Unique Identifier) – ***уникальный идентификатор***. IEEE присваивает один или несколько OUI каждому производителю сетевых адаптеров, что позволяет исключить совпадения адресов адаптеров от разных производителей. Всего возможно свыше 4 млн разных OUI, это означает, что теоретически может быть зарегистрировано 4 млн производителей. Вместе OUA и OUI называются UAA (Universally Administered Address) – *универсально управляемый адрес* или *IEEE-адрес*.

Два старших разряда адреса управляющие, они определяют тип адреса, способ интерпретации остальных 46 разрядов. Старший бит I/G (Individual/Group) указывает на тип адреса. Если он установлен в 0, то является индивидуальным, если в 1, – то групповым (многопунктовым или функциональным). Пакеты с групповым адресом получат все имеющие этот групповой адрес сетевые адаптеры. Причем групповой адрес определяется 46-ю младшими разрядами. Второй управляющий бит U/L (Universal/Local) называется флажком универсального/местного управления и определяет, как был присвоен адрес данному сетевому адаптеру. Обычно он установлен в 0. Установка бита U/L в 1 означает, что адрес задан не производителем сетевого адаптера, а организацией, использующей данную сеть. Это случается довольно редко.

Для широковещательной передачи (т. е. передачи всем абонентам сети одновременно) применяется специально выделенный ***сетевой адрес***, все 48 битов которого установлены в 1. Его принимают все абоненты сети независимо от их индивидуальных и групповых адресов.

Данной системы адресов придерживаются такие популярные сети, как Ethernet, Fast Ethernet, Token Ring, FDDI, 100VG-AnyLAN. Ее недостатки – высокая сложность аппаратуры сетевых адаптеров, а также большая доля служебной информации в передаваемом пакете (адреса источника и приемника вместе требуют уже 96 битов пакета или 12 байтов).

Во многих сетевых адаптерах предусмотрен так называемый циркулярный режим. В этом режиме адаптер принимает все пакеты, приходящие к нему, независимо от значения поля адреса приемника. Такой режим используется, например, для проведения диагностики сети, измерения ее производительности, контроля ошибок передачи.

При этом один компьютер принимает и контролирует все пакеты, проходящие по сети, но сам ничего не передает. В данном режиме работают сетевые адаптеры мостов и коммутаторы, которые должны обрабатывать перед ретрансляцией все пакеты, приходящие к ним.

5.2. Сетевой адрес

5.2.1. Представление IP-адреса

Адрес IP представляет собой 32-разрядное двоичное число, разделенное на группы по 8 битов, называемые **октетами**. Например:

00010001 11101111 00101111 01011110.

Обычно IP-адреса записываются в виде четырех десятичных октетов и разделяются точками. Таким образом, приведенный выше IP-адрес можно записать в следующей форме: 17.239.47.94.

Следует заметить, что максимальное значение октета равно 11111111_2 (двоичная система счисления), что соответствует в десятичной системе 255_{10}. Поэтому IP-адреса, в которых хотя бы один октет превышает это число, являются недействительными. Пример: 172.16.123.1 – действительный адрес, а 172.16.123.256 – несуществующий адрес, поскольку 256 выходит за пределы допустимого диапазона: от 0 до 255.

IP-адрес состоит из двух логических частей – ***номера подсети*** (ID подсети) и ***номера узла*** (ID хоста) в этой подсети. При передаче пакета из одной подсети в другую используется ID подсети. Когда пакет попал в подсеть назначения, ID хоста указывает на конкретный узел в рамках этой подсети.

Чтобы записать ID подсети, в поле номера узла в IP-адресе ставят нули. Чтобы записать ID хоста, в поле номера подсети ставят нули. Например, если в IP-адресе 172.16.123.1 первые два байта отводятся под номер подсети, остальные два байта – под номер узла, то номера записываются следующим образом: ID подсети 172.16.0.0; ID хоста 0.0.123.1.

По числу разрядов, отводимых для представления номера узла (или номера подсети), можно определить общее количество узлов (или подсетей) по простому правилу: если число разрядов для

представления номера узла равно N, то общее количество узлов равно $2^N - 2$. Два узла вычитаются вследствие того, что адреса со всеми разрядами, равными нулям или единицам, являются особыми и используются в специальных целях.

Например, если под номер узла в некоторой подсети отводится два байта (16 битов), то общее количество узлов в такой подсети равно $2^{16} - 2 = 65\,534$ узла.

Для определения того, какая часть IP-адреса отвечает за ID подсети, а какая за ID хоста, применяются два способа: с помощью классов и с помощью масок.

Общее правило: под ID подсети отводятся *первые* несколько битов IP-адреса, оставшиеся биты обозначают ID хоста.

Рассмотрим конфигурирование IP-адресации (v4) в операционных системах типа *Windows.*

Пример 5.1. Рассмотрим настройку протокола TCP/IPv4.

1. Запустите папку *Сетевые подключения*. Для этого в операционных системах типа Windows Seven нажмите кнопку *Пуск*, введите в строке поиска начальные буквы слова «Центр». Из списка выберите пункт *Центр управления сетями и общим доступом* (рис. 5.2).

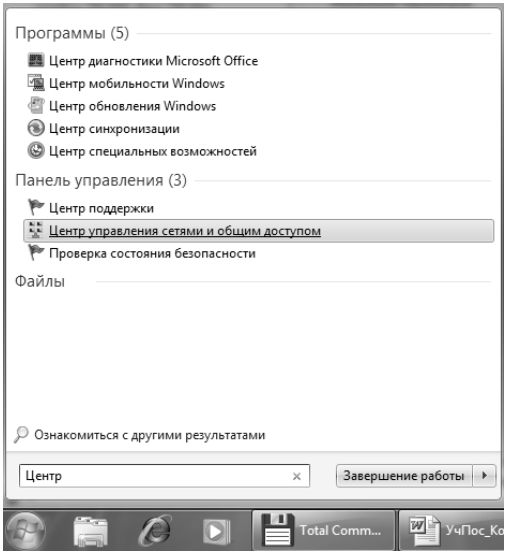


Рис. 5.2. Пример вызова центра управления сетями и общим доступом

2. В окне центра управления сетями и общим доступом щелкните по пункту *Изменение параметров адаптера* (рис. 5.3). Далее откроется окно с сетевыми подключениями (рис. 5.4).

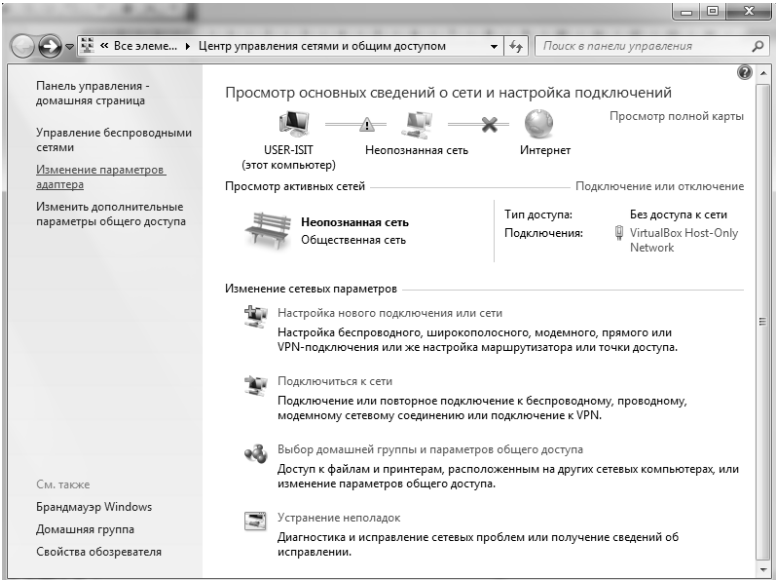


Рис. 5.3. Общий вид центра управления сетями и общим доступом

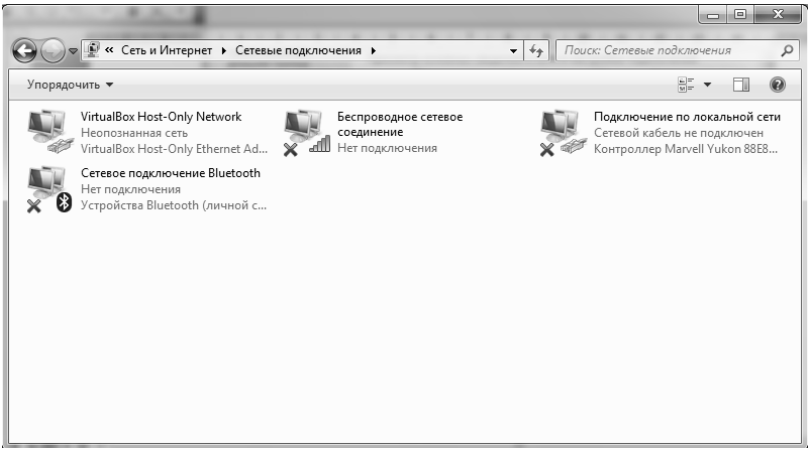


Рис. 5.4. Общий вид папки «Сетевые подключения»

3. Щелкните правой кнопкой мыши по подключению, которое требуется настроить, а затем выберите команду *Свойства*. Если появится диалоговое окно *Управление учетной записью пользователя*, убедитесь, что действие, указанное в окне, совпадает с тем, которое вы хотите выполнить, и нажмите *Продолжить*.

4. Далее выполните одно из указанных ниже действий:

– в случае подключения по локальной сети на вкладке *Общие* в списке *Компоненты*, используемые этим подключением, выберите пункт *Протокол Интернета версии 4* (*TCP/IPv4*) и нажмите кнопку *Свойства*;

– в случае подключения удаленного доступа, VPN-подключения или высокоскоростного подключения на вкладке *Сеть* в списке *Компоненты*, используемые этим подключением, выберите пункт *Протокол Интернета версии 4* (*TCP/IPv4*) и нажмите кнопку *Свойства*. В результате откроется окно с настройками протокола TCP/IP (рис. 5.5).

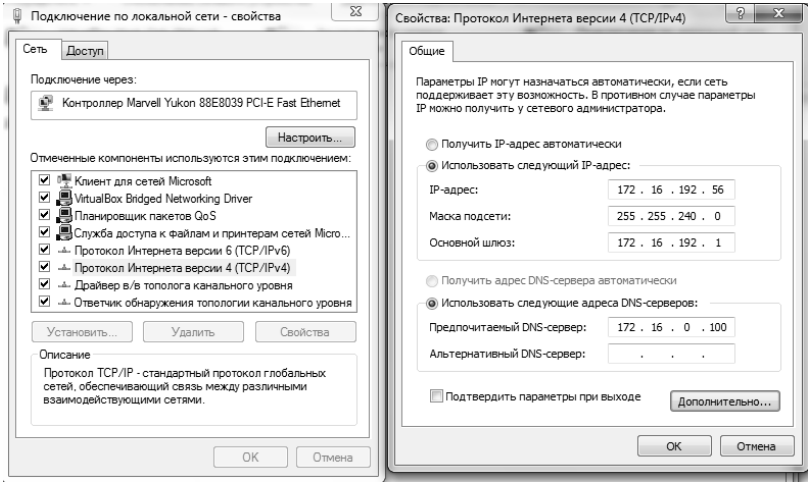


Рис. 5.5. Свойства протокола Интернета версии 4 (TCP/IPv4)

5. Выполните далее одно из указанных ниже действий:

– если необходимо, чтобы параметры IP-адреса назначались автоматически, выберите пункт *Получить IP-адрес автоматически* и нажмите кнопку *ОК*;

– если необходимо указать IP-адрес IPv4 или адрес DNS-сервера, выполните следующие действия:

а) выберите пункт *Использовать следующий IP-адрес* и в открывшемся окне введите IP-адрес, соответствующую маску подсети и адрес шлюза по умолчанию (в примере на рис. 5.5 IP-адрес: 172.16.192.56; маска подсети: 255.255.240.0; основной шлюз: 172.16.192.1);

б) выберите пункт *Использовать следующие адреса DNS-серверов* и в полях *Предпочитаемый DNS-сервер* и *Альтернативный DNS-сервер* введите адреса основного и дополнительного DNS-серверов (в примере на рис. 5.5 IP-адрес предпочитаемого DNS сервера: 172.16.0.100;

в) для настройки параметров DNS, WINS и IP необходимо использовать вкладку *Дополнительно* (рис. 5.6).

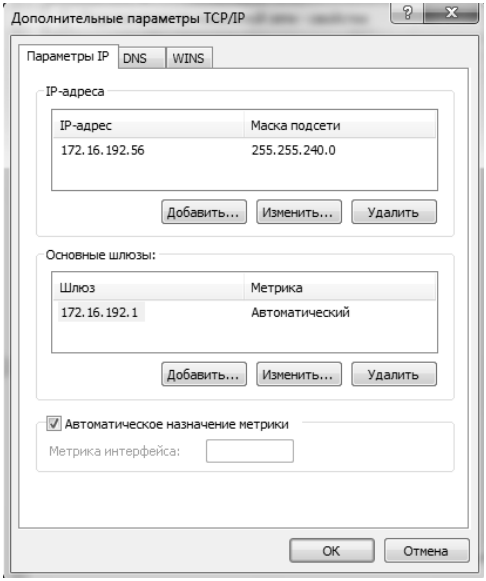


Рис. 5.6. Окно с дополнительными параметрами TCP/IP

6. В подключении по локальной сети при выборе параметра *Получить IP-адрес автоматически* включается вкладка *Альтернативная конфигурация*. Если компьютер используется более чем в одной сети, воспользуйтесь этой вкладкой для ввода альтернативных параметров IP-адреса. Для настройки параметров DNS, WINS и IP откройте вкладку *Настраиваемый пользователем* или *Альтернативная конфигурация.*

Дополнительные рекомендации. Если это возможно, используйте автоматическую настройку параметров протокола IP (DHCP), поскольку при этом устраняется необходимость настройки таких параметров, как IP-адрес, адрес DNS-сервера и адрес WINS-сервера.

Параметры *Альтернативная конфигурация* определяют второй набор параметров протокола IP, который используется при недоступности DHCP-сервера. Это весьма полезно для пользователей портативных компьютеров, которые часто перемещаются между двумя различными сетевыми средами (например, между средой со службой DHCP и средой со статическими IP-адресами).

Аналогично осуществляется конфигурирование TCP/IPv6 (используются свойства *Протокол Интернета версии 6* (*TCP/IPv6*)).

5.2.2. Классы IP-адресов

Существует пять классов IP-адресов: *A*, *B*, *C*, *D* и *E* (рис. 5.7).

Рис. 5.7. Классы IP-адресов

За принадлежность к тому или иному классу отвечают первые биты IP-адреса. Деление сетей на классы описано в RFC 791 (документ описания протокола IP). Целью такого деления являлось создание малого числа больших сетей (***класс A***), умеренного числа средних сетей (***класс B***) и большого числа малых сетей (***класс C***).

Если адрес начинается с 0, то сеть относится к *классу A* и номер сети занимает 1 байт, остальные 3 байта интерпретируются как номер узла в сети. Сети класса A имеют номера в диапазоне от 1 до 126. Сетей класса *A* немного, зато количество узлов в них может достигать $2^{24} - 2$, т. е. 16 777 214 узлов.

Если первые два бита адреса равны 10, то сеть относится к *классу B.* В сетях класса B под номер сети и под номер узла отводится по 16 битов, т. е. по 2 байта. Таким образом, сеть класса *B* является сетью средних размеров с максимальным числом узлов $2^{16} - 2$, что составляет 65 534 узлов.

Если адрес начинается с последовательности 110, то это сеть *класса C.* В данном случае под номер сети отводится 24 бита, а под номер узла – 1 байт. Сети этого класса наиболее распространены, число узлов в них ограничено $2^8 - 2$, т. е. 254 узлами.

Адрес, начинающийся с 1110, обозначает особый, ***групповой адрес*** (Multicast). Пакет с таким адресом направляется всем узлам, которым присвоен данный адрес.

Адреса ***класса E*** в настоящее время не используются (зарезервированы для будущих применений).

Характеристики адресов разных классов представлены в табл. 5.1.

Таблица 5.1

Характеристики IP-адресов разных классов

Класс	Первые биты	Наименьший номер сети	Наибольший номер сети	Количество сетей	Максимальное число узлов в сети
A	0	1.0.0.0	126.0.0.0	126	$2^{24} - 2 = 16\,777\,214$
B	10	128.0.0.0	191.255.0.0	16384	$2^{16} - 2 = 65\,534$
C	110	192.0.0.0	223.255.255.0	2097152	$2^8 - 2 = 254$
D	1110	224.0.0.0	239.255.255.255	Групповой адрес	
E	11110	240.0.0.0	247.255.255.255	Зарезервирован	

Применение классов удовлетворительно решало задачу деления на подсети в начале развития Интернета. В 1990-е годы с увеличением числа подсетей стал ощущаться дефицит IP-адресов. Это связано

с неэффективностью распределения при классовой схеме адресации. Например, если организации требуется тысяча IP-адресов, ей выделяется сеть класса B, при этом 64 534 адреса не будут использоваться.

Существует два основных способа решения этой проблемы:

– более эффективная схема деления на подсети с использованием масок (RFC 950);

– применение протокола IP версии 6 (IPv6).

Поэтому в настоящее время использовать принцип классов для определения идентификаторов сети и хоста можно и нужно только в случае отсутствия маски подсети.

5.2.3. Использование масок

Маска подсети (Subnet Mask) – это число, которое используется в паре с IP-адресом; двоичная запись маски содержит единицы в тех разрядах, которые должны в IP-адресе интерпретироваться как номер сети.

Для стандартных классов сетей маски имеют следующие значения:

– класс A – 11111111.00000000.00000000.00000000 (255.0.0.0);

– класс B – 11111111.11111111.00000000.00000000 (255.255.0.0);

– класс C – 11111111.11111111.11111111.00000000 (255.255.255.0).

Маска подсети записывается либо в виде, аналогичном записи IP-адреса, например 255.255.255.0, либо совместно с IP-адресом с помощью указания числа единичных разрядов в записи маски, например 192.168.1.1/24, т. е. в маске содержится 24 единицы (255.255.255.0).

При использовании масок можно вообще отказаться от понятия классов.

Пример 5.2. Пусть задан IP-адрес 17.239.47.94, маска подсети 255.255.0.0 (другая форма записи: 17.239.47.94/16).

Требуется определить ID подсети и ID хоста в обеих схемах адресации.

1. Адресация с использованием классов. Двоичная запись IP-адреса имеет вид:

00010001.11101111.00101111.01011110.

Так как первый бит равен нулю, адрес относится к классу A. Следовательно, первый байт отвечает за ID подсети, остальные три байта – за ID хоста.

ID подсети: 17.0.0.0. ID хоста: 0.239.47.94.

2. Адресация с использованием масок. Запишем IP-адрес и маску подсети в двоичном виде:

IP-address: 17.239.47.94 = 00010001.11101111.00101111.01011110;

Subnet mask: 255.255.0.0 = 11111111.11111111.00000000.00000000.

Вспомнив определение маски подсети, можно интерпретировать номер подсети как те биты, которые в маске равны 1, т. е. первые два байта. Оставшаяся часть IP-адреса будет номером узла в данной подсети.

ID подсети: 17.239.0.0. ID хоста: 0.0.47.94.

Номер подсети можно получить другим способом, применив к IP-адресу и маске операцию логического умножения или *конъюнкции* (AND):

AND 00010001.11101111.00101111.01011110

11111111.11111111.00000000.00000000

00010001.11101111.00000000.00000000

17 239 0 0

В масках количество единиц в последовательности, определяющей границу номера сети, не обязательно должно быть кратным 8.

Пример 5.3. Задан IP-адрес: 192.168.89.16, маска подсети – 255.255.192.0 (другая форма записи: 192.168.89.16/18).

Требуется определить ID подсети и ID хоста. Воспользуемся операцией AND:

IP-address: 192.168.89.16 = 11000000.10101000.01011001.00010000

Subnet mask: 255.255.0.0 = 11111111.11111111.11000000.00000000

Subnet ID: 11000000.10101000.01000000.00000000

192 168 64 0

Чтобы получить номер узла, нужно в битах, отвечающих за номер подсети, поставить нули:

Host ID: 00000000.00000000.00011001.00010000 = 0.0.25.16.

Ответ: ID подсети = 192.168.64.0, ID хоста = 0.0.25.16.

Для масок существует важное правило: разрывы в последовательности единиц или нулей недопустимы.

Например, не существует маски подсети, имеющей следующий вид:

11111111.11110111.00000000.00001000 (255.247.0.8),

так как последовательности единиц и нулей не являются непрерывными.

Также необходимо добавить, что маска, состоящая только из 0 (0.0.0.0), хоть и допустима (обозначает, что адрес локален), но не имеет смысла в контексте выделения из IP-адреса номера подсети и узла. Поэтому возможно лишь 32 варианта правильных масок – от 1 единицы до 32, причем маска 255.255.255.255 (32 единицы) формально может существовать и будет указывать единичный узел сети.

> **!** С помощью *масок* администратор может структурировать свою сеть, не требуя от поставщика услуг дополнительных номеров сетей.

Пример 5.4. Допустим, организации выделена сеть класса B: 160.95.0.0 (рис. 5.8).

Рис. 5.8. Сеть класса B до деления на подсети

В такой сети может находиться до 65 534 узлов. Однако организации требуется 3 независимые сети с числом узлов в каждой не более 254. В этой ситуации можно применить деление на подсети с помощью масок. Например, при использовании маски 255.255.255.0 третий байт адреса будет определять номер внутренней подсети, а четвертый байт – номер узла (рис. 5.9).

Маршрутизаторы во внешней сети (Интернет) ничего «не знают» о делении сети 160.95.0.0 на подсети, все пакеты направляются на маршрутизатор организации, который переправляет их в требуемую внутреннюю подсеть.

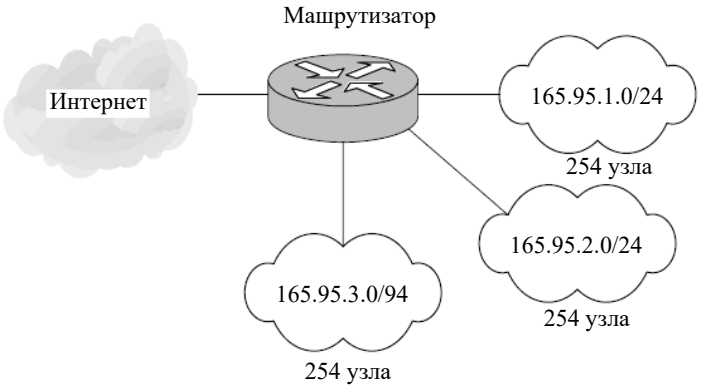


Рис. 5.9. Сеть класса B после деления на подсети

5.2.4. Особые IP-адреса

Некоторые IP-адреса являются особыми, они не должны применяться для идентификации обычных сетей.

1. Если первый октет ID сети начинается со 127, такой адрес считается адресом машины – источника пакета. В этом случае пакет не выходит в сеть, а возвращается на компьютер-отправитель.

Такие адреса называются **loopback** (*петля*, *замыкание на себя*) и используются для проверки функционирования стека TCP/IP.

2. Если все биты IP-адреса равны нулю, адрес обозначает узел-отправитель и используется в некоторых сообщениях ICMP.

3. Если все биты ID сети равны 0, а все биты ID хоста равны 1, то адрес называется **ограниченным широковещательным** (**Limited Broadcast**).

Пакеты, направленные по такому адресу, рассылаются всем узлам той подсети, в которой находится отправитель пакета. К данному же типу адреса можно относить и IP-адрес, в котором 32 разряда заполнены 1, так как пакет с таким адресом не будет «выпущен» за пределы сети отправителя устройством, связующим сети, например коммутатром или маршрутизатором.

4. Если все биты ID хоста равны 1, а биты ID сети не равны 0, т. е. однозначно идентифицируют адрес подсети, то адрес называется **широковещательным** (Broadcast); пакеты, имеющие широковещательный адрес, доставляются всем узлам подсети назначения.

5. Если все биты ID хоста равны 0, адрес считается **идентификатором подсети** (Subnet ID).

Наличие особых IP-адресов объясняет, почему из диапазона доступных адресов исключаются два адреса – это случаи, когда все биты ID хоста равны 1 или 0. Например, в сети класса C не 256, а 254 узла.

5.2.5. Распределение IPv4-адресов. Частные и публичные адреса

Поскольку каждый узел сети Интернет должен обладать уникальным IP-адресом, то, безусловно, важной является задача координации распределения адресов отдельным сетям и узлам. Такую координирующую роль выполняет интернет-корпорация по распределению имен и адресов (the Internet Corporation for Assigned Names and Numbers, ICANN).

Естественно, что ICANN не решает задач выделения IP-адресов конечным пользователям и организациям, а занимается распределением диапазонов адресов между крупными организациями – поставщиками услуг по доступу к сети Интернет (Internet Service Provider), которые, в свою очередь, могут взаимодействовать не только с более мелкими поставщиками, но и с конечными пользователями. Так, например, функции по распределению IP-адресов в Европе корпорация ICANN делегировала Координационному центру RIPE (RIPE NCC – фр. Reseaux IP Europeens + англ. Network Coordination Centre). В свою очередь, этот центр делегирует часть своих функций региональным организациям.

Служба распределения номеров IANA (Internet Assigned Numbers Authority) зарезервировала для частных сетей следующие три блока адресов:

10.0.0.0–10.255.255.255 (1 сеть класса A);

172.16.0.0–172.31.255.255 (16 сетей класса B);

192.168.0.0–192.168.255.255 (256 сетей класса C).

Любая организация может использовать IP-адреса из этих блоков без согласования с ICANA или Internet-регистраторами. В результате эти адреса используются во множестве организаций. Таким образом, уникальность адресов сохраняется только в масштабе одной или нескольких организаций, которые согласованно используют

общий блок адресов. В такой сети каждая рабочая станция может обмениваться информацией с любой другой рабочей станцией частной сети.

> **!** Если какой-либо организации требуются уникальные адреса для связи с внешними сетями, то их следует получать обычным путем через *регистраторов Internet*. Такие адреса никогда не будут входить ни в один из указанных выше блоков частных адресов.

Перед распределением адресов из частного и публичного блоков следует определить, какие из рабочих станций сети должны иметь связь с внешними системами на сетевом уровне. Для таких рабочих станций следует использовать публичные адреса, остальным же – можно присваивать адреса из частных блоков, это не помешает им взаимодействовать со всеми рабочими станциями частной сети организации, независимо от того, какие адреса используются (частные или публичные). Однако прямой доступ во внешние сети для рабочих станций с адресами из частного блока невозможен. Для организации их доступа во внешние шлюзы придется использовать прокси-серверы.

Перемещение рабочей станции из *частной сети* в публичную (и обратно) связано со сменой IP-адреса, соответствующих записей DNS и изменением конфигурационных файлов на других рабочих станцях, которые их идентифицируют по IP-адресам. Поскольку *частные адреса* не имеют глобального значения, маршрутная информация о частных сетях не должна выходить за пределы этих сетей, а пакеты с частными адресами отправителей или получателей не должны передаваться через межсетевые каналы. Предполагается, что маршрутизаторы в публичных сетях (особенно маршрутизаторы провайдеров Internet) будут отбрасывать маршрутную информацию из частных сетей. Если маршрутизатор публичной сети получает такую информацию, ее отбрасывание не должно трактоваться как ошибка протокола маршрутизации.

Также в качестве частной можно выделить еще одну сеть класса B – 169.254.0.0, адреса которой используются для автоматической конфигурации сетевых адаптеров операционной системой Windows при отсутствии либо не работающем DHCP-сервере.

5.2.6. Общие сведения о протоколе IPv6

Использование масок является временным решением проблемы дефицита IP-адресов, так как адресное пространство протокола IP не увеличивается, а количество хостов в Интернете растет с каждым днем. Для принципиального решения проблемы требуется существенное увеличение количества IP-адресов. Для преодоления ограничений IPv4 был разработан ***протокол IP 6-й версии*** – ***IPv6*** (RFC 2373, 2460).

Протокол IPv6 имеет следующие основные особенности:

– длина адреса 128 битов – такая длина обеспечивает примерно $3{,}4\times10^{38}$ адресов; данное количество адресов позволит присваивать в обозримом будущем уникальные IP-адреса любым устройствам;

– автоматическая конфигурация – протокол IPv6 предоставляет средства автоматической настройки IP-адреса и других сетевых параметров даже при отсутствии таких служб, как DHCP;

– встроенная безопасность – для передачи данных является обязательным использование *протокола защищенной передачи*, IPsec (протокол IPv4 также может использовать IPsec).

В настоящее время многие производители сетевого оборудования включают поддержку *протокола IPv6* в свои продукты, однако преобладающим остается протокол IPv4. Связано это с тем, что IPv6 обратно несовместим с IPv4 и процесс перехода сопряжен с определенными трудностями.

5.2.7. Архитектура адресации IPv6

Существует три типа адресов:

– *unicast*: идентификатор одиночного интерфейса. Пакет, посланный по unicast-адресу, доставляется интерфейсу, указанному в адресе. Под интерфейсом в контексте IPv6 следует понимать это средство подключения узла к каналу.

– *anycast*: идентификатор набора интерфейсов (принадлежащих разным узлам). Пакет, посланный по anycast-адресу, доставляется одному из интерфейсов, указанному в адресе (ближайшему, в соответствии с мерой, определенной протоколом маршрутизации).

– *multicast*: идентификатор набора интерфейсов (обычно принадлежащих разным узлам). Пакет, посланный по multicast-адресу, доставляется всем интерфейсам, заданным этим адресом.

В отличии от IPv4-протокола, в IPv6 не существует широковещательных адресов – их функции переданы multicast-адресам. Также надо отметить, что в IPv6 все нули и все единицы являются допустимыми кодами для любых полей, если не оговорено исключение.

5.2.8. Модель адресации

Выделяют следующие аспекты модели адресации протокола IPv6:

1) IPv6-адреса всех типов ассоциируются с интерфейсами, а не узлами. Так как каждый интерфейс принадлежит только одному узлу, юникастный адрес интерфейса может идентифицировать узел;

2) юникастный адрес IPv6 соотносится только с одним интерфейсом. Одному интерфейсу могут соответствовать много IPv6-адресов различного типа (*юникастные*, *эникастные* и *мультикастные*). Существует два исключения из этого правила:

– одиночный адрес может приписываться нескольким физическим интерфейсам, если приложение рассматривает их как единое целое.

– маршрутизаторы могут иметь ненумерованные интерфейсы (например, интерфейсу не присваивается никакого IPv6-адреса) для соединений «точка-точка», чтобы исключить необходимость вручную конфигурировать и объявлять эти адреса. Адреса не нужны для соединений точка-точка маршрутизаторов, если эти интерфейсы не используются в качестве точки отправления или назначения при посылке IPv6-дейтограмм.

3) IPv6 соответствует модели IPv4, где подсеть ассоциируется с каналом. Одному каналу могут соответствовать несколько подсетей.

5.2.9. Представление записи IPv6-адресов

Существует три стандартные формы для представления IPv6-адресов в виде текстовых строк:

1. *Основная форма записи* имеет вид x:x:x:x:x:x:x:x, где 'x' – шестнадцатеричные шестнадцатибитные числа.

fedc:ba98:7654:3210:FEDC:BA98:7654:3210
1080:0:0:0:8:800:200C:417A

Необходимо отметить, что не нужно писать начальные нули в каждом из конкретных полей, но в каждом поле должна быть хоть одна цифра (за исключением случая, описанного в пункте 2).

2. *Сокращенная форма записи.* Из-за метода записи некоторых типов IPv6-адресов последние часто содержат длинные последовательности нулевых битов. Для того чтобы сделать запись адресов, содержащих нулевые биты, более удобной, имеется специальный синтаксис для удаления лишних нулей. Использование записи «::» указывает на наличие групп из 16 нулевых битов. Данная комбинация «::» может появляться только при записи адреса.

1080:0:0:0:8:800:200c:417a	anycast-адрес
ff01:0:0:0:0:0:0:43	multicast-адрес

Эти адреса могут быть представлены в следующем виде:

1080::8:800:200c:417a	anycast-адрес
ff01::43	multicast-адрес

Последовательность «::» может также использоваться для удаления из записи начальных или завершающих последовательностей нулей в адресе. Расмотрим пример основной и сокращенной формы записи IPv6-адреса обратной связи и неспецифицированного IPv6-адреса.

Основная форма записи

0:0:0:0:0:0:0:1	адрес обратной связи
0:0:0:0:0:0:0:0	неспецифицированный адрес

Сокращенная форма записи

::1	адрес обратной связи
::	не специфицированный адрес

3. *Альтернативная форма записи*, которая более удобна при работе с IPv4 и IPv6, выглядит так: x:x:x:x:x:x:d.d.d.d, где 'x' – шестнадцатеричные шестнадцатибитные числа адреса, а 'd' – десятичные восьмибитные чила, составляющие младшую часть адреса (стандартное IPv4-представление). Примерами такой записи могут быть следующие IPv6-адреса:

0:0:0:0:0:0:13.1.68.3
0:0:0:0:0:FFFF:129.144.52.38

или в сокращенном виде:

::13.1.68.3
::FFFF:129.144.52.38

5.2.10. Представление типа IPv6-адреса

Специфический тип IPv6-адресов идентифицируется ***лидирующими битами адреса***. Поле переменной длины, которое содержит эти лидирующие биты, называется *префиксом формата* (Format Prefix, FP).

Некоторые примеры исходных назначений данных префиксов представлены в табл. 5.2.

Таблица 5.2

Префиксы IPv6-адресов

Назначение	Префикс (двоичный)	Часть адресного пространства
Зарезервировано для NSAP	0000001	1/128
Зарезервировано для IPX	0000010	1/128
Провайдерские unicast-адреса	010	1/8
Зарезервировано для географических unicast-адресов	100	1/8
Локальные канальные адреса	1111111010	1/1024
Локальная подсеть	1111111011	1/1024
Multicast-адреса	11111111	1/256

Данное распределение адресов поддерживает прямое выделение адресов провайдера, адресов локального применения и multicast-адресов. Зарезервировано место для адресов NSAP, IPX, географических адресов и т. д.

Unicast-адреса отличаются от multicast-адресов значением старшего октета: значение FF (11111111) идентифицирует multicast-адрес, а любые другие значения – адрес типа unicast. Anycast-адреса берутся из пространства адресов unicast и синтаксически неотличимы от них. Обычно говорят, что как только один и тот же unicast-адрес присвоен двум и более интерфейсам, то он становится *anycast-адресом*.

5.2.11. Unicast IPv6-адреса

Unicast-адрес служит для определения интерфейса устройства под управлением протокола IPv6. Пакет, который отправляется на unicast-адрес, будет получен интерфейсом, присвоенным для этого адреса. Как и в случае с протоколом IPv4, IPv6-адрес должен быть индивидуальным.

Существует шесть типов unicast-адресов:

1. *Global unicast-адрес.* Адреса данного типа уникальны по всему миру и доступны для маршрутизации через Интернет IPv6. Они эквивалентны публичным IPv4-адресам. В настоящее время назначаются только глобальные индивидуальные адреса с первыми тремя битами 001 или 2000::/3. Это лишь 1/8 часть от всего доступного адресного пространства IPv6. Так, например, адрес 2001:0DB8::/32 был зарезервирован для документации. Глобальные индивидуальные адреса могут быть настроены статически или присвоены динамически.

В общем случае глобальный индивидуальный адрес состоит из трех частей (рис. 5.10).

Рис. 5.10. Структура global unicast IPv6-адреса

Префикс глобальной маршрутизации – это префиксальная или сетевая часть адреса, назначаемая интернет-провайдером заказчику или узлу. В настоящее время /48 является префиксом глобальной маршрутизации, который интернет-регистраторы назначают своим заказчикам – корпоративным сетям и индивидуальным пользователям. Этого адресного пространства более чем достаточно для большинства заказчиков.

Идентификатор подсети используется организациями для обозначения подсетей в каждом узле.

Идентификатор интерфейса эквивалентен узловой части IPv4-адреса. Данный термин используется в том случае, когда один узел может иметь несколько интерфейсов, каждый из которых обладает одним или более IPv6-адресами.

2. *Link Local-адрес.* Local IPv6-адрес канала (Link Local-адрес) позволяет устройству обмениваться данными с другими устройствами под управлением IPv6 по одному и тому же каналу и только по данному каналу (подсети). Пакеты с локальным адресом канала источника или назначения не могут быть направлены за пределы того канала, в котором пакет создается. В отличие от локальных IPv4-адресов канала локальные адреса канала IPv6 играют важную роль в различных аспектах сети. Глобальный индивидуальный адрес не обязателен. Однако для содержания локального адреса канала необходим сетевой интерфейс под управлением протокола IPv6. Если локальный адрес канала не настроен вручную на интерфейсе, устройство автоматически создает собственный адрес, не обращаясь к DHCP-серверу. Узлы под управлением IPv6-протокола создают локальный IPv6-адрес канала даже тогда, когда устройству не был назначен глобальный IPv6-адрес. Это позволяет устройствам под управлением IPv6 обмениваться данными с другими устройствами под управлением IPv6 в одной подсети, в том числе со шлюзом по умолчанию (маршрутизатором). Локальные IPv6-адреса канала находятся в диапазоне FE80::/10.

3. *Loopback-адрес.* Данный адрес используется узлом для отправки пакета самому себе и не может быть назначен физическому интерфейсу. Как и на loopback-адрес IPv4, для проверки настроек TCP/IP на локальном узле можно послать эхо-запрос на loopback-адрес IPv6. Loopback-адрес IPv6 состоит из нулей, за исключением последнего бита, и выглядит как ::1/128 или просто ::1 в сокращенном формате.

4. *Unspecified-адрес* (неопределенный адрес). Он состоит из нулей и в сокращенном формате представлен как ::/128 или просто :: Данный адрес не может быть назначен интерфейсу и используется только в качестве адреса источника в IPv6-пакете. Неопределенный адрес применяется в качестве адреса источника в том случае, когда устройству еще не назначен постоянный IPv6-адрес или когда источник пакета не относится к месту назначения.

5. *Unique local-адрес.* Это IPv6-адреса, которые имеют некоторые общие особенности с частными адресами RFC 1918 для IPv4, но при этом между ними имеются и значительные различия. Уникальные локальные адреса используются для локальной адресации в пределах узла или между ограниченным количеством узлов. Эти адреса не следует маршрутизировать в глобальном протоколе IPv6.

Уникальные локальные адреса находятся в диапазоне от FC00::/7 до FDFF::/7. В случае с IPv4 частные адреса объединены с NAT для обеспечения преобразования адресов из частных в публичные. Это делается из-за недостатка адресного пространства IPv4. На многих сайтах также используют частный характер адресов RFC 1918, чтобы обеспечить безопасность или защитить сеть от потенциальных угроз. Однако такая мера никогда не была целью использования данных технологий, и организация IETF всегда рекомендовала предпринимать правильные меры предосторожности при работе маршрутизатора в Интернете. Хотя протокол IPv6 обеспечивает особую адресацию для сайтов, он не предназначен для того, чтобы скрывать внутренние устройства под управлением IPv6 от анализа из сети Интернет IPv6. IETF рекомендует ограничивать доступ к устройствам с помощью специальных (более эффективных) мер безопасности.

6. *IPv4 embedded-адрес.* Это тип индивидуальных адресов, являющийся адресом со встроенным IPv4-адресом. Использование этих адресов способствует переходу с протокола IPv4 на IPv6.

5.2.12. Multicast IPv6-адреса

Мультикаст-адрес IPv6 (multicast-адрес) является идентификатором для группы узлов. Узел может принадлежать к любому числу мультикаст-групп. Мультикаст-адреса имеют формат, отраженный на рис. 5.11:

Рис. 5.11. Структура multicast-адреса

Префикс 11111111 в начале адреса идентифицирует адрес как multicast-адрес. Структура флагов представлена на рис. 5.12.

Рис. 5.12. Структура флагов multicast-адреса

Старшие 3 флага зарезервированы и должны быть обнулены. Если четвертый флаг T установлен в «0», то это означает, что IPv6-адрес является стандартным (*well-known*) multicast-адресом, официально выделенным для глобального использования в Интернете, а T = 1 указывает, что данный multicast-адрес присвоен временно (*transient*).

Поле *Scope* представляет собой 4-битный код, предназначенный для определения предельной области действия multicast-группы. Допустимые значения поля *Scope* представлены в табл. 5.3.

Таблица 5.3

Допустимые значения поля Scope

Значение поля	Область действия
0	Зарезервировано
1	Область действия ограничена локальным узлом
2	Область действия ограничена локальным каналом
3	Не определено
4	Не определено
5	Область действия ограничена локальной сетью
6	Не определено
7	Не определено
8	Область действия ограничена локальной организацией
9	Не определено
A	Не определено
B	Не определено
C	Не определено
D	Не определено
E	Глобальные пределы (global scope)
F	Зарезервировано

Значение постоянно присвоенного multicast-адреса не зависит от значения поля *scope*. Например, если *NTP servers group* присвоен постоянный multicast-адрес с идентификатором группы 43 (hex), тогда:

– FF01:0:0:0:0:0:0:43 означает, что все NTP-серверы одного и того же узла рассматриваются как отправители;

– FF02:0:0:0:0:0:0:43 означает, что все NTP-серверы работают с тем же каналом, что и отправитель;

– FF05:0:0:0:0:0:0:43 означает, что все NTP-серверы принадлежат той же сети, что и отправитель;

– FF0E:0:0:0:0:0:0:43 означает, что все NTP-серверы находятся в Интернете.

Непостоянно выделенные multicast-адреса имеют значение только в пределах данного ограничения (scope). Например, группа, определенная непостоянным локальным multicast-адресом FF15:0:0:0:0:0:0:43, не имеет никакого смысла для другой локальной сети или непостоянной группы, использующей тот же групповой идентификатор с другим *scope*, или для постоянной группы с тем же групповым ID.

Multicast-адреса не должны использоваться в качестве адреса отправителя в IPv6-пакетах или встречаться в любых заголовках маршрутизации.

5.2.13. Автоматизация назначения IP-адресов узлам сети – протокол DHCP

Как уже было сказано, IP-адреса могут назначаться администратором сети вручную. Это представляет для администратора утомительную процедуру. Ситуация усложняется еще тем, что многие пользователи не обладают достаточными знаниями для того, чтобы конфигурировать свои компьютеры для работы в интерсети, и должны полагаться на администраторов.

Протокол динамической настройки узла, DHCP (Dynamic Host Configuration Protocol), был разработан для того, чтобы освободить администратора от этих проблем. Основным назначением DHCP является динамическое назначение IP-адресов. Однако, кроме динамического, DHCP может поддерживать и более простые способы ручного и автоматического статического назначения адресов.

В ручной процедуре назначения адресов активное участие принимает администратор, предоставляющий DHCP-серверу информацию

о соответствии IP-адресов физическим адресам или другим идентификаторам клиентов. Эти адреса сообщаются клиентам в ответ на их запросы к DHCP-серверу.

При автоматическом статическом способе DHCP-сервер присваивает IP-адрес (и, возможно, другие параметры конфигурации клиента) из пула наличных IP-адресов без вмешательства оператора. Границы пула назначаемых адресов задает администратор при конфигурировании DHCP-сервера.

Между идентификатором клиента и его IP-адресом, как и при ручном назначении, существует постоянное соответствие. Оно устанавливается в момент первичного назначения сервером DHCP IP-адреса клиенту. При последующих запросах сервер возвращает тот же IP-адрес. При динамическом распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время, что позволяет повторно использовать IP-адреса другими компьютерами. Служба DHCP обеспечивает надежный и простой способ конфигурации сети TCP/IP, гарантируя отсутствие конфликтов адресов за счет централизованного управления их распределением. Администратор управляет процессом назначения адресов с помощью параметра «продолжительности аренды», определяющего, как долго компьютер может использовать назначенный IP-адрес, перед тем как снова запросить его от сервера DHCP в аренду.

Примером работы протокола DHCP может служить ситуация, когда компьютер, являющийся клиентом DHCP, удаляется из подсети. При этом назначенный ему IP-адрес автоматически освобождается. Когда компьютер подключается к другой подсети, то ему автоматически назначается новый адрес. Ни пользователь, ни сетевой администратор не вмешиваются в этот процесс. Это свойство очень важно для мобильных пользователей.

Пример 5.5. Рассмотрим настройку DHCP-сервера на примере ОС Windows Server 2012.

1. ***Установка и авторизация сервера DHCP.*** Установка службы DHCP выполняется так же, как и установка любой другой компоненты Windows Server: *Пуск – Панель управления – Установка и удаление программ – Установка компонентов Windows – Сетевые службы* – кнопка *Состав* – выбрать пункт *DHCP* – кнопки *OK*, *Далее* и *Готово* (если потребуется, то указать путь к дистрибутиву системы).

Также можно установить DHCP-сервер, используя *Server Manager* (*Диспетчер серверов*), а именно *Start* (*Пуск*) – *Server Manager* (*Диспетчер серверов*), общий вид которого показан на рис 5.13.

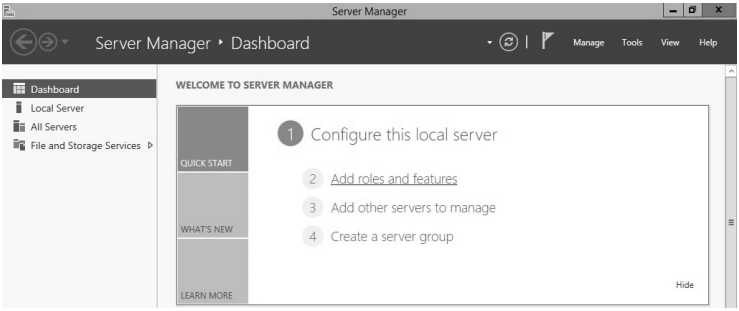


Рис. 5.13. Общий вид *Server Manager*

Далее выбираем *Add roles and features* (*Добавить роль сервера*) – или непосредственно через быстрый запуск, или через меню *Управление* – и на странице приветствия жмем *Next* (*Далее*) (рис. 5.14).

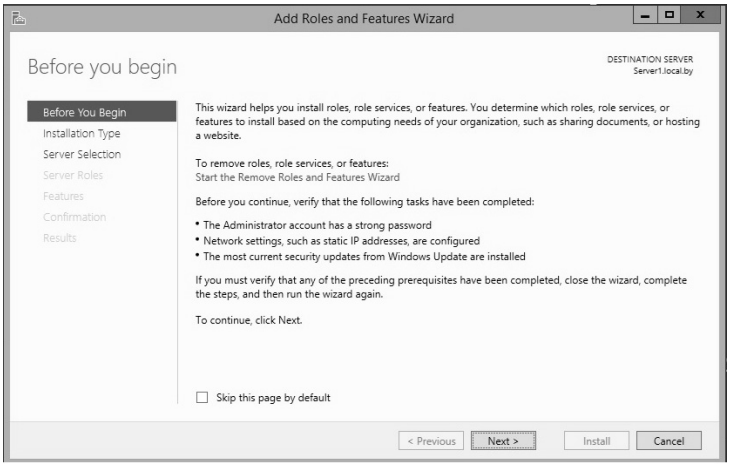


Рис. 5.14. Старт работы мастера установки роли сервера

Далее уже по умолчанию выбран необходимый пункт, т. е. *Role-based or feature-based installation* (*Установка ролей или компонентов*), и поэтому жмем *Далее* (рис. 5.15).

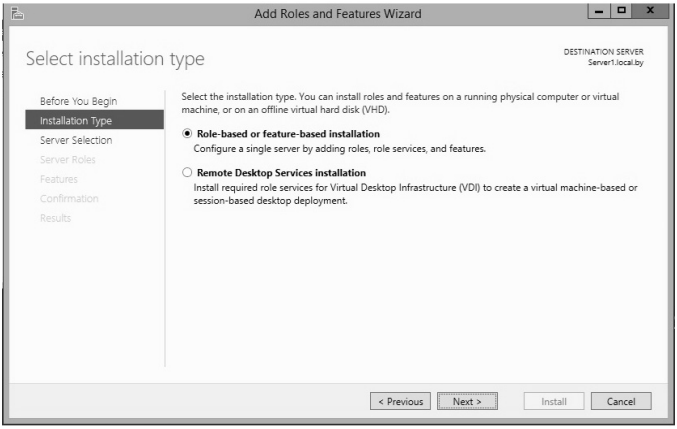


Рис. 5.15. Выбор опции установки ролей и компонент

Затем выбираем, на какой сервер или виртуальный жесткий диск будет устанавливаться DHCP-сервер (в нашем случае локально, т. е. этот же сервер). Далее определяем, какую роль будем устанавливать, и соответственно выбираем DHCP-сервер (рис. 5.16).

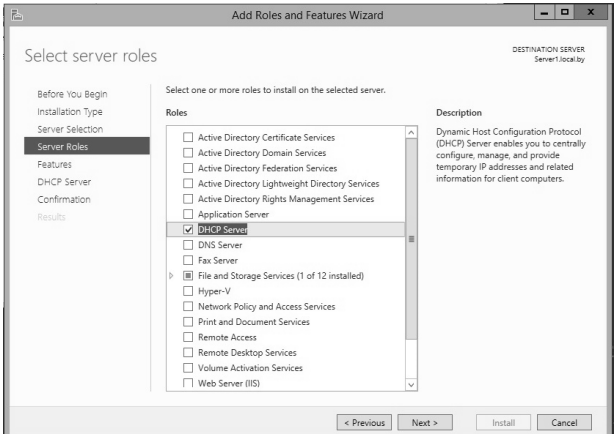


Рис. 5.16. Выбор устанавливаемой роли

После нажатия откроется окно, в котором сразу предложат выбрать для установки средства администрирования DHCP-сервера. Необходимо соглашаться, иначе все равно придется их выбирать, так

как администрироваться DHCP будут с данного компьютера. Затем жмем *Add Features* (*Добавить компоненты*) (рис. 5.17).

Далее будет предложено выбрать необходимые компоненты. Если на прошлом шаге были выбраны *Добавить компоненты*, то необходимые компоненты уже будут выбраны, а поэтому жмем *Next* (*Далее*) (рис. 5.18).

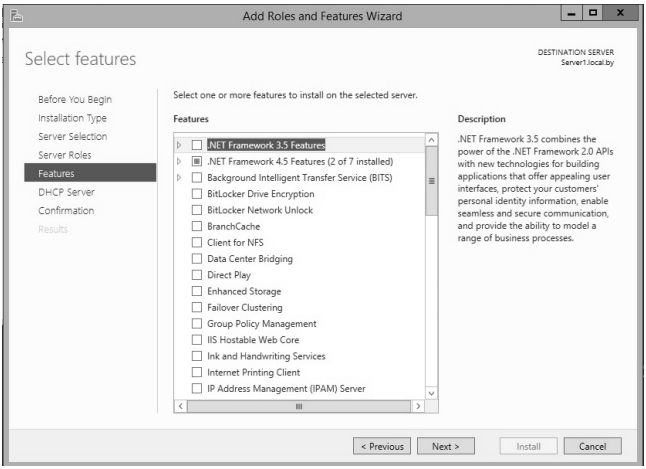


Рис. 5.17. Выбор средств администрирования

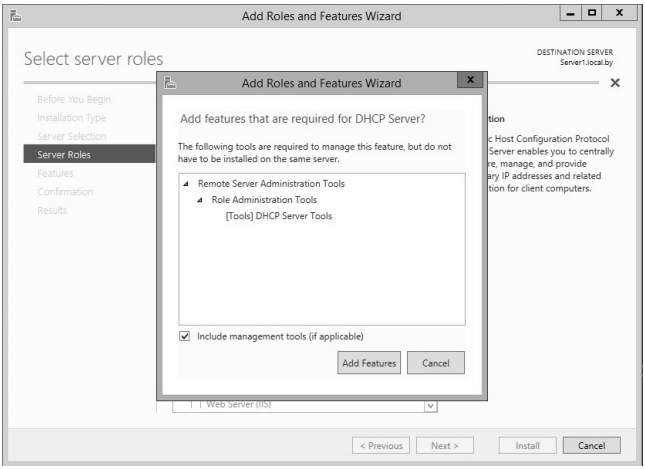


Рис. 5.18. Выбор компонент устанавливаемой роли

Еще на нескольких последующих этапах также жмем *Next* (*Далее*), и затем начнется установка DHCP-сервера (рис. 5.19).

После завершения установки будет предложено выполнить предварительную настройку параметров DHCP-сервера.

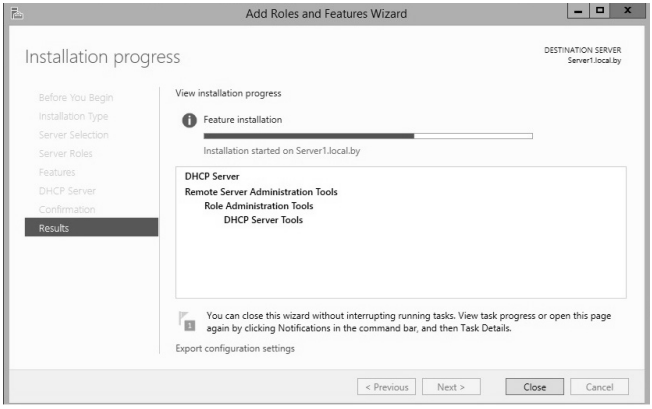


Рис. 5.19. Установка DHCP-сервера

2. ***Настройка параметров DHCP-сервера.*** После установки DHCP-сервер и средства его администрирования необходимо настроить. Для этого запускаем оснастку управления DHCP-сервером через *Server Manager* (*Диспетчер серверов*), меню *Tools* (*Средства*) (рис. 5.20).

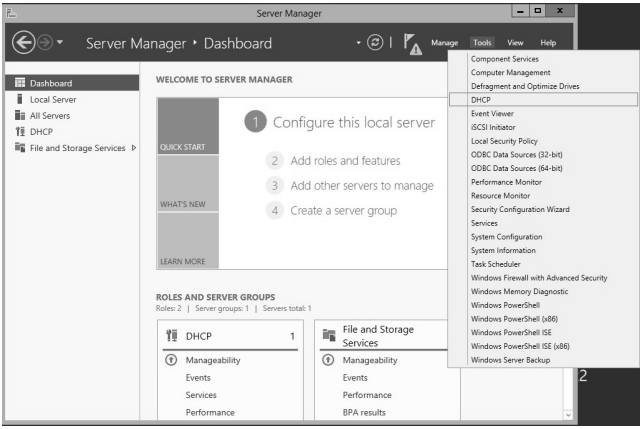


Рис. 5.20. Запуск DHCP-сервера

Создать область можно, щелкнув правой кнопкой мыши на имени сервера и выбрав пункт меню *New Scope* (*Создать область*) (или аналогичный пункт в меню *Действие* консоли DHCP) (рис. 5.21). Консоль запустит процесс создания области (*Мастер создания области*), который позволит по шагам определить все необходимые параметры.

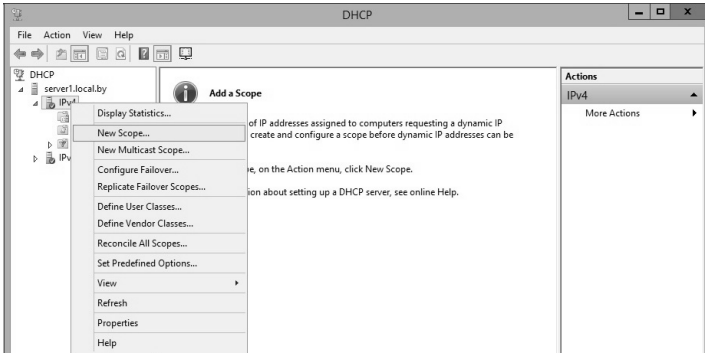


Рис. 5.21. Создание новой области DHCP-сервера

Имя и описание области. В больших сетях именование областей и задание их краткого описания облегчает работу администратора за счет более наглядного отображения в консоли всех созданных областей (рис. 5.22).

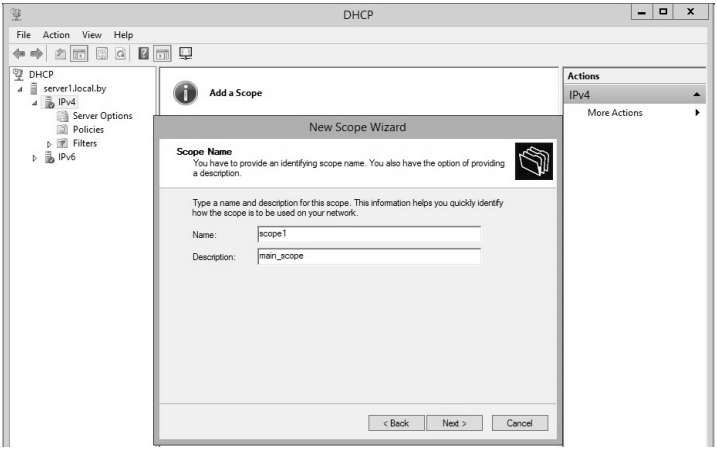


Рис. 5.22. Создание области DHCP-сервера

Дальнейший процесс создания и настройки области в Windows Server 2012 сводится к необходимости определить диапазон IP-адресов и маски подсети (в данном примере используется подсеть с Network ID 192.168.1.0 и маской 24 бита) (рис. 5.23).

Добавление исключений. На данном шаге задаются диапазоны IP-адресов, которые будут исключены из процесса выдачи адресов клиентам (все статические IP-адреса обязательно исключаются из действующего диапазона адресов). В рассмотренном на рис. 5.24 примере исключаются адреса обоих серверов: 192.168.100 и 192.168.1.101.

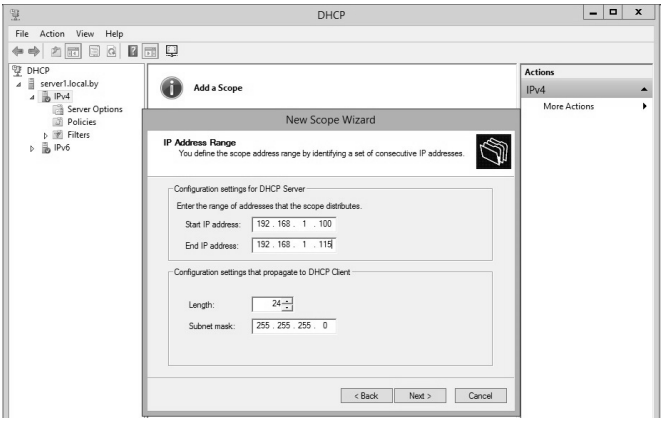


Рис. 5.23. Определение диапазона адресов области

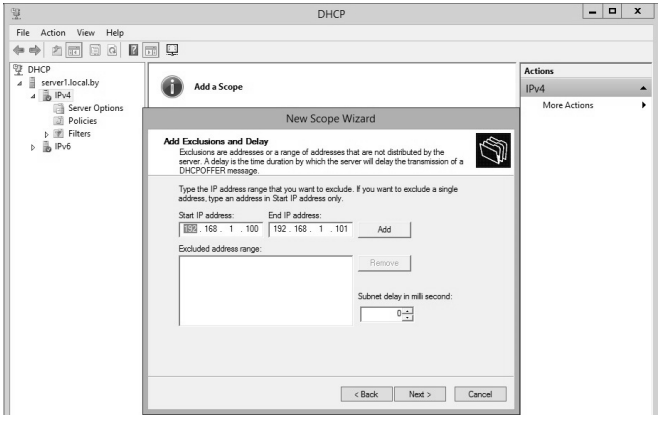


Рис. 5.24. Добавление исключающего диапазона адресов области

Срок действия аренды. Стандартный срок действия – 8 дней (рис. 5.25).

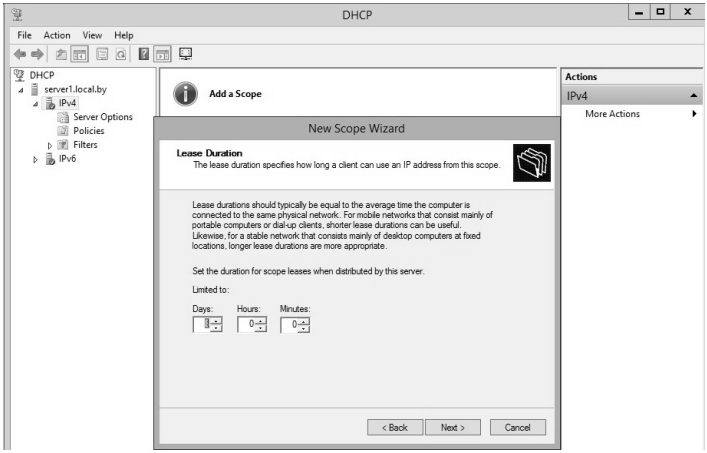


Рис. 5.25. Определение срока аренды клиентом адресов

Если в сети редко происходят изменения (добавление или удаление сетевых узлов, перемещение сетевых узлов из одной подсети в другую), то срок действия можно увеличить, это сократит количество запросов на обновление аренды. Если же сеть более динамичная, то срок аренды можно сократить, это позволит быстрее возвращать в пул IP-адреса, которые принадлежали компьютерам, уже удаленным из данной подсети.

Далее мастер предложит настроить параметры, специфичные для узлов IP-сети, относящихся к данной области, например маршрутизатор (основной шлюз), адрес DNS-сервера (можно назначить несколько адресов, рис. 5.26); адрес WINS-сервера (аналогично серверу DNS; можно также назначить несколько адресов).

Запрос на активацию области. IP-адреса, заданные в созданной области, не будут выдаваться клиентам, пока область не будет активирована (рис. 5.27).

Далее завершаем работу мастера, и область готова к использованию. Если какие-либо параметры (например, адреса серверов DNS или WINS) являются общими для всех областей, управляемых данным DHCP-сервером, то такие параметры лучше определить не в разделе параметров каждой области, а в разделе параметров самого сервера.

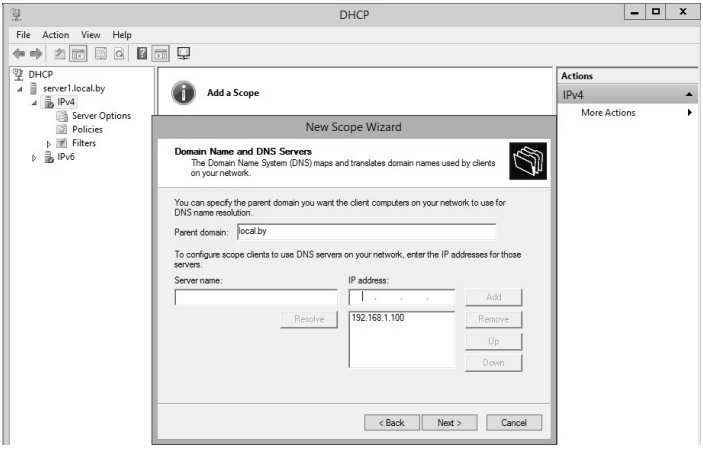


Рис. 5.26. Добавление адреса DNS-сервера, распределяемого областью

Рис. 5.27. Запрос на активацию области DHCP-сервера

Однако использование DHCP несет в себе как положительные моменты (централизация управления), так и некоторые проблемы.

Во-первых, это проблема согласования информационной адресной базы в службах DHCP и DNS. Как известно, DNS служит для преобразования символьных имен в IP-адреса. Если IP-адреса будут динамически изменяться сервером DHCP, то эти изменения необходимо также динамически вносить в базу данных сервера DNS.

Во-вторых, нестабильность IP-адресов усложняет управление сетью – необходимо выполнить резервирование IP-адресов по MAC-адресам через заполнение поля *Reservations* в соответствующей области (Scope) DHCP-сервера, тем самым формируется таблица соответствия MAC- и IP-адресов. Подобные проблемы возникают и при конфигурировании фильтров маршрутизаторов, оперирующих с IP-адресами.

Наконец, централизация процедуры назначения адресов снижает надежность системы: при отказе DHCP-сервера его клиенты не могут получить IP-адрес и другую информацию о конфигурации. Последствия такого отказа могут быть уменьшены путем использования в сети нескольких серверов DHCP (полноценное резервирование DHCP-серверов реализовано лишь начиная с Windows Server 2012).

5.3. Символьный адрес

! В стеке протоколов TCP/IP используются три типа адресов – *физические*, *IP-адреса* и *символьные доменные имена*.

Последние кажутся в этом ряду необязательными; действительно, сеть будет работать и без них. Но надо отметить, что пользователю сети неудобно запоминать числовые IP-адреса, ассоциируя их с конкретными сетевыми объектами. Люди привыкли к символьным именам, и именно поэтому в стек TCP/IP была введена система доменных имен DNS (Domain Name System). Она описывается в RFC 1034 и RFC 1035. Полное название доменных имен – FQDN (Fully Qualified Domain Name – полностью определенное имя домена). Кроме DNS-имен Windows Server поддерживает символьные имена NetBIOS.

5.3.1. Система доменных имен

Система доменных имен DNS основана на иерархической древовидной структуре, называемой пространством доменных имен. Доменом является каждый узел и лист этой структуры.

На рис. 5.28 приведен фрагмент пространства доменных имен Интернета.

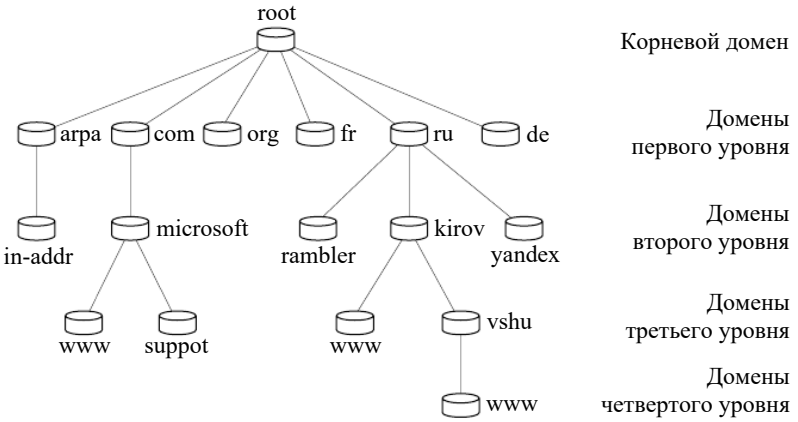


Рис. 5.28. Фрагмент пространства доменных имен Интернета

Самый верхний домен – *корневой* (Root Domain) – как реальный узел не существует, он исполняет роль вершины дерева. Ближайшие его потомки (*поддомены*) – *домены первого уровня TLD* (Top-Level Domain – домены верхнего уровня). Их можно разделить на три группы:

1. Особый домен *.arpa*, используемый для преобразования IP-адресов в доменные имена (обратное преобразование). Содержит единственный дочерний домен – *in-addr*;

2. Домены организаций:

– *com* – коммерческие организации (например, *microsoft.com*);

– *edu* – образовательные (например, mit.edu);

– *gov* – правительственные организации (например, *nsf.gov*);

– *org* – некоммерческие организации (например, *fidonet.org*);

– *net* – организации, поддерживающие сети (например, *nsf.net*);

– другие домены;

3. Домены стран (географические домены): *by* (Беларусь), *ru* (Россия), *fr* (Франция), *de* (Германия) и т. д.

Домены первого уровня включают только домены второго уровня, записи об отдельных хостах могут содержаться в доменах, начиная со второго уровня. Созданием и управлением доменами первого уровня с 1998 г. занимается международная некоммерческая организация ICANN (Internet Corporation for Assigned Names and

Numbers – интернет-корпорация по присвоению имен и адресов, *www.icann.org*). Домены второго уровня, находящиеся в географических доменах, распределяются специальными национальными организациями, которым ICANN передало полномочия в этом вопросе. Управлением доменами третьего и следующего уровней занимаются владельцы соответствующих доменов второго уровня. Полностью определенное доменное имя FQDN записывается следующим образом. Сначала идет *имя хоста* (лист в дереве пространства имен), затем через точку следует *DNS-суффикс* – последовательность доменных имен всех уровней до первого включительно. Запись оканчивается точкой, под которой подразумевается корневой домен. Например:

www.belstu.by.

В этой записи *www* – имя хоста, *belstu.by*. – DNS-суффикс. Точку в конце FQDN допускается опускать.

5.3.2. Служба DNS

Пользователь работает с доменными именами, компьютеры пересылают пакеты, пользуясь IP-адресами.

Для *согласования двух систем адресаций* необходима специальная служба, которая занимается переводом доменного имени в IP-адрес и обратно. Такая *служба в TCP/IP называется Domain Name Service* – служба доменных имен (аббревиатура DNS совпадает с аббревиатурой *системы доменных имен*). Процесс преобразования доменного имени в IP-адрес называется *разрешением доменного имени*.

Когда в сети ARPANET было несколько десятков компьютеров, задача преобразования символьного имени в IP-адрес решалась просто – создавался текстовый файл *HOSTS*, в котором хранились соответствия IP-адреса символьному имени. Этот файл должен был присутствовать на всех узлах сети. С увеличением числа узлов объем файла становился очень большим и администраторы не успевали отслеживать все изменения, происходящие в сети. Потребовалась автоматизация процесса разрешения имен, которую взяла на себя служба DNS.

Служба доменных имен поддерживает распределенную базу данных, которая хранится на специальных компьютерах – *DNS-серверах*. Термин *распределенная* означает, что вся информация не хранится в одном месте, ее части распределены по отдельным DNS-серверам.

Например, за домены первого уровня отвечают 13 корневых серверов, имеющих имена от *A.ROOT-SERVERS.NET* до *M.ROOT-SERVERS.NET*, расположенных по всему миру (большинство в США). Такие части пространства имен называются *зонами* (Zone).

Пространство имен делится на зоны исходя из удобства администрирования. Одна зона может содержать несколько доменов, так же как информация о домене может быть рассредоточена по нескольким зонам. На DNS-сервере могут храниться несколько зон. В целях повышения надежности и производительности зона может быть размещена одновременно на нескольких серверах, в этом случае один из серверов является *главным* и хранит *основную копию зоны* (Primary Zone), остальные серверы являются *дополнительными*, на них содержатся *вспомогательные копии зоны* (Secondary Zone).

Для преобразования IP-адресов в доменные имена существуют *зоны обратного преобразования* (Reverse Lookup Zone). На верхнем уровне пространства имен Интернета этим зонам соответствует домен *in-addr.arpa*. Поддомены этого домена формируются из IP-адресов, как показано на рис. 5.29.

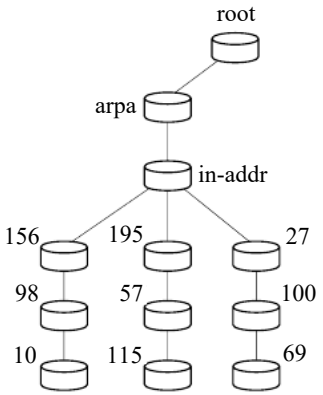


Рис. 5.29. Формирование поддоменов домена *arpa*

Следуя правилам формирования DNS-имен, зона обратного преобразования, соответствующая подсети 156.98.10.0, будет называться *10.98.156.in-addr.arpa*.

5.3.3. Процесс разрешения имен

Служба DNS построена по модели *клиент-сервер*, т. е. в процессе разрешения имен участвуют DNS-клиенты и DNS-серверы. Системный компонент DNS-клиента, называемый *DNS-распознавателем*, отправляет запросы на DNS-серверы. Запросы бывают двух видов:

– *итеративные* – DNS-клиент обращается к DNS-серверу с просьбой разрешить имя без обращения к другим DNS-серверам;

– *рекурсивные* – DNS-клиент перекладывает всю работу по разрешению имени на DNS-сервер. Если запрашиваемое имя отсутствует в базе данных и в кэше сервера, он отправляет итеративные запросы на другие DNS-серверы.

В основном DNS-клиентами используются рекурсивные запросы. На рис. 5.30 проиллюстрирован процесс разрешения доменного имени с помощью рекурсивного запроса.

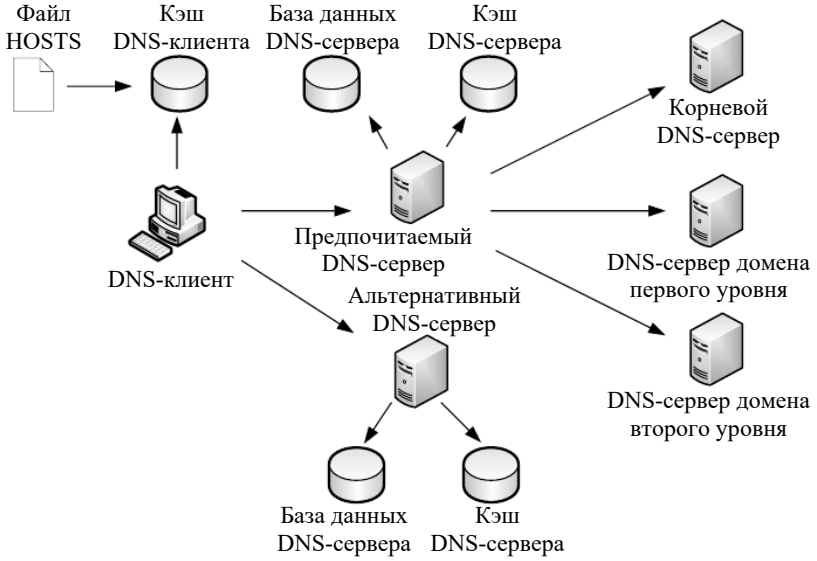


Рис. 5.30. Процесс обработки рекурсивного DNS-запроса

Сначала DNS-клиент осуществляет поиск в собственном локальном кэше DNS-имен. Это память для временного хранения ранее разрешенных запросов. В эту же память переносится содержимое файла *HOSTS* (каталог *windows/system32/drivers/etc*). Утилита *IPconfig* с ключом */displaydns* отображает содержимое DNS-кэша. Если кэш не содержит требуемой информации, DNS-клиент обращается с рекурсивным запросом к *предпочитаемому DNS-серверу* (Preferred DNS-server), адрес которого указывается при настройке стека TCP/IP. DNS-сервер просматривает собственную базу данных, а также кэш-память, в которой хранятся ответы на предыдущие запросы, отсутствующие в базе данных. В том случае, если запрашиваемое доменное имя не найдено, DNS-сервер осуществляет итеративные запросы к DNS-серверам верхних уровней, начиная с корневого DNS-сервера.

Рассмотрим процесс разрешения доменного имени на примере. Пусть требуется разрешить имя *www.microsoft.com*. Корневой домен содержит информацию о DNS-сервере, содержащем зону *.com*. Следующий запрос происходит к этому серверу, на котором хранятся данные о всех поддоменах зоны *.com*, в том числе о домене *microsoft* и его DNS-сервере. Сервер зоны *microsoft.com* может непосредственно разрешить имя *www.microsoft.com* в IP-адрес.

Иногда бывает, что предпочитаемый DNS-сервер недоступен. Тогда происходит запрос по той же схеме к *альтернативному DNS-серверу*, если, конечно, при настройке стека TCP/IP был указан его адрес.

5.3.4. Записи о ресурсах

База данных DNS-сервера содержит ***записи о ресурсах*** (Resource Record), в которых содержится информация, необходимая для разрешения доменных имен и правильного функционирования службы DNS. Существует более 20 типов записей о ресурсах, приведем самые важные:

– *A* (Host Address – адрес хоста) – основная запись, используемая для непосредственного преобразования доменного имени в IP-адрес;

– *CNAME* (Canonical Name – псевдоним) – запись определяет псевдоним хоста и позволяет обращаться по разным именам (псевдонимам) к одному и тому же IP-адресу;

– *MX* (Mail Exchanger – почтовый обменник) – запись для установления соответствия имени почтового сервера IP-адресу;

– *NS* (Name Server – сервер имен) – запись для установления соответствия имени DNS-сервера IP-адресу;

– *PTR* (Pointer – указатель) – запись для обратного преобразования IP-адреса в доменное имя;

– *SOA* (Start of Authority – начало авторизации) – запись для определения DNS-сервера, который хранит основную копию зоны;

– *SRV* (Service Locator – определитель служб) – запись для определения серверов некоторых служб (например, POP3, SMTP, LDAP).

5.3.5. Настройка DNS-адресации

Рассмотрим организацию DNS-адресации в локальной сети на примере Windows Server 2012 R2. Для организации DNS-адресации необходимо выполнить определенные действия на двух серверах (например, с именами *Server1* и *Server2*) и клиенте.

1. *Установка DNS-сервера*. Установка службы DNS производится в целом аналогично установке DHCP-сервера, с той лишь разницей, что выбирается установка службы DNS.

2. *Создание основной зоны прямого просмотра*. На сервере *Server1* создадим стандартную основную зону, например, с именем *world.ru*:

– открыть консоль *DNS* (рис. 5.31);

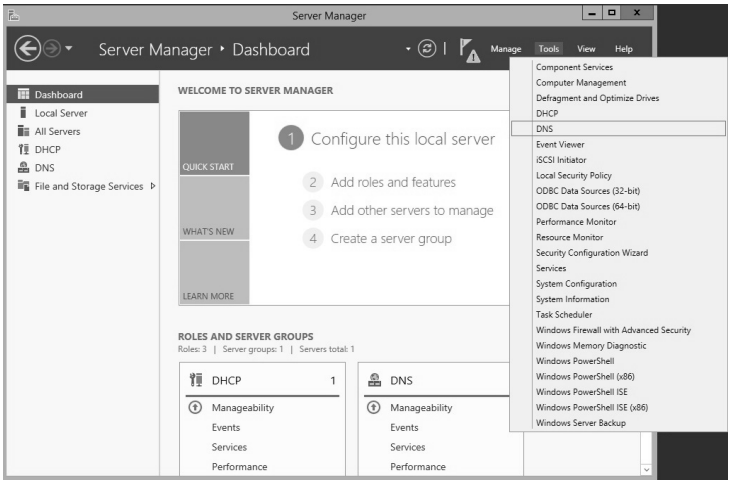


Рис. 5.31. Открытие консоли DNS-сервера

– выбрать раздел *Forward Lookup Zones* (*Зоны прямого просмотра*) и запустить мастера для создания зоны (тип зоны – *Primary* (*Основная*), динамические обновления – *Разрешить*, остальные параметры – по умолчанию (рис. 5.32);

– ввести тип зоны и имя – в примере используется название *local.by* (рис. 5.33 и 5.34); имя файла, хранящего информацию о зоне, сформируется автоматически (рис. 5.35);

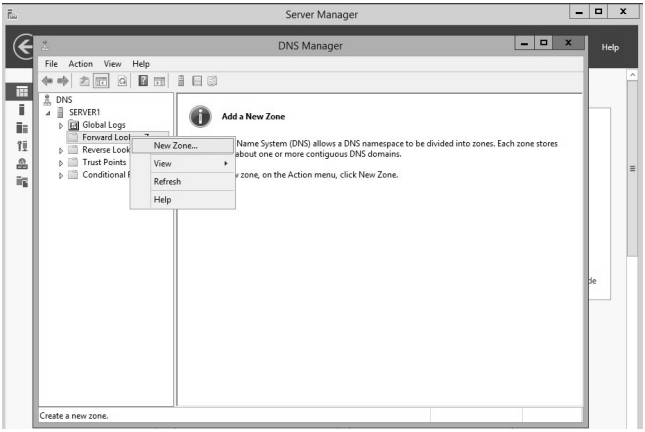


Рис. 5.32. Запуск мастера для создания новой зоны DNS-сервера

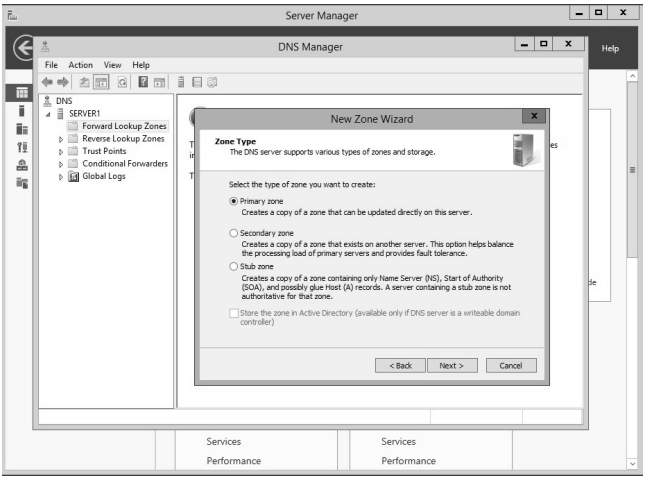


Рис. 5.33. Выбор типа новой зоны DNS-сервера

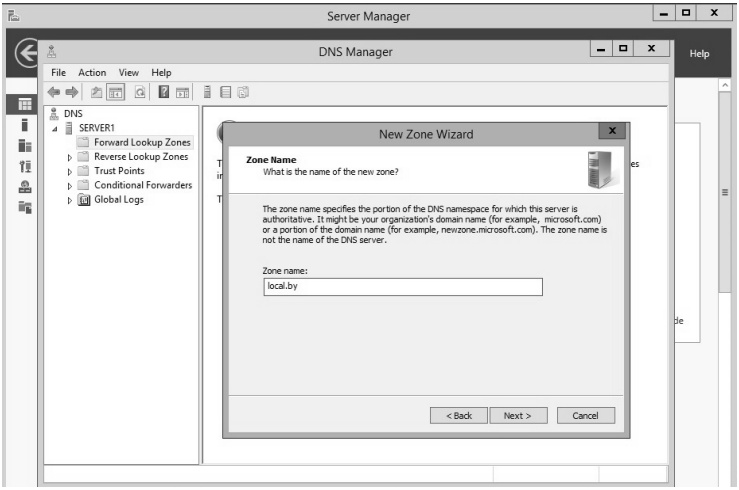


Рис. 5.34. Выбор названия новой зоны DNS-сервера

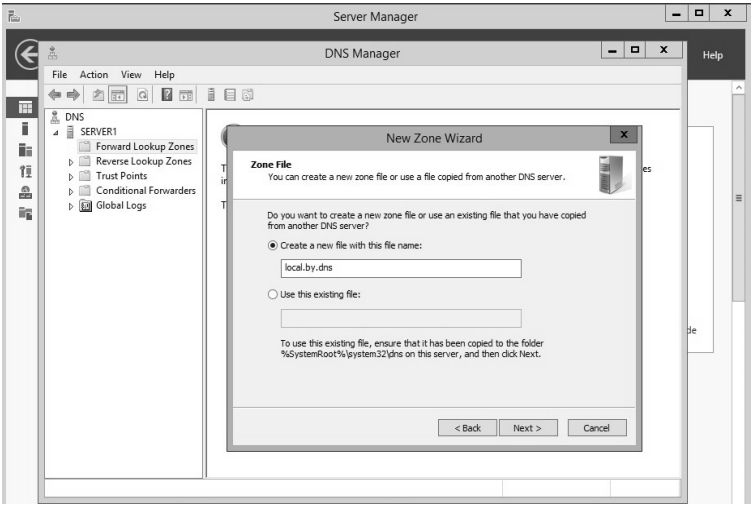


Рис. 5.35. Название файла новой зоны DNS-сервера

– разрешить передачу данной зоны на любой сервер DNS (*консоль DNS* – зона *local.by* – *Properties* (*Свойства*) – закладка *Zone Transfers* (*Передачи зон*) – Отметить *Allow zone transfers* (*Разрешить передачи*) и *To any server* (*На любой сервер*)) (рис. 5.36).

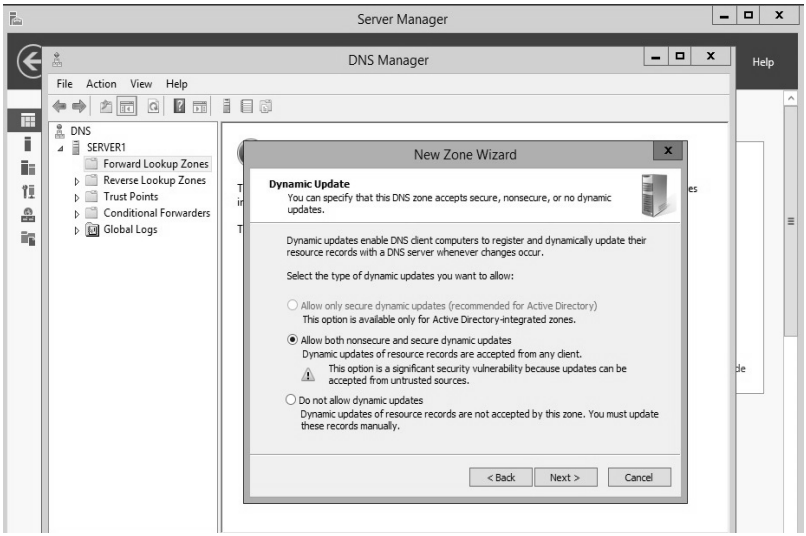


Рис. 5.36. Разрешение на передачу зоны
на другой сервер

В итоге получим зону прямого просмотра DNS-сервера, как показано на рис. 5.37.

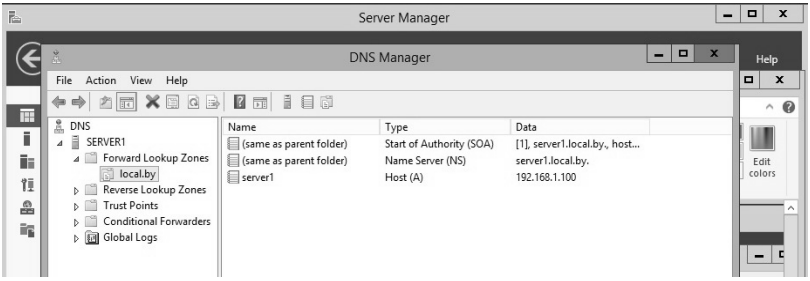


Рис. 5.37. DNS-сервер
с созданной зоной прямого просмотра

Чтобы на DNS-сервере автоматически зарегистрировалось имя сервера (в нашем случае *Server1*), необходимо указать в свойствах компьютера DNS-суффикс (рис. 5.38), а также в IP-конфигурации должен быть указан адрес DNS-сервера (в нашем случае это все тот же 192.168.1.100).

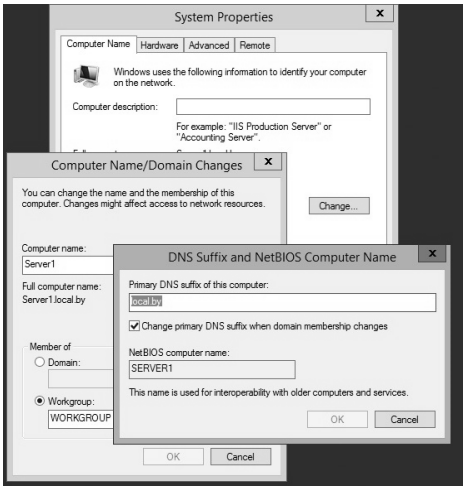


Рис. 5.38. DNS-суффикс
для символьного имени компьютера

3. *Создание дополнительной зоны прямого просмотра*. На втором сервере создадим стандартную дополнительную зону с именем *local.by* (все действия выполняются на втором сервере аналогично установке службы DNS на первом сервере с отличием типа зоны прямого просмотра):

– открыть консоль *DNS*;

– выбрать раздел *Primary Zone* (*Зона прямого просмотра*);

– запустить мастера для создания зоны (выбрать: тип зоны – *Secondary Zone* (*Дополнительная зона*), IP-адрес master-сервера (с которого будет копироваться зона) – адрес сервера *Server1*, остальные параметры – по умолчанию);

– ввести имя зоны – *local.by*.

В итоге получим совместную работу DNS-серверов с реализацией функции резервирования.

4. *Настройка узлов для выполнения динамической регистрации на сервере DNS*. Для выполнения данной задачи нужно произвести ряд действий как на серверах (если требуемые настройки не были выполнены ранее), так и в настройках клиента DNS. Рассмотрим пример настройки клиента с его регистрацией в DNS-сервере.

На сервере DNS должна быть создана соответствующая зона, а также разрешены динамические обновления.

На клиенте DNS необходимо сделать следующее:

– указать в настройках протокола TCP/IP адрес предпочитаемого DNS-сервера – тот сервер, на котором разрешены динамические обновления (в нашем примере – сервер с адресом 192.168.1.100);

– в полном имени компьютера указать соответствующий DNS-суффикс (в нашем примере – *local.by*). Для этого последовательно инициировать: *Мой компьютер – Свойства* – закладка *Имя компьютера* – кнопка *Изменить* – кнопка *Дополнительно* – в пустом текстовом поле вписать название домена *local* – кнопка *OK* (3 раза)) (рис. 5.39).

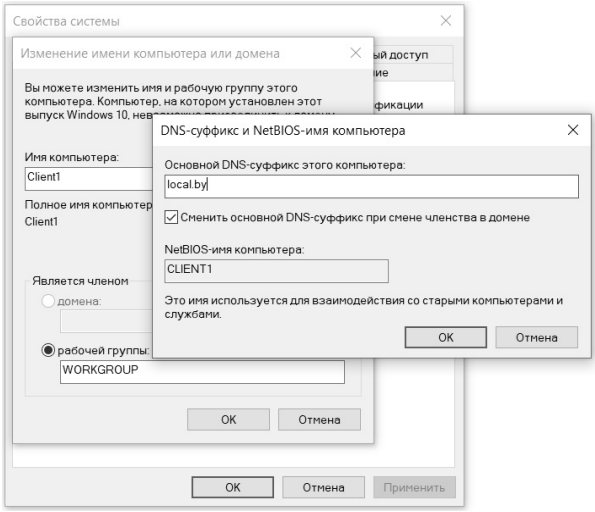


Рис. 5.39. Заполнение поля *DNS-суффикс* на клиенте

Далее система предложит перезагрузить компьютер. После выполнения перезагрузки на сервер DNS в зоне *local.by* автоматически создадутся записи типа A для серверов (рис. 5.40). В случае несоздания записи для клиента (в нашем примере это *client1*) можно на стороне клиента в командной строке выполнить команду *ipconfig/registerdns.*

Аналогичные операции необходимо выполнить на всех компьютерах сети.

Если автоматически записи не создались, то их можно создать вручную (рис. 5.41), однако при этом могут возникнуть сложности с автоматическим обновлением записей при изменении IP-адресов.

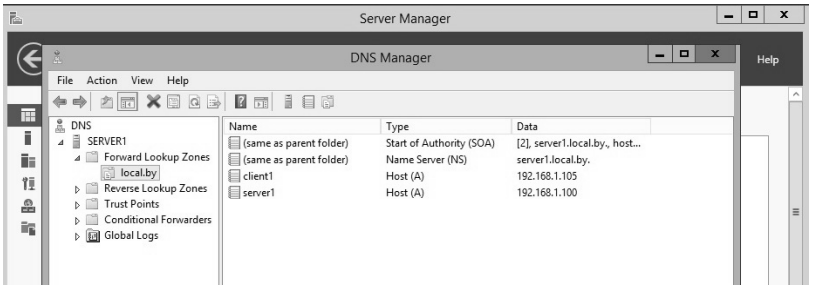


Рис. 5.40. Пример DNS-сервера
с записями для клиента и сервера

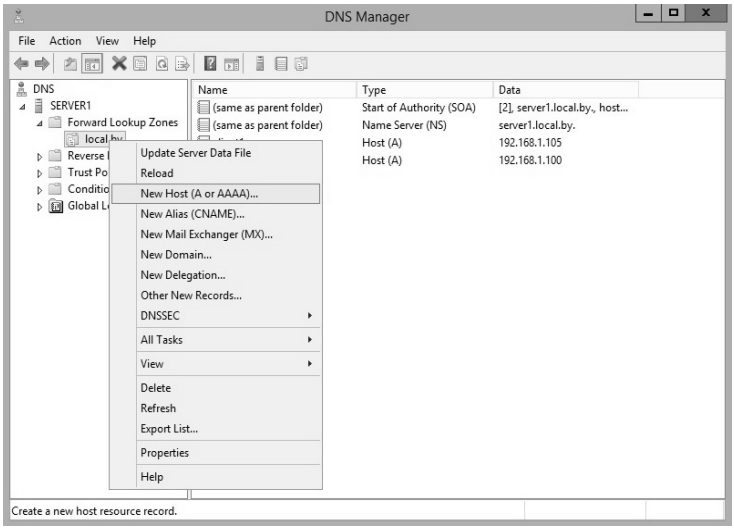


Рис. 5.41. Создание записи типа A
на DNS-сервере вручную

5. *Создание зоны обратного просмотра.* Выполняется по следующим шагам:

– открыть консоль DNS;

– выбрать раздел *Reverse Lookup Zones* (*Зоны обратного просмотра*);

– запустить мастер создания зоны (выбрать: тип зоны – *Primary* (*Основная*), динамические обновления – *Разрешить*, остальные параметры – по умолчанию) (рис. 5.42);

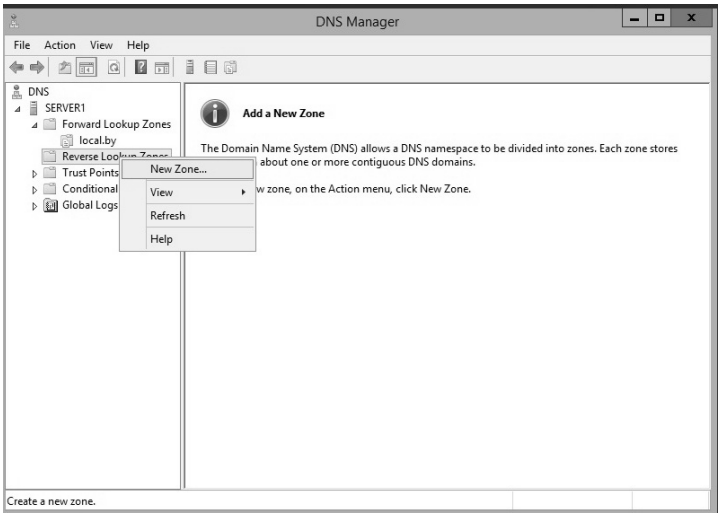


Рис. 5.42. Создание зоны обратного просмотра

– в поле *Код сети* (*ID*) ввести параметры идентификатора сети – 192.168.1, а затем выполнить команду принудительной регистрации компьютеров на сервере DNS – *ipconfig*/*registerdns*.

В итоге компьютеры зарегистрируются в обратной зоне DNS.

5.3.6. Имена NetBIOS

Протокол NetBIOS (Network Basic Input Output System – сетевая базовая система ввода-вывода) был разработан в 1984 г. для корпорации IBM как сетевое дополнение стандартной BIOS на компьютерах IBM PC. В операционных системах Microsoft Windows NT, а также в Windows 98 протокол и имена NetBIOS являлись основными сетевыми компонентами. Начиная с Windows 2000, операционные системы Microsoft ориентируются на глобальную сеть Интернет, в связи с чем фундаментом сетевых решений стали протоколы TCP/IP и доменные имена.

Имена NetBIOS не образуют никакой иерархии в своем пространстве, это простой линейный список имен компьютеров, точнее, работающих на компьютере служб. Имена компьютеров состоят из 15 видимых символов плюс 16-й служебный символ. Если видимых

символов меньше 15, то оставшиеся символы заполняются нулями (не символ нуля, а байт, состоящий из двоичных нулей). 16-й символ соответствует службе, работающей на компьютере с данным именем.

Просмотреть список имен пространства NetBIOS, которые имеются на данном компьютере, можно с помощью команды *nbtstat –n*.

Рассмотрим пример на рис. 5.43. На рисунке изображен вывод команды *nbtstat –n* на сервере *dc1.world.ru*. Результатом является список NetBIOS-имен, сгенерированных данным сервером.

```
C:\nbtstat -n

Подключение по локальной сети:
Адрес IP узла: [192.168.0.1] Код области: []

                Локальная таблица NetBIOS-имен

       Имя                Тип          Состояние
    ---------------------------------------------------
    DC1             <00>  Уникальный  Зарегистрирован
    WORLD           <00>  Группа      Зарегистрирован
    WORLD           <1C>  Группа      Зарегистрирован
    DC1             <20>  Уникальный  Зарегистрирован
    WORLD           <1B>  Уникальный  Зарегистрирован
    WORLD           <1E>  Группа      Зарегистрирован
    WORLD           <1D>  Уникальный  Зарегистрирован
    ..__MSBROWSE__.<01>  Группа      Зарегистрирован
```

Рис. 5.43. Пример информации о пространстве имен NetBIOS

В угловых скобках указан шестнадцатеричный код 16-го служебного символа какого-либо имени. Например, имя DC1 и 16-й символ «00» соответствуют службе *Рабочая станция*, которая выполняет роль клиента при подключении к ресурсам файлов и печати, предоставляемых другими компьютерами сети. А то же имя DC1 и символ с кодом «20» соответствуют службе *Сервер*, которая предоставляет ресурсы файлов и печати данного сервера для других компьютеров сети. Имя *WORLD* соответствует либо NetBIOS-имени домена *world.ru* (вспомните установку первого контроллера домена), либо имени так называемой сетевой рабочей группы, отображаемой в *Сетевом окружении* любого компьютера с Windows.

Имя *..__MSBROWSE__.* говорит о том, что данный компьютер является *Обозревателем сетевого окружения* по протоколу TCP/IP, т. е. если на каком-либо компьютере с системой Windows открыть *Сетевое окружение*, то данный компьютер будет запрашивать список компьютеров, сгруппированных в сетевой рабочей группе *WORLD*, именно с сервера *DC1*. Все эти имена, перечисленные в таблице,

будут автоматически регистрироваться в базе данных сервера *WINS* (Windows Internet Naming Service – служба имен в Интернете для Windows) после того, как данный сервер будет установлен в сети и данный компьютер станет клиентом сервера *WINS*.

5.3.7. Процесс разрешения имен в пространстве NetBIOS

Когда один компьютер пытается использовать ресурсы, предоставляемые другим компьютером через интерфейс NetBIOS поверх протокола TCP/IP, то первый компьютер, называемый *клиентом*, вначале должен определить IP-адрес второго компьютера, называемого *сервером*, по имени этого компьютера. Это может быть сделано одним из трех способов:

– *широковещательный запрос*;

– *обращение к локальной базе данных NetBIOS-имен*, хранящейся в файле *LMHOSTS* (этот файл хранится в той же папке, что и файл *HOSTS*, отображающий FQDN-имена);

– *обращение к централизованной БД имен NetBIOS*, хранящейся на сервере *WINS*.

В зависимости от типа узла NetBIOS разрешение имен осуществляется различными комбинациями перечисленных способов:

– ***b-узел*** (Broadcast Node – широковещательный узел) – разрешает имена в IP-адреса посредством широковещательных сообщений (компьютер, которому нужно разрешить имя, рассылает по локальной сети широковещательное сообщение с запросом IP-адреса по имени компьютера);

– ***p-узел*** (Peer Node – узел *точка-точка*) – разрешает имена в IP-адреса с помощью *WINS-сервера* (когда клиенту нужно разрешить имя компьютера в IP-адрес, клиент отправляет серверу имя, а тот в ответ посылает адрес);

– ***m-узел*** (Mixed Node – смешанный узел) – комбинирует запросы *b*- и *p*-узла (*WINS-клиент* смешанного типа сначала пытается применить широковещательный запрос, а в случае неудачи обращается к *WINS-серверу*; поскольку разрешение имени начинается с широковещательного запроса, m-узел загружает сеть широковещательным трафиком в той же степени, что и *b*-узел);

– ***h-узел*** (Hybrid Node – гибридный узел) – также комбинирует запросы *b*- и *p*-узла, но при этом сначала используется запрос к

WINS-серверу и лишь в случае неудачи начинается рассылка широковещательного сообщения, поэтому в большинстве сетей *h*-узлы работают быстрее и меньше засоряют сеть широковещательными пакетами.

С точки зрения производительности, объема сетевого трафика и надежности процесса разрешения NetBIOS-имен самым эффективным является *h*-узел.

Если в свойствах протокола TCP/IP Windows-компьютера нет ссылки на WINS-сервер, то данный компьютер является *b-узлом*. Если в свойствах протокола TCP/IP имеется ссылка хотя бы на один WINS-сервер, то данный компьютер является *h*-узлом. Другие типы узлов настраиваются через реестр системы Windows.

Функции сервера NetBIOS-имен описаны в RFC 1001 и 1002. В больших сетях для распределения нагрузки по регистрации и разрешению NetBIOS-имен необходимо использовать несколько серверов WINS (рекомендации Microsoft – один WINS-сервер на каждые 10 000 сетевых узлов). При этом одна часть клиентов будет настроена на регистрацию и обращение для разрешения имен на один WINS-сервер, другая часть – на второй сервер и т. д. Для того чтобы все серверы WINS имели полную информацию обо всех имеющихся в корпоративной сети NetBIOS-узлах, необходимо настроить репликацию баз данных серверов WINS между собой. После завершения репликации каждый сервер WINS будет иметь полный список NetBIOS-узлов всей сети. И любой клиент, регистрируясь на ближайшем к нему WINS-сервере, при этом может послать запрос «своему» серверу на разрешение имен NetBIOS-узлов, зарегистрированных не только на данном WINS-сервере, но и на всех остальных серверах WINS.

5.4. Утилиты диагностики TCP/IP и DNS

Любая операционная система имеет набор диагностических утилит для тестирования сетевых настроек и функционирования коммуникаций. Большой набор диагностических средств есть и в системах семейства Windows (как графических, так и в режиме командной строки).

Утилиты командной строки, являющиеся инструментами первой необходимости для проверки настроек протокола TCP/IP и работы сетей и коммуникаций, представлены в табл. 5.4. Подробное описание данных утилит содержится в системе интерактивной помощи Windows (вызывается нажатием кнопки *F1*). В табл. 5.4 указаны основные и наиболее часто используемые параметры этих команд, а также дано их краткое описание.

Таблица 5.4

Утилиты диагностики TCP/IP и DNS

Название утилиты	Параметры	Комментарии
ipconfig	*/?* – отобразить справку по команде */all* – отобразить полную информацию о настройке параметров всех адаптеров */release* – освободить динамическую IP-конфигурацию */renew* – обновить динамическую IP-конфигурацию с DHCP-сервера */flushdns* – очистить кэш разрешений DNS */registerdns* – обновить регистрацию на DNS-сервере */displaydns* – отобразить содержимое кэша разрешений DNS	Служит для отображения всех текущих параметров сети TCP/IP и обновления параметров DHCP и DNS. При вызове команды *ipconfig* без параметров выводятся IP-адрес, маска подсети и основной шлюз для каждого сетевого адаптера
arp	*-a* – отображает текущие ARP-записи	Отображение и изменение ARP-таблиц
ping	Формат команды: *ping <сетевой узел> параметры* Параметры: *-t* – бесконечная (до нажатия клавиш <Ctrl> + <Break>) отправка пакетов на указанный узел *-a* – определение имени узла по IP-адресу *-n <число>* – число отправляемых запросов *-l <размер>* – размер буфера отправки *-w <таймаут>* – таймаут ожидания каждого ответа в миллисекундах	Мощный инструмент диагностики (с помощью протокола ICMP). Команда *ping* позволяет проверить: работоспособность IP-соединения; правильность настройки протокола TCP/IP на узле; работоспособность маршрутизаторов; работоспособность системы разрешения имен FQDN или NetBIOS; доступность и работоспособность какого-либо сетевого ресурса

Окончание табл. 5.4

Название утилиты	Параметры	Комментарии
Tracert	*-d* – без разрешения IP-адресов в имена узлов *-h <макс Число>* – максимальное число прыжков при поиске узла *-w <таймаут>* – таймаут каждого ответа в миллисекундах	Служебная программа для трассировки маршрутов, используемая для определения пути, по которому IP-дейтаграмма доставляется по месту назначения
pathping	*-n* – без разрешения IP-адресов в имена узлов *-h <макс Число>* – максимальное число прыжков при поиске узла *-q <число_запросов>* – число запросов при каждом прыжке *-w <таймаут>* – таймаут каждого ответа в миллисекундах	Средство трассировки маршрута, сочетающее функции программ *ping* и *tracert* и обладающее дополнительными возможностями. Данная команда показывает степень потери пакетов на любом маршрутизаторе или канале, с ее помощью легко определить, какие маршрутизаторы или каналы вызывают неполадки в работе сети
netstat	*-a* – отображение всех подключений и ожидающих (слушающих) портов *-n* – отображение адресов и номеров портов в числовом формате *-o* – отображение кода (ID) процесса каждого подключения *-r* – отображение содержимого локальной таблицы маршрутов	Используется для отображения статистики протокола и текущих TCP/IP-соединений
nbtstat	*-n* – выводит имена пространства имен NetBIOS, зарегистрированные локальными процессами *-c* – отображает кэш имен NetBIOS (разрешение NetBIOS-имен в IP-адреса) *-R* – очищает кэш имен и перезагружает его из файла Lmhosts *-RR* – освобождает имена NetBIOS, зарегистрированные на WINS-сервере, а затем обновляет их регистрацию	Средство диагностики разрешения имен NetBIOS
hostname	Никаких ключей для данной утилиты не предусмотрено	Это самая простая утилита – она выводит на экран имя компьютера

Рассмотрим несколько примеров использования утилит командной строки для диагностики протокола TCP/IP и символьной адресации (DNS).

Пример 5.6. Использование команды *ipconfig* (без параметров и с параметром */all*) представлено на рис. 5.44.

```
Настройка протокола IP для Windows

   Имя компьютера  . . . . . . . . . : dc1
   Основной DNS-суффикс  . . . . . . : world.ru
   Тип узла. . . . . . . . . . . . . : неизвестный
   IP-маршрутизация включена . . . . : нет
   WINS-прокси включен . . . . . . . : нет
   Порядок просмотра суффиксов DNS . : world.ru

Подключение по локальной сети - Ethernet адаптер:

   DNS-суффикс этого подключения . . :
   Описание  . . . . . . . . . . . . . : Realtek RTL8139 Family PCI Fast
Ethernet NIC
   Физический адрес. . . . . . . . . : 00-11-D8-E7-14-F4
   DHCP включен. . . . . . . . . . . : нет
   IP-адрес  . . . . . . . . . . . . : 192.168.0.1
   Маска подсети . . . . . . . . . . : 255.255.255.0
   Основной шлюз . . . . . . . . . . :
   DNS-серверы . . . . . . . . . . . : 192.168.0.1
```

Рис. 5.44. Использование команды *ipconfig*

Пример 5.7. Рассмотрим использование команды *arp*. Пусть в сети только два узла (сервер *DC1* и сервер *DC2*). Тогда в кэше сервера *DC1* будет только одна запись – отображение IP-адреса сервера *DC2* на MAC-адрес сетевого адаптера (рис. 5.45).

```
C:\>arp -a
Интерфейс: 192.168.0.1 --- 0x10003
  IP-адрес              Физический адрес      Тип
  192.168.0.2           00-03-ff-e7-14-f4     динамический
```

Рис. 5.45. Использование команды *arp*

Пример 5.8. Рассмотрим использование команды *ping*.

Существуют различные варианты использования данной утилиты (в сети имеются два компьютера с именами *DC1* и *DC2*, настроена DNS-адресация):

– *ping* <IP-адрес> (рис. 5.46);

– *ping* <NetBIOS-имя узла>, когда в зоне сервера DNS нет записи для сервера *DC2* (поиск IP-адреса производится широковещательным запросом) (рис. 5.47);

– *ping* <NetBIOS-имя узла>, когда в зоне сервера DNS есть запись для сервера *DC2* (надо обратить внимание на подстановку клиентом DNS-суффикса домена в запросе на имя узла, т. е. в команде используется краткое NetBIOS-имя сервера, а в статистике команды выводится полное имя) (рис. 5.48);

```
C:\>ping 192.168.0.2

Обмен пакетами с 192.168.0.2 с 32 байт данных:

Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128

Статистика Ping для 192.168.0.2:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0 мсек, Максимальное = 0 мсек, Среднее = 0 мсек
```

Рис. 5.46. Использование команды *ping* с заданным IP-адресом

```
C:\>ping dc2

Обмен пакетами с dc2 [192.168.0.2] с 32 байт данных:

Ответ от 192.168.0.2: число байт=32 время=16мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128

Статистика Ping для 192.168.0.2:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 16 мсек, Среднее = 4 мсек
```

Рис. 5.47. Использование команды *ping* с заданным NetBIOS-именем узла (с широковещательным запросом)

```
C:\>ping dc2

Обмен пакетами с dc2.world.ru [192.168.0.2] с 32 байт данных:

Ответ от 192.168.0.2: число байт=32 время=16мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128

Статистика Ping для 192.168.0.2:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 16 мсек, Среднее = 4 мсек
```

Рис. 5.48. Использование команды *ping* с заданным NetBIOS-именем узла (при условии существования записи на DNS-сервере)

– *ping* <FQDN-имя узла>, когда в зоне сервера DNS нет записи для сервера *DC2* (узел DC2 не будет найден в сети) (рис. 5.49);

– *ping* <FQDN-имя узла>, когда в зоне сервера DNS есть запись для сервера *DC2* (узел успешно найден) (рис. 5.50);

– *ping –a* <IP-адрес> (обратное разрешение IP-адреса в имя узла) (рис. 5.51).

```
C:\>ping dc2.world.ru

При проверке связи не удалось обнаружить узел dc2.world.ru. Проверьте имя
узла и

повторите попытку.
```

Рис. 5.49. Использование команды *ping*
с заданным FQDN-именем узла
(при условии отсутствия записи на DNS-сервере)

```
C:\>ping dc2.world.ru

Обмен пакетами с dc2.world.ru [192.168.0.2] с 32 байт данных:

Ответ от 192.168.0.2: число байт=32 время=16мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128

Статистика Ping для 192.168.0.2:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 16 мсек, Среднее = 4 мсек
```

Рис. 5.50. Использование команды *ping* с заданным FQDN-именем узла
(при условии существования записи на DNS-сервере)

```
C:\>ping -a 192.168.0.2

Обмен пакетами с dc2.world.ru [192.168.0.2] с 32 байт данных:

Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128
Ответ от 192.168.0.2: число байт=32 время<1мс TTL=128

Статистика Ping для 192.168.0.2:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 0 мсек, Среднее = 0 мсек
```

Рис. 5.51. Использование команды *ping*
с обратным разрешением IP-адреса в имя узла

Пример 5.9. Рассмотрим использование команды *tracert*. На рис. 5.52 приведен пример трассировки маршрута до узла *www.ru* (если в вашем распоряжении только одна IP-сеть, то изучить работу данной команды будет невозможно).

```
C:\>tracert -d www.ru

Трассировка маршрута к www.ru [194.87.0.50]
с максимальным числом прыжков 30:

  1    17 ms    <1 мс    <1 мс  192.168.0.1
  2     1 ms    <1 мс     1 ms  217.1.1.33
  3     3 ms     3 ms     3 ms  217.1.10.1
  4     *        *        *     Превышен интервал ожидания для запроса.
  5     *        *        *     Превышен интервал ожидания для запроса.
  6    10 ms    11 ms    10 ms  217.150.36.190
  7     *        *        *     Превышен интервал ожидания для запроса.
  8    13 ms    13 ms    15 ms  194.87.0.83
  9    17 ms    12 ms    12 ms  194.87.0.50

Трассировка завершена.
```

Рис. 5.52. Использование команды *tracert*

Пример 5.10. Рассмотрим пример использования команды *pathping*. Пусть поставлена задача, аналогичная предыдущему примеру (трассировка маршрута до узла www.ru). Выполним ее командой *pathping* (рис. 5.53).

```
C:\> pathping -n www.ru

Трассировка маршрута к www.ru [194.87.0.50]
с максимальным числом прыжков 30:
  0  192.168.0.1
  1  217.1.1.33
  2  217.1.10.1
  3     *        *        *
Подсчет статистики за: 100 сек. ...
           Исходный узел     Маршрутный узел
Прыжок  RTT    Утер./Отпр.   %   Утер./Отпр.  %   Адрес
  0    0мс      0/ 100 =  0%     0/ 100 =  0%  192.168.0.1
                                 0/ 100 =  0%   |
  1    2мс      0/ 100 =  0%     0/ 100 =  0%  217.1.1.33
                                 0/ 100 =  0%   |
  2    5мс      0/ 100 =  0%     0/ 100 =  0%  217.1.10.1
                                100/ 100 =100%   |
  3   ---     100/ 100 =100%     0/ 100 =  0%  0.0.0.0

Трассировка завершена.
```

Рис. 5.53. Использование команды *pathping*

Пример 5.11. Рассмотрим пример использования команды *netstat* (с параметрами *–an* – отображение в числовой форме списка активных подключений и слушающих портов) (рис. 5.54).

```
C:\> netstat -an

Активные подключения

  Имя    Локальный адрес         Внешний адрес          Состояние
  TCP    0.0.0.0:53              0.0.0.0:0              LISTENING
  TCP    0.0.0.0:135             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1029            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1030            0.0.0.0:0              LISTENING
  TCP    192.168.0.1:139      0.0.0.0:0              LISTENING
  UDP    0.0.0.0:445             *:*
  UDP    0.0.0.0:500             *:*
  UDP    0.0.0.0:1025            *:*
  UDP    0.0.0.0:1028            *:*
  UDP    0.0.0.0:1035            *:*
  UDP    0.0.0.0:4500            *:*
  UDP    127.0.0.1:53            *:*
  UDP    127.0.0.1:123           *:*
  UDP    127.0.0.1:1026          *:*
  UDP    127.0.0.1:1027          *:*
  UDP    192.168.0.1:53       *:*
  UDP    192.168.0.1:123      *:*
  UDP    192.168.0.1:137      *:*
  UDP    192.168.0.1:138      *:*
```

Рис. 5.54. Использование команды *netstat*

Пример 5.12. Рассмотрим пример диагностики имен NetBIOS при помощи команды *nbtstat* (с параметром *–n* – отображение локальных имен NetBIOS) (рис. 5.55).

```
C:\> nbtstat -n

Подключение по локальной сети:
Адрес IP узла: [192.168.0.1] Код области: []

               Локальная таблица NetBIOS-имен

      Имя                Тип          Состояние
   ---------------------------------------------------
   DC1            <00>  Уникальный  Зарегистрирован
   DC1            <20>  Уникальный  Зарегистрирован
   WORLD          <00>  Группа      Зарегистрирован
   WORLD          <1E>  Группа      Зарегистрирован
```

Рис. 5.55. Диагностика имен NetBIOS при помощи команды *nbtstat*

5.5. Маршрутизация в IP-сетях

5.5.1. Задача маршрутизации

Раскроем суть задачи *маршрутизации* (Routing). Пусть имеется составная сеть, задача состоит в том, чтобы доставить пакет из одной подсети в другую. Известны IP-адрес и маска подсети узла-отправителя

(иными словами, ID подсети и ID хоста), IP-адрес узла-получателя. Сложность заключается в многочисленности возможных путей передачи пакета. Например, даже в простой сети, показанной на рис. 5.56, для передачи сообщения из подсети 1 в подсеть 3 существует несколько способов.

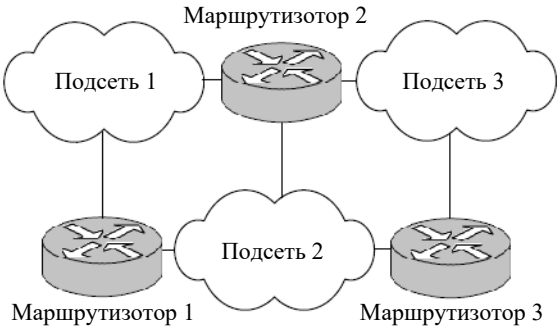


Рис. 5.56. Пример составной сети

Еще одной проблемой является то, что из существующих путей требуется выбрать *оптимальный* (по времени или по уровню надежности).

> **!** В сетях TCP/IP задача маршрутизации решается с помощью специальных устройств – *маршрутизаторов*, которые содержат *таблицы маршрутизации* (Routing Table).

Компьютер с операционной системой Windows Server также может выступать в роли маршрутизатора. Вообще говоря, любой хост, на котором действует стек TCP/IP, имеет свою таблицу маршрутизации (естественно, гораздо меньших размеров, чем на маршрутизаторе).

5.5.2. Таблица маршрутизации

Таблица маршрутизации, которая создается по умолчанию на компьютере с Windows Server 2003 (одна сетевая карта, IP-адрес: 192.168.1.1, маска подсети: 255.255.255.0), имеет вид, соответствующий табл. 5.5.

Таблица 5.5

Таблица маршрутизации

Адрес назначения (Network Destination)	Маска подсети (Netmask)	Шлюз (Gateway)	Интерфейс (Interface)	Метрика (Metric)
0.0.0.0	0.0.0.0	192.168.1.2	192.168.1.1	20
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
192.168.1.0	255.255.255.0	192.168.1.1	192.168.1.1	20
192.168.1.1	255.255.255.255	127.0.0.1	127.0.0.1	20
192.168.1.255	255.255.255.255	192.168.1.1	192.168.1.1	20
224.0.0.0	240.0.0.0	192.168.1.1	192.168.1.1	20
255.255.255.255	255.255.255.255	192.168.1.1	192.168.1.1	1

В приведенной таблице имеются следующие поля:

– *адрес назначения* (Network Destination) – адрес хоста или подсети, для которых задан маршрут в таблице;

– *маска подсети* (Netmask) – маска подсети для адреса назначения;

– *шлюз*, *маршрутизатор* (Gateway) – адрес для передачи пакета;

– *интерфейс* (Interface) – адрес собственного порта маршрутизатора (сетевой карты), на который следует передать пакет, при этом любой маршрутизатор содержит не менее двух портов (в компьютере в роли маршрутизатора с Windows Server портами являются сетевые карты);

– *метрика* (Metric) – число маршрутизаторов (число хопов), которые необходимо пройти для достижения хоста назначения. Для двух маршрутов с одинаковыми адресами назначения выбирается маршрут с наименьшей метрикой. Значение 20 в таблице соответствует 100-мегабитной сети Ethernet.

Кратко опишем записи в таблице по умолчанию:

– 0.0.0.0 – *маршрут по умолчанию* (Default Route). Эта запись выбирается в случае отсутствия совпадений с адресом назначения. В приведенной таблице маршруту по умолчанию соответствует шлюз 192.168.1.2 – это адрес порта маршрутизатора, который связывает данную подсеть с другими подсетями;

– 127.0.0.0 – *маршрут обратной связи* (Loopback Address), все пакеты с адресом, начинающимся на 127, возвращаются на узел-источник;

– 192.168.1.0 – *адрес собственной подсети* узла;

– 192.168.1.1 – *собственный адрес узла* (совпадает с маршрутом обратной связи);

– 192.168.1.255 – *адрес широковещательной рассылки* (пакет с таким адресом попадает всем узлам данной подсети);

– 224.0.0.0 – *маршрут для групповых адресов*;

– 255.255.255.255 – *адрес ограниченной широковещательной рассылки*.

5.5.3. Принципы маршрутизации в TCP/IP

Рассмотрим, каким образом решается задача *маршрутизации* на примере *составной сети*, показанной на рис. 5.56, добавив некоторые подробности – IP-адреса и MAC-адреса узлов (рис. 5.57).

Рассмотрим пример выбора маршрутов.

Пример 5.13. Предположим, что роль маршрутизатора будет выполнять компьютер с ОС Windows Server, который содержит четыре сетевые карты (четыре порта). Каждая карта имеет собственные MAC-адрес и IP-адрес, принадлежащий той подсети, к которой порт подключен. Приведем часть таблицы маршрутизации для этого компьютера (табл. 5.6).

Таблица 5.6

Таблица маршрутизации сервера с четырьмя сетевыми картами

Адрес назначения (Network Destination)	Маска подсети (Netmask)	Шлюз (Gateway)	Интерфейс (Interface)	Метрика (Metric)
0.0.0.0	0.0.0.0	160.95.0.2	160.95.0.1	20
160.95.0.0	255.255.255.0	160.95.0.1	160.95.0.1	20
160.95.0.1	255.255.255.255	127.0.0.1	127.0.0.1	20
160.95.0.255	255.255.255.255	160.95.0.1	160.95.0.1	20
160.95.1.0	255.255.255.0	160.95.1.1	160.95.1.1	20
160.95.1.1	255.255.255.255	127.0.0.1	127.0.0.1	20
160.95.1.255	255.255.255.255	160.95.1.1	160.95.1.1	20
160.95.2.0	255.255.255.0	160.95.2.1	160.95.2.1	20
160.95.2.1	255.255.255.255	127.0.0.1	127.0.0.1	20
160.95.2.255	255.255.255.255	160.95.2.1	160.95.2.1	20
160.95.3.0	255.255.255.0	160.95.3.1	160.95.3.1	20
160.95.3.1	255.255.255.255	127.0.0.1	127.0.0.1	20
160.95.3.255	255.255.255.255	160.95.3.1	160.95.3.1	20

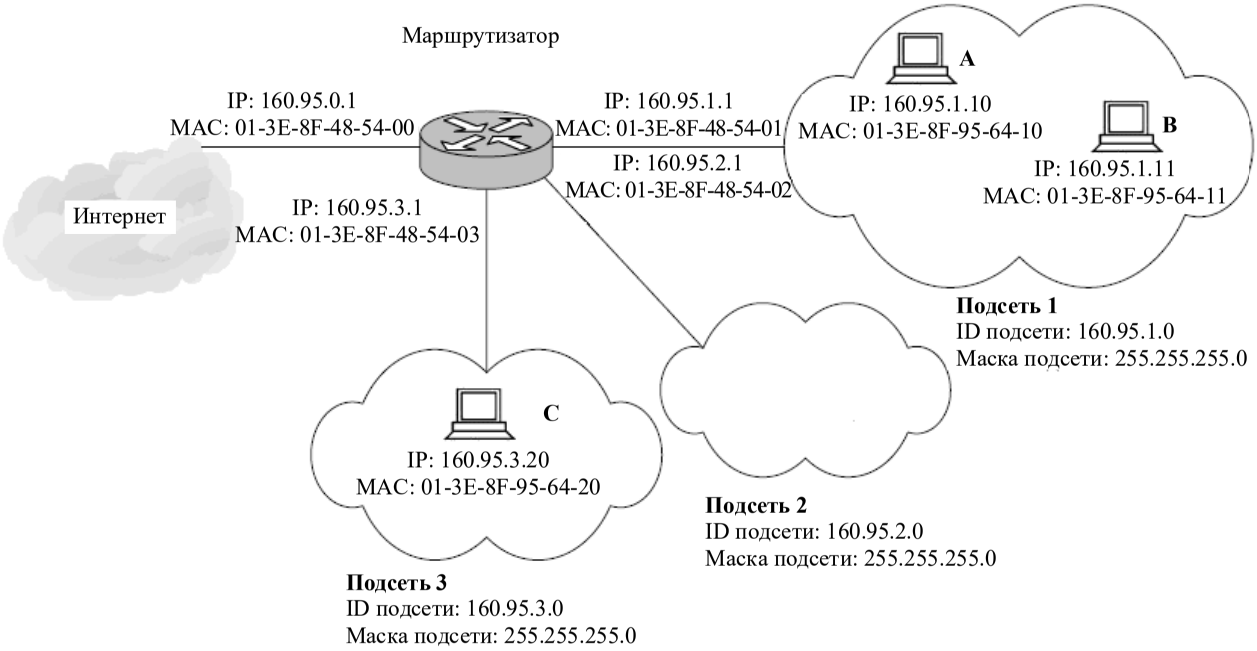


Рис. 5.57. Пример составной сети

Будем считать, что пакеты передает хост А. Его таблица маршрутизации может иметь вид, показанный в табл. 5.7.

Таблица 5.7

Таблица маршрутизации хоста А

Адрес назначения (Network Destination)	Маска подсети (Netmask)	Шлюз (Gateway)	Интерфейс (Interface)	Метрика (Metric)
0.0.0.0	0.0.0.0	160.95.1.1	160.95.1.10	20
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
160.95.1.0	255.255.255.0	160.95.1.10	160.95.1.10	20
160.95.1.10	255.255.255.255	127.0.0.1	127.0.0.1	20
160.95.1.255	255.255.255.255	160.95.1.10	160.95.1.10	20
224.0.0.0	240.0.0.0	160.95.1.10	160.95.1.10	20
255.255.255.255	255.255.255.255	160.95.1.10	160.95.1.10	1

Проанализируем, каким образом будет происходить передача пакетов от хоста А. Возможны три варианта местонахождения получателя:

– подсеть 1 (хост А – хост В);
– подсеть 2 или подсеть 3 (хост А – хост С);
– внешняя сеть (хост А – Интернет).

Если узлом назначения является хост В, пакет не должен попадать на маршрутизатор, так как получатель находится в той же сети, что и отправитель. Хост А ищет в своей таблице маршрутизации подходящий маршрут. При этом для каждой строки на адрес назначения (IP хоста В: 160.95.1.11) накладывается *маска подсети* (операция логического умножения, AND) и результат сравнивается с полем *Network Destination*. Подходящими оказываются два маршрута: 0.0.0.0 и 160.95.1.0. Из них выбирается маршрут с наибольшим числом двоичных единиц – 160.95.1.0, т. е. пакет отправляется непосредственно хосту В. IP-адрес хоста В разрешается с помощью протокола ARP в MAC-адрес. В пересылаемом пакете будет указана следующая информация:

IP-адрес отправителя:	160.95.1.10
MAC-адрес отправителя:	01-3E-8F-95-64-10
IP-адрес получателя:	160.95.1.11
MAC-адрес получателя:	01-3E-8F-95-64-11

Если окажется, что количество единиц совпадает, выбирается маршрут с наименьшей метрикой. Предположим теперь, что узел А

отправляет пакет узлу С (подсеть 3). Поиск в собственной таблице маршрутизации не дает подходящих результатов, кроме маршрута по умолчанию – 0.0.0.0. Для этого маршрута указан адрес порта маршрутизатора 160.95.1.1 (Default Gateway – *шлюз по умолчанию*). Протокол ARP помогает определить MAC-адрес порта. Именно на него отправляется пакет сначала, причем указывается IP-адрес конечного получателя (узла С):

IP-адрес отправителя:	160.95.1.10
MAC-адрес отправителя:	01-3E-8F-95-64-10
IP-адрес получателя:	160.95.3.20
MAC-адрес получателя:	01-3E-8F-48-54-01

Модуль маршрутизации *Windows Server* анализирует полученный пакет, выделяет из него адрес узла С, осуществляет поиск в своей таблице маршрутизации (поиск происходит так же, как на хосте А). Находятся две подходящие записи: 160.95.3.0 и 0.0.0.0. Выбирается первый маршрут, так как в нем больше двоичных единиц. Пакет в подсеть 3 отправляется с порта 160.95.3.1:

IP-адрес отправителя:	160.95.1.10
MAC-адрес отправителя:	01-3E-8F-48-54-03
IP-адрес получателя:	160.95.3.20
MAC-адрес получателя:	01-3E-8F-95-64-20

Наконец, в случае когда хост А осуществляет передачу во внешнюю сеть, пакет сначала попадает на маршрутизатор. Поиск в таблице маршрутизации дает единственный подходящий результат: 0.0.0.0. Поэтому пакет отправляется на порт внешнего маршрутизатора 160.95.0.2. Дальнейшее продвижение пакета выполняют маршрутизаторы Интернета.

5.5.4. Настройка таблиц маршрутизации

Для построения таблиц маршрутизации существует два метода: *статический* и *динамический*.

Статический метод заключается в том, что администратор вручную создает и удаляет записи в таблице. В состав операционной системы Windows Server входит утилита *route.* Она может использоваться с четырьмя командами:

– *print* – печать текущего содержимого таблицы;
– *add* – добавление новой записи;
– *delete* – удаление устаревшей записи;
– *change* – редактирование существующей записи.

Запись должна определяться следующим образом: *<destination> MASK <netmask> <gateway> METRIC <metric> IF <interface>*. Например, *route add 160.95.1.0 mask 255.255.255.0 160.95.1.1 metric 20 IF 1*.

Кроме того, можно использовать два ключа:

– *f* – удаление из таблицы всех записей, кроме записей по умолчанию;

– *p* – создание постоянной записи (т. е. не исчезающей после перезагрузки). По умолчанию создаются временные записи.

Достоинством статического метода является простота. Но для сетей с быстро меняющейся конфигурацией этот метод не подходит, так как администратор может не успевать отслеживать все изменения. В этом случае применяют *динамический метод* построения таблицы маршрутизации, основанный на протоколах маршрутизации.

5.5.5. Протоколы обмена маршрутной информацией

Протоколы обмена маршрутной информацией (Routing Information Protocol, RIP) стека TCP/IP относятся к классу адаптивных, которые, в свою очередь, делятся на две группы, каждая из которых связана с одним из следующих типов алгоритмов:

– *дистанционно-векторный алгоритм* (Distance Vector Algorithms, DVA),

– *алгоритм состояния связей* (Link State Algorithms, LSA).

В *алгоритмах дистанционно-векторного типа* каждый маршрутизатор периодически и широковещательно рассылает по сети вектор расстояний от себя до всех известных ему сетей. Под расстоянием обычно понимается число промежуточных маршрутизаторов, через которые пакет должен пройти прежде, чем попадет в соответствующую сеть.

Может использоваться и другая метрика, учитывающая не только число транзитных точек, но и время прохождения пакетов по связи

между соседними маршрутизаторами. Получив вектор от соседнего маршрутизатора, каждый из них добавляет к вектору информацию о других известных ему сетях, о которых он узнал непосредственно (если они подключены к его портам) или из аналогичных объявлений других маршрутизаторов, а затем новое значение вектора опять рассылается по сети. В итоге каждый маршрутизатор узнает информацию об имеющихся сетях и о расстоянии до них через соседние маршрутизаторы.

Дистанционно-векторные алгоритмы хорошо работают только в небольших сетях. В больших сетях они «засоряют» каналы связи интенсивным широковещательным трафиком, к тому же изменения конфигурации могут отрабатываться по этому алгоритму не всегда корректно, так как маршрутизаторы не имеют точного представления о топологии связей в сети, а располагают только обобщенной информацией − вектором дистанций, к тому же полученной через посредников.

Наиболее распространенным протоколом, основанным на дистанционно-векторном алгоритме, является протокол *RIP* (*Routing Information Protocol*, RFC 1723). Это, по сути, один из старейших протоколов обмена маршрутной информацией. Однако он до сих пор используется.

Алгоритмы состояния связей обеспечивают каждый маршрутизатор информацией, достаточной для построения точного графа связей сети. Все маршрутизаторы работают на основании одинаковых графов, что делает процесс маршрутизации более устойчивым к изменениям конфигурации.

Широковещательная рассылка используется здесь только при изменениях состояния связей, что происходит в надежных сетях не так часто.

Для того чтобы понять, в каком состоянии находятся линии связи, подключенные к его портам, маршрутизатор периодически обменивается короткими пакетами со своими ближайшими соседями. Этот трафик также широковещательный, но он передается только между соседями и поэтому не так «засоряет» сеть.

Протоколом, основанным на алгоритме состояния связей, в стеке TCP/IP является протокол *OSPF* (*Open Shortest Path Firs*, RFC 2328).

Он принят в 1991 г.) и обладает многими особенностями, ориентированными на применение в больших сильно разветвленных (гетерогенных) сетях.

Протокол OSPF вычисляет маршруты в IP-сетях, сохраняя при этом другие протоколы обмена маршрутной информацией.

На практике также применяются комбинированные протоколы: более старый – *EGP* (*Exterior Gateway Protocol* – протокол внешнего шлюза) и его современная версия – *BGP* (*Border Gateway Protocol* – протокол граничного шлюза). Именно последний, т. е. протокол BGP, является основным протоколом динамической маршрутизации в сети Интернет.

Выводы

1. Физический, или локальный, адрес узла определяется технологией, с чьей помощью построена сеть, в которую входит узел. Для узлов, входящих в локальные сети, – это MAC-адрес сетевого адаптера или порта маршрутизатора.

2. В стеке TCP/IP используется три типа адресов: локальные (MAC-адреса), IP-адреса и доменные имена. IP-адрес действует на сетевом уровне и позволяет объединять разнородные локальные и глобальную сети в единую составную сеть.

3. IP-адрес стоит из 4 байтов (октетов), разделенных точками. В его структуре выделяют две части – номер подсети и номер узла. Определение того, какая часть адреса отводится под номер подсети, осуществляется двумя способами – с помощью классов и с помощью масок. В схеме классовой адресации существует 5 классов, основными являются классы A, B и C.

4. Поле номера подсети определяется по первым битам адреса. При использовании масок номер подсети находится при помощи логического умножении маски на IP-адрес. Адресация с применением масок является более гибкой по сравнению с классами.

5. Уже довольно давно возникла проблема дефицита IP-адресов. Решение данной проблемы с помощью масок является временным. Принципиально другой подход заключается в существенном расширении адресного пространства и реализуется в протоколе IPv6.

6. Некоторые IP-адреса являются особыми и не используются при адресации конкретных узлов. Это нужно учитывать при назначении IP-адресов.

7. Для преобразования IP-адресов в аппаратные MAC-адреса применяется протокол ARP, для обратного преобразования – протокол RARP.

8. Для диагностики и управления стеком TCP/IP в операционной системе Microsoft Windows Server существуют специальные утилиты – *IPconfig*, *ping*, *tracert*, *netstat*, *arp*, *hostname* и др.

9. Задача маршрутизации заключается в определении оптимального пути передачи сообщения в составных сетях с меняющейся топологией. В сетях TCP/IP эту задачу решают маршрутизаторы на основе таблиц маршрутизации. В таблицы маршрутизации входит информация о номерах и масках подсетей назначения, адресах шлюзов и собственных портов маршрутизатора, а также о метриках. Решение о передаче пакета на тот или иной порт принимается на основании совпадения адреса назначения из пакета с адресом из таблицы, при этом оптимальный маршрут выбирается на основе метрики. Для адресов, отсутствующих в таблице, применяется специальный адрес – адрес шлюза по умолчанию.

10. Для создания таблиц маршрутизации в Windows Server 2003 используют два метода – статический, с помощью утилиты *route*, и динамический, с применением протоколов маршрутизации RIP и OSPF.

Контрольные вопросы

1. Что такое хост?

2. Опишите структуру MAC-адреса.

3. Перечислите виды и приведите примеры адресов, используемых в стеке TCP/IP.

4. Из каких частей состоит IP-адрес?

5. Как определяется номер подсети в IP-адресе?

6. Каков диапазон возможных адресов у сети класса C?

7. Определите номер подсети на основе маски: 116.98.04.39/27.

8. Каковы основные особенности протокола IPv6?

9. Поясните принцип работы протокола ARP.
10. В чем заключается задача маршрутизации?
11. Для чего нужна таблица маршрутизации?
12. Назовите основные поля в таблице маршрутизации.
13. Что такое *default gateway*?
14. Перечислите ключи утилиты *route* (приведите примеры).
15. Назовите преимущества и недостатки протокола RIP.
16. Перечислите преимущества и недостатки протокола OSPF.
17. Поясните принцип работы утилиты *ping* (на примерах).
18. Дайте объяснение принципа работы утилиты *tracert* (на примерах).

6

БАЗОВЫЕ ТЕХНОЛОГИИ ЛОКАЛЬНОЙ СЕТИ

За время, прошедшее с появления первых локальных сетей, было разработано несколько сотен самых разных сетевых технологий, однако заметное распространение получили только некоторые из них, что связано, прежде всего, с поддержкой этих сетей известными фирмами и высоким уровнем стандартизации принципов их организации.

Увеличение скорости передачи в локальных сетях до 1000 Мбит/с и более требует применения самых передовых технологий, проведения серьезных и дорогих научных исследований. Это могут позволить себе только крупнейшие фирмы, которые поддерживают свои стандартные сети и их более совершенные разновидности. К тому же большинство потребителей уже установило у себя какие-то сети и вовсе не желает сразу и полностью заменять все сетевое оборудование на другое, пусть даже в чем-то лучшее. Поэтому в ближайшем будущем вряд ли стоит ожидать принятия принципиально новых стандартов.

В настоящее время стандартные сети обеспечивают большой диапазон допустимых размеров сети, допустимого количества ее абонентов и, что не менее важно, большой диапазон цен на аппаратуру. Но проблема выбора сети все равно остается трудноразрешимой.

6.1. Сети Ethernet и Fast Ethernet

6.1.1. Основные характеристики сетей Ethernet

Наибольшее распространение среди стандартных сетей получила сеть *Ethernet.* Впервые она появилась в 1972 г. (разработчиком выступила известная фирма Xerox). Сеть оказалась довольно удачной, и вследствие этого ее в 1980 г. поддержали такие крупнейшие фирмы,

как DEC и Intel (объединение этих фирм, поддерживающих Ethernet, назвали DIX по первым буквам их названий). Стараниями данных фирм в 1985 г. сеть Ethernet стала международным стандартом, ее приняли крупнейшие международные организации по стандартам: комитет 802 IEEE (Institute of Electrical and Electronic Engineers) и ECMA (European Computer Manufacturers Association).

Стандарт получил название IEEE 802.3. Он определяет *множественный доступ* к моноканалу типа *шина* с обнаружением конфликтов и контролем передачи, т. е. с уже упоминавшимся методом доступа CSMA/CD. Этому стандарту удовлетворяют и некоторые другие сети, так как он не сильно детализирован. В результате сети стандарта IEEE 802.3 нередко несовместимы между собой как по конструктивным, так и по электрическим характеристикам.

Основные характеристики стандарта IEEE 802.3: топология – шина, среда передачи – коаксиальный кабель, скорость передачи – 10 Мбит/с, максимальная длина – 5 км, максимальное количество абонентов – до 1024, длина сегмента сети – до 500 м, количество абонентов на одном сегменте – до 100, метод доступа – CSMA/CD, передача узкополосная, т. е. без модуляции (моноканал).

Между стандартами IEEE 802.3 и Ethernet существуют небольшие отличия. Сеть Ethernet сейчас достаточно популярна в мире, и нет сомнения, что таковой она и останется в ближайшие годы. Этому в немалой степени способствовало то, что с самого начала все характеристики, параметры, протоколы сети были открыты для всех, в результате чего огромное число производителей во всем мире стало выпускать аппаратуру Ethernet, полностью совместимую между собой.

В *классической сети Ethernet* применяется 50-омный коаксиальный кабель двух видов (толстый и тонкий). Однако в последнее время (с начала 1990-х гг.) все большее распространение получает версия Ethernet, использующая в качестве среды передачи витые пары. Определен также стандарт для применения в сети оптоволоконного кабеля.

В стандарты были внесены соответствующие добавления. В 1995 г. появился стандарт на более быструю версию Ethernet, работающую на скорости 100 Мбит/с (так называемый Fast Ethernet, стандарт IEEE 802.3u), использующую в качестве среды передачи витую

пару или оптоволоконный кабель. Появилась и версия на скорость 1000 Мбит/с (Gigabit Ethernet, стандарт IEEE 802.3z).

Помимо стандартной топологии *шина* применяются также топологии *пассивная звезда* и *пассивное дерево*. При этом предполагается использование *репитеров* (репитер – повторитель) и *пассивных* (репитерных) *концентраторов*, соединяющих между собой различные части (сегменты) сети (рис. 6.1).

В качестве сегмента может также выступать единичный абонент. Коаксиальный кабель используется для шинных сегментов, а витая пара и оптоволоконный кабель – для лучей пассивной звезды (для присоединения к концентратору одиночных компьютеров). Главное – чтобы в полученной топологии не было замкнутых путей (*петель*). Фактически абоненты соединены в физическую шину, так как сигнал от каждого из них распространяется сразу во все стороны и не возвращается назад (как в кольце). Максимальная длина кабеля всей сети в целом (максимальный путь сигнала) теоретически может достигать 6,5 км, но практически не превышает 2,5 км.

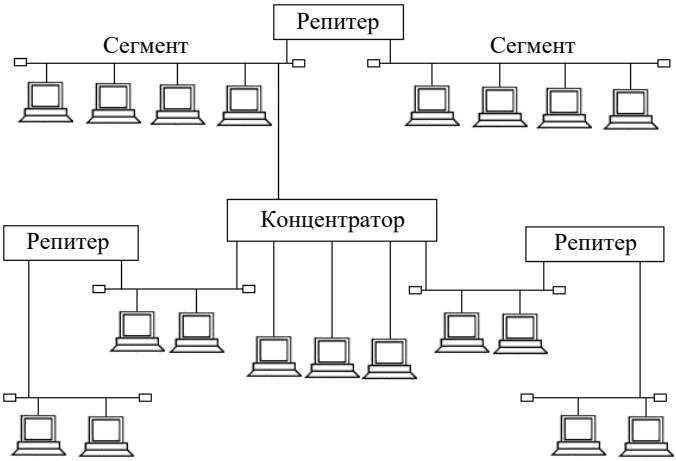


Рис. 6.1. Топология сети Ethernet

! В сети *Fast Ethernet* не предусмотрена физическая топология *шина*, используются только *пассивная звезда* или *пассивное дерево*. К тому же в Fast Ethernet гораздо более жесткие требования к предельной длине сети.

При увеличении в 10 раз скорости передачи и сохранении формата пакета его минимальная длина становится в 10 раз короче (5,12 мкс против 51,2 мкс в Ethernet) – допустимая величина *двойного времени прохождения сигнала по сети* уменьшается в 10 раз.

Для сети Ethernet, работающей на скорости 10 Мбит/с, стандарт определяет четыре основных типа среды передачи информации:

– 10BASE5 (толстый коаксиальный кабель);
– 10BASE2 (тонкий коаксиальный кабель);
– 10BASE-T (витая пара);
– 10BASE-FL (оптоволоконный кабель).

Обозначение среды передачи включает в себя следующих три элемента: цифра «10» означает скорость передачи 10 Мбит/с. Слово BASE означает передачу в основной полосе частот (без модуляции высокочастотного сигнала), а последний элемент означает допустимую длину сегмента: 5 – 500 м, 2 – 200 м (точнее, 185 м) или тип линии связи: «T» – витая пара (Twisted Pair), «F» – оптоволоконный кабель (Fiber Optic).

Точно так же для сети Ethernet, которая работает на скорости 100 Мбит/с (Fast Ethernet), стандарт определяет три типа среды передачи:

– 100BASE-T4 (счетверенная витая пара);
– 100BASE-TX (сдвоенная витая пара);
– 100BASE-FX (оптоволоконный кабель).

В вышеприведенном списке цифра «100» означает скорость передачи 100 Мбит/с, буква «T» – витую пару, буква «F» – оптоволоконный кабель. Типы 100BASE-TX и 100BASE-FX иногда объединяют под именем 100BASE-X, а 100BASE-T4 и 100BASE-TX – под именем 100BASE-T.

Отметим, что сеть Ethernet не отличается ни рекордными характеристиками, ни оптимальными алгоритмами, она уступает по ряду параметров другим стандартным сетям. Но благодаря мощной поддержке, высочайшему уровню стандартизации, огромным объемам выпуска технических средств, Ethernet резко выделяется среди других стандартных сетей, и поэтому любую другую сетевую технологию принято сравнивать именно с Ethernet.

Для передачи информации в сети Ethernet применяется стандартный код *Манчестер-II*. При этом один уровень сигнала нулевой, а другой – отрицательный, т. е. постоянная составляющая сигнала не равна нулю. При отсутствии передачи потенциал в сети нулевой.

Гальваническая развязка осуществляется аппаратурой адаптеров, репитеров и концентраторов. При этом приемопередатчик сети гальванически изолирован от остальной аппаратуры с помощью трансформаторов и изолированного источника питания, а с кабелем сети соединен напрямую.

6.1.2. Структура пакета в сетях Ethernet

Доступ к сети Ethernet осуществляется по методу CSMA/CD, который обеспечивает полное равноправие абонентов. В сети используются пакеты переменной длины со структурой, представленной на рис. 6.2. Длина кадра Ethernet (т. е. пакета без преамбулы) должна быть не менее 512-битных интервалов, или 51,2 мкс (именно такова предельная величина двойного времени прохождения в сети). Предусмотрена *индивидуальная*, *групповая* и *широковещательная адресация*.

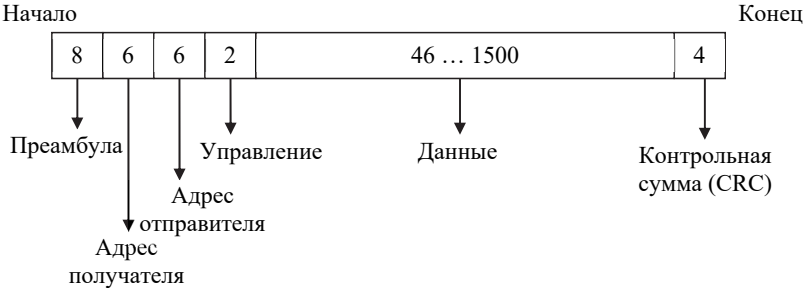


Рис. 6.2. Структура пакета сети Ethernet (цифры показывают количество байтов)

В пакет Ethernet входят следующие поля.

1. *Преамбула* состоит из 8 байтов, первые семь из которых представляют собой код 10101010, а последний восьмой – код 10101011. В стандарте IEEE 802.3 этот последний байт называется **признаком начала кадра** (SFD – Start of Frame Delimiter) и образует отдельное поле пакета.

2. *Адрес получателя* (приемника) и *адрес отправителя* (передатчика) включают по 6 байтов. Эти адресные поля обрабатываются аппаратурой абонентов.

3. *Поле управления* (L/T – Length/Type) содержит информацию о длине поля данных. Оно может также определять тип используемого протокола. Принято считать, что если значение этого поля не больше 1500, то оно определяет длину поля данных. Если же его значение больше 1500, то оно определяет тип кадра. Поле управления обрабатывается программно.

4. *Поле данных* должно включать в себя от 46 до 1500 байтов данных. Если пакет должен содержать менее 46 байтов данных, то поле данных дополняется байтами заполнения. В соответствии со стандартом IEEE 802.3 в структуре пакета выделяется специальное поле заполнения (Pad Data – *незначащие данные*), которое может иметь нулевую длину в том случае, если данных достаточно (больше 46 байтов).

5. *Поле контрольной суммы* (FCS – Frame Check Sequence) содержит 32-разрядную *циклическую контрольную сумму пакета* (Cyclic Redundancy Check, CRC) и служит для проверки правильности передачи пакета (*обнаружение и исправление ошибок*).

Таким образом, *минимальная длина кадра* (пакета без преамбулы) составляет 64 байта (512 битов). Именно эта величина определяет максимально допустимую двойную задержку распространения сигнала по сети в 512-битных интервалах (51,2 мкс – для Ethernet, 5,12 мкс – для Fast Ethernet). Стандарт предполагает, что преамбула может уменьшаться при прохождении пакета через различные сетевые устройства, поэтому она не учитывается. Максимальная длина кадра составляет 1518 байтов (12 144 бита, т. е. 1214,4 мкс – для Ethernet, 121,44 мкс – для Fast Ethernet). Это важно для выбора размера буферной памяти сетевого оборудования и для оценки общей загруженности сети.

На практике в сетях Ethernet на канальном уровне используются кадры 4 различных форматов (типов).

Один и тот же тип кадра может иметь разные названия, поэтому ниже для каждого типа кадра приведено по несколько наиболее употребительных названий:

– кадр 802.3/LLC (кадр 802.3/802.2 или кадр Novell 802.2);

– кадр Raw 802.3 (или кадр Novell 802.3);

– кадр Ethernet DIX (или кадр Ethernet II);

– кадр Ethernet SNAP (SubNetwork Access Protocol – протокол доступа к подсети).

Выводы

1. Ethernet – это самая распространенная на сегодняшний день технология локальных сетей. В широком смысле Ethernet – это целое семейство технологий, включающее различные фирменные и стандартные варианты, из которых наиболее известны фирменный вариант Ethernet DIX, 10-мегабитные варианты стандарта IEEE 802.3, а также новые высокоскоростные технологии Fast Ethernet и Gigabit Ethernet. Почти все виды технологий Ethernet используют один и тот же метод разделения среды передачи данных – метод случайного доступа CSMA/CD, который определяет облик технологии в целом.

2. В узком смысле Ethernet – это 10-мегабитная технология, описанная в стандарте IEEE 802.3.

3. Одно из важных явлений в сетях Ethernet – *коллизия* – ситуация, когда две станции одновременно пытаются передать кадр данных по общей среде. Наличие коллизий – это неотъемлемое свойство сетей Ethernet, являющееся следствием принятого случайного метода доступа. Возможность четкого распознавания коллизий обусловлена правильным выбором параметров сети, в частности соблюдением соотношения между минимальной длиной кадра и максимально возможным диаметром сети.

4. Максимально возможная *пропускная способность* сегмента Ethernet в кадрах в секунду достигается при передаче кадров минимальной длины и составляет 14 880 кадр./с. При этом полезная пропускная способность сети составляет всего 5,48 Мбит/с, что лишь ненамного превышает половину номинальной пропускной способности – 10 Мбит/с.

5. Технология Ethernet поддерживает 4 разных типа кадров, которые имеют общий формат адресов узлов. Существуют формальные признаки, по которым сетевые адаптеры автоматически распознают тип кадра.

6. В зависимости от типа физической среды стандарт IEEE 802.3 определяет различные спецификации: 10Base-5, 10Base-2, 10Base-T, 10Base-FL, 10Base-FX и т. д. Для каждой спецификации определяются тип кабеля, максимальные длины непрерывных отрезков кабеля, а также правила использования повторителей для увеличения диаметра сети.

6.2. Сеть Token Ring

6.2.1. Основные характеристики сетей Token Ring

Сеть ***Token Ring*** предложена фирмой IBM в 1985 г. (первый вариант появился в 1980 г.). Назначением сети является объединение в сеть всех типов компьютеров, выпускаемых IBM (от персональных до больших).

По сравнению с аппаратурой Ethernet аппаратура Token Ring оказывается заметно дороже, так как использует более сложные методы управления обменом, поэтому распространена сеть Token Ring значительно меньше. Однако ее применение становится оправданным, когда требуются большие интенсивности обмена (например, при связи с большими компьютерами) и ограниченное время доступа.

Сеть Token Ring имеет топологию *кольцо*, хотя внешне она больше напоминает *звезду*. Это связано с тем, что отдельные абоненты (компьютеры) присоединяются к сети не прямо, а через специальные *концентраторы*, или *многостанционные устройства доступа* (MSAU или MAU – Multistation Access Unit).

Поэтому физически сеть образует *звездно-кольцевую топологию* (рис. 6.3).

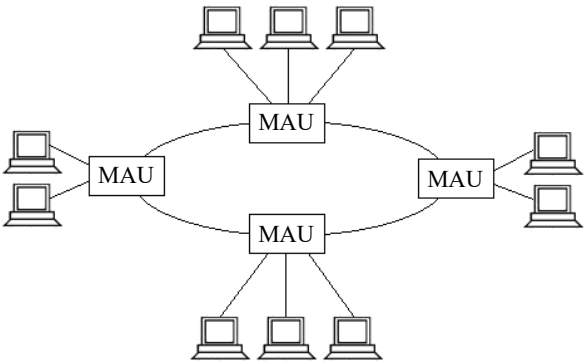


Рис. 6.3. Звездно-кольцевая топология сети Token Ring

В действительности же абоненты объединяются все-таки в кольцо, т. е. каждый из них передает информацию одному соседнему абоненту, а принимает информацию от другого соседнего абонента.

Концентратор (MAU) при этом только позволяет централизовать задание конфигурации, отключение неисправных абонентов, контроль за работой сети и т. д. (рис. 6.4). Для присоединения кабеля к концентратору применяются специальные разъемы, которые обеспечивают постоянство замкнутости кольца даже при отключении абонента от сети. Концентратор в сети может быть и единственным – в кольцо замыкаются только абоненты, подключенные к нему.

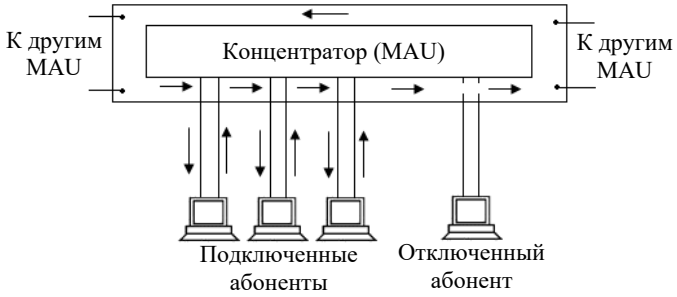


Рис. 6.4. Соединение абонентов сети Token Ring в кольцо с помощью концентратора (MAU)

В каждом кабеле, соединяющем *адаптеры* и *концентратор* (*адаптерные кабели*, Adapter Cable), находятся на самом деле две разнонаправленные линии связи. Такими же двумя разнонаправленными линиями связи, входящими в *магистральный кабель* (Path Cable), объединяются между собой в кольцо различные концентраторы (рис. 6.5), хотя для этой же цели может использоваться и единственная однонаправленная линия связи (рис. 6.6).

Конструктивно концентратор представляет собой автономный блок с восьмью разъемами для подключения абонентов (компьютеров) с помощью *адаптерных кабелей* и двумя (крайними) разъемами для подключения к другим концентраторам с помощью специальных магистральных кабелей.

Несколько концентраторов могут конструктивно объединяться в группу, ***кластер*** (Cluster), внутри которой абоненты также соединены в единое кольцо. Применение кластеров позволяет увеличивать количество абонентов, подключенных к одному центру.

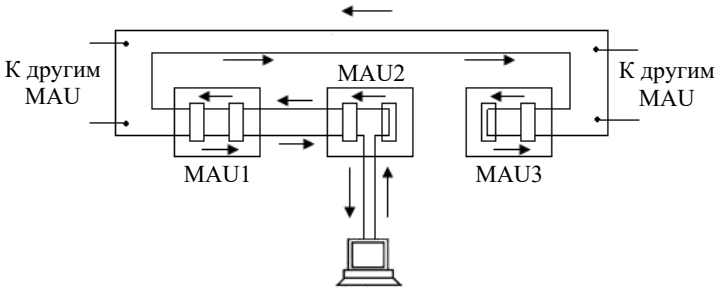


Рис. 6.5. Объединение концентраторов двунаправленной линией связи

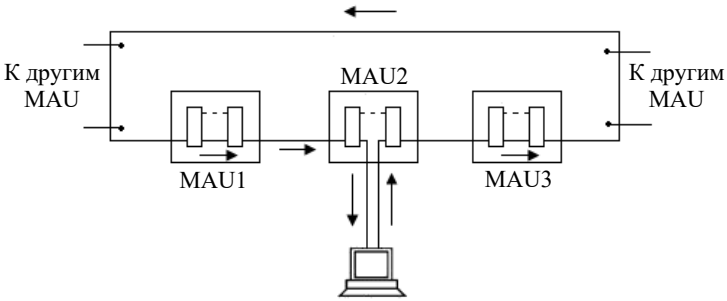


Рис. 6.6. Объединение концентраторов однонаправленной линией связи

В качестве среды передачи в сети IBM Token Ring сначала применялась витая пара, но затем появились варианты аппаратуры для коаксиального кабеля, а также для оптоволоконного кабеля в стандарте FDDI. В данных сетях может применяться кабель витая пара как неэкранированного (UTP), так и экранированного (STP) типов.

Основные технические характеристики сети Token Ring следующие:

– максимальное количество концентраторов типа IBM 8228 (MAU) – 12;

– максимальное количество абонентов в сети – 96;

– максимальная длина кабеля между абонентом и концентратором – 45 м;

– максимальная длина кабеля между концентраторами – 45 м;

– максимальная длина кабеля, соединяющего все концентраторы – 120 м;

– скорость передачи данных – 4 и 16 Мбит/с.

Все приведенные характеристики относятся к *неэкранированной витой паре*. В случае применения другой среды передачи характеристики сети могут отличаться. Например, при использовании экранированной витой пары количество абонентов может быть увеличено до 200 (вместо 96), длина кабеля – до 100 м (вместо 45), количество концентраторов – до 33, а полная длина кольца, соединяющего концентраторы – до 200 м. Оптоволоконный кабель позволяет увеличивать длину кабеля до 1 км. Как видим, сеть Token Ring уступает сети Ethernet как по допустимому размеру сети, так и по максимальному количеству абонентов.

Для передачи информации в Token Ring используется вариант кода *Манчестер-II*. Как и в любой звездообразной топологии, никаких дополнительных мер по электрическому согласованию и внешнему заземлению не требуется.

Для присоединения кабеля к сетевому адаптеру используется внешний девятиконтактный разъем типа DIN. Так же как и адаптеры Ethernet, адаптеры Token Ring имеют на своей плате переключатели или перемычки для настройки адресов и прерываний системной шины. Если сеть Ethernet можно построить только на адаптерах и кабеле, то для сети Token Ring обязательно нужно приобретать концентраторы. Это также увеличивает стоимость аппаратуры Token Ring.

В то же время в отличие от Ethernet сеть Token Ring лучше «держит» большую нагрузку (больше 30~40%) и обеспечивает гарантированное время доступа. Это крайне необходимо, например, в сетях производственного назначения, в которых задержка реакции на внешнее событие может привести к серьезным последствиям.

В сетях с *маркерным методом доступа* (а к ним, кроме сетей Token Ring, относятся сети FDDI, а также сети, близкие к стандарту 802.4, ArcNet, сети производственного назначения MAP) право на доступ к среде передается циклически от станции к станции по логическому кольцу (рис. 6.7): Token Ring. IEEE 802.5 и FDDI представляют собой кольцевые сети. Оба используют **маркер**, или **токен** (Token), – особый вид сетевого кадра для регулирования сетевого доступа отдельных узлов. Токен передается от узла к узлу, и каждый узел может отправить сообщение только после того, как он получил (захватил) токен. Основное различие между сетями IEEE 802.5 и FDDI заключается в правилах, которые диктуют, когда узел может получить токен и на какой срок.

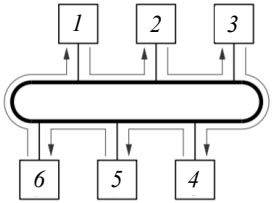


Рис. 6.7. Взаимодействие узлов в сети
с передачей маркера (токена)

В сети Token Ring кольцо образуется отрезками кабеля, соединяющими соседние станции. Таким образом, каждая станция связана с предшествующей и последующей станциями и может непосредственно обмениваться данными только с ними. Для обеспечения доступа станций к физической среде по кольцу циркулирует кадр специального формата и назначения – ***маркер***, или ***токен***.

В сети Token Ring любая станция всегда непосредственно получает данные только от одной станции – той, которая является предыдущей в кольце. Такая станция называется ***ближайшим активным соседом***, расположенным выше по потоку (данных) – Nearest Active Upstream Neighbour, NAUN. Передачу же данных станция всегда осуществляет своему ближайшему соседу вниз по потоку данных.

Получив маркер, станция анализирует его и при отсутствии у нее данных для передачи обеспечивает его продвижение к следующей станции. Станция, которая имеет данные для передачи, при получении маркера изымает его из кольца, что дает ей право доступа к физической среде и передачи своих данных. Затем эта станция выдает в кольцо кадр данных установленного формата последовательно по битам. Переданные данные проходят по кольцу всегда в одном направлении от одной станции к другой. Кадр снабжен адресом назначения и адресом источника.

Все станции кольца ретранслируют кадр побитно, как повторители. Если кадр проходит через станцию назначения, то, распознав свой адрес, эта станция копирует кадр в свой внутренний буфер и вставляет в кадр признак подтверждения приема. Станция, выдавшая кадр данных в кольцо, при обратном его получении с подтверждением приема изымает этот кадр из кольца и передает в сеть новый

маркер для обеспечения возможности другим станциям сети передавать данные. Такой алгоритм доступа применяется в сетях Token Ring со скоростью работы 4 Мбит/с, описанных в стандарте 802.5.

Время владения разделяемой средой в сети Token Ring ограничивается ***временем удержания маркера*** (Token Holding Time), после истечения которого станция обязана прекратить передачу собственных данных (текущий кадр разрешается завершить) и передать маркер далее по кольцу.

Станция может успеть передать за время удержания маркера один или несколько кадров в зависимости от размера кадров и величины времени удержания маркера. Обычно время удержания маркера по умолчанию равно 10 мс, а максимальный размер кадра в стандарте IEEE 802.5 не определен. Для сетей 4 Мбит/с он обычно равен 4 Кбайт, а для сетей 16 Мбит/с – 16 Кбайт. Это связано с тем фактом, что за время удержания маркера станция должна успеть передать хотя бы один кадр. При скорости 4 Мбит/с за время 10 мс можно передать 5000 байтов, а при скорости 16 Мбит/с – соответственно 20 000 байтов. Максимальные размеры кадра выбраны с некоторым запасом.

В сетях Token Ring 16 Мбит/с используется также несколько другой алгоритм доступа к кольцу, называемый **алгоритмом раннего освобождения маркера** (Early Token Release, ETR).

В соответствии с ETR станция передает маркер доступа следующей станции сразу же после окончания передачи последнего бита кадра, не дожидаясь возвращения по кольцу этого кадра с битом подтверждения приема. В этом случае пропускная способность кольца используется более эффективно, так как по нему одновременно продвигаются кадры нескольких станций. Тем не менее, свои кадры в каждый момент времени может генерировать только одна станция, которая в данный момент владеет маркером доступа. Остальные станции в это время только повторяют чужие кадры, так что принцип разделения кольца во времени сохраняется, ускоряется только процедура передачи владения кольцом.

Для различных видов сообщений могут назначаться различные приоритеты: от 0 (низший) до 7 (высший). Решение о приоритете конкретного кадра принимает передающая станция (протокол Token Ring

получает этот параметр через межуровневые интерфейсы от протоколов верхнего уровня, например прикладного). Маркер также всегда имеет некоторый уровень текущего приоритета. Станция имеет право захватить переданный ей маркер только в том случае, если приоритет кадра, который она хочет передать, выше приоритета маркера или равен ему. В противном случае станция обязана передать маркер следующей по кольцу станции.

За наличие в сети маркера, причем единственной его копии, отвечает *активный монитор*. Если активный монитор не получает маркер в течение длительного времени, то он порождает новый маркер.

6.2.2. Форматы кадров Token Ring

В сетях Token Ring существует три различных формата кадров:
– *маркера*;
– *данных*;
– *прерывающей последовательности*.

Кадр маркера состоит из трех полей, каждое длиной в 1 байт.

1. *Начальный ограничитель* (Start Delimiter, SD) появляется в начале маркера, а также в начале любого кадра, проходящего по сети. Поле представляет собой следующую уникальную последовательность символов манчестерского кода: JKOJKOOO. Поэтому начальный ограничитель нельзя спутать ни с какой битовой последовательностью внутри кадра.

2. *Управление доступом* (Access Control) состоит из четырех подполей: *PPP*, *T*, *M* и *RRR*, где *PPP* – биты приоритета, *T* – бит маркера, *M* – бит монитора, *RRR* – резервные биты приоритета (рис. 6.8).

Бит *T*, установленный в 1, указывает на то, что данный кадр является маркером доступа. Бит монитора устанавливается в 1 активным монитором и в 0 любой другой станцией, передающей маркер или кадр. Если активный монитор видит маркер или кадр, содержащий бит монитора со значением 1, то активный монитор «знает», что этот кадр или маркер уже однажды обошел кольцо и не был обработан станциями. Если это кадр, то он удаляется из кольца. Если это маркер, то активный монитор передает его дальше по кольцу. Использование полей приоритетов будет рассмотрено ниже.

Рис. 6.8. Формат маркера

3. *Конечный ограничитель* (End Delimeter, ED) – последнее поле маркера. Точно так как и поле начального ограничителя, это поле содержит уникальную последовательность манчестерских кодов JK1JK1, а также два однобитовых признака: *I* и *E*. Признак *I* (Intermediate) показывает, является ли кадр последним в серии кадров (1-0) или промежуточным (1-1). Признак *E* (Error) – это признак ошибки. Он устанавливается в 0 станцией-отправителем, и любая станция кольца, через которую проходит кадр, должна установить этот признак в 1, если она обнаружит ошибку по контрольной сумме или другую некорректность кадра.

Кадр данных состоит из нескольких групп полей:

– *последовательность начала кадра;*
– *адрес получателя;*
– *адрес отправителя;*
– *данные;*
– *последовательность контроля кадра;*
– *последовательность конца кадра.*

Кадр данных может переносить данные либо для управления кольцом (данные MAC-уровня), либо пользовательские данные LLC-уровня (Logical Link Control, LLC – *логический контроль связи*). Функции LLC по обеспечению надежной передачи данных в LAN напоминают функции транспортного уровня моделей OSI и TCP/IP: при передаче данных снизу вверх LLC принимает от уровня MAC пакет сетевого уровня, пришедший из сети. Соотношение функций протоколов LLC и MAC во многом подобно соотношению функций протоколов UDP/TCP и IP. Как и протоколы транспортного уровня

UDP/TCP, протокол LLC не занимается непосредственно доставкой кадров узлам сети. Передачу данных между узлами, подобно IP, выполняет после получения доступа к разделяемой среде уровень MAC. MAC, так же как и IP, обеспечивает доставку в дейтаграммном режиме, т. е. без установления соединения и без восстановления потерянных или поврежденных кадров. В том случае, когда протоколы верхних уровней запрашивают у LLC надежный транспортный сервис, LLC устанавливает соединение с узлом назначения и организует повторную доставку кадров.

Кадр данных включает те же три поля, что и маркер, и имеет кроме них еще несколько дополнительных полей.

1. Каждый кадр данных (MAC или LLC) начинается с *последовательности начала кадра*, которая содержит три поля:

– *начальный ограничитель*, такой же, как и для кадра маркера;

– *управление доступом*, также совпадает для кадров и для маркеров;

– *контроль кадра* – это однобайтное поле, которое содержит два подполя: *тип кадра* и *идентификатор управления MAC*: 2 бита типа кадра имеют значения 00 – для кадров MAC и 01 – для кадров LLC.

Биты идентификатора управления MAC определяют тип кадра управления кольцом из списка 6 управляющих кадров MAC-уровня. Назначение этих шести типов кадров следующее:

а) чтобы удостовериться, что ее адрес уникальный, станция, когда впервые присоединяется к кольцу, посылает кадр *тест дублирования адреса* (Duplicate Address Test, DAT);

б) чтобы сообщить другим станциям, что он работоспособен, активный монитор периодически посылает в кольцо кадр *существует активный монитор* (Active Monitor Present, AMP);

в) кадр *существует резервный монитор* (Standby Monitor Present, SMP) отправляется любой станцией, не являющейся активным монитором;

г) резервный монитор отправляет кадр *Маркер заявки* (Claim Token, CT), когда подозревает, что активный монитор отказал, затем резервные мониторы договариваются между собой, какой из них станет новым активным монитором;

д) станция отправляет кадр *Сигнал* (Beacon, BCN) в случае возникновения серьезных сетевых проблем, таких как обрыв кабеля, обнаружение станции, передающей кадры без ожидания маркера, выход

станции из строя. Определяя, какая станция отправляет кадр сигнала, диагностирующая программа (ее существование и функции не определяются стандартами Token Ring) может локализовать проблему. Каждая станция периодически передает кадры BCN до тех пор, пока не примет кадр BCN от своего предыдущего (NAUN) соседа. В результате в кольце только одна станция продолжает передавать кадры BCN – та, у которой «имеются проблемы с предыдущим соседом». В сети Token Ring каждая станция знает MAC-адрес своего предыдущего соседа, поэтому Beacon-процедура приводит к выявлению адреса некорректно работающей станции;

е) кадр *Очистка* (Purge, PRG) используется новым активным монитором для того, чтобы перевести все станции в исходное состояние и очистить кольцо от всех ранее посланных кадров.

2. *Адрес получателя* (Destination Address, DA). Может иметь длину либо 2 байта, либо 6. Первый бит адреса назначения определяет групповой или индивидуальный адрес как для 2-байтных, так и для 6-байтных адресов. Второй бит в 6-байтных адресах говорит о том, назначен адрес локально или глобально. Адрес, состоящий из всех единиц, является *широковещательным*.

3. *Адрес отправителя* (Source Address, SA). Имеет тот же размер и формат, что и адрес получателя. Однако признак группового адреса используется в нем особым способом. Так как адрес источника не может быть групповым, то наличие единицы в этом разряде говорит о том, что в кадре имеется специальное *поле маршрутной информации* (Routing Information Field, RIF). Эта информация требуется при работе мостов, связывающих несколько колец Token Ring, в режиме маршрутизации от источника.

В стандарте 802.5 используются адреса той же структуры, что и в стандарте 802.3.

4. *Данные* (Info). Поле *данных* кадра может содержать данные одного из описанных управляющих кадров уровня MAC или пользовательские данные, упакованные в кадр уровня LLC. Это поле, как уже отмечалось, не имеет определенной стандартом максимальной длины, хотя существуют практические ограничения на его размер, основанные на соотношениях между временем удержания маркера и временем передачи кадра.

5. *Последовательность контроля кадра* используется для обнаружения ошибок. Состоит из четырех байтов остатка циклически избыточной контрольной суммы, вычисляемой по алгоритму CRC-32

(используется порождающий полином степени 32; см. п. 12.4.4), осуществляющему циклическое суммирование по модулю 32.

6. *Последовательность конца кадра* состоит из двух полей: *Конечный ограничитель* (End Delimeter, ED) и *Статус кадра* (Frame Status, FS).

Поле статуса FS имеет длину 1 байт и содержит 4 резервных бита и 2 подполя: *бит распознавания адреса A* и *бит копирования кадра C*. Так как это поле не сопровождается вычисляемой суммой CRC, то используемые биты для надежности дублируются: поле статуса FS имеет вид *ACxxACxx*. Если бит распознавания адреса не установлен во время получения кадра, это означает, что станция назначения больше не присутствует в сети (возможно, вследствие неполадок, а возможно, станция находится в другом кольце, связанном с данным с помощью моста). Если оба бита опознавания адреса и копирования кадра установлены и бит обнаружения ошибки также установлен, то исходная станция знает, что ошибка случилась после того, как этот кадр был корректно получен.

Кадр прерывающей последовательности состоит из двух байтов, содержащих начальный и конечный ограничители. Прерывающая последовательность может появиться в любом месте потока битов и сигнализирует о том, что текущая передача кадра или маркера отменяется.

6.2.3. Приоритетный доступ к кольцу

Станция может воспользоваться маркером, если только у нее есть кадры для передачи с приоритетом, равным или большим, чем приоритет маркера.

Каждый *кадр данных* или *маркер* имеет приоритет, устанавливаемый *битами приоритета* (значение от 0 до 7, причем 7 – наивысший приоритет).

Сетевой адаптер станции с кадрами, у которых приоритет ниже, чем приоритет маркера, не может захватить маркер, но может поместить наибольший приоритет своих ожидающих передачи кадров в резервные биты маркера, но только в том случае, если записанный в

резервных битах приоритет ниже его собственного. В результате в резервных битах приоритета устанавливается наивысший приоритет станции, которая пытается получить доступ к кольцу, но не может этого сделать из-за высокого приоритета маркера.

Станция, сумевшая *захватить маркер*, передает свои кадры с приоритетом маркера, а затем передает маркер следующему соседу. При этом она переписывает значение резервного приоритета в поле приоритета маркера, а резервный приоритет обнуляется. Поэтому при следующем проходе маркера по кольцу его захватит станция, имеющая наивысший приоритет.

При инициализации кольца основной и резервный приоритет маркера устанавливаются в 0. Хотя механизм приоритетов в технологии Token Ring имеется, но он начинает работать только в том случае, когда приложение или прикладной протокол решают его использовать. Иначе все станции будут иметь равные права доступа к кольцу, что в основном и происходит на практике, так как большая часть приложений этим механизмом не пользуется. Это связано с тем, что приоритеты кадров поддерживаются не во всех технологиях, например, в сетях Ethernet они отсутствуют, поэтому приложение будет вести себя по-разному, в зависимости от технологии нижнего уровня, что нежелательно.

В современных сетях приоритетность обработки кадров обычно обеспечивается коммутаторами или маршрутизаторами, которые поддерживают их независимо от используемых протоколов канального уровня.

6.2.4. Физический уровень технологии Token Ring

Стандарт Token Ring фирмы IBM изначально предусматривал построение связей в сети с помощью концентраторов, называемых MAU (Multistation Access Unit) или MSAU (Multi-Station Access Unit), т. е. устройств *многостанционного доступа*. Сеть Token Ring может включать до 260 узлов.

Концентратор Token Ring может быть *активным* или *пассивным*.

Пассивный концентратор просто соединяет порты внутренними связями так, чтобы станции, подключаемые к этим портам, образовывали кольцо.

Ни усиление сигналов, ни их *ресинхронизацию* пассивный MSAU не выполняет. Такое устройство можно считать простым кроссовым блоком за одним исключением – MSAU обеспечивает обход какого-либо порта, когда присоединенный к этому порту компьютер выключают. Такая функция необходима для обеспечения связности кольца вне зависимости от состояния подключенных компьютеров. Обычно обход порта выполняется за счет релейных схем, которые питаются постоянным током от сетевого адаптера, а при выключении сетевого адаптера нормально замкнутые контакты реле соединяют вход порта с его выходом.

Активный концентратор выполняет функции регенерации сигналов и поэтому иногда называется повторителем, как в стандарте Ethernet.

Возникает следующий вопрос: если концентратор является пассивным устройством, то каким образом обеспечивается качественная передача сигналов на большие расстояния, которые возникают при включении в сеть нескольких сот компьютеров? Ответ состоит в том, что роль усилителя сигналов в этом случае берет на себя каждый сетевой адаптер, а функцию ресинхронизирующего блока выполняет сетевой адаптер активного монитора кольца. Каждый сетевой адаптер Token Ring имеет блок повторения, который умеет регенерировать и ресинхронизировать сигналы, однако последнюю функцию выполняет в кольце только блок повторения активного монитора.

Блок ресинхронизации состоит из 30-битного буфера, который принимает манчестерские сигналы с несколько искаженными за время оборота по кольцу интервалами следования. При максимальном количестве станций в кольце (260) вариация задержки циркуляции бита по кольцу может достигать 3-битных интервалов. Активный монитор «вставляет» свой буфер в кольцо и синхронизирует битовые сигналы, выдавая их на выход с требуемой частотой.

В общем случае сеть Token Ring имеет комбинированную звездно-кольцевую конфигурацию. Конечные узлы подключаются к MSAU по топологии звезды, а сами MSAU объединяются через специальные порты Ring In (RI) и Ring Out (RO) для образования магистрального физического кольца.

Все станции в кольце должны работать на одной скорости – либо 4, либо 16 Мбит/с. Кабели, соединяющие станцию с концентратором, называются *ответвительными* (Lobe Cable), а кабели, соединяющие концентраторы – *магистральными* (Trunk Cable).

Технология Token Ring позволяет использовать для соединения конечных станций и концентраторов различные типы кабеля: STP Type I, UTP Type 3, UTP Type 6, а также волоконно-оптический кабель.

При использовании экранированной витой пары STP Type 1 из номенклатуры кабельной системы IBM в кольцо допускается объединять до 260 станций при длине ответвительных кабелей до 100 м, а при использовании неэкранированной витой пары максимальное количество станций сокращается до 72 при длине ответвительных кабелей до 45 м.

Расстояние между пассивными MSAU может достигать 100 м при использовании кабеля STP Type 1 и 45 м при использовании кабеля UTP Type 3. Между активными MSAU максимальное расстояние увеличивается соответственно до 730 или 365 м в зависимости от типа кабеля.

Максимальная длина кольца Token Ring составляет 4000 м. Ограничения на максимальную длину кольца и количество станций в кольце в технологии Token Ring не являются такими жесткими, как в технологии Ethernet. Здесь эти ограничения во многом связаны со временем оборота маркера по кольцу (но не только – есть и другие соображения, диктующие выбор ограничений). Так, если кольцо состоит из 260 станций, то при времени удержания в 10 мс маркер вернется в активный монитор в худшем случае через 2,6 с, а это время как раз составляет тайм-аут контроля оборота маркера. В принципе, все значения тайм-аутов в сетевых адаптерах узлов сети Token Ring можно настраивать, поэтому можно построить сеть Token Ring с большим количеством станций и с большей длиной кольца.

Существует большое разнообразие аппаратуры для сетей Token Ring, которое улучшает некоторые стандартные характеристики этих сетей: максимальную длину сети, расстояние между концентраторами, надежность (путем использования двойных колец).

Компания IBM предложила новый вариант технологии Token Ring, названный High-Speed Token Ring, HSTR. Эта технология поддерживает скорости в 100 и 155 Мбит/с, сохраняя основные особенности технологии Token Ring 16 Мбит/с.

Выводы

1. В сетях Token Ring используется маркерный метод доступа, который гарантирует каждой станции получение доступа к разделяемому кольцу в течение времени оборота маркера. Из-за этого свойства метод иногда называют *детерминированным*.

2. Метод доступа основан на приоритетах: от 0 (низший) до 7 (высший). Станция сама определяет приоритет текущего кадра и может захватить кольцо только в том случае, когда в нем нет более *приоритетных кадров*.

3. Современные сети Token Ring работают на скоростях до 1000 Мбит/с и могут использовать в качестве физической среды экранированную витую пару, неэкранированную витую пару, а также волоконно-оптический кабель. Максимальное количество станций в кольце – 260, а максимальная длина кольца – 4 км.

4. Технология Token Ring обладает элементами отказоустойчивости. За счет обратной связи кольца одна из станций – активный монитор – непрерывно контролирует наличие маркера, а также время оборота маркера и кадров данных. При некорректной работе кольца запускается процедура его повторной инициализации, а если она не помогает, то для локализации неисправного участка кабеля или неисправной станции используется процедура *beaconing* (*iBeacon* – API сервиса iOS, начиная с версии 7), которая позволяет передавать данные между беспроводными устройствами – маяками (*beacon*) – и устройствами, поддерживающими Bluetooth.

5. Максимальный размер поля данных кадра Token Ring зависит от скорости работы кольца. Для скорости 4 Мбит/с он равен около 5000 байтов, а при скорости 16 Мбит/с – около 16 Кбайт. Минимальный размер поля данных кадра не определен, т. е. может быть равен 0.

6. В сети Token Ring станции объединяют в кольцо с помощью концентраторов, которые называются MSAU. Пассивный концентратор MSAU выполняет роль кроссовой панели, соединяющей выход

предыдущей станции в кольце со входом последующей. Максимальное расстояние от станции до MSAU – 100 м для STP и 45 м для UTP.

7. Кольцо может быть построено на основе активного концентратора MSAU, который в этом случае называют повторителем.

8. Сеть Token Ring может строиться на основе нескольких колец, разделенных мостами, маршрутизирующими кадры по принципу «от источника», для чего в кадр Token Ring добавляется специальное поле с маршрутом прохождения колец.

6.3. Сети FDDI

6.3.1. Основные характеристики сетей FDDI

Сеть FDDI (Fiber Distributed Data Interface – *оптоволоконный распределенный интерфейс данных*) – это одна из последних разработок стандартов локальных сетей.

Стандарт FDDI, предложенный Американским национальным институтом стандартов ANSI (спецификация ANSI X3T9.5), изначально ориентировался на высокую скорость передачи (100 Мбит/с) и на применение перспективного оптоволоконного кабеля (длина волны света – 850 нм). Поэтому в данном случае разработчики не были стеснены рамками стандартов, ориентированных на низкие скорости и электрический кабель.

За основу стандарта FDDI был взят метод *маркерного доступа*, предусмотренный международным стандартом IEEE 802.5. Небольшие отличия от этого стандарта (упоминались выше) определяются необходимостью обеспечить высокую скорость передачи информации на большие расстояния.

Топология сети FDDI *– кольцо*, причем применяется два разнонаправленных оптоволоконных кабеля, что позволяет в принципе использовать полнодуплексную передачу информации с удвоенной эффективной скоростью в 200 Мбит/с (при этом каждый из двух каналов работает на скорости 100 Мбит/с). Применяется и *звездно-кольцевая* топология с концентраторами, включенными в кольцо.

Основные технические характеристики сети FDDI:

– максимальное количество абонентов сети – 1000;

– максимальная протяженность кольца сети – 20 км;

– максимальное расстояние между абонентами сети – 2 км;

– среда передачи – многомодовый оптоволоконный кабель (возможно применение электрической витой пары);

– метод доступа – маркерный;

– скорость передачи информации – 100 Мбит/с (200 Мбит/с – для дуплексного режима передачи).

Как видим, FDDI имеет некоторые преимущества по сравнению со всеми рассмотренными ранее сетями. Даже сеть Fast Ethernet, имеющая такую же пропускную способность (100 Мбит/с), не может сравниться с FDDI по допустимым размерам сети и допустимому количеству абонентов.

Маркерный метод доступа FDDI обеспечивает в отличие от CSMA/CD гарантированное время доступа и отсутствие конфликтов при любом уровне нагрузки.

Отметим, что ограничение на общую длину сети в 20 км связано не с затуханием сигналов в кабеле, а с необходимостью ограничения времени полного прохождения сигнала по кольцу для обеспечения предельно допустимого времени доступа. А вот максимальное расстояние между абонентами (2 км при многомодовом кабеле) определяется как раз затуханием сигналов в кабеле (оно не должно превышать 11 дБ). Предусмотрена также возможность применения одномодового кабеля, и в этом случае расстояние между абонентами может достигать 45 км, а полная длина кольца – 100 км.

Имеется и реализация FDDI на электрическом кабеле (CDDI – Copper Distributed Data Interface или TPDDI – Twisted Pair Distributed Data Interface). При этом используется кабель категории 5 с разъемами RJ-45. Максимальное расстояние между абонентами в этом случае должно быть не более 100 м. Стоимость оборудования сети на электрическом кабеле в несколько раз меньше. Но эта версия сети уже не имеет столь очевидных преимуществ перед своими конкурентами, как изначальная FDDI.

Для передачи данных в FDDI применяется код 4B/5B (табл. 6.1), специально разработанный для этого стандарта.

Таблица 6.1

Схема кода 4B/5B

Информация	Код 4B/5B
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100
1111	11101

Он обеспечивает скорость передачи 100 Мбит/с при пропускной способности кабеля 125 млн сигналов в секунду (или 125 МБод), а не 200 МБод, как в случае кода *Манчестер-II*. При этом каждым четырем битам передаваемой информации (каждому полубайту) ставится в соответствие 5 передаваемых по кабелю битов. Это позволяет приемнику восстанавливать синхронизацию приходящих данных один раз на четыре принятых бита, т. е. достигается компромисс между простейшим кодом NRZ (Non Return to Zero – без возвращения к нулю) и самосинхронизирующимся на каждом бите кодом *Манчестер-II*.

6.3.2. Структура сети FDDI

Стандарт FDDI для достижения высокой гибкости сети предусматривает включение в кольцо абонентов двух типов.

1. Абоненты (станции) класса A (они же *абоненты двойного подключения* – Dual-Attachment Stations, DAS) подключаются к обоим

(внутреннему и внешнему) кольцам сети. При этом реализуется возможность обмена со скоростью до 200 Мбит/с или же возможность резервирования кабеля сети (при повреждении основного кабеля используется резервный). Аппаратура этого класса используется в самых критичных частях сети.

2. Абоненты (станции) класса B (они же *абоненты одинарного подключения* – Single-Attachment Stations, SAS) подключаются только к одному (внешнему) кольцу сети. Естественно, они могут быть более простыми и дешевыми, чем адаптеры класса A, но, к сожалению, не имеют их возможностей. В сеть данные адаптеры могут включаться только через концентратор или обходной коммутатор, отключающий их в случае аварии.

Кроме собственно абонентов (компьютеров, терминалов и т. д.) в сети используются ***связные концентраторы*** (Wiring Concentrators). Включение связных концентраторов позволяет собрать в одно место все точки подключения для того, чтобы контролировать работу сети, диагностировать неисправности и упрощать реконфигурацию. К тому же при применении кабелей разных типов (например, оптоволоконного кабеля и витой пары) концентратор выполняет функцию преобразования электрических сигналов в оптические и наоборот.

Концентраторы также бывают двойного (Dual-Attachment Concentrator, DAC) и одинарного (Single-Attachment Concentrator, SAC) подключения.

Пример простейшей конфигурации сети FDDI представлен на рис. 6.9.

FDDI включает четыре типа портов абонентов.

1. *Порт A* используется только для устройств двойного подключения, его вход присоединяется к первичному кольцу, а выход – ко вторичному.

2. *Порт B* применяется только для устройств двойного подключения, его вход подсоединяется ко вторичному кольцу, а выход – к первичному.

3. *Порт M* (Master – основной) используется для концентраторов и соединяет два концентратора между собой или концентратор с абонентом.

4. *Порт S* (Slave – подчиненный) определен только для устройств одинарного подключения и используется для соединения двух абонентов или абонента и концентратора.

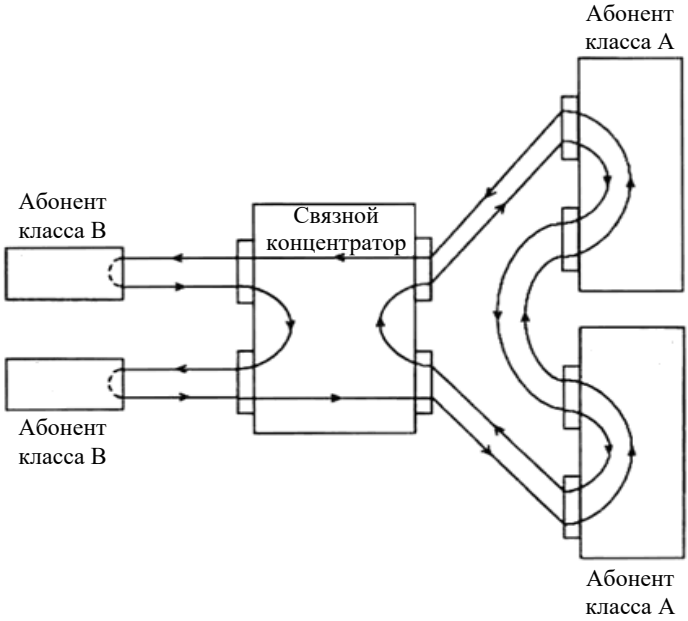


Рис. 6.9. Пример конфигурации сети FDDI

Стандарт FDDI предусматривает также возможность реконфигурации сети с целью сохранения ее работоспособности в случае повреждения кабеля.

В показанном на рис. 6.10 случае поврежденный участок кабеля исключается из кольца, но целостность сети вследствие перехода на одно кольцо вместо двух не нарушается (т. е. абоненты класса А начинают работать как абоненты класса В).

В отличие от метода доступа, который предлагается стандартом IEEE 802.5, в FDDI применяется так называемая *множественная передача маркера*. Если в случае сети Token Ring новый (свободный) маркер передается абонентом только после возвращения к нему его пакета, то в FDDI новый маркер передается абонентом сразу же после окончания передачи им пакета.

Последовательность действий здесь следующая. Абонент, желающий передавать информацию, ждет маркер, который «идет» за каждым пакетом. Когда маркер пришел, абонент удаляет его из сети и передает свой пакет. Сразу после передачи пакета абонент посылает сгенерированный маркер.

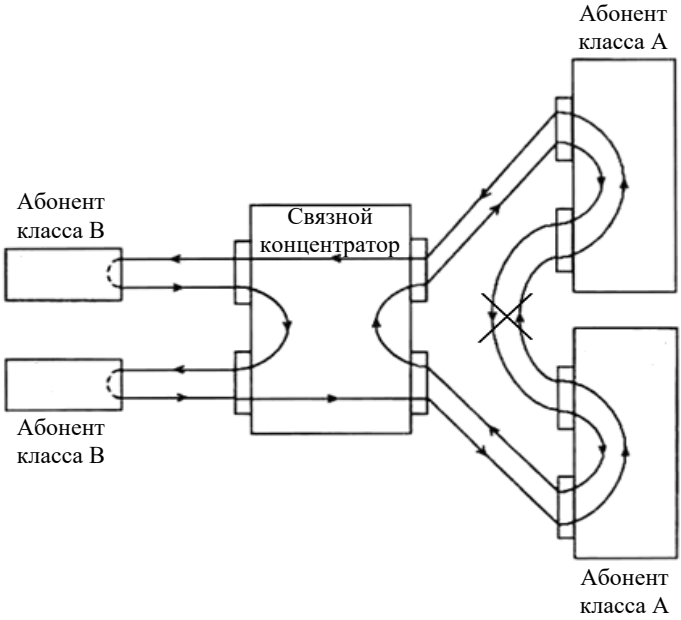


Рис. 6.10. Реконфигурация сети FDDI при повреждении кабеля

Одновременно каждый абонент ведет свой отсчет времени, сравнивая реальное *время обращения маркера* (Token Rotation Time, TRT) с заранее установленным *контрольным временем его прибытия* (Target Token Rotation Time, TTRT). Если маркер возвращается раньше, чем установлено TTRT, то делается вывод, что сеть загружена мало, и, следовательно, абонент может спокойно передавать всю свою информацию. Если же маркер возвращается позже, то сеть загружена сильно, и абонент может передавать только самую необходимую информацию. При этом величины контрольного времени TTRT могут устанавливаться различными для разных абонентов. Такой механизм позволяет абонентам гибко реагировать на загрузку сети и автоматически под-держивать ее на оптимальном уровне.

6.3.3. Структура пакета в сетях FDDI

Стандарт FDDI в отличие от стандарта IEEE 802.5 не предусматривает возможности установки *приоритетов пакетов* и *резервирования*. Вместо этого все абоненты разделяются на две группы: *асинхронные* и *синхронные*.

Асинхронные абоненты – это те, для которых время доступа к сети не слишком критично.

Синхронные – это те, для которых время доступа должно быть жестко ограничено. В стандарте предусмотрен специальный алгоритм, обслуживающий эти типы абонентов.

Форматы маркера (рис. 6.11) и пакета (рис. 6.12) сети FDDI несколько отличаются от форматов, используемых в сети Token Ring (на рис. 6.11 и 6.12 цифры означают количество байтов).

Рис. 6.11. Формат маркера FDDI

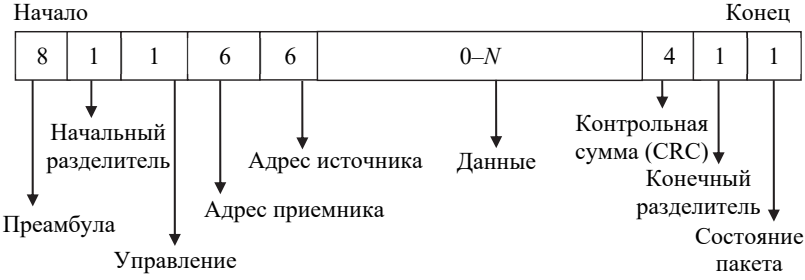


Рис. 6.12. Формат пакета FDDI

Общая длина пакета не может превышать 4500 байтов. Рассмотрим назначение каждого из полей.

1. *Преамбула* используется для синхронизации. Первоначально она содержит 64 бита, но абоненты, через которых проходит пакет, могут менять ее размер.

2. *Начальный разделитель* выполняет функцию признака начала кадра.

3. *Адреса приемника* и *источника* могут быть 6-байтными (аналогично Ethernet и Token Ring) или 2-байтными.

4. *Поле данных* может быть переменной длины, но суммарная длина пакета не должна превышать 4500 байтов.

5. *Поле контрольной суммы* содержит 32-битную циклическую контрольную сумму (CRC) пакета.

6. *Конечный разделитель* определяет конец кадра.

7. *Байт состояния пакета* включает в себя бит обнаружения ошибки, бит распознавания адреса и бит копирования (все аналогично Token Ring).

Формат байта управления сети FDDI, который представлен на рис. 6.13, имеет следующую структуру.

1. *Бит класса пакета* определяет, синхронный или асинхронный это пакет.

2. *Бит длины адреса* указывает, какой адрес (6-байтный или 2-байтный) используется в данном пакете.

3. *Поле формата кадра* показывает, управляющий это кадр или информационный.

4. *Поле типа кадра* определяет, к какому типу относится данный кадр.

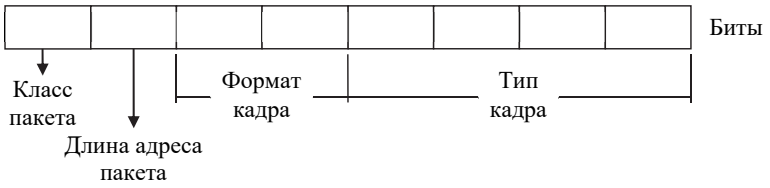


Рис. 6.13. Формат байта управления

В заключение отметим, что несмотря на очевидные преимущества сети FDDI, темпы ее распространения связаны главным образом со стоимостью аппаратуры (достаточно высока). Основная область применения FDDI – это *базовые*, *опорные* (Backbone) сети, объединяющие несколько сетей. Применяется FDDI и для соединения мощных рабочих станций или серверов, требующих высокоскоростного обмена.

Выводы

1. Технология FDDI первой использовала волоконно-оптический кабель в локальных сетях, а также работу на скорости 100 Мбит/с.

2. Существует значительная преемственность между технологиями Token Ring и FDDI: для обеих характерны кольцевая топология и маркерный метод доступа.

3. Технология FDDI является наиболее отказоустойчивой технологией локальных сетей. При однократных отказах кабельной системы или станции сеть за счет «сворачивания» двойного кольца в одинарное остается вполне работоспособной.

4. Маркерный метод доступа FDDI работает по-разному для *синхронных* и *асинхронных кадров* (тип кадра определяет станция). Для передачи первого станция всегда может захватить пришедший маркер на фиксированное время, для передачи второго маркер захватывается только тогда, когда он выполнил оборот по кольцу достаточно быстро, что говорит об отсутствии перегрузок кольца. Такой метод доступа предпочитает синхронные кадры и регулирует загрузку кольца, притормаживая передачу несрочных асинхронных кадров.

5. В качестве физической среды технология FDDI использует волоконно-оптические кабели и UTP категории 5 (этот вариант физического уровня называется TP-PMD).

6. Максимальное количество станций двойного подключения в кольце – 500, максимальный диаметр двойного кольца – 100 км. Максимальные расстояния между соседними узлами для многомодового кабеля равны 2 км, для витой пары UPT категории 5 – 100 м, а для одномодового оптоволокна зависят от его качества.

6.4. Сети 100VG-AnyLAN

6.4.1. Основные характеристики сетей 100VG-AnyLAN

Сеть 100VG-AnyLAN – это одна из разработок высокоскоростных локальных сетей фирм Hewlett-Packard и IBM. Соответствует стандарту IEEE 802.12, так что уровень ее стандартизации достаточно

высокий. Главными ее достоинствами являются большая скорость обмена, сравнительно невысокая стоимость аппаратуры, централизованный метод управления обменом без конфликтов и совместимость на уровне пакетов с популярными сетями Ethernet и Token Ring. В названии сети цифра 100 соответствует скорости 100 Мбит/с, буквы VG обозначают дешевую витую пару категории 3 (Voice Grade), а AnyLAN (любая сеть) обозначает то, что сеть совместима с двумя самыми распространенными сетями.

Основные технические характеристики сети 100VG-AnyLAN:

– скорость передачи – 100 Мбит/с;

– топология «звезда» с возможностью наращивания;

– метод доступа – централизованный, бесконфликтный (Demand Priority – с *запросом приоритета*);

– среда передачи – счетверенная неэкранированная витая пара (кабели UTP категории 3, 4 или 5), сдвоенная витая пара (кабель UTP категории 5), сдвоенная экранированная витая пара (STP), а также оптоволоконный кабель (сейчас в основном распространена счетверенная витая пара);

– максимальная длина кабеля между концентратором и абонентом и между концентраторами – 100 м (для кабеля UTP категории 3), 150 м (для кабеля UTP категории 5 и экранированного кабеля), 2 км (для оптоволоконного кабеля).

Таким образом, параметры сети 100VG-AnyLAN довольно близки к параметрам сети Fast Ethernet. Однако главное преимущество Fast Ethernet – это полная совместимость с наиболее распространенной сетью Ethernet (в случае 100VG-AnyLAN для этого обязательно требуется коммутатор или мост). В то же время централизованное управление 100VG-AnyLAN исключает конфликты и гарантирует предельную величину времени доступа (чего не предусмотрено в сети Ethernet).

6.4.2. Структура сети 100VG-AnyLAN

Пример структуры сети 100VG-AnyLAN показан на рис. 6.14.

Сеть 100VG-AnyLAN состоит из центрального (основного) концентратора уровня 1, к которому могут подключаться как отдельные

абоненты, так и концентраторы уровня 2, к которым, в свою очередь, подключаются абоненты и концентраторы уровня 3. При этом сеть может иметь не более трех таких уровней. Получается, что максимальный размер сети может составлять 600 м для *неэкранированной витой пары*.

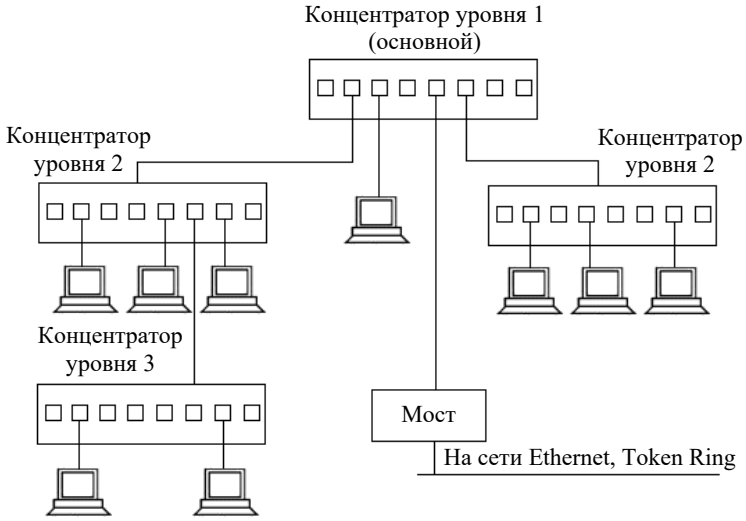


Рис. 6.14. Структура сети 100VG-AnyLAN

В отличие от неинтеллектуальных концентраторов других сетей (например, Ethernet), концентраторы сети l00VG-AnyLAN – это *интеллектуальные контроллеры*, которые управляют всем доступом к сети. Для этого они непрерывно контролируют запросы, поступающие на все порты. Концентраторы принимают все приходящие пакеты и отправляют их только тем абонентам, которым они адресованы. Однако никакой обработки информации они не производят, т. е. в данном случае получается все-таки не настоящая (активная) звезда, но и не пассивная звезда.

Каждый из концентраторов может быть настроен на работу с форматами пакетов Ethernet или Token Ring, при этом концентраторы всей сети должны работать с пакетами какого-нибудь одного формата. Для связи с сетями Ethernet и Token Ring необходимы мосты, но они довольно простые. Концентраторы имеют один порт верхнего уровня (для присоединения его к концентратору более высокого уровня) и

несколько портов нижнего уровня (для присоединения абонентов). В качестве абонента может выступать компьютер (рабочая станция), сервер, мост, маршрутизатор, коммутатор или другой концентратор.

Каждый порт концентратора может быть установлен в один из двух возможных режимов работы: *нормальный режим*, предполагающий пересылку абоненту, присоединенному к порту, только пакетов, адресованных лично ему, и *мониторный режим*, предполагающий пересылку абоненту, присоединенному к порту, всех пакетов, приходящих на концентратор. Этот режим позволяет одному из абонентов контролировать работу всей сети в целом (выполнять функцию мониторинга).

6.4.3. Метод доступа в сетях 100VG-AnyLAN

В сетях 100VG-AnyLAN используется метод доступа Demand Priority (описан в п. 3.2.2).

Каждый абонент, желающий передавать информацию, посылает концентратору свой запрос на передачу. Концентратор циклически прослушивает всех абонентов по очереди и дает право передачи абоненту, следующему по порядку за тем, который закончил передачу. Но этот простейший алгоритм усложнен в сети 100VG-AnyLAN, так как запросы могут иметь два уровня приоритета: *нормальный уровень приоритета*, используемый для обычных приложений, и *высокий уровень приоритета*, используемый для приложений, требующих быстрого обслуживания.

Запросы с высоким уровнем приоритета обслуживаются раньше, чем запросы с нормальным приоритетом. Если приходит запрос высокого приоритета, то нормальный порядок обслуживания прерывается, и после окончания приема текущего пакета обслуживается запрос высокого приоритета. Если таких высокоприоритетных запросов несколько, то возврат к нормальной процедуре обслуживания происходит только после полной обработки всех этих запросов. При этом концентратор следит за тем, чтобы не была превышена установленная величина гарантированного времени доступа. Если высокоприоритетных запросов слишком много, то запросы с нормальным приоритетом автоматически переводятся им в ранг высокоприоритетных. Таким образом, даже низкоприоритетные запросы не будут ждать своей очереди слишком долго.

Концентраторы более низких уровней также анализируют запросы абонентов, присоединенных к ним, и в случае необходимости пересылают их запросы к концентратору более высокого уровня. За один раз концентратор более низкого уровня может передать концентратору более высокого уровня не один пакет (как обычный абонент), а столько пакетов, сколько абонентов присоединено к нему. Так, для примера на рис. 6.15 в случае одновременного возникновения заявок на передачу у всех абонентов (компьютеров) порядок обслуживания будет такой: компьютер 1-2, затем 1-3, потом 2-1, 2-4, 2-8 и далее 1-6.

Однако так будет только при одинаковом (нормальном) приоритете всех запросов. Если же, например, от компьютеров 1-2, 2-4 и 2-8 поступят высокоприоритетные запросы, то порядок обслуживания будет таким: 1-2, 2-4, 2-8, 1-3, 2-1,1-6.

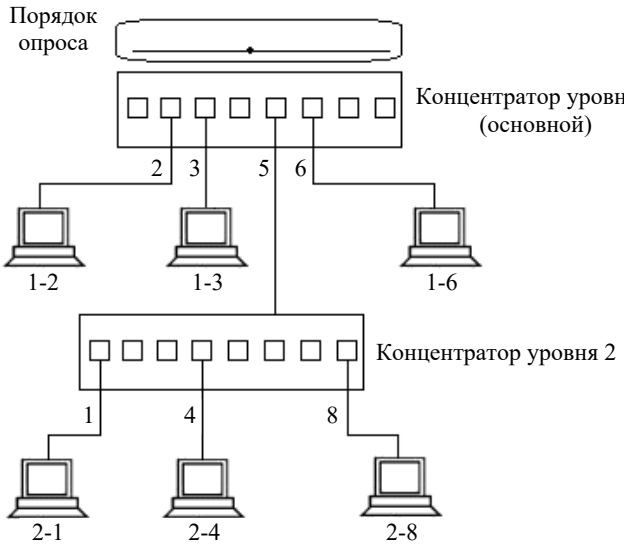


Рис. 6.15. Порядок обслуживания запросов абонентов на различных уровнях сети

Помимо собственно передачи пакетов и пересылки запросов на передачу, в сети применяется также *специальная процедура подготовки к связи* (Link Training). Во время данной процедуры концентратор и абоненты обмениваются между собой управляющими

пакетами. При этом проверяется правильность присоединения линий связи и их исправность. Одновременно концентратор получает информацию об особенностях абонентов, подключенных к нему, об их назначении и сетевых адресах. Запускается данная процедура самим абонентом при включении питания или после подключения к концентратору, а также автоматически при большом уровне ошибок.

6.4.4. Кодирование информации в сетях 100VG-AnyLAN

Проблема кодирования передаваемых данных решена в сети l00VG-AnyLAN достаточно интересно. Вся передаваемая информация проходит следующие этапы обработки:

1) *разделение на квинтеты* (группы по 5 битов);

2) *перемешивание*, *скремблирование* (Scrambling) полученных квинтетов;

3) *кодирование квинтетов* специальным кодом 5B6B (этот код обеспечивает в выходной последовательности не более трех единиц или нулей подряд, что используется для обнаружения ошибок);

4) *добавление начального и конечного разделителей кадра*.

Сформированные таким образом кадры передаются в 4 линии передачи (при использовании счетверенной витой пары). При сдвоенной витой паре и оптоволоконном кабеле применяется временное *мультиплексирование информации в каналах*.

В результате этих действий достигается **рандомизация сигналов** – выравнивание количества передаваемых единиц и нулей, снижение взаимовлияния кабелей друг на друга и самосинхронизация передаваемых сигналов без удвоения требуемой полосы пропускания, как в случае кода *Манчестер-II*.

При использовании *счетверенной витой пары* передача по каждой из четырех пар свитых проводов производится со скоростью 30 Мбит/с (рис. 6.16).

Суммарная скорость передачи составляет 120 Мбит/с. Однако полезная информация (вследствие использования специального кода 5B6B) передается при этом всего лишь со скоростью 100 Мбит/с. Таким образом, частота сигнала в кабеле должна быть не менее 15 МГц. Этому требованию удовлетворяет кабель с витыми парами категории 3.

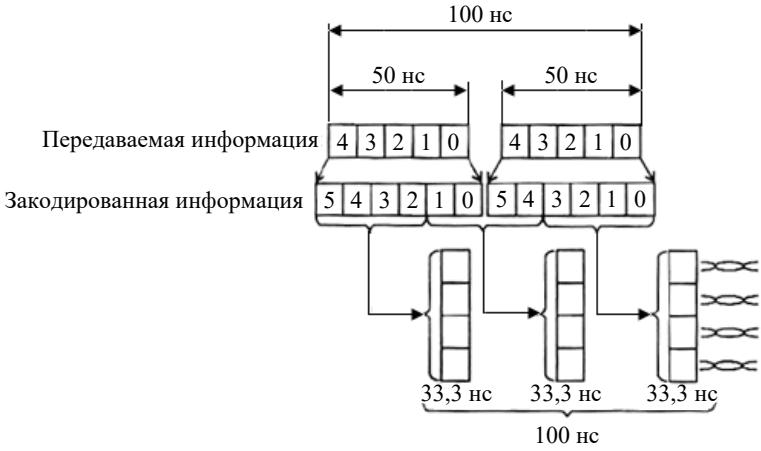


Рис. 6.16. Кодирование информации в сетях 100VG-AnyLAN

В сети 100VG-AnyLAN предусмотрены два режима обмена: ***полудуплексный*** и ***полнодуплексный*** (***дуплексный***).

При *полудуплексном обмене* все четыре витые пары используются для передачи одновременно в одном направлении (от абонента к концентратору или наоборот). Он используется для передачи пакетов.

При *полнодуплексном обмене* две витые пары передают в одном направлении, а две другие – в другом направлении. Он используется для передачи управляющих сигналов. Для управления используются два тональных сигнала. Первый из них представляет собой последовательность из 16 логических единиц и 16 логических нулей, следующих со скоростью 30 Мбит/с (в результате частота сигнала получается равной 0,9375 МГц). Второй тональный сигнал имеет вдвое большую частоту (1,875 МГц) и образуется чередованием восьми логических единиц и восьми логических нулей. Все управление сетью осуществляется комбинацией этих двух тональных сигналов.

В табл. 6.2 приведена расшифровка различных комбинаций этих сигналов, передаваемых абоненту и концентратору.

Когда ни у абонента, ни у концентратора нет информации для передачи, оба они посылают по обеим линиям первый тоновый сигнал (1-1). Если принимаемый концентратором пакет может быть адресован данному абоненту, ему посылается комбинация сигналов 1-2.

При этом абонент должен прекратить передачу управляющих сигналов концентратору и освободить эти две линии связи для пересылки информационных пакетов. Такая же комбинация 1-2, полученная концентратором, означает *запрос на передачу пакета с нормальным приоритетом*.

Запрос на передачу пакета с высоким приоритетом передается комбинацией 2-1. Наконец, комбинация 2-2 сообщает как абоненту, так и концентратору о необходимости перейти к процедуре подготовки к связи.

Таблица 6.2

Расшифровка комбинаций сигналов

Передаваемые сигналы	Расшифровка абонентом	Расшифровка концентратором
1-1	Нет информации для передачи	Нет информации для передачи
1-2	Концентратор принимает пакет	Запрос нормального приоритета
2-1	Зарезервировано	Запрос с высоким приоритетом
2-2	Запрос процедуры подготовки к связи	Запрос процедуры подготовки к связи

В целом сеть 100VG-AnyLAN представляет собой довольно доступное решение со скоростью передачи до 100 Мбит/с. Однако она не обладает полной совместимостью ни с одной из стандартных сетей, поэтому ее дальнейшая судьба проблематична. К тому же в отличие от сети FDDI она не имеет никаких рекордных параметров.

Выводы

1. Согласно стандарту IEEE 802.12, технология 100VG-AnyLAN сохраняла формат кадра Ethernet, но существенно изменяла метод доступа.

2. В технологии 100VG-AnyLAN арбитром, решающим вопрос о предоставлении станциям доступа к разделяемой среде, является концентратор, поддерживающий метод Demand Priority – приоритетные требования. Метод Demand Priority оперирует с двумя уровнями приоритетов, выставляемыми станциями, причем приоритет

станции, долго не получающей обслуживания, повышается динамически.

3. Концентраторы VG могут объединяться в иерархию, при этом порядок доступа к среде не зависит от уровня концентратора, к которому подключена станция, а зависит только от приоритета кадра и времени подачи заявки на обслуживание.

4. Технология 100VG-AnyLAN поддерживает кабель UTP категории 3, причем для обеспечения скорости 100 Мбит/с передает данные одновременно по 4 парам. Имеется также физический стандарт для кабеля UTP категории 5, кабеля STP Type 1 и волоконно-оптического кабеля.

Контрольные вопросы

1. Приведите основные характеристики сетей Ethernet.
2. Какой метод доступа используется в сетях Ethernet?
3. Приведите стуктуру пакета сетей Ethernet.
4. Какова минимальная длина кадра в сетях Ethernet?
5. Приведите основные характеристики сетей Token Ring.
6. Опишите используемую топологию в сетях Token Ring.
7. Какие функции выполняет концентратор в сетях Token Ring?
8. Какие форматы кадров существуют в сетях Token Ring?
9. Опишите процедуру приорететного доступа к кольцу в сетях Token Ring.
10. Приведите основные характеристики сетей FDDI.
11. Какой метод доступа используется в сетях FDDI?
12. Опишите используемую топологию в сетях FDDI.
13. Приведите стуктуру пакета сетей FDDI.
14. Опишите стуктуру сети FDDI.
15. Назовите основные характеристики сетей 100VG-AnyLAN.
16. Какой метод доступа используется в сетях 100VG-AnyLAN?
17. Охарактеризуйте процедуру кодирования информации в сетях 100VG-AnyLAN.
18. Опишите стуктуру сети 100VG-AnyLAN.
19. Какой тип кабеля может быть использован в сетях 100VG-AnyLAN?

7

ФИЗИЧЕСКАЯ СРЕДА ПЕРЕДАЧИ ДАННЫХ

Физическая среда является основой, на которой строятся физические средства соединения. Сопряжение с физическими средствами соединения посредством физической среды обеспечивает *физический уровень*.

В качестве физической среды широко используются эфир, металлы, оптическое стекло и кварц. Среда передачи данных может включать как кабельные, так и беспроводные технологии. Хотя физические кабели являются наиболее распространенными носителями для сетевых коммуникаций, беспроводные технологии все больше внедряются благодаря их способности связывать глобальные сети.

На этом уровне для физических кабелей определяются механические и электрические (оптические) свойства среды передачи, которые включают:

– тип кабелей и разъемов;
– разводку контактов в разъемах;
– схему кодирования сигналов для значений 0 и 1.

Канальный уровень определяет доступ к среде и управление передачей посредством процедуры передачи данных по каналу. В локальных сетях протоколы канального уровня используются компьютерами, мостами, коммутаторами и маршрутизаторами. В компьютерах функции канального уровня реализуются совместными усилиями сетевых адаптеров и их драйверов.

7.1. Кабели, линии и каналы связи

Для организации связи в сетях используются следующие понятия:
– *кабели связи*;
– *линии связи*;
– *каналы связи*.

Кабель связи – это длинномерное изделие электротехнической промышленности. Из кабелей связи и других элементов (монтаж, крепеж, кожухи и т. д.) строят ***линии связи*** между узлами сети.

Прокладка линии внутри – задача достаточно серьезная. Длина линий связи колеблется от десятков метров до десятков тысяч километров. В любую более-менее серьезную линию связи кроме кабелей входят траншеи, колодцы, муфты, переходы через реки, моря и океаны, а также грозозащита (равно, как и другие виды защиты) линий. Большую сложность представляют собой юридические вопросы, включающие согласование прокладки линий связи, особенно в городе.

При наличии кабелей связи создаются линии связи, а уже по линиям связи создаются ***каналы связи***. Линии и каналы связи заводятся на *узлы связи*. Линии, каналы и узлы образуют ***первичные сети связи***.

7.2. Кабельные системы

7.2.1. Типы кабелей и структурированные кабельные системы

В качестве среды передачи данных используются различные виды кабелей: *коаксиальный*, *кабели на основе экранированной* и *неэкранированной витой пары* (они относятся к классу электрических), а также *оптоволоконный кабель*.

Наиболее популярным видом среды передачи данных на небольшие расстояния (до 100 м) является *неэкранированная витая пара*, которая включена практически во все современные стандарты и технологии локальных сетей и обеспечивает пропускную способность до 100 Мбит/с (на кабелях категории 5) и выше. Также отметим, что *оптоволоконный кабель* применяется как для построения локальных связей, так и для образования магистралей глобальных сетей. Оптоволоконный кабель может обеспечить очень высокую пропускную способность канала (до нескольких десятков гигабит в секунду) и передачу на значительные расстояния (до нескольких десятков километров без промежуточного усиления сигнала).

В качестве среды передачи данных в вычислительных сетях (*беспроводные сети* – Wireless Networks или Wi-Fi – Wireless Fidelity –

беспроводное соответствие) используются также электромагнитные волны различных частот – *КВ* (короткие волны – диапазон радиоволн с частотой от 3 до 30 МГц), *УКВ* (ультракороткие волны – от 30 МГц до 3000 ГГц) или *СВЧ* (сверхвысокие частоты – от 300 МГц до 300 ГГц. Для построения глобальных каналов этот вид среды передачи данных используется широко – на нем построены спутниковые каналы связи и наземные радиорелейные каналы, работающие в зонах прямой видимости в СВЧ-диапазонах. Об основных особенностях технологии Wi-Fi речь пойдет ниже (см. подглавы 7.4, 9.5 и 11.2).

Очень важно правильно построить фундамент сети – ***кабельную систему***. В последнее время в качестве такой надежной основы все чаще используется структурированная кабельная система.

! **Структурированная кабельная система** (Structured Cabling System, SCS) – это набор коммутационных элементов (кабелей, разъемов, коннекторов, кроссовых панелей и шкафов), а также методика их совместного использования, которая позволяет создавать регулярные, легко расширяемые структуры связей в вычислительных сетях.

Преимущества структурированной кабельной системы

1. *Универсальность*. Структурированная кабельная система при продуманной организации может стать единой средой для передачи компьютерных данных в локальной вычислительной сети.

2. *Увеличение срока службы*. Срок старения хорошо структурированной кабельной системы может составлять 8–10 лет.

3. *Уменьшение стоимости добавления новых пользователей и изменения их мест размещения*. Стоимость кабельной системы в основном определяется не стоимостью кабеля, а стоимостью работ по его прокладке.

4. *Возможность легкого расширения сети*. Структурированная кабельная система является модульной, поэтому ее легко наращивать, позволяя легко и ценой малых затрат переходить на более совершенное оборудование, удовлетворяющее растущим требованиям к системам коммуникаций.

5. *Обеспечение более эффективного обслуживания*. Структурированная кабельная система облегчает обслуживание и поиск неисправностей.

6. *Надежность*. Структурированная кабельная система имеет повышенную надежность, поскольку обычно производство всех ее компонентов и техническое сопровождение осуществляется одной фирмой-производителем.

7.2.2. Стандарты кабелей

Кабель – это достаточно сложное изделие, состоящее из проводников, слоев экрана и изоляции.

Обычно кабели присоединяются к оборудованию с помощью разъемов. Кроме того, для обеспечения быстрой перекоммутации кабелей и оборудования используются различные электромеханические устройства, называемые ***кроссовыми секциями***, ***кроссовыми коробками*** или ***шкафами***.

В компьютерных сетях применяются кабели, удовлетворяющие определенным стандартам, что позволяет строить кабельную систему сети из кабелей и соединительных устройств разных производителей. Сегодня наиболее употребительными в мировой практике являются следующие стандарты.

1. Американский стандарт EIA/TIA-568A, который был разработан совместными усилиями нескольких организаций: ANSI, EIA/TIA и лабораторией Underwriters Labs (UL). Стандарт EIA/TIA-568A разработан на основе предыдущей версии стандарта EIA/TIA-568 и дополнений к этому стандарту TSB-36 и TSB-40A).
2. Международный стандарт ISO/IEC 11801.
3. Европейский стандарт EN50173.

Эти стандарты близки между собой и по многим позициям предъявляют к кабелям идентичные требования. Однако есть и различия между этими стандартами, например, в международные стандарты ISO11801 и европейский EN50173 вошли некоторые типы кабелей, которые отсутствуют в стандарте EIA/TIA-568A.

До появления стандарта EIA/TIA большую роль играл американский стандарт системы категорий кабелей Underwriters Labs, разработанный совместно с компанией Anixter. Позже этот стандарт вошел в стандарт EIA/TIA-568.

Кроме этих открытых стандартов, многие компании в свое время разработали свои фирменные стандарты, из которых до сих пор имеет практическое значение только один – стандарт компании IBM.

При стандартизации кабелей принят протокольно-независимый подход. Это означает, что в стандарте оговариваются электрические, оптические и механические характеристики, которым должен удовлетворять тип кабеля или соединительного изделия – разъема, кроссовой коробки и т. п. Однако для какого протокола предназначен данный кабель, стандарт не оговаривает. Поэтому нельзя приобрести кабель для протокола Ethernet или FDDI, нужно просто знать, какие типы стандартных кабелей поддерживают протоколы Ethernet и FDDI.

В ранних версиях стандартов определялись только характеристики кабелей, без соединителей. В последних версиях стандартов появились требования к соединительным элементам (документы TSB-36 и TSB-40A, вошедшие затем в стандарт 568A), а также к линиям (каналам), представляющим типовую сборку элементов кабельной системы, состоящую из шнура от рабочей станции до розетки, самой розетки, основного кабеля (длиной до 90 м для витой пары), точки перехода (например, еще одной розетки или жесткого кроссового соединения) и шнура до активного оборудования, например концентратора или коммутатора.

Мы остановимся только на основных требованиях к самим кабелям, не рассматривая характеристик соединительных элементов и собранных линий.

В стандартах кабелей оговаривается достаточно много характеристик, из которых наиболее важные перечислены ниже.

1. *Затухание* (Attenuation). Затухание измеряется в децибелах на метр для определенной частоты или диапазона частот сигнала (напоминание: увеличение/уменьшение величины тока или напряжения на 1 дБ означает соответственно их увеличение/уменьшение в 1,122 раза).

2. *Перекрестные наводки на ближнем конце* (Near End Cross Talk, NECT). Измеряются в децибелах для определенной частоты сигнала.

3. *Импеданс* (Impedance – волновое сопротивление) – это полное (активное и реактивное) сопротивление в электрической цепи. Импеданс измеряется в омах (Ом) и является относительно постоянной величиной для кабельных систем (например, для коаксиальных кабелей, используемых в стандартах Ethernet, импеданс кабеля должен составлять 50 Ом). Для неэкранированной витой пары наиболее часто используемые значения импеданса – 100 и 120 Ом. В области высоких частот (100–200 МГц) импеданс зависит от частоты.

4. *Активное сопротивление* – это сопротивление постоянному току в электрической цепи. Активное сопротивление, в отличие от

импеданса, не зависит от частоты и возрастает с увеличением длины кабеля.

5. *Емкость* – это свойство металлических проводников накапливать энергию. Два электрических проводника в кабеле, разделенные диэлектриком, представляют собой конденсатор, способный накапливать заряд. Емкость является нежелательной величиной, поэтому следует стремиться к тому, чтобы она была как можно меньше (иногда применяют термин «паразитная емкость»). Высокое значение емкости в кабеле приводит к искажению сигнала и ограничивает полосу пропускания линии.

6. *Уровень внешнего электромагнитного излучения*, или *электрический шум.* **Электрический шум** – нежелательное переменное напряжение в проводнике. Он бывает двух типов: ***фоновый*** и ***импульсный***.

Электрический шум можно также разделить на низко-, средне- и высокочастотный. Источниками *фонового электрического шума* в диапазоне до 150 кГц являются линии электропередачи, телефоны и лампы дневного света; в диапазоне от 150 кГц до 20 МГц – компьютеры, принтеры, ксероксы; в диапазоне от 20 МГц до 1 ГГц – телевизионные и радиопередатчики, микроволновые печи. Основными источниками импульсного электрического шума являются моторы, переключатели и сварочные агрегаты. Электрический шум измеряется в милливольтах.

7. *Диаметр*, или *площадь сечения проводника*. Для медных проводников достаточно употребительной является американская система AWG (American Wire Gauge), которая вводит некоторые условные типы проводников, например 22 AWG, 24 AWG, 26 AWG. Чем больше номер типа проводника, тем меньше его диаметр. В вычислительных сетях наиболее употребительными являются типы проводников, приведенные выше в качестве примеров. В европейских и международных стандартах диаметр проводника указывается в миллиметрах.

Основное внимание в современных стандартах уделяется кабелям на основе витой пары и волоконно-оптическим кабелям.

7.2.3. Кабель «витая пара»

Витой парой (Twisted Pair) называется кабель, в котором изолированная пара проводников скручена с небольшим числом витков на единицу длины (рис. 7.1).

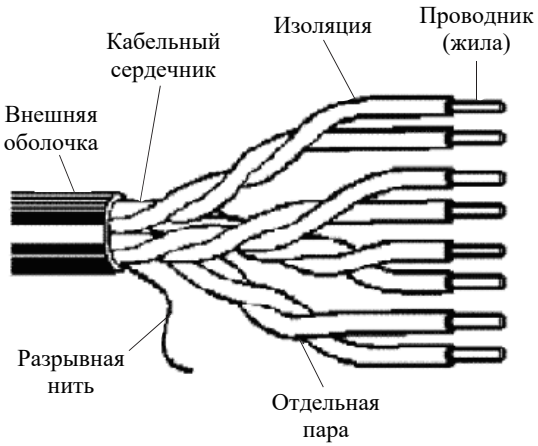


Рис. 7.1. Структура кабеля типа «витая пара»

Скручивание проводов уменьшает электрические помехи извне при распространении сигналов по кабелю, а экранированные витые пары еще более увеличивают степень помехозащищенности сигналов.

Кабель типа «витая пара» используется во многих сетевых технологиях, включая Ethernet, ARCNet и IBM Token Ring.

Кабели «витая пара» подразделяются на *неэкранированные* (Unshielded Twisted Pair, UTP) и *экранированные* медные кабели. Последние подразделяются на две разновидности: с *экранированием каждой пары* и *общим экраном* (Shielded Twisted Pair, STP) и *с одним только общим экраном* (Foiled Twisted Pair, FTP) (рис. 7.2).

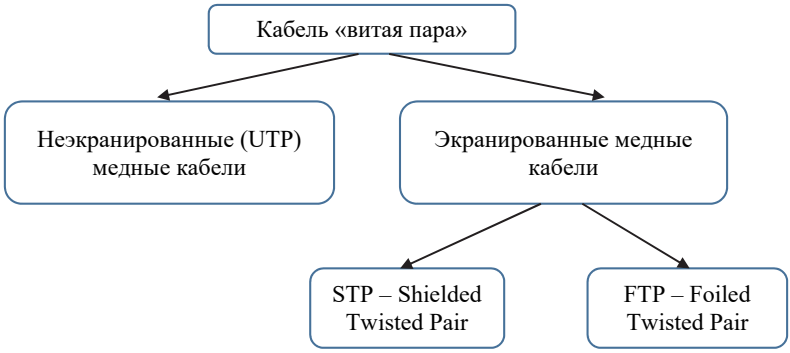


Рис. 7.2. Типы кабелей «витая пара»

Наличие или отсутствие экрана у кабеля говорит не о защите передаваемых данных, а о различных подходах к подавлению помех. Отсутствие экрана делает неэкранированные кабели более гибкими и устойчивыми к изломам. Кроме того, они не требуют дорогостоящего контура заземления для эксплуатации в нормальном режиме, как экранированные. Неэкранированные кабели идеально подходят для прокладки в помещениях внутри офисов, а экранированные лучше использовать для установки в местах с особыми условиями эксплуатации, например рядом с очень сильными источниками электромагнитных излучений, которых в офисах обычно нет.

Кабели на основе неэкранированной витой пары. Медный неэкранированный кабель UTP в зависимости от электрических и механических характеристик разделяется на 5 категорий (Category 1 – Category 5). Кабели категорий 1 и 2 были определены в стандарте EIA/TIA-568 (Electronic Industries Alliance, EIA – альянс отраслей электронной промышленности США), но в стандарт 568A уже не вошли как устаревшие.

Кабели категории 1 применяются там, где требования к скорости передачи минимальны. Обычно это кабель для цифровой и аналоговой передачи голоса и низкоскоростной (до 20 Кбит/с) передачи данных. До 1983 г. это был основной тип кабеля для телефонной разводки.

Кабели категории 2 были впервые применены фирмой IBM при построении собственной кабельной системы. Главное требование к ним – способность передавать сигналы со спектром до 1 МГц.

Кабели категории 3 были стандартизованы в 1991 г., когда был разработан *Стандарт телекоммуникационных кабельных систем для коммерческих зданий* (EIA-568), на основе которого затем был создан действующий стандарт EIA-568A. Данный стандарт определил электрические характеристики кабелей категории 3 для частот в диапазоне до 16 МГц, поддерживающих, таким образом, высокоскоростные сетевые приложения. Кабель категории 3 предназначен как для передачи данных, так и для передачи голоса. Шаг скрутки проводов равен примерно 3 витка на 1 фут (30,5 см). Кабели категории 3 сейчас составляют основу многих кабельных систем зданий, в которых они используются для передачи и голоса, и данных.

Кабели категории 4 представляют собой несколько улучшенный вариант кабелей категории 3. Кабели категории 4 обязаны выдерживать тесты на частоте передачи сигнала 20 МГц и обеспечивать повышенную помехоустойчивость и низкие потери сигнала. Кабели

категории 4 хорошо подходят для применения в системах с увеличенными расстояниями (до 135 м) и в сетях Token Ring с пропускной способностью 16 Мбит/с. На практике используются редко.

Кабели категории 5 были разработаны для поддержки высокоскоростных протоколов, поэтому характеризуются частотой до 100 МГц. Большинство новых высокоскоростных стандартов ориентируются на использование витой пары категории 5. На этом кабеле работают протоколы со скоростью передачи данных 100 Мбит/с – FDDI (с физическим стандартом TP-PMD), Fast Ethernet, 100VG-AnyLAN, а также более скоростные протоколы – ATM на скорости 155 Мбит/с и Gigabit Ethernet на скорости 1000 Мбит/с (вариант Gigabit Ethernet на витой паре категории 5 стал стандартом в июне 1999 г.). Кабель категории 5 пришел на замену кабелю категории 3, и сегодня многие новые кабельные системы крупных зданий строятся именно на этом типе кабеля (в сочетании с волоконно-оптическим).

Кабели CAT5 выпускаются в 4-парном исполнении. Каждая из четырех пар кабеля имеет определенный цвет и шаг скрутки. Обычно две пары предназначены для передачи данных, а две – для передачи голоса. Во время эксплуатации двух пар, скорость передачи будет 100 Мбит/с. Частотная полоса – 100 МГц. CAT5e – наиболее используемый вид кабеля, насчитывает четыре пары, применяется при конструировании сетей 100/1000 Мбит/с. Во время задействования двух пар, скорость передачи составляет 100 Мбит/с, а если задействованы все четыре пары – 1000 Мбит/с. Частотная полоса – 100 МГц.

CAT6 – находит применение в сетях Fast Ethernet (100 Мбит/с), Gigabit Ethernet (1000 Мбит/с), передает сигнал на скорости до 10 Гбит/с. Частотная полоса – 250 МГц. Также существует подкатегория CAT6a, которая характеризуется частотной полосой до 500 МГц и обеспечивает скорость передачи до 10 Гбит/с на расстоянии 100 м.

В кабелях CAT7 во время работы на частоте до 600 МГц скорость передачи доходит до 10 Гбит/с. Максимальная длина передачи сигнала обеспечивается тем, что кабели категории 7 обязательно экранируются, причем используется двойное экранирование, т. е. как каждая пара, так и весь кабель в целом. Также известна подкатегория CAT7a, характеризующаяся частотой до 1200 МГц и скоростью передачи до 40 Гбит/с при условии использования кабеля длиной не более 50 м, а также скоростью передачи до 100 Гбит/с на расстоянии до 15 м.

В табл. 7.1 представлены обобщенные характеристики кабеля типа «витая пара» в зависимости от категории.

Таблица 7.1

Категории кабеля «витая пара»

Категория кабеля	Полоса частот, МГц, не более	Скорость передачи данных	Тип кабеля	Назначение и конструкция
CAT1	0,1	–	UTP	Передача речевого сигнала, телефонный кабель типа «лапша»
CAT2	1	4 Мбит/с	UTP	2 пары проводников, сейчас не применяется
CAT3	16	10 Мбит/с на расстояние до 100 м	UTP	4-парный кабель для телефонных и локальных сетей
CAT4	20	16 Мбит/с	UTP	4-парный кабель, сейчас не применяется. Применялся в сетях Token Ring
CAT5	100	100 Мбит/с при использовании 2 пар 1000 Мбит/с при использовании 4 пар	UTP	4-парный кабель для телефонных и локальных сетей
CAT5e	125	100 Мбит/с при использовании 2 пар 1000 Мбит/с при использовании 4 пар на расстояние до 100 м 10 000 Мбит/с при использовании 4 пар на расстояние до 55 м	UTP	4-парный кабель для компьютерных сетей, представляющий доработанный вариант CAT5. Обеспечивает аналогичную скорость при меньших габаритах
CAT6	250	1000 Мбит/с при использовании 4 пар 10 000 Мбит/с – до 50 м	UTP	4-парный кабель для компьютерных сетей
CAT6a	500	До 10 Гбит/с на расстояние 100 м	UTP	4-парный кабель высокоскоростных линий Интернета
CAT7	600	До 10 Гбит/с	S/FTP	4-парный кабель высокоскоростных линий Интернета
CAT7a	1200	До 40 Гбит/с на расстояние до 50 м До 100 Гбит/с на расстояние до 15 м	S/FTP	4-парный кабель высокоскоростных линий Интернета

Для соединения кабелей с оборудованием используются вилки и розетки RJ-45, представляющие 8-контактные разъемы, похожие на обычные телефонные разъемы RJ-11.

Кабели на основе экранированной витой пары. Экранированная витая пара STP хорошо защищает передаваемые сигналы от внешних помех, а также меньше излучает электромагнитных колебаний, что защищает пользователей сетей от вредного для здоровья излучения.

Наличие заземляемого экрана удорожает кабель и усложняет его прокладку, так как требует выполнения качественного заземления. Экранированный кабель применяется только для передачи данных, а голос по нему не передают.

Основным стандартом, определяющим параметры экранированной витой пары, является фирменный стандарт IBM. В этом стандарте кабели делятся не на категории, а на типы: *Type 1*, *Type 2*, ..., *Type 9*.

Основным типом экранированного кабеля является кабель Type 1 стандарта IBM. Он состоит из двух пар скрученных проводов, экранированных проводящей оплеткой, которая заземляется. Электрические параметры кабеля Type 1 примерно соответствуют параметрам кабеля UTP категории 5. Некоторые стандарты поддерживают кабель STP Type 1, например, 100VG-AnyLAN, а также Fast Ethernet (хотя основным типом кабеля для Fast Ethernet является UTP категории 5). В случае если технология может использовать UTP и STP, нужно убедиться, на какой тип кабеля рассчитаны приобретаемые трансиверы. Сегодня кабель STP Type 1 включен в стандарты EIA/TIA-568A, ISO 11801 и EN 50173, т. е. приобрел международный статус.

Экранированные витые пары используются также в кабеле IBM Type 2, который представляет кабель Type 1 с добавленными двумя парами неэкранированного провода для передачи голоса.

Для присоединения экранированных кабелей к оборудованию используются разъемы конструкции IBM.

7.2.4. Коаксиальные кабели

Коаксиальные кабели (рис. 7.3) используются в радио- и телевизионной аппаратуре.

Коаксиальные кабели могут передавать данные со скоростью 10 Мбит/с на расстояние до 500 м. Они разделяются на *толстые* и *тонкие* – в зависимости от толщины.

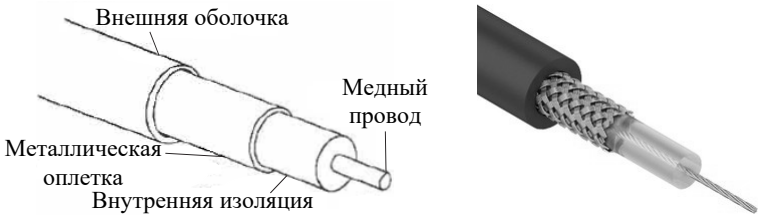


Рис. 7.3. Структура коаксиального кабеля

Типы коаксиальных кабелей приведены в табл. 7.2.

Таблица 7.2

Типы коаксиальных кабелей

Тип	Название, значение сопротивления
RG-8 и RG-11	Thicknet, 50 Ом
RG-58 U	Thinnet, 50 Ом, сплошной центральный медный проводник
RG-58 A/U	Thinnet, 50 Ом, центральный многожильный проводник
RG-59	Broadband/Cable television (широковещательное и кабельное телевидение), 75 Ом
RG-59 U	Broadband/Cable television (широковещательное и кабельное телевидение), 50 Ом
RG-62	ARCNet, 93 Ом

Кабель Thinnet, известный как кабель RG-58, является наиболее широко используемым физическим носителем данных. Сети при этом не требуют дополнительного оборудования и являются простыми и недорогими. Хотя *тонкий коаксиальный кабель* (Thin Ethernet) позволяет осуществлять передачу на меньшее расстояние, чем толстый, для соединений с тонким кабелем применяются стандартные байонетные разъемы BNC типа CP-50, и ввиду его небольшой стоимости он становится фактически стандартным для офисных ЛВС. Используется в технологии Ethernet 10Base2, описанной ниже.

Толстый коаксиальный кабель (Thick Ethernet) имеет большую степень помехозащищенности, большую механическую прочность, но требует специального приспособления для прокалывания кабеля, чтобы создать ответвления для подключения к ЛВС. Он более дорогой и менее гибкий, чем тонкий. Используется в технологии Ethernet 10Base5, описанной ниже. Сети ARCNet с передачей маркера обычно используют кабель RG-62 A/U.

Рассмотрим основные параметры систем на основе коаксиальных кабелей.

1. Характеристики спецификации 10Base2:
– тонкий коаксиальный кабель;
– характеристики кабеля: диаметр 0,2 дюйма, RG-58 A/U, 50 Ом;
– приемлемые разъемы – BNC;
– максимальная длина сегмента – 185 м;
– минимальное расстояние между узлами – 0,5 м;
– максимальное число узлов в сегменте – 30.
2. Характеристики спецификации 10Base5:
– толстый коаксиальный кабель;
– волновое сопротивление – 50 Ом;
– максимальная длина сегмента – 500 м;
– минимальное расстояние между узлами – 2,5 м;
– максимальное число узлов в сегменте – 100.

7.2.5. Оптоволоконный кабель. Общие принципы

Волоконно-оптические линии связи – это вид связи, при котором информация передается по оптическим диэлектрическим волноводам, известным под названием *оптическое волокно*.

Оптическое волокно в настоящее время считается самой совершенной физической средой для передачи информации, а также самой перспективной средой для передачи больших потоков информации на значительные расстояния.

Основания так считать вытекают из ряда особенностей, присущих оптическим волноводам.

Физические особенности. Широкополосность оптических сигналов обусловлена чрезвычайно высокой частотой несущей (F_0 = = 1014 Гц). Это означает, что по оптической линии связи можно передавать информацию со скоростью порядка 1000 Мбит/с. Говоря другими словами, по одному волокну можно передать одновременно 10 млн телефонных разговоров и миллион видеосигналов. Скорость передачи данных может быть увеличена за счет передачи информации сразу в двух направлениях, так как световые волны могут распространяться в одном волокне независимо друг от друга. На сегодняшний день предел по плотности передаваемой информации по оптическому волокну не достигнут.

Очень малое (по сравнению с другими средами) затухание светового сигнала в волокне. Лучшие образцы российского волокна имеют затухание 0,22 дБ/км на длине волны 1,55 мкм, что позволяет строить линии связи длиной до 100 км без *регенерации сигналов* (промежуточного усиления). Для сравнения, лучшее волокно Sumitomo на длине волны 1,55 мкм имеет затухание 0,154 дБ/км. В лабораториях разрабатываются еще более «прозрачные», так называемые *фторцирконатные волокна* с теоретическим пределом порядка 0,02 дБ/км на длине волны 2,5 мкм. Лабораторные исследования показали, что на основе таких волокон могут быть созданы линии связи с регенерационными участками через 4600 км при скорости передачи порядка 1 Гбит/с.

Технические особенности и преимущества оптических волокон. Волокно изготовлено из кварца, основу которого составляет двуокись кремния, широко распространенного, а потому недорогого материала, в отличие от меди.

Оптические волокна имеют диаметр около 100 мкм, т. е. очень компактны и легки, что делает их перспективными для использования в авиации, приборостроении, кабельной технике.

Применяя особо прочный пластик, на кабельных заводах изготавливают самонесущие подвесные кабели, не содержащие металла и тем самым безопасные в электрическом отношении. Такие кабели можно монтировать на мачтах существующих линий электропередач, как отдельно, так и встроенными в фазовый провод, экономя значительные средства на прокладку кабеля через реки и другие преграды.

Системы связи на основе оптических волокон устойчивы к электромагнитным помехам, а передаваемая по световодам информация защищена от несанкционированного доступа. Волоконно-оптические линии связи нельзя прослушивать, не разрушив поверхность канала. Всякие воздействия на волокно могут быть зарегистрированы методом мониторинга (непрерывного контроля) целостности линии. Важное свойство оптического волокна – долговечность. Время жизни волокна, т. е. сохранение им своих свойств в определенных пределах, превышает 25 лет, что позволяет проложить оптико-волоконный кабель один раз и по мере необходимости наращивать пропускную способность канала путем замены приемников и передатчиков на более быстродействующие.

Недостатки волоконной технологии. При создании линии связи требуются высоконадежные активные элементы, преобразующие

электрические сигналы в свет и свет в электрические сигналы. Необходимы также оптические коннекторы (соединители) с малыми оптическими потерями и большим ресурсом на подключение-отключение. Точность изготовления таких элементов линии связи должна соответствовать длине волны излучения, т. е. погрешности должны быть порядка доли микрона. Поэтому производство таких компонентов оптических линий связи очень дорогостоящее. Другой недостаток заключается в том, что для монтажа оптических волокон требуется прецизионное (высокоточное), а потому дорогое технологическое оборудование. Как следствие, при аварии (обрыве) оптического кабеля затраты на восстановление выше, чем при работе с медными кабелями.

Преимущества от применения волоконно-оптических линий связи (ВОЛС) настолько значительны, что, несмотря на перечисленные недостатки оптического волокна, эти линии связи все шире используются для передачи информации.

Структура оптоволокна. Волоконно-оптический кабель состоит из тонких (5–60 мкм) волокон, по которым распространяются световые сигналы. Это наиболее качественный тип кабеля – он обеспечивает передачу данных с очень высокой скоростью и лучше других типов передающей среды обеспечивает защиту данных от внешних помех.

Волоконно-оптические кабели состоят из *центрального проводника света* (сердцевины) – *стеклянного волокна*, окруженного другим слоем стекла – оболочкой, обладающей меньшим показателем преломления, чем сердцевина (рис. 7.4). Распространяясь по сердцевине, лучи света не выходят за ее пределы, отражаясь от покрывающего слоя оболочки.

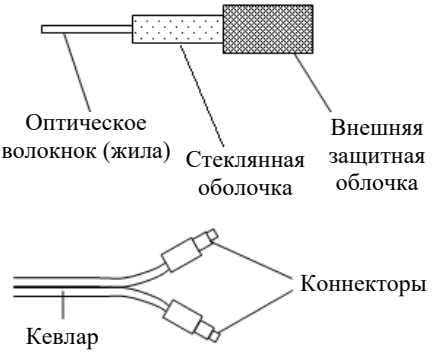


Рис. 7.4. Структура оптоволокна и оптоволоконного кабеля

В зависимости от распределения показателя преломления и величины диаметра сердечника различают:

– *многомодовое волокно со ступенчатым изменением показателя преломления* (рис. 7.5, *а*);

– *многомодовое волокно с плавным изменением показателя преломления* (рис. 7.5, *б*);

– *одномодовое волокно* (рис. 7.5, *в*).

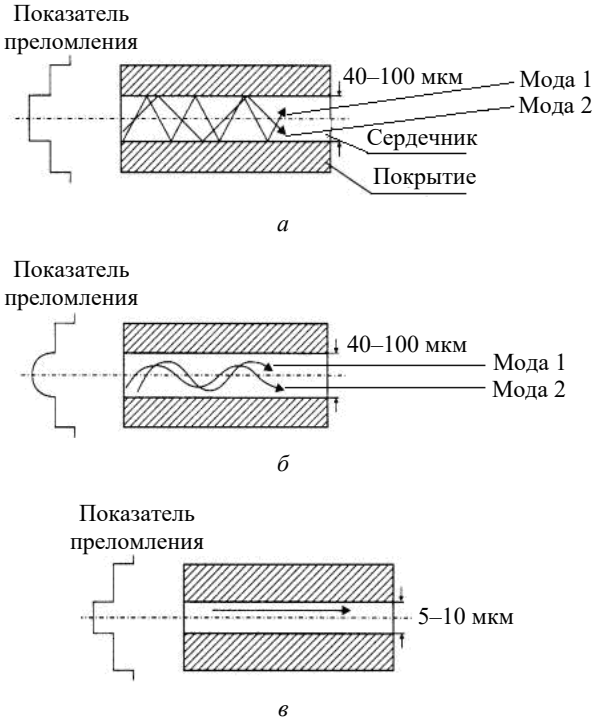


Рис. 7.5. Типы оптоволоконного кабеля:
а – многомодовое волокно со ступенчатым изменением показателя преломления; *б* – многомодовое волокно с плавным изменением показателя преломления; *в* – одномодовое оптоволокно

Понятие «мода» описывает режим распространения световых лучей во внутреннем сердечнике кабеля.

В *одномодовом кабеле* (Single Mode Fiber, SMF) используется центральный проводник очень малого диаметра, соизмеримого с

длиной волны света от 5 до 10 мкм. При этом практически все лучи света распространяются вдоль оптической оси световода, не отражаясь от внешнего проводника.

В *многомодовых кабелях* (Multi Mode Fiber, MMF) используются более широкие внутренние сердечники, которые легче изготовить технологически. В стандартах определены два наиболее употребительных многомодовых кабеля: 62,5/125 и 50/125 мкм, где 62,5 или 50 мкм – это диаметр центрального проводника, а 125 мкм – диаметр внешнего проводника.

В многомодовых кабелях во внутреннем проводнике одновременно существует несколько световых лучей, отражающихся от внешнего проводника под разными углами. Угол отражения луча называется модой луча.

Параметры оптоволоконных кабелей. Оба типа волокна характеризуются двумя важнейшими параметрами: *затуханием* и *дисперсией*.

Затухание – параметр, определяемый потерями на поглощение и на рассеяние излучения в оптическом волокне (измеряется в децибелах на километр (дБ/км)). Потери на поглощение зависят от чистоты материала, потери на рассеяние – от неоднородностей показателя преломления материала и т. д.

Затухание также зависит от длины волны излучения, вводимого в волокно. В настоящее время передачу сигналов по волокну осуществляют в трех диапазонах: 0,85, 1,3 и 1,55 мкм, так как именно в этих диапазонах кварц имеет повышенную прозрачность.

Дисперсия – это рассеяние во времени спектральных и модовых составляющих оптического сигнала.

7.2.6. Виды оптоволоконных кабелей

В сравнении с витой парой, которая вне зависимости от места применения имеет примерно одну и ту же конструкцию, оптоволоконные кабели связи могут иметь значительные отличия исходя из сферы применения и места укладки.

Можно выделить следующие основные виды оптоволоконных кабелей для передачи данных исходя из области применения:

– для прокладки внутри зданий;

– для кабельной канализации небронированный;

– для кабельной канализации бронированный;
– для укладки в грунт;
– подвесной самонесущий;
– с тросом;
– подводный.

Наиболее простой конструкцией обладают кабели для прокладки внутри зданий и канализационный небронированный, а самыми сложными – для прокладки в землю и подводные.

Кабель для прокладки внутри зданий. Оптические кабели для прокладки внутри зданий (рис. 7.6) разделяют на *распределительные*, из которых формируется сеть в целом, и *абонентские*, которые используются непосредственно для прокладки по помещению к конечному потребителю.

Как и витую пару, оптику прокладывают в кабельных лотках, кабель-каналах, а некоторые марки могут быть протянуты и по внешним фасадам зданий.

Рис. 7.6. Оптоволоконный кабель
для прокладки внутри зданий

Конструкция оптоволоконных кабелей для прокладки в зданиях включает в себя оптическое волокно, защитное покрытие и центральный силовой элемент, например пучок арамидных нитей. К оптоволокну, которое прокладывается в помещениях, есть особые требования по противопожарной безопасности, такие как нераспространение горения и низкое дымовыделение, поэтому в качестве оболочки для них используется не полиэтилен, а полиуретан.

Другие требования – это низкая масса кабеля, гибкость и небольшой размер. По этой причине многие модели имеют облегченную конструкцию, иногда с дополнительной защитой от влаги. Так как протяженность оптики внутри зданий обычно невелика, то и затухание сигнала незначительно, и влияние на передачу данных оно не оказывает. Число оптических волокон в таких кабелях не превышает двенадцати.

Также существует и оптоволоконный кабель, который содержит в себе дополнительно еще и витую пару.

Небронированный канализационный кабель. Небронированное оптоволокно (рис. 7.7) используется для укладки в канализации при условии, что на нее не будет внешних механических воздействий. Также подобный кабель прокладывается в тоннелях, коллекторах и зданиях.

Рис. 7.7. Небронированный оптоволоконный кабель

Но даже в случаях отсутствия внешнего воздействия на кабель в канализации, его могут укладывать в защитные полиэтиленовые трубы. Характерной особенностью данного типа оптоволоконного кабеля можно назвать наличие гидрофобного наполнителя, который гарантирует возможность эксплуатации в условиях канализации и дает некоторую защиту от влаги.

Бронированный канализационный кабель. Бронированные оптоволоконные кабели (рис. 7.8) используются при наличии больших внешних нагрузок, в особенности на растяжение.

Рис. 7.8. Бронированный оптоволоконный кабель

Бронирование может быть различным, ленточным или проволочным, последнее подразделяется на одно- и двухповивное. Кабели с ленточным бронированием используются в менее агрессивных условиях, например при прокладке в кабельной канализации, трубах, тоннелях, на мостах. Ленточное бронирование представляет собой стальную гладкую или гофрированную трубку толщиной 0,15–0,25 мм. Гофрирование при условии, что это единственный слой защиты кабеля, является предпочтительным, так как оберегает оптоволокно от грызунов и в целом повышает гибкость

кабеля. При более суровых условиях эксплуатации, например при закладке в грунт или на дно рек, используются кабели с проволочной броней.

Кабель для укладки в грунт. Для прокладки в грунт используют оптические кабели (рис. 7.9) с проволочной одноповивной или двухповивной броней. Также применяются и усиленные кабели с ленточным бронированием, но значительно реже. Прокладка оптического кабеля осуществляется в траншею или с помощью кабелеукладчиков.

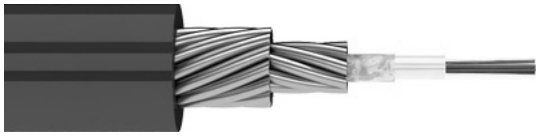


Рис. 7.9. Бронированный оптоволоконный кабель для укладки в грунт

В условиях влажного грунта используется модель кабеля, оптоволоконная часть которого заключена в герметичную металлическую трубку, а бронеповивы проволоки пропитаны специальным водоотталкивающим компаундом. Инженеры, работающие на укладке кабеля, не должны допускать превышения растягивающих и сдавливающих нагрузок сверх допустимых. В противном случае сразу или со временем могут быть повреждены оптические волокна, что приведет кабель в негодность.

Броня влияет и на значение допустимого усилия на растяжение. Оптоволоконные кабели с двухповивной броней могут выдержать усилие от 80 кН, одноповивные – от 7 до 20 кН, а ленточная броня гарантирует «выживание» кабеля при нагрузке не менее 2,7 кН (напомним: кН – килоньютон – единица силы в системе СИ; 1 Н – величина, определяемая как сила, изменяющая за 1 с скорость тела массой 1 кг на 1 м/с в направлении действия силы; таким образом, 1 Н = 1 кг·м/с^2).

Подвесной самонесущий кабель. Подвесные самонесущие кабели (рис. 7.10) монтируются на уже существующих опорах воздушных линий связи и высоковольтных ЛЭП.

Подвесные самонесущие кабели имеют стандартную круглую форму, благодаря которой снижаются ветровые нагрузки на конструкцию, а расстояние пролета между опорами может достигать ста и

более метров. В конструкции самонесущих подвесных оптических кабелей обязательно присутствует центральный силовой элемент, изготовленный из стеклопластика или арамидных нитей. Благодаря последним оптоволоконный кабель выдерживает высокие продольные нагрузки. Подвесные самонесущие кабели с арамидным нитями используют в пролетах до одного километра. Еще одно преимущество арамидных нитей, кроме их прочности и малом весе, заключается в том, что арамид по природе своей является диэлектриком, т. е. кабели, изготовленные на его основе безопасны, например при попадании молнии.

Рис. 7.10. Подвесной самонесущий кабель

В зависимости от строения сердечника различают несколько типов подвесного кабеля:

– *кабель с профилированным сердечником* – содержит оптические волокна или модули с этими волокнами – кабель устойчив к растяжению и сдавливанию;

– *кабель со скрученными модулями* – содержит оптические волокна, свободно уложенные, кабель устойчив к растяжениям;

– *кабель с одним оптическим модулем* – сердечник данного типа кабеля не имеет силовых элементов, поскольку они находятся в оболочке. Такие кабели обладают недостатком, связанным с неудобством идентификации волокон. Тем не менее они обладают меньшим диаметром и более доступной ценой.

Оптический кабель с тросом. Оптические кабели с тросом (рис. 7.11) – это разновидность самонесущих кабелей, которые также используются для воздушной прокладки.

Рис. 7.11. Подвесной оптоволоконный кабель с тросом

В таком изделии трос может быть несущим и навивным. Еще существуют модели, в которых оптика встроена в грозозащитный трос.

Усиление оптического кабеля тросом (профилированным сердечником) считается достаточно эффективным методом. Сам трос представляет собой стальную проволоку, заключенную в отдельную оболочку, которая в свою очередь соединяется с оболочкой кабеля. Свободное пространство между ними заполняется гидрофобным заполнителем.

Часто такую конструкцию оптического кабеля с тросом называют «восьмеркой» из-за внешнего сходства. «Восьмерки» применяют для прокладки воздушных линий связи с пролетом не более 50–70 м. В эксплуатации подобных кабелей есть некоторые ограничения, например, «восьмерку» со стальным тросом нельзя подвешивать на ЛЭП.

Но кабели с навивным грозозащитным тросом (грозотросом) спокойно монтируются на высоковольтных ЛЭП, крепясь при этом к проводу заземления.

Подводный оптический кабель. Данный тип оптических кабелей (рис. 7.12) стоит особняком, так как прокладывается в принципиально иных условиях. Почти все типы подводных кабелей так или иначе бронированы, а степень бронирования уже зависит от рельефа дна и глубины залегания.

Рис. 7.12. Подводный оптоволоконный кабель

Различают следующие основные типы подводных кабелей (по типу бронирования):

– *не бронирован*;

– *одинарное* (одноповивное) *бронирование*;

– *усиленное* (одноповивное) *бронирование*;

– *усиленное стальное* (двухповивное) *бронирование*.

Общая схема бронированных кабелей выглядит следующим образом (рис. 7.13).

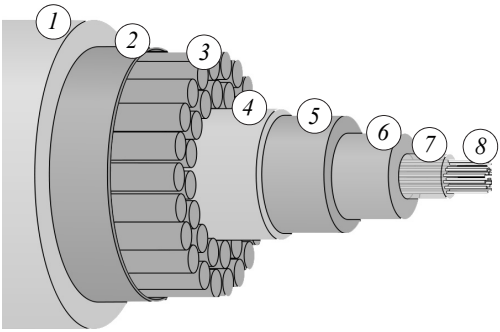


Рис. 7.13. Схема бронирования подводного оптоволоконного кабеля:
1 – полиэтиленовая изоляция; *2* – майларовое покрытие;
3 – двухповивное бронирование стальной проволокой; *4* – алюминиевая гидроизоляционная трубка; *5* – поликарбонат; *6* – центральная медная или алюминиевая трубка; *7* – внутримодульный гидрофобный заполнитель;
8 – оптические волокна

Как не парадоксально, прямой корреляции бронирования кабеля с глубиной залегания нет, так как армирование защищает оптику не от высоких давлений на глубине, а от воздействия морских обитателей, а также сетей, тралов и якорей рыболовецких судов.

7.3. Параметры кабельных систем Ethernet

7.3.1. Параметры систем на основе неэкранированной витой пары

Неэкранированная витая пара (Unshielded Twisted Pair, UTP) – это кабель из четырех скрученных пар проводов без защитного экрана.

Характеристики кабеля*:*

– диаметр проводников 0,4–0,6 мм (22~26 AWG), 4 скрученных пары: 8 проводников, из которых для 10Base-T, 100Base-TX, 1000Base-TX используют одну, две или четыре пары (кабель должен иметь категорию 3, 5 или 6 и качество *data grade* или выше);

– максимальная длина сегмента 100 м;

– разъемы восьмиконтактные RJ-45 (рис. 7.14).

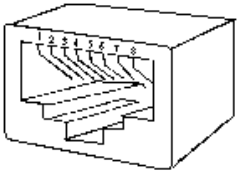


Рис. 7.14. Восьмиконтактные разъемы RJ-45

В табл. 7.3 приведены сигналы (спецификация 100Base-T), соответствующие номерам контактов разъема RJ-45 (для спецификации 1000Base-TX используются все восемь контактов).

Таблица 7.3

Сигналы, соответствующие номерам контактов разъема RJ-45

Тип	Каскадирование	Нормальный режим
1	RD+ (прием)	TD+ (передача)
2	RD– (прием)	TD– (передача)
3	TD+ (передача)	RD+ (прием)
4	Не используется	Не используется
5	Не используется	Не используется
6	TD– (передача)	RD– (прием)
7	Не используется	Не используется
8	Не используется	Не используется

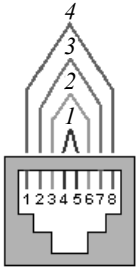
7.3.2. Стандартные разводки кабеля типа «витая пара»

В настоящее время наиболее популярны две схемы – T568A и T568B (рис. 7.15). Они идентичны в случае, если не используются вторая и третья пары.

T568A T568B

Рис. 7.15. Схемы разводки витой пары

Предпочтительна первая схема, поскольку она совместима с *однопарной* и *двупарной конфигурацией системы* USOC (Universal Service Ordering Code, рис. 7.16).



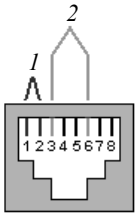
USOC (4 пары)

Рис. 7.16. Двупарная конфигурация системы USOC

Однако обе схемы могут использоваться для линий ISDN (Integrated Servises Digital Network), а также в высокоскоростных сетях. Дело в том, что схемы разработаны таким образом, чтобы свести к минимуму взаимные наводки в парах. А это необходимое условие для категорий 3, 4, 5, 5e и 6. Поэтому при реализации высокоскоростных сетей используют именно эти конфигурации.

7.3.3. Реализация сетевых топологий на основе стандартной разводки

Топология сети 10Base-T реализуется с помощью 8-пинового разъема по схожей схеме с T568A и T568B, однако на контакты выводятся другие пары (рис. 7.17).

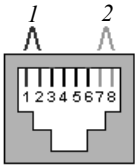


10 BASE-T (IEEE-802.3)

Рис. 7.17. Схема восьмипинового разъема для сетей 10BaseT

Если все же пользоваться стандартами T568, то в случае первой пары (по версии 10Base-T) необходимо использовать 3/2-ю пару разводки T568A/T568B, а в качестве второй 2/3-ю пару T568A/T568B.

Как и 10Base-T, ATM и TP-PMD (Twisted Pair-Physical Medium Dependent) реализуются только на 8-пиновом разъеме с использованием двух пар, и точно также схема схожа со стандартными разводками T568A и T568B (рис. 7.18).



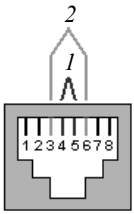
TP-PMD (IEEE 802.3) ATM

Рис. 7.18. Схема восьмипинового разъема для стандарта TP-PMD

В данной разводке в случае первой пары (по версии 10Base-T) необходимо использовать 3/2-ю пару разводки T568A/T568B, а в качестве второй 4-ю пару T568A или T568B.

Еще одна разводка, косвенно совместимая с T568A/B, а также и со схемой USOC – Token Ring (рис. 7.19).

Она строится на двух парах, занимающих центральные контакты. Причем Token Ring может сразу строиться на основе схемы T568A и USOC без каких-либо модификаций. В случае же использования T568B необходимо в качестве второй пары применять 3-ю. MMJ – частный стандарт для оборудования DEC, реализуется на 6-пиновом модифицированном разъеме. Разводка не совместима ни с USOC, ни с T568A/B. Первая пара выводится на 2-й и 3-й контакты, вторая – на 4-й и 5-й, а третья пара занимает внешние, 1-й и 6-й, пины.

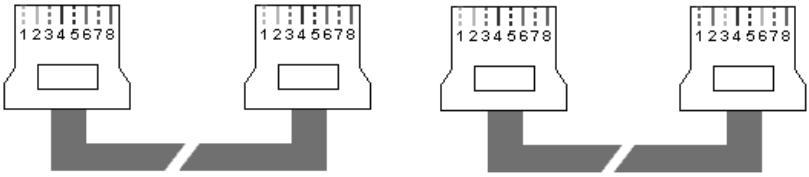


Token Ring (IEEE 802.5)

Рис. 7.19. Схема восьмипинового разъема для сетей Token Ring

7.3.4. Кросс-разводка кабеля типа «витая пара»

Термин *кросс-разводка* используется применительно к разводке пар в *патч-кордах*. Всего существует две базовые кросс-разводки – *прямая* и *перекрестная* (рис. 7.20).



Прямая кросс-разводка Перекрестная кросс-разводка

Рис. 7.20. Схема прямой
и перекрестной кросс-разводки

Их названия говорят сами за себя. В первом случае каждый проводник выводится строго на один и тот же контакт разъемов с обоих концов кабеля. Таким образом, 1-й контакт с одного конца соединен с 1-м на другом, 2-й со 2-м и так все пины. Патч-корды с подобной разводкой используются в кроссировочных узлах. При соединении с сетью оконечного оборудования, будь то персональный компьютер, факс и т. п., применяются патч-корды с перекрестной кросс-разводкой. Конкретная схема перекрестной кросс-разводки зависит от конкретной реализуемой сети, в частности, один из стандартов предполагает образование перекрестной разводки за счет перемены местами четырех пинов. Первый меняется с третьим, а второй с шестым.

7.4. Беспроводные технологии передачи данных

Методы *беспроводной* (Wireless) *технологии* передачи данных являются удобным, а иногда незаменимым средством связи. Беспроводные технологии различаются по типам сигнала, частоте (большая частота означает большую скорость передачи) и расстоянию передачи. Большое значение имеют помехи и стоимость.

Можно выделить три основных типа беспроводной технологии:
– *радиосвязь*;
– *связь в микроволновом диапазоне*;
– *инфракрасная связь*.

Технологии радиосвязи основаны на передаче данных на радиочастотах и практически не имеют ограничений по дальности. Радиосвязь используется для соединения локальных сетей на больших географических расстояниях.

Радиопередача в целом имеет высокую стоимость и чувствительна к электронному и атмосферному наложению, а также подвержена перехватам, поэтому требует шифрования для обеспечения уровня безопасности.

Передача данных в микроволновом диапазоне (Microwaves) использует высокие частоты и применяется как на коротких, так и на больших расстояниях. Главное ограничение заключается в том, чтобы передатчик и приемник были в зоне прямой видимости.

Применяется в местах, где использование физического носителя затруднено. Передача данных в микроволновом диапазоне при использовании спутников может быть очень дорогой.

Инфракрасные технологии (Infrared Transmission), функционируют на очень высоких частотах, приближающихся к частотам видимого света. Они могут быть использованы для установления двусторонней или широковещательной передачи на близких расстояниях.

При инфракрасной связи обычно используют светодиоды (LED – Light Emitting Diode) для передачи инфракрасных волн приемнику. Инфракрасная передача ограничена малым расстоянием в прямой зоне видимости и может быть использована в офисных зданиях.

В настоящее время наибольшее распространение получила так называемая ***Wi-Fi-связь***, базирующаяся на стандарте IEEE 802.11.

Wi-Fi-сеть (Wireless Local Area Network, WLAN) – это радиосеть, позволяющая передавать информацию между объектами по радиоволнам (без проводов). Разработкой стандартов в этой области занимается Wi-Fi Alliance.

WLAN-сети имеют ряд преимуществ перед обычными кабельными сетями:

– WLAN-сеть можно очень быстро развернуть, что очень удобно при проведении презентаций или в условиях работы вне офиса;

– пользователи мобильных устройств при подключении к локальным беспроводным сетям могут легко перемещаться в рамках действующих зон сети;

– современные сети обладают довольно высокой скоростью (до 300 Мб/с), что позволяет использовать их для очень широкого спектра задач;

– с помощью дополнительного оборудования беспроводная сеть может быть успешно соединена с кабельными сетями;

– WLAN-сеть может оказаться единственным выходом, если невозможна прокладка кабеля для обычной сети.

Несмотря на все достоинства, WLAN-сети обладают рядом недостатков, главный из которых – возможность легкого ***перехвата данных и взлома сети***.

7.4.1. Требования к беспроводным локальным сетям

Беспроводные сети должны удовлетворять некоторым требованиям, типичным для всех локальных сетей, в том числе: высокая пропускная способность, возможность охвата небольших расстояний, связность подключенных станций и возможность широковещания. Кроме того, существует набор требований, характерных только для беспроводных локальных сетей. Перечислим важнейшие из них.

1. *Производительность*. Протокол управления доступом к среде должен максимально эффективно использовать беспроводную среду для максимизации пропускной способности.

2. *Число узлов*. От беспроводных локальных сетей может требоваться поддержка сотен узлов из множества ячеек.

3. *Соединение с магистральной локальной сетью*. В большинстве случаев требуется взаимосвязь со станциями магистральной локальной сети. Для беспроводных локальных сетей, имеющих внутреннюю инфраструктуру, это требование легко удовлетворяется посредством использования модулей управления, присоединяемых к локальным сетям обоих типов. Может также понадобиться специальное помещение для мобильных пользователей и организация эпизодических беспроводных сетей.

4. *Обслуживаемая область*. Типичная сфера охвата беспроводной локальной сети имеет диаметр 100–300 м.

5. *Потребление питания от батарей*. Мобильные сотрудники используют рабочие станции с питанием от батарей, потребление которого не должно быть большим при использовании беспроводных адаптеров. Это делает неприменимым протокол MAC, требующий, чтобы мобильные узлы постоянно следили за точками доступа или часто связывались с основной станцией.

6. *Устойчивость передачи и безопасность*. Беспроводные сети, если они разработаны неправильно, могут быть подвержены интерференции (наложение сигналов) и легко прослушиваться. Структура беспроводной локальной сети должна обеспечивать надежную передачу даже в обстановке шума, а также некоторый уровень зашиты от прослушивания.

7. *Совместная работа в сети*. С ростом популярности беспроводных сетей повысилась вероятность того, что две или более сетей будут работать в одной области или в нескольких областях, допускающих интерференцию разных локальных сетей. Такая интерференция может мешать нормальной работе алгоритма MAC и способствовать несанкционированному доступу к отдельной локальной сети.

8. *Работа без лицензии*. Пользователи желали бы приобретать продукты рынка беспроводных локальных сетей и работать с ними на нелицензируемой полосе частот.

9. *Переключение/роуминг*. Протокол MAC, используемый в беспроводных локальных сетях, должен позволять мобильным станциям перемещаться из одной ячейки в другую.

10. *Динамическая конфигурация*. MAC-адресация и сетевое управление локальной сети должны обеспечивать динамическое и автоматическое добавление, удаление и передислокацию конечных систем, не причиняя неудобств другим пользователям.

7.4.2. Стандарты беспроводных сетей IEEE 802.11

Существует несколько разновидностей WLAN-сетей, которые различаются схемой организации сигнала, скоростями передачи данных, радиусом охвата сети, а также характеристиками радиопередатчиков и приемных устройств, параметрами передачи (шифрование, кодирование и т. д.), методами взаимодействия оборудования.

Стандарт Wi-Fi 802.11a. Стандарт был принят в 1999 г., тем не менее нашел свое применение только с 2001 г. Данный стандарт используется в основном в США и Японии. В России и Европе он не получил широкого распространения.

В стандарте 802.11a применяется схема модуляции сигнала – *мультиплексирование с разделением по ортогональным частотам* (Orthogonal Frequency Division Multiplexing, OFDM). Основной поток данных разделяется на несколько параллельных субпотоков с относительно низкой скоростью передачи, и затем для их модуляции применяется соответствующее число несущих. Стандартом определены три обязательные скорости передачи данных (6, 12 и 24 Мбит/с) и пять дополнительных (9, 18, 24, 48 и 54 Мбит/с). Также имеется возможность одновременного использования двух каналов, что повышает скорость передачи данных в 2 раза.

Стандарт Wi-Fi 802.11b. Стандарт основан на методе *широкополосной модуляции с прямым расширением спектра* (Direct Sequence Spread Spectrum, DSSS). Весь рабочий диапазон делится на 14 каналов, разнесенных на 25 МГц для исключения взаимных помех. Данные передаются по одному из этих каналов без переключения на другие. Возможно одновременное использование всего 3 каналов. Скорость передачи данных может автоматически меняться в зависимости от уровня помех и расстояния между передатчиком и приемником.

Стандарт IEEE 802.11b реализует максимальную теоретическую скорость передачи 11 Мбит/с, что сравнимо с кабельной сетью 10 BaseT Ethernet. Следует учитывать, что такая скорость возможна при передаче данных одним WLAN-устройством. Если в среде одновременно функционирует большее число абонентских станций, то полоса пропускания распределяется между всеми и скорость передачи данных на одного пользователя падает.

Стандарт Wi-Fi 802.11g. Стандарт окончательно был утвержден в июне 2003 г. Он является дальнейшим усовершенствованием

спецификации IEEE 802.11b и реализует передачу данных в том же частотном диапазоне. Главным преимуществом этого стандарта является повышенная пропускная способность – скорость передачи данных в радиоканале достигает 54 Мбит/с по сравнению с 11 Мбит/с в 802.11b. Как и IEEE 802.11b, новая спецификация функционирует в диапазоне 2,4 ГГц, однако для повышения скорости используется та же схема модуляции сигнала, что и в 802.11a – ортогональное частотное мультиплексирование (OFDM).

Стандарт 802.11g совместим с 802.11b. Так, адаптеры 802.11b могут работать в сетях 802.11g (но при этом не быстрее 11 Мбит/с), а адаптеры 802.11g могут снижать скорость передачи данных до 11 Мбит/с для работы в старых сетях 802.11b.

Стандарт Wi-Fi 802.11n (Wi-Fi 4). Стандарт был ратифицирован 11 сентября 2009 г. Он увеличивает скорость передачи данных практически в 4 раза по сравнению с устройствами стандартов 802.11g (максимальная скорость которых равна 54 Мбит/с), при условии использования в режиме 802.11n с другими устройствами 802.11n. Максимальная теоретическая скорость передачи данных составляет 600 Мбит/с, применяя передачу данных сразу по четырем антеннам. По одной антенне – до 150 Мбит/с.

Устройства 802.11n функционируют в частотных диапазонах 2,4–2,5 или 5,0 ГГц.

В основе стандарта IEEE 802.11n лежит технология OFDM-MIMO. Большинство функционала позаимствовано из стандарта 802.11a, тем не менее в стандарте IEEE 802.11n имеется возможность применения как частотного диапазона, принятого для стандарта IEEE 802.11a, так и частотного диапазона, принятого для стандартов IEEE 802.11b/g. Таким образом, устройства, поддерживающие стандарт IEEE 802.11n, могут функционировать в частотном диапазоне либо 5, либо 2,4 ГГц, причем конкретная реализация зависит от страны.

Увеличение скорости передачи в стандарте IEEE 802.11n достигается за счет удвоения ширины канала с 20 до 40 МГц, а также вследствие реализации технологии MIMO.

Стандарт 802.11ac. Стандарт 802.11ac (Wi-Fi 5) – новый стандарт, который работает только в диапазоне 5 ГГц. Теоретическая скорость передачи данных – до 6,9 Гбит/с (при наличии 8 антенн и в режиме MU-MIMO). Данный режим есть только на двухдиапазонных маршрутизаторах, которые могут транслировать сеть в диапазоне 2.4 и 5 ГГц.

В данной технологии используется узконаправленное излучение антенн, более широкие каналы, несколько антенн для передачи и приема данных. Все это позволяет довести реальное быстродействие до 1,3 Гбит/с и увеличить расстояние связи. Новый стандарт обеспечивает также лучшее прохождение сигналов через стены домов, поэтому сеть на базе технологии 802.11ac будет надежно работать в пределах целого здания.

В настоящее время данный стандарт в основном работает с ограниченной скоростью передачи данных до 300 Мбит/с. Ожидается, что повышение быстродействия будет достигнуто в первую очередь благодаря тому, что устройства смогут работать не только с каналами шириной 20–40 МГц, но и 80–160 МГц, особенно в частотном диапазоне 5 ГГц.

Стандарт 802.11ax. Стандарт IEEE 802.11ax (Wi-Fi 6), анонсированный осенью 2019 г. и утвержденный в 2020 г.) оказался ответом на актуальные запросы рынка к большей емкости беспроводной сети, подключению большего количества пользователей при гарантированной производительности устройств и приложений. Технология Wi-Fi 6 повысит скорость передачи данных и пропускную способность как новых, так и уже существующих беспроводных сетей. Ориентированный на корпоративных пользователей стандарт обеспечит гибкость и масштабируемость инфраструктуры для быстрого запуска цифровых сервисов.

Wi-Fi 6 отличается от предшественника – Wi-Fi 5 – увеличенной почти в полтора раза скоростью передачи данных (9,6 против 6,9 Гбит/с). Однако реальный прирост скорости у конечных пользователей будет составлять порядка 30–40%.

Кроме того, новый стандарт предусматривает более совершенную систему шифрования WPA3 (Wi-Fi Protected Access III) и способен обеспечивать более высокую стабильность работы в местах скопления устройств с поддержкой Wi-Fi. Технология работает в диапазонах частот 2,4 и 5 ГГц, что обеспечивает большую пропускную способность.

К другим возможностям стандарта можно отнести реализацию технологи MU-MIMO, которая позволяет роутеру принимать и отправлять несколько сигналов одновременно. В целом в табл. 7.4 представлены основные параметры стандартов Wi-Fi 802.11.

Отметим, что базовые стандарты беспроводной связи могут быть расширены за счет реализации дополнительных стандартов.

Таблица 7.4

Обобщенные характеристики стандартов Wi-Fi

Название стандарта	Частота передачи, ГГц	Комментарий
802.11a	5	Не совместим с сетями b или g. Это один из самых старых стандартов, но сегодня он используется многими устройствами. Максимальная скорость передачи – 54 Мбит/с, но обычно от 6 до 24 Мбит/с
802.11b	2,4	Совместим с g-сетями. В реальности, g была сделана обратно совместимой с b-сетью для поддержки большего количества устройств. Максимальная скорость передачи – 11 Мбит/с
802.11g	2,4	Самый популярный тип сети. Сочетание скорости и обратной совместимости делает его подходящим для современных сетей. Максимальная скорость передачи – 54 Мбит/с
802.11n	2,4 и 5	Максимальная теоретическая скорость передачи данных составляет 600 Мбит/с, при этом используется сразу четыре антенны. По одной антенне – до 150 Мбит/с
802.11ac	5	Стандарт 802.11ac обеспечивает обратную совместимость с 802.11b/g/n, характеризуется скоростью до 1300 Мбит/с в полосе 5 ГГц и до 450 Мбит/с на 2,4 ГГц. Максимальная теоретическая скорость передачи данных – 6,9 Гбит/с (при наличии 8 антенн в режиме MU-MIMO)
802.11ax	2,4 и 5	Увеличена почти в полтора раза по сравнению с 802.11ac теоретическая скорость передачи данных 9,6 Гбит/с

Рабочая группа IEEE дополнила существующие спецификации 802.11 MAC (уровень доступа к среде передачи) и 802.11a PHY (физический уровень в сетях 802.11a) *алгоритмами эффективного выбора частот для офисных и уличных беспроводных сетей*, а также *средствами управления использованием спектра*, *контроля за излучаемой мощностью* и *генерации отчетов*. Получился так называемый стандарт ***IEEE 802.11h*.**

Решение этих задач базируется на использовании протоколов Dynamic Frequency Selection (DFS) и Transmit Power Control (TPC), предложенных Европейским институтом стандартов по телекоммуникациям (ETSI).

Указанные протоколы предусматривают динамическое реагирование клиентов беспроводной сети на интерференцию радиосигналов путем перехода на другой канал, снижения мощности либо двумя способами одновременно.

Разработанный стандарт ***IEEE 802.11i*** призван расширить возможности протокола 802.11 MAC, предусмотрев средства *шифрования передаваемых данных*, а также *централизованной аутентификации* пользователей и рабочих станций. В результате масштабы беспроводных локальных сетей можно будет наращивать до сотен и тысяч рабочих станций.

В основе 802.11i лежит *протокол аутентификации* Extensible Authentication Protocol (EAP), базирующийся на PPP. Сама процедура аутентификации предполагает участие в ней трех сторон – *вызывающей* (клиента), *вызываемой* (точки доступа) и *сервера аутентификации* (как правило, сервера RADIUS). В то же время новый стандарт, судя по всему, оставит на усмотрение производителей реализацию алгоритмов управления ключами.

Разработанные средства защиты данных должны найти применение не только в беспроводных, но и в других локальных сетях – Ethernet и Token Ring.

Повышение уровня *безопасности беспроводных* сетей явилось целью создания данной спецификации. В ней реализован набор защитных функций при обмене информацией через беспроводные сети. Примером является технология AES (Advanced Encryption Standard) – алгоритм шифрования, поддерживающий ключи длиной 128, 192 и 256 битов.

Стандарт ***IEEE 802.11j*** оговаривает существование в одном диапазоне сетей стандартов 802.11a и HiperLAN2. Спецификация предназначена для Японии и расширяет стандарт 802.11a добавочным каналом 4,9 ГГц.

Стандарт ***IEEE 802.11d*** определяет требования к физическим параметрам каналов (мощность излучения и диапазоны частот) и устройств беспроводных сетей с целью обеспечения их соответствия законодательным нормам различных стран.

Спецификации стандарта ***IEEE 802.11e*** позволяют создавать *мультисервисные беспроводные локальные сети*, ориентированные не только на корпоративных пользователей, но и на индивидуальных. При сохранении полной совместимости с уже принятыми стандартами 802.11a и b он позволяет расширить их функциональность за счет поддержки потоковых мультимедиаданных и гарантированного *качества услуг* (QoS).

Создание данного стандарта связано с использованием средств *мультимедиа*. Он определяет механизм назначения приоритетов разным видам трафика, таким как аудио- и видеоприложения.

Спецификации ***IEEE 802.11f*** описывают *протокол обмена служебной информацией между точками доступа* (Inter-Access Point Protocol, IAPP). Данный протокол необходим для построения распределенных беспроводных сетей передачи данных. Дата утверждения спецификаций ***IEEE 802.11f*** в качестве стандарта пока не определена.

Данный стандарт, связанный с аутентификацией, определяет механизм взаимодействия точек связи между собой при перемещении клиента между сегментами сети. Другое название стандарта – Inter Access Point Protocol.

Характеристику основного оборудования для беспроводной связи рассмотрим в подглаве 9.5.

Краткое описание дополнительных стандартов представлено в табл. 7.5.

7.4.3. Принципы организации беспроводных сетей

Выделяют два основных вида организации беспроводных сетей: *Ad-Hoc* (Independent Basic Service Set (IBSS) или Peer-to-Peer) и *Infrastructure Mode*.

Таблица 7.5

Дополнительные стандарты беспроводных сетей

Наименование стандарта	Назначение
802.11h	Дополняет спецификации IEEE 802.11 алгоритмами эффективного выбора частот для офисных и уличных беспроводных сетей, а также средствами управления спектра
802.11i	Предусматривает для стандартов IEEE 802.11 средства шифрования передаваемых данных, а также централизованной аутентификации пользователей и рабочих станций
802.11j	Данный стандарт оговаривает существование в одном диапазоне сетей стандартов 802.11a и HiperLAN2. Спецификация предназначена для Японии и расширяет стандарт 802.11a добавочным каналом 4,9 ГГц
802.11d	Стандарт определяет требования к физическим параметрам каналов (мощность излучения, диапазон частот) и устройств беспроводных сетей с целью обеспечения их соответствия законодательным нормам различных стран
802.11e	При сохранении полной совместимости с используемыми стандартами 802.11a и b позволяет расширить их функциональность за счет поддержки потоковых мультимедиаданных и гарантированного качества услуг (QoS)
802.11f	Данный стандарт определяет механизм взаимодействия точек связи между собой при перемещении клиента между сегментами сети

Режим Ad-Hoc является простейшей структурой локальной сети, при которой узлы сети (ноутбуки или компьютеры) связываются напрямую друг с другом.

Такая структура удобна для быстрого развертывания сетей. Для ее организации требуется минимум оборудования – каждый узел должен быть оборудован адаптером WLAN.

Сеть характеризуется изменяющейся структурой. Клиентские устройства соединяются «на лету», образуя собой сеть. Каждый узел сети пытается переслать данные, предназначенные другим узлам. При этом определение того, какому узлу пересылать данные, производится динамически, на основании связности сети. Это является отличием от проводных сетей и управляемых беспроводных сетей, в которых задачу управления потоками данных выполняют маршрутизаторы (в проводных сетях) или точки доступа (в управляемых беспроводных сетях).

Первыми беспроводными самоорганизующимися сетями были сети *Packet Radio*, финансируемые с 1970-х годов управлением DARPA после проекта ALOHAnet.

В ***режиме Infrastructure Mode*** узлы сети связаны друг с другом не напрямую, а через точку доступа, так называемую *Access Point*. Различают два режима взаимодействия с точками доступа – *BSS* (Basic Service Set) и *ESS* (Extended Service Set).

В режиме BSS все узлы связаны между собой через одну точку доступа, которая может играть роль моста для соединения с внешней кабельной сетью.

Режим ESS представляет собой объединение нескольких точек доступа, т. е. объединяет несколько сетей BSS. В этом случае точки доступа могут взаимодействовать и друг с другом. Расширенный режим удобно использовать тогда, когда необходимо объединить в одну сеть несколько пользователей или подключить несколько проводных сетей.

Важным вопросом при организации WLAN-сетей является дальность покрытия. На этот параметр влияет сразу несколько факторов.

1. Используемая частота (чем она больше, тем меньшая дальность действия радиоволн).

2. Наличие преград между узлами сети (различные материалы по-разному поглощают и отражают сигналы).

3. Режим функционирования – *Infrastructure Mode* или *Ad-Hoc*.

4. Мощность оборудования.

Если рассматривать идеальные условия, то зона покрытия с одной точкой доступа будет иметь следующий средний радиус покрытия:

– для сети стандарта IEEE 802.11a – 50 м;

– для сетей 802.11b и 802.11g – порядка 100 м.

За счет увеличения количества точек доступа (в режиме Infrastructure ESS) можно расширять зоны покрытия сети на всю необходимую область охвата.

Выводы

1. В компьютерных сетях применяются кабели, удовлетворяющие определенным стандартам. Современные стандарты определяют характеристики не отдельного кабеля, а полного набора элементов, необходимого для создания кабельного соединения, например шнура от рабочей станции до розетки, самой розетки, основного кабеля, жесткого кроссового соединения и шнура до концентратора. Сегодня наиболее употребительными стандартами являются американский стандарт EIA/TIA-568A, международный стандарт ISO/IEC 11801, европейский стандарт EN50173, а также фирменный стандарт компании IBM.

2. Стандарты определены для четырех типов кабеля: на основе неэкранированной витой пары, на основе экранированной витой пары, коаксиального и волоконно-оптического.

3. Кабель на основе неэкранированной витой пары в зависимости от электрических и механических характеристик разделяется на 7 категорий. Кабели категории 1 применяются там, где требования к скорости передачи минимальны. Главная особенность кабелей категории 2 – способность передавать сигналы со спектром до 1 МГц. Кабели категории 3 широко распространены и предназначены как для передачи данных, так и для передачи голоса. Кабели категории 4 представляют собой несколько улучшенный вариант кабелей категории 3 и на практике используются редко. Кабели категории 5 были специально разработаны для поддержки высокоскоростных протоколов FDDI, Fast Ethernet, 100VG-AnyLAN, ATM и Gigabit Ethernet.

4. Особое место занимают кабели категорий 6 и 7. Для кабеля категории 6 характеристики определяются до частоты 200 МГц, а для кабелей категории 7 – до 600 МГц. Кабели категории 7 обязательно экранируются, причем как каждая пара, так и весь кабель в целом.

Кабель категории 6 может быть как экранированным, так и неэкранированным. Основное назначение этих кабелей – поддержка высокоскоростных протоколов на отрезках кабеля большей длины, чем кабель UTP категории 5.

5. Кабель на основе экранированной витой пары хорошо защищает передаваемые сигналы от внешних помех, а пользователей сетей – от вредного для здоровья излучения. Наличие заземляемого экрана удорожает кабель и усложняет его прокладку. Экранированный кабель применяется только для передачи данных. Основным стандартом, определяющим параметры экранированной витой пары, является фирменный стандарт IBM.

6. Коаксиальные кабели представлены в большом разнообразии вариантов: «толстый» коаксиальный кабель, различные виды «тонкого» коаксиального кабеля (сюда же относится телевизионный кабель), которые обладают худшими механическими и электрическими характеристиками по сравнению с «толстым» коаксиальным кабелем, зато за счет своей гибкости более удобны при монтаже.

7. Волоконно-оптические кабели обладают отличными электромагнитными и механическими характеристиками, недостаток их состоит в сложности и высокой стоимости монтажных работ.

8. Беспроводная связь предусматривает использование трех основных технологий: радиосвязи; связи в микроволновом диапазоне; инфракрасной связи. Технологии радиосвязи пересылают данные на радиочастотах и практически не имеют ограничений по дальности. Сегодня именно они получили наибольшее распространение в виде так называемой Wi-Fi-связи, базирующейся на стандарте IEEE 802.11.

9. Wi-Fi-связь характеризуется множеством преимуществ по сравнению с проводными системами (легко разворачиваются; пользователи свободно перемещаются в пределах зоны действия сети и т. д.), но есть существенный недостаток – возможность легкого перехвата данных и взлома сети, что в некоторой степени сдерживает их распространение.

Контрольные вопросы

1. Что может быть использовано в качестве физической среды передачи данных?

2. Какие вопросы при организации сети решаются на физическом уровне?

3. Что такое кабель?

4. Что такое линии связи?

5. Дайте определение каналов связи.

6. Какие проблемы существуют при организации каналов связи?

7. Дайте определение стуктурированной кабельной системы.

8. Перечислите преимущества использования стуктурированной кабельной системы.

9. Каково назначение структурированной кабельной системы?

10. Перечислите типы кабелей для передачи данных в сети.

11. Какими основными стандартами определены характеристики кабеля?

12. Перечислите основные характеристики электрических кабелей, определяемые стандартами.

13. На какие классы подразделяются кабельные системы?

14. Какие типы кабелей используются для передачи данных в сети?

15. Какие известны кабельные системы Ethernet?

16. Приведите основные характеристики кабеля «витая пара» в зависимости от категории.

17. Опишите стандартные разводки кабеля «витая пара».

18. Какие существуют типы оптоволоконных кабелей?

19. Опишите физические особенности оптоволоконных кабелей.

20. Охарактеризуйте технические особенности оптоволоконных кабелей.

21. Назовите основные недостатки оптоволоконных кабелей.

22. Приведите основные параметры оптоволоконных кабелей.

23. Какие известны технологии беспроводной передачи данных?

24. В каких случаях используется инфракрасная связь?

25. В чем заключаются преимущества использования радиосвязи?

26. Перечислите стандарты беспроводных сетей (802.11).

27. Опишите стандарт IEEE 802.11a.

28. Охарактеризуйте стандарт IEEE 802.11b.

29. Дайте описание стандарта IEEE 802.11g.

30. Приведите характеристики стандарта IEEE 802.11n.

31. Опишите стандарт IEEE 802.11ac.

32. Расскажите о стандарте IEEE 802.11ax.

33. Охарактеиризуйте режим работы беспроводной сети IBSS.

34. Дайте описание режимам работы беспроводной сети BSS.

35. Опишите режим работы беспроводной сети ESS.

8

СЕТЕВЫЕ ОПЕРАЦИОННЫЕ СИСТЕМЫ

Сетевые операционные системы (Network Operating System, NOS) – это комплекс программ, обеспечивающих обработку, хранение и передачу данных в сети.

Сетевая операционная система (СОС) составляет основу любой компьютерной сети. Каждый компьютер в сети автономен. Поэтому под ***сетевой операционной системой*** в широком смысле можно понимать совокупность операционных систем отдельных компьютеров, взаимодействующих с целью обмена сообщениями и разделения ресурсов по единым правилам – протоколам.

В узком смысле **сетевая ОС** – это операционная система отдельного компьютера, которая обеспечивает ему возможность работать в сети.

СОС выполняет функции прикладной платформы, предоставляет разнообразные виды сетевых служб и поддерживает работу прикладных процессов в абонентских системах. СОС использует клиент-серверную либо одноранговую архитектуру. Компоненты СОС располагаются на всех рабочих станциях, включенных в сеть.

СОС определяет взаимосвязанную группу протоколов верхних уровней, обеспечивающих выполнение основных функций сети. К ним, в первую очередь, относятся:

– адресация объектов сети;
– функционирование сетевых служб;
– обеспечение безопасности данных;
– управление сетью.

При выборе СОС необходимо рассматривать множество факторов. Среди них:

– набор сетевых служб, которые предоставляет сеть;

– возможность наращивания имен, определяющих хранимые данные и прикладные программы;

– механизм рассредоточения ресурсов по сети;

– способ модификации сети и сетевых служб;

– надежность функционирования и быстродействие сети;

– используемые или выбираемые физические средства соединения;

– типы компьютеров, объединяемых в сеть, их ОС;

– предлагаемые системы, обеспечивающие управление сетью;

– используемые средства защиты данных;

– совместимость с уже созданными прикладными процессами;

– число серверов, которое может работать в сети;

– перечень ретрансляционных систем, обеспечивающих сопряжение локальных сетей с различными территориальными сетями;

– способ документирования работы сети, организация подсказок и поддержек.

8.1. Структура сетевой операционной системы

Общая структура СОС представлена на рис. 8.1.

Рис. 8.1. Структура сетевой ОС

В соответствии со структурой, приведенной на рис. 8.1, в сетевой операционной системе отдельной машины можно выделить несколько частей.

1. *Средства управления локальными ресурсами компьютера*: функции распределения оперативной памяти между процессами, планирования и диспетчеризации процессов, управления процессорами, управления периферийными устройствами и другие функции управления ресурсами локальных ОС.

2. *Средства предоставления собственных ресурсов и услуг в общее пользование* – серверная часть ОС (сервер). Эти средства обеспечивают, например, блокировку файлов и записей, ведение справочников имен сетевых ресурсов; обработку запросов удаленного доступа к собственной файловой системе и базе данных; управление очередями запросов удаленных пользователей к своим периферийным устройствам.

3. *Средства запроса доступа к удаленным ресурсам и услугам* – клиентская часть ОС (редиректор). Эта часть выполняет распознавание и перенаправление в сеть запросов к удаленным ресурсам от приложений и пользователей. Клиентская часть также осуществляет прием ответов от серверов и преобразование их в локальный формат, так что для приложения выполнение локальных и удаленных запросов не различимо.

4. *Коммуникационные средства ОС*, с помощью которых происходит обмен сообщениями в сети. Эта часть обеспечивает адресацию и буферизацию сообщений, выбор маршрута передачи сообщения по сети, надежность передачи и т. п., т. е. является средством транспортировки сообщений.

8.2. Клиентское программное обеспечение

Для работы с сетью на клиентских рабочих станциях должно быть установлено клиентское программное обеспечение, которое обеспечивает доступ к ресурсам, расположенным на сетевом сервере. Тремя наиболее важными компонентами клиентского программного обеспечения являются *редиректоры*, *распределители* и *имена UNC*.

8.2.1. Редиректоры

Редиректор (Redirector) – сетевое программное обеспечение, которое принимает запросы ввода-вывода для *удаленных файлов*, *именованных каналов* или *почтовых слотов* и затем переназначает их сетевым сервисам другого компьютера.

Редиректор перехватывает все запросы, поступающие от приложений, и анализирует их. Фактически существуют два типа редиректоров, используемых в сети:

– *клиентский редиректор* (Client Redirector);

– *серверный редиректор* (Server Redirector).

Оба редиректора функционируют на представительском уровне модели OSI. Когда клиент делает запрос к сетевому приложению или службе, редиректор перехватывает этот запрос и проверяет, является ли ресурс локальным (находящимся на запрашивающем компьютере) или удаленным (в сети). Если редиректор определяет, что это локальный запрос, то направляет его центральному процессору для немедленной обработки. Если запрос предназначен для сети, то редиректор направляет его по сети к соответствующему серверу. По существу, редиректоры скрывают от пользователя сложность доступа к сети. После того как сетевой ресурс определен, пользователи могут получить к нему доступ без знания его точного расположения.

8.2.2. Распределители

Распределитель (Designator) – это часть программного обеспечения, которая управляет присвоением букв накопителя (Drive Letter) как локальным, так и удаленным сетевым ресурсам или разделяемым дисководам, что помогает во взаимодействии с сетевыми ресурсами.

Когда между сетевым ресурсом и буквой локального накопителя создана ассоциация, известная также как отображение дисковода (Mapping a Drive), распределитель отслеживает присвоение такой буквы дисковода сетевому ресурсу. Затем, когда пользователь или приложение получат доступ к диску, распределитель заменит букву дисковода на сетевой адрес ресурса, прежде чем запрос будет послан редиректору.

8.2.3. Имена UNC

Редиректор и распределитель являются не единственными методами, используемыми для доступа к сетевым ресурсам.

Большинство современных сетевых операционных систем распознают ***имена UNC*** (Universal Naming Convention (Pathnames) – универсальное соглашение по наименованию). UNC представляют собой стандартный способ именования сетевых ресурсов.

Эти имена имеют форму: *Имя_сервера\имя_ресурса.* Способные работать с UNC приложения и утилиты командной строки используют имена UNC вместо отображения сетевых дисков.

8.3. Серверное программное обеспечение

Для того чтобы компьютер мог выступать в роли сетевого сервера, необходимо установить серверную часть сетевой ОС, которая позволяет поддерживать ресурсы и распространять их среди сетевых клиентов.

Важным вопросом для сетевых серверов является возможность ограничить доступ к сетевым ресурсам. Это **называется сетевой безопасностью** (Network Security). Она предоставляет средства управления ресурсами, к которым могут получить доступ пользователи, степень этого доступа, а также сколько пользователей смогут получить такой доступ одновременно.

В дополнение к обеспечению контроля над сетевыми ресурсами сервер выполняет следующие функции:

– предоставляет проверку *регистрационных имен* (Logon Identification) для пользователей;

– управляет пользователями и группами;

– хранит инструменты сетевого администрирования для управления, контроля и аудита;

– обеспечивает отказоустойчивость для защиты целостности сети.

Некоторые из сетевых ОС, в том числе Windows, имеют программные компоненты, обеспечивающие компьютеру как клиентские, так и серверные возможности. Это позволяет компьютерам поддерживать и использовать сетевые ресурсы. В общем, этот тип сетевых операционных систем не так мощен и надежен, как законченные сетевые операционные системы. Главное преимущество комбинированной клиентско-серверной сетевой операционной системы заключается в том, что важные ресурсы, расположенные на отдельной рабочей станции, могут быть разделены с остальной частью сети. Недостаток состоит в том, что если рабочая станция поддерживает много активно используемых ресурсов, она испытывает серьезное падение производительности. Если такое происходит, то необходимо перенести эти ресурсы на сервер для увеличения общей производительности.

В зависимости от функций, возлагаемых на конкретный компьютер, в его операционной системе может отсутствовать либо клиентская, либо серверная часть.

На рис. 8.2 компьютер 1 выполняет функции клиента, а компьютер 2 – функции сервера, соответственно на первой машине отсутствует серверная часть, а на второй – клиентская.

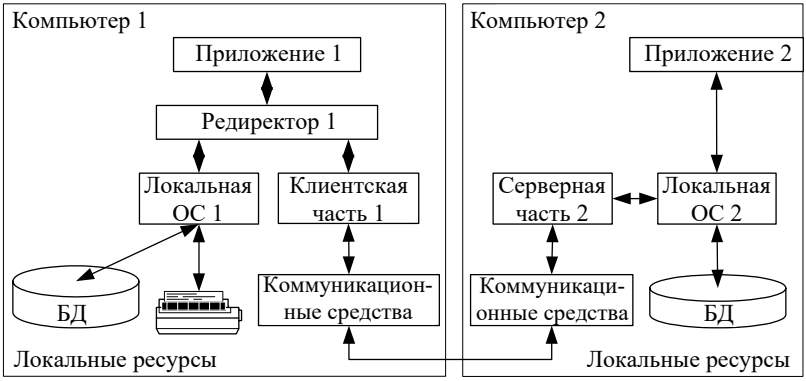


Рис. 8.2. Взаимодействие компонентов сетевой ОС

Если выдан запрос к ресурсу данного компьютера, то он переадресовывается локальной операционной системе. Если же это запрос к удаленному ресурсу, то он перенаправляется в клиентскую часть, где преобразуется из локальной формы в сетевой формат, а затем передается коммуникационным средствам. Серверная часть ОС

компьютера 2 принимает запрос, преобразует его в локальную форму и передает для выполнения своей локальной ОС. После того как результат получен, сервер обращается к транспортной подсистеме и направляет ответ клиенту, выдавшему запрос. Клиентская часть преобразует результат в соответствующий формат и адресует его тому приложению, которое выдало запрос.

При выборе сетевой операционной системы необходимо учитывать:

– совместимость оборудования;
– тип сетевого носителя;
– размер сети;
– требования к серверу;
– операционные системы на клиентах и серверах;
– сетевую файловую систему;
– соглашения об именах в сети;
– организацию сетевых устройств хранения.

8.4. Одноранговые и серверные сетевые операционные системы

В зависимости от того как распределены функции между компьютерами сети, сетевые ОС могут выполнять функции как *клиента*, так и *сервера*.

Если компьютер предоставляет свои ресурсы другим пользователям сети, то он играет роль сервера. При этом компьютер, обращающийся к ресурсам другой машины, является клиентом. Компьютер, работающий в сети, может выполнять функции либо клиента, либо сервера, либо совмещать обе эти функции. На рис. 8.3 и 8.4 приведены примеры использования структур сетевых ОС в одноранговых сетях и сетях с выделенными серверами.

Если выполнение каких-либо серверных функций является основным назначением компьютера, то такой компьютер называется **выделенным сервером**.

В зависимости от того, какой ресурс сервера является **разделяемым**, сервер называется *файл-сервером*, *факс-сервером*, *принт-сервером*, *сервером приложений*, *сервером БД*, *Web-сервером* и т. д.

На выделенных серверах устанавливается ОС для выполнения тех или иных серверных функций. Выделенный сервер не принято использовать в качестве компьютера для выполнения текущих задач, не связанных с его основным назначением, так как это может уменьшить производительность его работы как сервера.

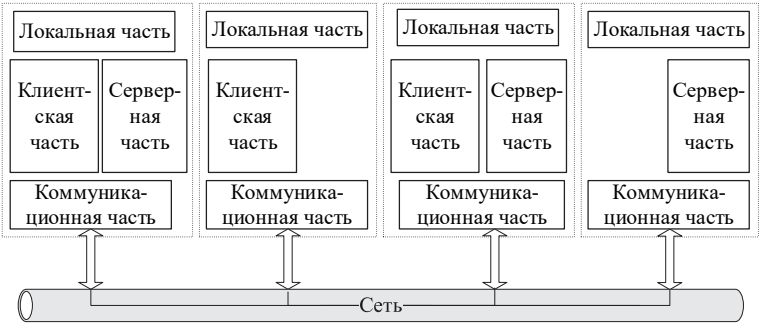


Рис. 8.3. Структура ОС для одноранговой сети

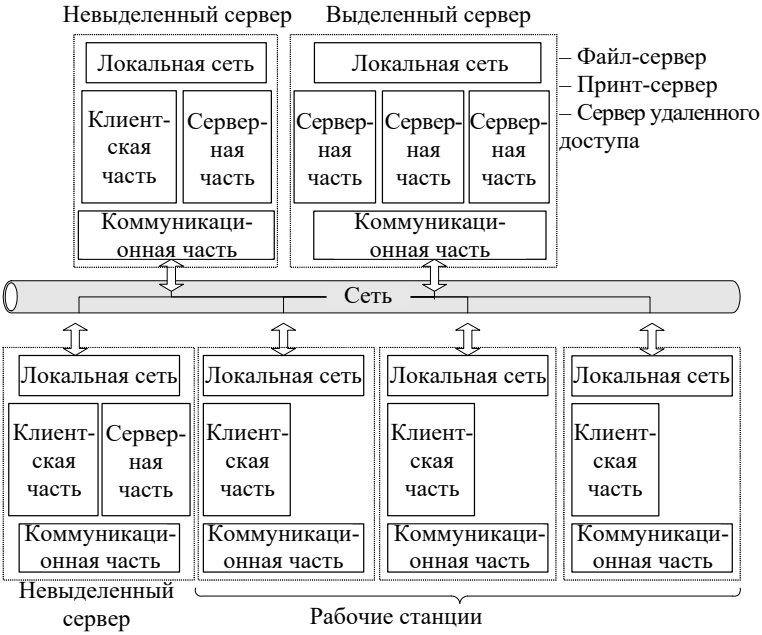


Рис. 8.4. Структура ОС для клиент-серверной сети

В *одноранговых сетях* все компьютеры равны в правах доступа к ресурсам друг друга. Каждый пользователь может по своему желанию объявить какой-либо ресурс своего компьютера разделяемым, после чего другие пользователи могут его эксплуатировать. В таких сетях на всех компьютерах устанавливается одна и та же ОС, которая предоставляет всем компьютерам в сети *потенциально* равные возможности. Одноранговые сети могут быть построены, например, на базе ОС LANtastic, Personal Ware, Windows (типа XP, Vista, Seven). Одноранговые сети проще в организации и эксплуатации. Но они применяются в основном для объединения небольших групп пользователей, не предъявляющих больших требований к объемам хранимой информации, ее защищенности от несанкционированного доступа и к скорости доступа.

При повышенных требованиях к этим характеристикам более подходящими являются сети с выделенными серверами, где сервер лучше решает задачу обслуживания пользователей своими ресурсами, так как его аппаратура и сетевая операционная система специально спроектированы для этой цели.

В *сетях с выделенными серверами* чаще всего используются сетевые ОС, в состав которых входит несколько вариантов ОС, отличающихся возможностями серверных частей. Например, сетевая операционная система Novell NetWare имеет серверный вариант, оптимизированный для работы в качестве файл-сервера, а также варианты оболочек для рабочих станций с различными локальными ОС, причем эти оболочки выполняют исключительно функции клиента. Другим примером ОС, ориентированной на построение сети с выделенным сервером, является операционная система Windows Server.

Выводы

Таким образом, сетевая операционная система – это операционная система, обеспечивающая компьютеру возможность работать в сети.

1. В общем, она состоит из серверной и клиентской части, а также коммуникационных средств.

2. Для работы с сетью на клиентских рабочих станциях должно быть установлено клиентское программное обеспечение, позволяющее получить доступ к ресурсам, расположенным на сетевом сервере. Тремя наиболее важными компонентами клиентского программного обеспечения являются редиректоры, распределители и имена UNC.

3. Серверная часть сетевой операционной системы позволяет поддерживать ресурсы и распространять их среди сетевых клиентов. Однако важнейшим вопросом для сетевых серверов является возможность предоставлять средства управления доступом пользователя к ним, а также определять его уровень. Этот контроль обеспечивает конфиденциальность и защиту предоставляемых ресурсов.

4. Отметим, что некоторые современные сетевые операционные системы, например Windows, имеют программные компоненты, обеспечивающие компьютеру как клиентские, так и серверные возможности, что позволяет компьютерам поддерживать и использовать сетевые ресурсы. Но в целом подобный тип сетевых операционных систем не так мощен и надежен, как законченные сетевые операционные системы.

Контрольные вопросы

1. Что такое сетевая операционная система и каково ее назначение?
2. Какие функции сети выполняет сетевая операционная система?
3. Опишите структуру сетевой операционной системы.
4. Что такое редиректор и каковы его функции?
5. Что такое распределитель и каковы его функции?
6. Как подразделяются сетевые операционные системы по правам доступа к ресурсам?
7. В чем заключаются преимущества и недостатки операционных систем, обеспечивающие как клиентские, так и серверные возможности?
8. Опишите структуру одноранговой сети.
9. Опишите структуру сетей с использованием клиент-серверных ОС.
10. Как подразделяются серверы по их назначению?

9

АППАРАТНЫЕ СРЕДСТВА ДЛЯ ПЕРЕДАЧИ ДАННЫХ

9.1. Сетевые адаптеры

Сетевые адаптеры – это сетевое оборудование, обеспечивающее функционирование сети на физическом и канальном уровнях.

> ! *Сетевой адаптер (Network Interface Controller, NIC) относится к периферийному устройству компьютера*, непосредственно взаимодействующему со сре-дой передачи данных, которая прямо или через другое комму-никационное оборудование связывает его с другими компью-терами.

Это устройство решает задачи надежного обмена двоичными данными, которые представлены соответствующими электромагнитными сигналами, по внешним линиям связи. Как и любой контроллер компьютера, сетевой адаптер работает под управлением *драйвера* операционной системы, и распределение функций между сетевым адаптером и драйвером может изменяться от реализации к реализации.

Компьютер, будь то сервер или рабочая станция, подключается к сети с помощью внутренней платы – сетевого адаптера (хотя бывают и внешние сетевые адаптеры, подключаемые к компьютеру через параллельный порт).

Сетевой адаптер вставляется в гнездо *материнской платы*. Карты сетевых адаптеров устанавливаются на каждой рабочей станции и на файловом сервере. Рабочая станция отправляет запрос к файловому серверу и получает ответ через сетевой адаптер, когда файловый сервер готов. Сетевые адаптеры преобразуют параллельные коды, используемые внутри компьютера и представленные маломощными

сигналами, в последовательный поток мощных сигналов для передачи данных по внешней сети. Сетевые адаптеры должны быть совместимы с кабельной системой сети, внутренней информационной шиной ПК и сетевой операционной системой.

9.1.1. Назначение и настройка

Для работы ПК в сети надо правильно установить и настроить сетевой адаптер. Для адаптеров, отвечающих *стандарту PnP* (Plug and Play), настройка производится автоматически. В ином случае необходимо настроить линию *запроса на прерывание, IRQ* (Interrupt Request Line) и *адрес ввода-вывода* (Input/Output Address).

Адрес ввода-вывода – это трехзначное шестнадцатеричное число, которое идентифицирует коммуникационный канал между аппаратными устройствами и центральным процессором.

Чтобы сетевой адаптер функционировал правильно, должны быть настроены линия IRQ и адрес ввода-вывода. Примеры некоторых запросов на прерывание IRQ и адреса ввода-вывода для основных устройств компьютера приведены в таблице.

Адреса ввода-вывода для основных устройств компьютера

Стандартное применение	Запрос на прерывание	Диапазон ввода-вывода
Системный таймер	IRQ0	–
Клавиатура	IRQ1	–
Вторичный контроллер IRQ или видеокарта	IRQ2	–
Прерывание от асинхронного последовательного порта COM2 и COM4	IRQ3	От 2F0 до 2FF
Прерывание от асинхронного последовательного порта COM1 и COM3	IRQ4	От 3F0 до 3FF
Обычно свободен (может быть занят параллельным портом LPT2)	IRQ5	–
Прерывание от параллельного принтерного порта LPT1	IRQ7	–
Обычно свободен	IRQ9	От 370 до 37F
Обычно свободен (может быть занят первичным контроллером SCSI)	IRQ10	–

Окончание таблицы

Стандартное применение	Запрос на прерывание	Диапазон ввода-вывода
Обычно свободен (может быть занят вторичным контроллером SCSI)	IRQ11	–
Мышь PS/2	IRQ12	–
Прерывание от первичного контроллера жесткого диска	IRQ14	–
Обычно свободен (может быть занят вторичным контроллером жесткого диска IDE)	IRQ15	–

Обычно сетевая карта обнаруживает конфликт, если двум устройствам назначен один и тот же ресурс (запрос на прерывание или адрес ввода-вывода). Сетевые карты поддерживают различные типы сетевых соединений.

Физический интерфейс между самой сетевой картой и сетью называют **трансивером** (Transceiver) – устройством, которое как получает, так и посылает данные.

Трансиверы на сетевых картах могут получать и посылать цифровые и аналоговые сигналы. Тип интерфейса, который использует сетевая карта, часто может быть физически определен на сетевой карте. Перемычки, или джамперы (маленькие перемычки, соединяющие два контакта), могут быть настроены для указания типа трансивера, который должна использовать сетевая карта в соответствии со схемой сети. Например, перемычка в одном положении может включить разъем RJ-45 для поддержки сети типа витая пара, в другом – поддержку внешнего трансивера.

9.1.2. Функции сетевых адаптеров

Сетевые адаптеры производят следующие основные операции при приеме или передаче сообщения.

1. *Гальваническая развязка* с коаксиальным кабелем или витой парой. Для этой цели используются импульсные трансформаторы. Иногда для развязки используются оптроны.

2. *Прием (передача) данных*. Данные передаются из ОЗУ ПК в адаптер или из адаптера в память ПК через программируемый канал ввода-вывода, канал прямого доступа или разделяемую память.

3. *Буферизация*. Для согласования скоростей пересылки данных в адаптер или из него со скоростью обмена по сети используются буферы. Во время обработки в сетевом адаптере данные хранятся в буфер, который позволяет адаптеру осуществлять доступ ко всему пакету информации. Использование буферов необходимо для согласования между собой скоростей обработки информации различными компонентами ЛВС.

4. *Формирование пакета*. Сетевой адаптер должен разделить данные на блоки в режиме передачи (или соединить их в режиме приема) и оформить в виде кадра определенного формата. Кадр включает несколько служебных полей, среди которых имеется адрес компьютера назначения и контрольная сумма кадра. По ней сетевой адаптер станции назначения делает вывод о корректности доставленной по сети информации.

5. *Доступ к каналу связи.* Набор правил, обеспечивающих доступ к среде передачи, выявление конфликтных ситуаций и контроль состояния сети.

6. *Идентификация своего адреса в принимаемом пакете*. Физический адрес адаптера может определяться установкой переключателей, храниться в специальном регистре или прошиваться в полупроводниковых запоминающих устройствах.

7. *Преобразование параллельного кода в последовательный код* при передаче данных и из последовательного кода в параллельный при приеме. В режиме передачи данные передаются по каналу связи в последовательном коде.

8. *Кодирование и декодирование данных.* На этом этапе должны быть сформированы электрические сигналы, используемые для представления данных. Большинство сетевых адаптеров для этой цели используют *манчестерское кодирование*. Этот метод не требует передачи синхронизирующих сигналов для распознавания единиц и нулей по уровням сигналов, а вместо этого для представления единиц и нулей используется перемена полярности сигнала.

9. *Передача или прием импульсов.* В режиме передачи закодированные электрические импульсы данных передаются в кабель (при приеме импульсы направляются на декодирование).

Сетевые адаптеры вместе с сетевым программным обеспечением способны распознавать и обрабатывать ошибки, которые могут возникнуть из-за электрических помех, коллизий или плохой работы оборудования.

Последние типы сетевых адаптеров поддерживают технологию Plug and Play. Если сетевую карту установить в компьютер, то при первой загрузке система определит тип адаптера и запросит для него драйверы.

Некоторые сетевые адаптеры имеют возможность использовать оперативную память ПК в качестве буфера для хранения входящих и исходящих пакетов данных.

Базовый адрес памяти (Base Memory Address) представляет собой шестнадцатеричное число, которое указывает на адрес в оперативной памяти, где находится этот буфер. Важно выбрать базовый адрес без конфликтов с другими устройствами.

9.1.3. Типы сетевых адаптеров

Сетевые адаптеры различаются по типу и разрядности используемой в компьютере внутренней шины данных – ISA, PCI, PCI-E.

Сетевые адаптеры различаются также по типу принятой в сети технологии – Ethernet, Token Ring, FDDI и т. п. Как правило, конкретная модель сетевого адаптера работает по определенной сетевой технологии (например, Ethernet). В связи с тем, что для каждой технологии сейчас имеется возможность использования различных сред передачи данных (тот же Ethernet поддерживает коаксиальный кабель, неэкранированную витую пару и оптоволоконный кабель), сетевой адаптер может поддерживать как одну, так и одновременно несколько сред. В случае когда сетевой адаптер поддерживает только одну среду передачи данных, а необходимо использовать другую, применяются трансиверы и конверторы.

Различные типы сетевых адаптеров отличаются не только методами доступа к среде и протоколами, но и следующими параметрами:

– скоростью передачи;
– объемом буфера для пакета;
– типом шины;
– быстродействием шины;
– совместимостью с различными микропроцессорами;
– использованием прямого доступа к памяти (DMA);
– адресацией портов ввода-вывода и запросов прерывания;
– конструкцией разъема.

Классификация сетевых адаптеров. В качестве примера классификации адаптеров можно привести подход, предложенный фирмой 3Com, заключающийся в выделении поколений развития Ethernet-адаптеров.

Сетевые адаптеры первого поколения были выполнены на дискретных логических микросхемах, в результате чего обладали низкой надежностью. Они имели буферную память только на один кадр, что приводило к низкой производительности адаптера, так как все кадры передавались из компьютера в сеть или из сети в компьютер последовательно. Кроме того, задание конфигурации адаптера первого поколения происходило вручную, с помощью перемычек. Для каждого типа адаптеров использовался свой драйвер, причем интерфейс между драйвером и сетевой операционной системой не был стандартизирован.

В *сетевых адаптерах второго поколения* для повышения производительности стали применять метод *многокадровой буферизации*, при котором следующий кадр загружается из памяти компьютера в буфер адаптера одновременно с передачей предыдущего кадра в сеть. В режиме приема, после того как адаптер полностью принял один кадр, он может начать передавать этот кадр из буфера в память компьютера одновременно с приемом другого кадра из сети.

В сетевых адаптерах второго поколения широко используются микросхемы с высокой степенью интеграции, что повышает надежность адаптеров. Кроме того, драйверы этих адаптеров основаны на стандартных спецификациях. Адаптеры второго поколения обычно поставляются с драйверами, работающими как в стандарте NDIS (Network Device Interface Specification – *спецификация интерфейса сетевого драйвера*), разработанном фирмами 3Com и Microsoft и одобренном IBM, так и в стандарте ODI (Open Data-Link Interface – *интерфейс открытого драйвера*), разработанном фирмой Novell.

Интерфейс ODI образуют модули драйвера и *уровня поддержки связи* (Link Support Layer, LSL) со стеком протоколов. Уровень поддержки связи (LSL) принимает пакет от драйвера, распознает тип пакета и выбирает соответствующий протокол из стека коммуникационных протоколов. Важно отметить, что стек транспортных протоколов рабочей станции является открытым: если новый протокол разрабатывался с использованием спецификаций ODI-интерфейса, то он может быть включен в стек.

В *сетевых адаптерах третьего поколения* (к ним фирма 3Com относит свои адаптеры семейства EtherLink III) осуществляется *конвейерная схема обработки кадров*. Она заключается в том, что процессы приема кадра из оперативной памяти компьютера и передачи его в сеть совмещаются во времени.

Таким образом, после приема нескольких первых байтов кадра начинается их передача. Это существенно (на 25–55%) повышает производительность цепочки *оперативная память – адаптер – физический канал – адаптер – оперативная память*. Такая схема очень чувствительна к порогу начала передачи, т. е. к количеству байтов кадра, которое загружается в буфер адаптера перед началом передачи в сеть. Сетевой адаптер третьего поколения осуществляет самонастройку этого параметра путем анализа рабочей среды, а также методом расчета без участия администратора сети. Самонастройка обеспечивает максимально возможную производительность для конкретного сочетания производительности внутренней шины компьютера, его системы прерываний и системы прямого доступа к памяти.

Адаптеры третьего поколения базируются на *специализированных интегральных схемах* (Application-Specific Integrated Circuit, ASIC), что повышает производительность и надежность адаптера при одновременном снижении его стоимости. Компания 3Com назвала свою технологию конвейерной обработки кадров Parallel Tasking, другие компании также реализовали похожие схемы в своих адаптерах. Повышение производительности канала *адаптер – память* очень важно для увеличения производительности сети в целом, так как производительность сложного маршрута обработки кадров, включающего, например, концентраторы, коммутаторы, маршрутизаторы, глобальные каналы связи и т. п., всегда определяется производительностью самого медленного элемента этого маршрута. Следовательно, если сетевой адаптер сервера или клиентского компьютера работает медленно, никакие быстрые коммутаторы не смогут повысить скорость работы сети.

Выпускаемые сейчас сетевые адаптеры можно отнести к *четвертому поколению*. В эти адаптеры обязательно входит ASIC, выполняющая функции MAC-уровня, а также большое количество высокоуровневых функций.

В набор таких функций может входить *поддержка агента удаленного мониторинга* (Remote Network MONitoring, RMON), схема приоритезации кадров, функции дистанционного управления компьютером и т. п. В серверных вариантах адаптеров почти обязательно наличие мощного процессора, разгружающего центральный процессор. Примером сетевого адаптера четвертого поколения может служить адаптер компании Intel – Intel PRO/1000 MT Desktop или Hardlink HA-32G фирмы MAS Elektronik AG.

Современные сетевые адаптеры, как правило, поддерживают следующие функции.

1. *PCI BUS-Mastering* – Peripheral Component Interconnect – компьютерная шина ввода-вывода, предназначена для подключения периферийных устройств к системной плате персонального компьютера; Bus-Mastering – режим, при котором устройство может самостоятельно (без участия центрального процессора) управлять шиной (пересылать данные, формировать команды и т. д.). На время обмена устройство захватывает шину и становится *master-устройством* (для примера, режим Bus-Mastering частично проявляется, если устройство работает с использованием канала DMA). Данная функция (PCI BUS-Mastering) означает возможность пересылки данных устройством без участия центрального процессора. На сетевой карте должны быть распаяны схемы, позволяющие осуществлять прямую передачу информации.

2. *BootROM* (или *Boot ROM*; ROM – тип полупроводниковой памяти). Возможность загрузки системы по сети заложена в виде BootROM сетевой карты. Это микросхема энергонезависимой памяти, где хранится код загрузчика. Он выполняет поиск в сети сервера и запрашивает у него IP-адрес, а также путь, где можно получить образ операционной системы. После того как образ загружен и размещен в оперативной памяти, дальнейшее управление загрузкой передается ему так же, как при работе с обычной загрузочной дискетой или диском. Таким образом, при соответствующей настройке ПК может работать вообще без жесткого диска. Загрузка через сеть настраивается в BIOS материнских плат, поддерживающих данную функцию.

3. *Wake-on-Lan*. Данная технология представляет собой включение (*Wake* – разбудить) удаленной системы через сеть. Адаптер отслеживает сетевой трафик в ожидании специального Wake-пакета и при его получении пробуждает систему. При этом требуется, чтобы в настройках BIOS была разрешена активация компьютера по запросу с порта, на который установлена карта.

Выводы

1. От производительности сетевых адаптеров зависит производительность любой сложной сети, так как данные всегда проходят не только через коммутаторы и маршрутизаторы сети, но и через адаптеры компьютеров, а результирующая производительность последовательно соединенных устройств определяется производительностью самого медленного устройства.

2. Сетевые адаптеры характеризуются производительностью, шиной компьютера, к которой они могут присоединяться, типом приемопередатчика, а также наличием собственного процессора (его характеристиками и возможностями), разгружающего центральный процессор компьютера.

3. Сетевые адаптеры для серверов обычно имеют собственный процессор, а клиентские сетевые адаптеры – нет.

4. Современные адаптеры умеют адаптироваться к временным параметрам шины и оперативной памяти компьютера для повышения производительности обмена *сеть – компьютер*.

9.2. Повторители и концентраторы

Основная функция *повторителя* (Repeater), как это следует из его названия, – повторение сигналов, поступающих на его порт. Повторитель улучшает электрические характеристики сигналов и их синхронность, и за счет этого появляется возможность увеличивать общую длину кабеля между самыми удаленными в сети узлами.

Многопортовый повторитель часто называют ***концентратором*** (Concentrator) или *хабом* (Hub). Это отражает тот факт, что данное устройство реализует не только функцию повторения сигналов, но и концентрирует в одном центральном устройстве функции объединения компьютеров в сеть.

Концентратор представляет собой сетевое устройство, действующее на физическом уровне сетевой модели OSI.

Отрезки кабеля, соединяющие два компьютера или какие-либо два других сетевых устройства, называются *физическими сегментами*, поэтому концентраторы и повторители, которые используются для добавления новых физических сегментов, являются средством физической структуризации сети.

Концентратор – устройство, у которого суммарная пропускная способность входных каналов выше пропускной способности выходного канала.

На рис. 9.1 показан внешний вид концентратора фирмы HP.

Рис. 9.1. Внешний вид концентратора фирмы HP

Так как потоки входных данных в концентраторе больше выходного потока, то главной его задачей является концентрация данных. При этом возможны ситуации, когда число блоков данных, поступающих на входы концентратора, превышает его возможности. Тогда концентратор ликвидирует часть этих блоков.

Ядром концентратора является *процессор*. Для объединения входной информации чаще всего используется *множественный доступ с разделением времени*.

Функции, выполняемые концентратором, близки к задачам, которые возложены на мультиплексор. Наращиваемые (модульные) концентраторы позволяют выбирать их компоненты, не анализируя

совместимость с уже используемыми. Современные концентраторы имеют порты для подключения к разнообразным локальным сетям.

Концентратор является активным оборудованием. Он служит центром (шиной) звездообразной конфигурации сети и обеспечивает подключение сетевых устройств. В концентраторе для каждого узла (ПК, принтеры, серверы доступа, телефоны и пр.) должен быть предусмотрен отдельный порт.

Наращиваемые концентраторы представляют собой отдельные модули, которые объединяются при помощи быстродействующей системы связи. Такие концентраторы предоставляют удобный способ поэтапного расширения возможностей и мощности ЛВС.

Концентратор осуществляет электрическую развязку отрезков кабеля до каждого узла, поэтому короткое замыкание на одном из отрезков не выведет из строя всю ЛВС.

Концентраторы образуют из отдельных физических отрезков кабеля общую среду передачи данных – **логический сегмент**.

Логический сегмент также называют *доменом коллизий*, поскольку при попытке одновременной передачи данных любых двух компьютеров этого сегмента, хотя бы и принадлежащих разным физическим сегментам, возникает блокировка передающей среды. Следует особо подчеркнуть, что, какую бы сложную структуру ни образовывали концентраторы, например при иерархическом соединении, все компьютеры, подключенные к ним, образуют единый логический сегмент, в котором любая пара взаимодействующих компьютеров полностью блокирует возможность обмена данными для других компьютеров.

Концентраторы поддерживают технологию Plug and Play и не требуют какой-либо установки параметров. Необходимо просто спланировать свою сеть и вставить разъемы в порты концентратора и компьютеров.

9.2.1. Планирование сети с концентратором

При выборе места для установки концентратора во внимание принимаются следующие аспекты:

– местоположение;
– расстояние;
– питание.

Выбор места установки концентратора является наиболее важным этапом планирования небольшой сети. Хаб разумно расположить вблизи геометрического центра сети (на одинаковом расстоянии от всех компьютеров). Такое расположение позволит минимизировать расход кабеля, длина которого от концентратора до любого из подключаемых к сети компьютеров или периферийных устройств не должна превышать 100 м.

Концентратор можно поставить на стол или закрепить его на стене с помощью входящих в комплект хаба скоб. Установка хаба на стене позволяет упростить подключение кабелей, если они уже проложены в офисе.

При планировании сети необходимо предусматривать возможность наращивания (каскадирования) хабов.

9.2.2. Преимущества концентратора

Концентраторы имеют много преимуществ. Во-первых, в сети используется топология «звезда», при которой соединения с компьютерами образуют лучи, а хаб является центром звезды. Такая топология упрощает установку и управление сети. Любые перемещения компьютеров или добавление в сеть новых узлов при такой топологии весьма несложно выполнить. Кроме того, эта топология значительно надежнее, поскольку при любом повреждении кабельной системы сеть сохраняет работоспособность (перестает работать лишь поврежденный луч). Светодиодные индикаторы хаба позволяют контролировать состояние сети и легко обнаруживать неполадки.

Различные производители концентраторов реализуют в своих устройствах разные наборы вспомогательных функций, но наиболее часто встречаются следующие:

– объединение сегментов с различными физическими средами (например, коаксиал, витая пара и оптоволокно) в единый логический сегмент;

– автосегментация портов – автоматическое отключение порта при его некорректном поведении (повреждение кабеля, интенсивная генерация пакетов ошибочной длины и т. п.);

– поддержка между концентраторами резервных связей, которые используются при отказе основных;

– защита передаваемых по сети данных от несанкционированного доступа (например, путем искажения поля данных в кадрах, повторяемых на портах, не содержащих компьютера с адресом назначения);

– поддержка средств управления сетями – протокола SNMP (Simple Network Management Protocol), *баз управляющей информации* MIB (Management Information Base).

9.2.3. Многосегментные концентраторы

При рассмотрении некоторых моделей концентраторов возникает вопрос: зачем в этой модели имеется такое большое количество портов, например 192 или 240? Имеет ли смысл разделять среду в 10 или 100 Мбит/с между таким большим количеством станций? В таких концентраторах имеется несколько несвязанных внутренних шин, которые предназначены для создания нескольких разделяемых сред. Например, концентратор имеет три внутренние шины Ethernet. Если в таком концентраторе 72 порта, то каждый из этих портов может быть связан с любой из трех внутренних шин. Между собой компьютеры, подключенные к разным сегментам, общаться через концентратор не могут, так как шины внутри концентратора никак не связаны.

Многосегментные концентраторы нужны для создания разделяемых сегментов, состав которых может легко изменяться. Большинство многосегментных концентраторов, таких как System 5000 компании Nortel Networks или PortSwitch Hub компании 3Com, позволяют выполнять операцию соединения порта с одной из внутренних шин чисто программным способом, например с помощью локального конфигурирования через консольный порт. В результате администратор сети может присоединять компьютеры пользователей к любым портам концентратора, а затем с помощью программы конфигурирования концентратора управлять составом каждого сегмента. Если завтра сегмент 1 станет перегруженным, то его компьютеры можно распределить между оставшимися сегментами концентратора.

Многосегментные концентраторы – это программируемая основа больших сетей.

Возможность многосегментного концентратора программно изменять связи портов с внутренними шинами называется **конфигурационной коммутацией** (Configuration Switching, CS).

Для соединения сегментов между собой нужны устройства другого типа – ***мосты*** или ***коммутаторы*** (см. подглаву 9.3). Такое межсетевое устройство должно подключаться к нескольким портам многосегментного концентратора, подсоединенным к разным внутренним шинам, и выполнять передачу кадров или пакетов между сегментами точно так же, как если бы они были образованы отдельными устройствами-концентраторами.

Для крупных сетей многосегментный концентратор играет роль *интеллектуального кроссового шкафа*, который выполняет новое соединение не за счет механического перемещения вилки кабеля в новый порт, а за счет программного изменения внутренней конфигурации устройства.

9.2.4. Конструктивное исполнение концентраторов

На конструктивное устройство концентраторов большое влияние оказывает их область применения. Концентраторы рабочих групп чаще всего выпускаются как устройства с фиксированным количеством портов, корпоративные концентраторы – как модульные устройства на основе шасси, а концентраторы отделов могут иметь стековую конструкцию. Такое деление не является жестким, поэтому в качестве корпоративного концентратора может использоваться, например, модульный концентратор.

Концентратор с фиксированным количеством портов – это наиболее простое конструктивное исполнение, когда устройство представляет собой отдельный корпус со всеми необходимыми элементами (портами, органами индикации и управления, блоком питания), и эти элементы заменять нельзя. Обычно все порты такого концентратора поддерживают одну среду передачи, общее количество портов изменяется от 4–8 до 24. Один порт может быть специально выделен для подключения концентратора к магистрали сети или же для объединения концентраторов (в качестве такого порта часто используется порт с интерфейсом AUI, в этом случае применение соответствующего трансивера позволяет подключить концентратор к практически любой физической среде передачи данных).

Модульный концентратор выполняется в виде отдельных модулей с фиксированным количеством портов, устанавливаемых на общее шасси.

Шасси имеет внутреннюю шину для объединения отдельных модулей в единый повторитель. Часто такие концентраторы являются многосегментными, тогда в пределах одного модульного концентратора работает несколько несвязанных между собой повторителей. Для модульного концентратора могут существовать различные типы модулей, отличающиеся количеством портов и типом поддерживаемой физической среды. Часто агент протокола SNMP выполняется в виде отдельного модуля, при установке которого концентратор превращается в интеллектуальное устройство. Модульные концентраторы позволяют более точно подобрать необходимую для конкретного применения конфигурацию концентратора, а также гибко и с минимальными затратами реагировать на изменения конфигурации сети.

Ввиду ответственной работы, которую выполняют корпоративные модульные концентраторы, они снабжаются модулем управления, системой терморегулирования, избыточными источниками питания и возможностью замены модулей «на ходу».

Недостатком концентратора на основе шасси является высокая начальная стоимость такого устройства для случая, когда предприятию на первом этапе создания сети нужно установить всего 1–2 модуля. Высокая стоимость шасси вызвана тем, что оно поставляется вместе со всеми общими устройствами, такими как избыточные источники питания и т. п. Поэтому для сетей средних размеров большую популярность завоевали стековые концентраторы.

Стековый концентратор, как и концентратор с фиксированным числом портов, выполнен в виде отдельного корпуса без возможности замены отдельных его модулей.

Однако стековыми эти концентраторы называются не потому, что они устанавливаются один на другой. Такая чисто конструктивная деталь вряд ли удостоилась бы особого внимания, так как установка нескольких устройств одинаковых габаритных размеров в общую стойку практикуется очень давно. Стековые концентраторы снабжены специальными портами и кабелями для объединения нескольких таких корпусов в единый повторитель, который имеет общий блок повторения, обеспечивает общую ресинхронизацию сигналов для всех своих портов и поэтому с точки зрения правила четырех хабов считается одним повторителем.

Если стековые концентраторы имеют несколько внутренних шин, то при соединении в стек эти шины объединяются и становятся общими для всех устройств стека. Число объединяемых в стек корпусов может быть достаточно большим (обычно до 8, но бывает и больше). Стековые концентраторы могут поддерживать различные физические среды передачи, что делает их почти такими же гибкими, как и модульные концентраторы, но при этом стоимость этих устройств в расчете на один порт получается обычно ниже, так как сначала предприятие может купить одно устройство без избыточного шасси, а потом нарастить стек еще несколькими аналогичными устройствами.

Стековые концентраторы, выпускаемые одним производителем, выполняются в едином конструктивном стандарте, что позволяет легко устанавливать их друг на друга, образуя единое настольное устройство, или помещать их в общую стойку. Экономия при организации стека происходит еще и за счет единого для всех устройств стека модуля SNMP-управления (который вставляется в один из корпусов стека как дополнительный модуль), а также общего избыточного источника питания.

Модульно-стековые концентраторы представляют собой модульные концентраторы, объединенные специальными связями в стек. Как правило, корпуса таких концентраторов рассчитаны на небольшое количество модулей (1–3). Эти концентраторы сочетают достоинства концентраторов обоих типов.

Выводы

1. Концентраторы кроме основной функции протокола (побитного повторения кадра на всех или последующем порту) всегда выполняют ряд полезных дополнительных функций, определяемых производителем концентратора. Автосегментация – одна из важнейших дополнительных функций, с помощью которой концентратор отключает порт при обнаружении разнообразных проблем с кабелем и конечным узлом, подключенным к данному порту.

2. В число дополнительных функций входят функции защиты сети от несанкционированного доступа, запрещающие подключение к концентратору компьютеров с неизвестными MAC-адресами,

а также заполняющие нулями поля данных кадров, поступающих не к станции назначения.

3. Стековые концентраторы сочетают преимущества модульных концентраторов и концентраторов с фиксированным количеством портов.

4. Многосегментные концентраторы позволяют делить сеть на сегменты программным способом, без физической перекоммутации устройств.

5. Сложные концентраторы, выполняющие дополнительные функции, обычно могут управляться централизованно по сети по протоколу SNMP.

9.3. Мосты и коммутаторы

9.3.1. Мосты

Мост (Bridge) – ретрансляционная система, соединяющая каналы передачи данных (рис. 9.2).

Рис. 9.2. Внешний вид беспроводного сетевого моста

В соответствии с базовой эталонной моделью взаимодействия открытых систем мост описывается протоколами физического и канального уровней, над которыми располагаются канальные процессы. Мост опирается на пару связываемых им физических средств соединения, которые в этой модели представляют физические каналы. Мост преобразует физический (1А, 1В) и канальный (2А, 2В) уровни различных типов (рис. 9.3). Что касается канального процесса, то он объединяет разнотипные каналы передачи данных в один общий.

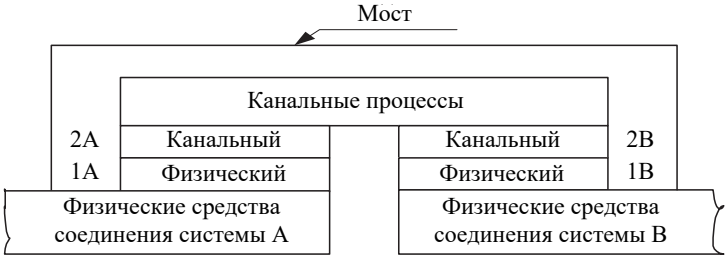


Рис. 9.3. Структура моста

Мост и его быстродействующий аналог – *коммутатор* (Switching Hub) – делят общую среду передачи данных на логические сегменты.

Логический сегмент образуется путем объединения нескольких физических сегментов (отрезков кабеля) с помощью одного или нескольких концентраторов. Каждый логический сегмент подключается к отдельному порту моста/коммутатора. При поступлении кадра на какой-либо из портов мост/коммутатор повторяет этот кадр, но не на всех портах, как это делает концентратор, а только на том порту, к которому подключен сегмент, содержащий компьютер-адресат.

Мосты могут соединять сегменты с разными типами носителей, например 10Base-T, 100Base-T, 1000Base-T (витая пара), 10Base2 (тонкий коаксиальный кабель) и 1000Base-FX (оптоволокно), а также сети с разными методами доступа к каналу, например сети Ethernet (метод доступа CSMA/CD) и Token Ring (метод доступа TPMA).

Мосты используются только для связи локальных сетей с глобальными, т. е. как средства удаленного доступа, поскольку в этом случае необходимости в параллельной передаче между несколькими парами портов просто не возникает (рис. 9.4).

По мере развития оборудование данного типа стало *многопортовым* и получило название ***коммутатор***. Некоторое время оба понятия существовали одновременно, а позднее термин *мост* заменился *коммутатором*. Далее в этой теме будет использоваться термин «коммутатор», поскольку сказанное ниже в равной степени относится и к мостам, и к коммутаторам. Следует отметить, что в последнее время локальные мосты полностью вытеснены коммутаторами.

Нередки случаи, когда необходимо соединить локальные сети, в которых различаются лишь протоколы физического и канального уровней. Протоколы остальных уровней в этих сетях приняты одинаковыми. Такие сети могут быть соединены мостом. Часто мосты

наделяются дополнительными функциями. Такие мосты обладают определенным интеллектом (интеллектом в сетях называют действия, выполняемые устройствами) и фильтруют сквозь себя блоки данных, адресованные абонентским системам, расположенным в той же сети. Для этого в памяти каждого моста имеются адреса систем, включенных в каждую из сетей. Блоки, проходящие через интеллектуальный мост, дважды проверяются, на входе и выходе. Это позволяет предотвращать появление ошибок внутри моста.

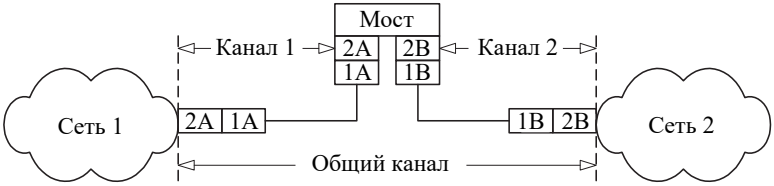


Рис. 9.4. Соединение двух сетей при помощи двух каналов

Мосты не имеют механизмов управления потоками блоков данных. Поэтому может оказаться, что входной поток блоков окажется большим, чем выходной. В этом случае мост не справится с обработкой входного потока, и его буферы могут переполняться. Чтобы этого не произошло, избыточные блоки выбрасываются. Специфические функции выполняет мост в радиосети. Здесь он обеспечивает взаимодействие двух радиоканалов, работающих на разных частотах. Его именуют ***ретранслятором***.

Таким образом, мосты оперируют данными на высоком уровне и имеют совершенно определенное назначение. Они предназначены для соединения сетевых сегментов, имеющих различные физические среды, например для соединения сегмента с оптоволоконным кабелем и сегмента с коаксиальным кабелем. Мосты также могут быть использованы для связи сегментов, имеющих различные протоколы низкого уровня (физического и канального).

9.3.2. Коммутаторы

Коммутатор (Switch, Switching Hub) – устройство, осуществляющее выбор одного из возможных вариантов направления передачи данных (рис. 9.5).

Общая структура коммутатора аналогична стуктуре моста (внешний вид одного из них показан на рис. 9.3), т. е. современные коммутаторы оперируют не только на физическом, но и на канальном уровне модели OSI.

В коммуникационной сети коммутатор является ретрансляционной системой (системой, предназначенной для передачи данных или преобразования протоколов), обладающей свойством прозрачности (т. е. коммутация осуществляется здесь без какой-либо обработки данных).

Рис. 9.5. Внешний вид коммутатора фирмы Cisco

Коммутатор не имеет буферов и не может накапливать данные. Поэтому при использовании коммутатора скорости передачи сигналов в соединяемых каналах передачи данных должны быть одинаковыми. Канальные процессы, реализуемые коммутатором, выполняются специальными интегральными схемами. В отличие от других видов ретрансляционных систем здесь, как правило, не используется программное обеспечение.

Коммутатор может соединять серверы в кластер и быть основой для объединения нескольких рабочих групп. Он направляет пакеты данных между узлами ЛВС. Каждый коммутируемый сегмент получает доступ к каналу передачи данных без конкуренции и видит только тот трафик, который направляется в его сегмент. Коммутатор должен предоставлять каждому порту возможность соединения с максимальной скоростью без конкуренции со стороны других портов (в отличие от совместно используемого концентратора). Обычно в коммутаторах имеются один или два высокоскоростных порта, а также хорошие инструментальные средства управления. Коммутатором можно заменить маршрутизатор, дополнить им наращиваемый маршрутизатор или использовать коммутатор в качестве основы для соединения нескольких концентраторов. Коммутатор может служить

отличным устройством для направления трафика между концентраторами ЛВС рабочей группы и загруженными файл-серверами.

Коммутатор локальной сети (Local Area Network Switch) – устройство, обеспечивающее взаимодействие сегментов одной либо группы локальных сетей.

Коммутатор локальной сети, как и обычный коммутатор, обеспечивает взаимодействие подключенных к нему локальных сетей (рис. 9.6).

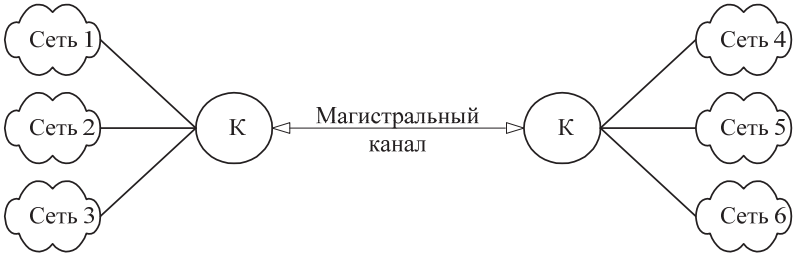


Рис. 9.6. Схема подключения локальных сетей к коммутаторам

В дополнение к основной функции он осуществляет преобразование интерфейсов, если соединяются различные типы сегментов локальной сети. Чаще всего это сети Ethernet, кольцевые сети IBM, сети с оптоволоконным распределенным интерфейсом данных. В перечень функций, выполняемых коммутатором локальной сети, входят:

– обеспечение сквозной коммутации;
– наличие средств маршрутизации;
– поддержка простого протокола управления сетью;
– имитация моста либо маршрутизатора;
– организация виртуальных сетей;
– скоростная ретрансляция блоков данных.

! Несмотря на сходство мостов и коммутаторов, ключевая разница между ними состоит в том, что *мост в каждый момент времени может осуществлять передачу кадров только между одной парой портов, а коммутатор одновременно поддерживает потоки данных между всеми своими портами*.

Другими словами, ***мост передает кадры последовательно, а коммутатор – параллельно***.

9.3.3. Техническая реализация и дополнительные функции коммутаторов

В настоящее время существует большое разнообразие моделей коммутаторов. Они отличаются как внутренней организацией, так и набором выполняемых дополнительных функций, таких как трансляция протоколов, поддержка алгоритма покрывающего дерева, образование виртуальных логических сетей и ряда других.

Современные коммутаторы используют в качестве базовой одну из трех схем, на которой строится такой узел обмена:

– коммутационная матрица;

– разделяемая многовходовая память;

– общая шина.

Часто эти три способа взаимодействия комбинируются в одном коммутаторе.

В конструктивном отношении коммутаторы делятся на следующие типы:

– автономные коммутаторы с фиксированным количеством портов;

– модульные коммутаторы на основе шасси;

– коммутаторы с фиксированным количеством портов, собираемые в стек.

Автономный коммутатор обычно предназначен для организации небольших рабочих групп.

Модульные коммутаторы на основе шасси чаще всего предназначены для применения на магистрали сети. Поэтому они выполняются на основе какой-либо комбинированной схемы, в которой взаимодействие модулей организуется по быстродействующей шине или же на основе быстрой разделяемой памяти большого объема. Модули такого коммутатора выполняются на основе технологии «hot swap», т. е. допускают замену на ходу, без выключения коммутатора, так как центральное коммуникационное устройство сети не должно иметь перерывов в работе. Шасси обычно снабжается резервированными источниками питания и резервированными вентиляторами в тех же целях.

С технической точки зрения определенный интерес представляют ***стековые коммутаторы***. Эти устройства представляют собой коммутаторы, которые могут работать автономно, так как выполнены в отдельном корпусе, но имеют специальные интерфейсы, которые позволяют их объединять в общую систему, работающую

как единый коммутатор. Говорят, что в этом случае отдельные коммутаторы образуют стек.

Стековые коммутаторы применяются для создания сетей рабочих групп или, например, отделов и т. д.

Выводы

1. Логическая структуризация сети необходима при построении сетей средних и крупных размеров. Использование общей разделяемой среды приемлемо только для сети, состоящей из 5–10 компьютеров. Деление сети на логические сегменты повышает производительность, надежность, гибкость построения и управляемость сети.

2. Для логической структуризации сети применяются мосты и их современные преемники – коммутаторы. Они позволяют разделить сеть на логические сегменты с помощью минимума средств – только на основе протоколов канального уровня. Кроме того, эти устройства не требуют конфигурирования.

3. Логические сегменты, построенные на основе коммутаторов, являются строительными элементами более крупных сетей, объединяемых маршрутизаторами.

4. Основное различие между коммутатором и мостом состоит в том, что мост в каждый момент времени может осуществлять передачу кадров только между одной парой портов, а коммутатор одновременно поддерживает потоки данных между всеми своими портами. Другими словами, мост передает кадры последовательно, а коммутатор параллельно.

5. Коммутаторы – наиболее быстродействующие современные коммуникационные устройства, они позволяют соединять высокоскоростные сегменты без блокирования (уменьшения пропускной способности) межсегментного трафика.

6. Коммутаторы связывают процессоры портов по трем основным схемам – коммутационная матрица, общая шина и разделяемая память. В коммутаторах с фиксированным количеством портов обычно используется коммутационная матрица, а в модульных коммутаторах – сочетание коммутационной матрицы в отдельных модулях с общей шиной и разделяемой памятью для связи модулей.

9.4. Маршрутизаторы и шлюзы

9.4.1. Структура маршрутизатора

Маршрутизатор (или **роутер** – Router) – это ретрансляционная система, которая соединяет две коммуникационные сети либо их части.

Каждый маршрутизатор реализует протоколы физического (1А, 1В), канального (2А, 2В) и сетевого (3А, 3В) уровней, как показано на рис. 9.7.

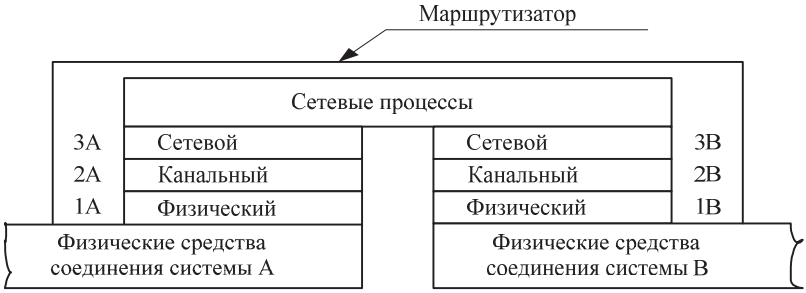


Рис. 9.7. Структура маршрутизатора

Специальные сетевые процессы соединяют части коммутатора в единое целое. Физический, канальный и сетевой протоколы в разных сетях различны. Поэтому соединение пар коммуникационных сетей происходит через маршрутизаторы, которые осуществляют необходимое преобразование указанных протоколов. Сетевые процессы выполняют взаимодействие соединяемых сетей.

Маршрутизатор работает с несколькими каналами, направляя в какой-нибудь из них очередной блок данных. Маршрутизаторы обмениваются информацией об изменениях структуры сетей, трафике. Благодаря этому выбирается оптимальный маршрут следования блока данных в разных сетях от абонентской системы-отправителя к системе-получателю. Маршрутизаторы обеспечивают также соединение административно независимых коммуникационных сетей.

Архитектура маршрутизатора также используется при создании узла коммутации пакетов.

9.4.2. Различие между маршрутизаторами и мостами

Маршрутизаторы превосходят мосты своей способностью фильтровать и направлять пакеты данных на сети.

Так как маршрутизаторы работают на сетевом уровне, они могут соединять сети, использующие разную сетевую архитектуру, методы доступа к каналам связи и протоколы.

Маршрутизаторы не обладают такой способностью к анализу сообщений как мосты, но зато могут принимать решение о выборе оптимального пути для данных между двумя сетевыми сегментами.

Мосты принимают решение по поводу адресации каждого из поступивших пакетов данных, переправлять его через мост или нет в зависимости от адреса назначения. Маршрутизаторы же выбирают из таблицы маршрутов наилучший для данного пакета.

В «поле зрения» маршрутизаторов находятся только пакеты, адресованные к ним предыдущими маршрутизаторами, в то время как мосты должны обрабатывать все пакеты сообщений в сегменте сети, к которому они подключены.

Тип топологии или протокола уровня доступа к сети не имеет значения для маршрутизаторов, так как они работают на уровень выше, чем мосты (сетевой уровень модели OSI). Маршрутизаторы часто используются для связи между сегментами с одинаковыми протоколами высокого уровня.

Наиболее распространенным транспортным протоколом, который используют маршрутизаторы, является IPX фирмы Novell или TCP фирмы Microsoft.

Важно отметить, что *для работы маршрутизаторов требуется один и тот же протокол во всех сегментах, с которыми он связан*.

Для управления загруженностью трафика сегмента сети лучше использовать мосты.

9.4.3. Шлюзы

Шлюз (Gateway) – ретрансляционная система, обеспечивающая взаимодействие информационных сетей. Структура шлюза представлена на рис. 9.8.

Шлюз является наиболее сложной ретрансляционной системой, обеспечивающей взаимодействие сетей с различными наборами протоколов всех семи уровней. В свою очередь, наборы протоколов могут опираться на различные типы физических средств соединения.

Рис. 9.8. Структура шлюза

При соединении информационных сетей часть уровней может иметь одни и те же протоколы. В этом случае сети соединяются не при помощи шлюза, а на основе более простых ретрансляционных систем, именуемых маршрутизаторами и мостами.

> ! *Шлюзы оперируют на верхних уровнях модели OSI* (сеансовом, представительском и прикладном) и представляют наиболее развитый метод подсоединения сетевых сегментов и компьютерных сетей. Необходимость в сетевых шлюзах возникает при объединении двух систем, имеющих различную архитектуру.

Например, шлюз приходится использовать для соединения сети (протокол TCP/IP) и большой ЭВМ со стандартом SNA. Эти две

архитектуры не имеют ничего общего, и потому требуется полностью переводить весь поток данных, проходящих между двумя системами.

В качестве шлюза обычно используется выделенный компьютер, на котором запущено программное обеспечение шлюза и производятся преобразования, позволяющие взаимодействовать нескольким системам в сети. Другой функцией шлюзов является преобразование протоколов. При получении сообщения IPX/SPX для клиента TCP/IP шлюз преобразует сообщения в протокол TCP/IP.

Выводы

1. Соединение коммуникационных сетей осуществляется через маршрутизаторы, которые выполняют необходимое преобразование определенных протоколов.

2. Маршрутизаторы в процессе работы обмениваются информацией об изменениях структуры сетей, трафике и их состоянии, благодаря чему может быть осуществлен выбор оптимального маршрута следования блока данных.

3. Тип топологии или протокола уровня доступа к сети не имеет значения для маршрутизаторов, так как они работают на сетевом уровне модели OSI.

4. Шлюз является ретрансляционной системой, обеспечивающей взаимодействие информационных сетей. Обычно шлюзом являются серверы с настроенной службой маршрутизации, поэтому шлюзы функционируют на всех семи уровнях модели OSI.

5. Шлюзы сложны в установке и настройке. Шлюзы работают медленнее, чем маршрутизаторы.

9.5. Оборудование для сетей Wi-Fi

Сети Wi-Fi (см. подглаву 7.4) отождествляются с аббревиатурой *WLAN* (Wireless Local Area Network). Для организации *сетей Wi-Fi* (Wireless Fidelity – беспроводное соответствие) необходимы сетевые

карты, точки доступа и антенны Wi-Fi. Наличие точек доступа отпадает, когда речь идет об очень малых сетях, размещенных в одном помещении. Использование точек доступа позволяет более гибко настроить сеть, объединить клиентов проводных и беспроводных сетей, а также установить связь с удаленными объектами (внешнее исполнение).

Сетевые Wi-Fi-карты, по сути, мало чем отличаются от обычных сетевых карт, за исключением некоторых особенностей настройки. Сетевые Wi-Fi-карты представлены в трех основных вариантах исполнения – внутренние PCI-карты, CardBus и USB-адаптеры. Также существуют адаптеры в CompactFlash форм-факторе.

Адаптеры различаются по платформе, в которой они используются: PCI – в настольном компьютере, CardBus – в ноутбуке, CompactFlash – в карманном компьютере, USB – универсален.

Принцип построения и настройки сетей един и не зависит от форм-фактора Wi-Fi-адаптера.

Необходимо отметить, что тип адаптера влияет лишь на излучаемую мощность передатчика и чувствительность приемника, а также возможность использования внешней антенны.

9.5.1. Wi-Fi-точки доступа

Wi-Fi-точки доступа – устройства, позволяющие объединять клиентов сети (как проводной, так и беспроводной) в единую сеть.

Другими словами, для Wi-Fi-клиентов точка доступа – это своеобразный хаб (концентратор). Для клиентов проводной сети – возможность выхода в сеть к беспроводным клиентам.

Wi-Fi-точки доступа представлены в двух основных вариантах исполнения – для использования внутри помещений и для внешнего использования. Существуют варианты исполнения точек доступа, совмещенных с панельными антеннами для внешнего использования.

При рассмотрении точек доступа исполнение играет очень важную роль, т. е. внутриофисные точки доступа нельзя использовать на улице, а внешние крайне нецелесообразно использовать внутри

помещений. Также исполнение Wi-Fi-точек доступа определяет их функциональные возможности.

В беспроводных сетях используются два частотных диапазона: 2,4 и 5 ГГц (см. табл. 7.4). Беспроводные сети стандарта 802.11b/g работают в диапазоне 2,4 ГГц, сети стандарта 802.11a – 5 ГГц, а сети стандарта 802.11n могут работать как в диапазоне 2,4, так и 5 ГГц.

Используемый частотный диапазон и эксплуатационные ограничения в разных странах могут быть различные.

Важным вопросом при организации WLAN-сетей является дальность покрытия. На этот параметр влияет сразу несколько факторов. Перечислим основные:

1) используемая частота (чем она больше, тем меньшая дальность действия радиоволн);

2) наличие преград между узлами сети (различные материалы по-разному поглощают и отражают сигналы);

3) режим функционирования (Infrastructure Mode или Ad-Hoc);

4) мощность электронного оборудования;

5) вид и качество антенн.

Если рассматривать идеальные условия, то зона покрытия с одной точкой доступа будет иметь следующий средний радиус:

– для сети стандарта IEEE 802.11a – порядка 50 м;

– для сетей 802.11b и 802.11g – порядка 100 м.

Внутриофисные точки доступа служат для объединения Wi-Fi-клиентов внутри помещений. Они оснащены функциями фильтров, создания виртуальных сетей и т. д. Но зачастую используются точки доступа с более широкими возможностями – WAN-порт, firewall, FTP-сервер и т. д.

Необходимо помнить, что беспроводные устройства Wi-Fi имеют ограниченный радиус действия. Например, домашний интернет-центр с точкой доступа Wi-Fi стандарта 802.11b/g имеет радиус действия до 60 м в помещении и до 400 м вне помещения. В помещении дальность действия беспроводной точки доступа может быть ограничена несколькими десятками метров – в зависимости от конфигурации комнат, наличия капитальных стен и их количества, а также других препятствий.

Различные препятствия (стены, потолки, мебель, металлические двери и т. д.), расположенные между Wi-Fi-устройствами, могут частично или значительно отражать/поглощать радиосигналы, что приводит к частичной или полной потере сигнала.

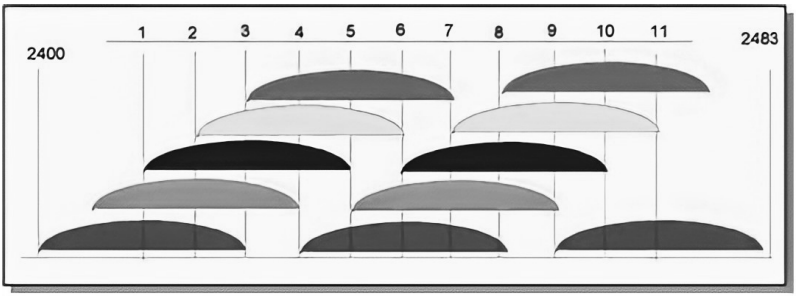
В городах с многоэтажной застройкой основным препятствием для радиосигнала являются здания. Наличие капитальных стен (бетон + арматура), листового металла, штукатурки на стенах, стальных каркасов и т. п. влияет на качество радиосигнала и может значительно ухудшать работу Wi-Fi-устройств. Внутри помещения причиной помех радиосигнала также могут являться зеркала и тонированные окна. Даже человеческое тело ослабляет сигнал примерно на 3 дБ.

Внешние Wi-Fi-точки доступа служат для объединения Wi-Fi-клиентов вне помещений, например в публичных местах. Внешние точки доступа имеют защищенное исполнение, более жесткие эксплуатационные характеристики и т. д. При использовании нескольких внешних точек доступа можно соединить достаточно удаленные объекты и создать публичный *хот-спот*. Внешние Wi-Fi-точки доступа отличаются и большей излучаемой мощностью. Ко всем внешним точкам доступа можно подключить дополнительные антенны, что позволяет расширить зону покрытия Wi-Fi-сети.

Вне помещений влиять на качество передаваемого сигнала может ландшафт местности (например, деревья, леса, холмы). Атмосферные помехи (дождь, гроза, снегопад) также могут являться причиной уменьшения производительности беспроводной сети (в случае если радиосигнал передается вне помещений).

В полосе частот 2,4 ГГц доступны 11 или 13 каналов шириной 20 МГц (802.11b/g/n) или 40 МГц (IEEE 802.11n) с интервалами 5 МГц между ними. Беспроводное устройство, использующее один из частотных каналов, создает значительные помехи на соседние каналы. Например, если точка доступа использует канал 6, то она оказывает сильные помехи на каналы 5 и 7, а также, уже в меньшей степени, – на каналы 4 и 8. Для исключения взаимных помех между каналами необходимо, чтобы их несущие частоты отстояли друг от друга на 25 МГц (5 межканальных интервалов).

На рис. 9.9 показаны спектры 11 каналов. Кодировка обозначает группы непересекающихся каналов – [1, 6, 11], [2, 7], [3, 8], [4, 9], [5, 10]. Беспроводные сети, расположенные в пределах одной зоны покрытия, рекомендуется настраивать на непересекающиеся каналы, на которых будет наблюдаться меньше *интерференции* и *коллизий* (конфликтов). Для протокола IEEE 802.11n номера непересекающихся каналов – 1, 6 и 11 (для ширины канала 20 МГц), с шириной канала 40 МГц – это каналы 3 и 11.



Источник: https://help.keenetic.com/hc/ru/articles/213968729.

Рис. 9.9. Выбор частотных характеристик
непересекающихся каналов

Интернет-центр (маршрутизатор или роутер) с беспроводной точкой доступа Wi-Fi позволяет организовать высокоскоростную беспроводную сеть для совместной работы в Интернете и домашней сети с ноутбуков, планшетов, смартфонов и других устройств. Доступ в Интернет по беспроводной сети можно обеспечить любым устройствам, оснащенным модулями и адаптерами Wi-Fi: телевизору с функцией Smart TV, игровой приставке, умным часам и др.

Следует обратить внимание на то, что многие точки доступа могут выступать в роли беспроводного клиента, что значительно расширяет область их применения.

> ***Нужно помнить, что любая общедоступная точка доступа Wi-Fi – рай для злоумышленников.***

Также необходимо принимать во внимание среду, в которой размещена точка доступа по отношению к приемному устройству. Например, если на открытом пространстве радиус действия сигнала Wi-Fi – до 200 м, то после прохождения одной межкомнатной стены он уменьшится до 200 м · 15% = 30 м. После второй еще раз: 30 м · 15% = 4,5 м. А после третьей: 4,5 м · 15% = 0,67 м. Таким образом, можно предположить, что через две межкомнатные стены (толщиной не более 15 см) сеть Wi-Fi будет работать, а вот через три стены скорее всего соединение установить не получится.

Необхоимо также помнить о влиянии на качество сигнала Wi-Fi различных электронных устройств.

Bluetooth-устройства, беспроводные клавиатуры и мыши работают в частотном диапазоне 2,4 ГГц, а следовательно, могут оказывать влияние на работу точки доступа и других Wi-Fi-устройств.

Микроволновые СВЧ-печи также могут ослаблять уровень сигнала Wi-Fi, так как обычно работают в диапазоне 2,4 ГГц, как и радионяни, в результате чего ухудшается качество беспроводной связи.

Внешние источники электрического напряжения, такие как линии электропередач и силовые подстанции, могут также являться источниками помех.

9.5.2. Wi-Fi-антенны

Внешние Wi-Fi-антенны служат для передачи и приема сигнала, усиление которого в режиме передачи позволяет увеличить зону покрытия Wi-Fi-сетей.

В основном распространены *пассивные антенны – круговые*, или *всенаправленные* (рис. 9.10), и *направленные* (рис. 9.11). Основное различие – характер распространения волн антенной. Круговая антенна излучает сигнал по кругу 360° (горизонталь), а направленная лишь на определенный сектор.

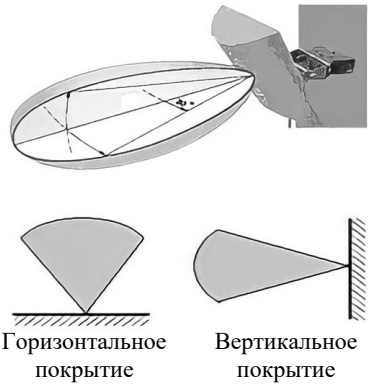


Рис. 9.10. Направленные Wi-Fi-антенны

Wi-Fi-антенны характеризуются следующими параметрами.

1. *Поляризация* – отражает специфику распространения радиоволн. Поляризация бывает горизонтальная (линейная) и вертикальная.

При проектировании сети данный аспект необходимо учитывать при подборе антенн, поляризация обязательно должна совпадать.

2. *HPBW* (Half-Power Beamwidth) *по горизонтали* – угол распространения волн по горизонтали. Для всех круговых антенн равен 360°. Для направленных Wi-Fi-антенн значительно меньше.

3. *HPBW по вертикали* – угол распространения волн по вертикали. При малом угле возможно возникновение мертвых зон.

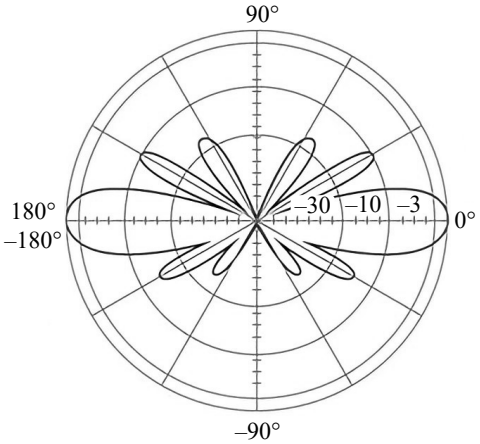


Рис. 9.11. Диаграмма направленности круговой Wi-Fi-антенны (0°)

4. *Усиление* – характеризует усиление сигнала. Чем больше данный параметр, тем на большем расстоянии можно установить связь с сетью.

Удлинительные провода для антенн используются, если антенна удалена от точки доступа или сетевой карты. Особое внимание следует уделить разъемам, так как у разных производителей они могут различаться. Провода применяются специальные – СВЧ, их длина должна быть как можно меньше.

9.5.3. Принципы организации беспроводных сетей

Используют два вида организации беспроводных сетей: *Ad-Hoc* (Independent Basic Service Set, IBSS или Peer-to-Peer) и *Infrastructure Mode* (см. п. 7.4.3).

Режим Ad-Hoc основан на использовании простейшей структуры локальной сети, при которой узлы сети (ноутбуки или компьютеры) связываются напрямую друг с другом. Такая структура удобна для быстрого развертывания сетей. Для ее организации требуется минимум оборудования – достаточно адаптера WLAN на каждом узле.

В ***режиме Infrastructure Mode*** узлы сети связаны друг с другом не напрямую, а через точку доступа, так называемый *Access Point*. Различают два режима взаимодействия с точками доступа – *BSS* (Basic Service Set) и *ESS* (Extended Service Set).

В режиме BSS все узлы связаны между собой через одну точку доступа, которая может играть роль моста для соединения с внешней кабельной сетью.

Режим ESS представляет собой объединение нескольких точек доступа, т. е. связывает несколько сетей BSS. В этом случае точки доступа могут взаимодействовать и друг с другом. Расширенный режим удобно использовать тогда, когда необходимо объединить в одну сеть несколько пользователей или подключить несколько проводных сетей.

За счет увеличения количества точек доступа (в режиме Infrastructure ESS) можно расширять зоны покрытия сети на всю необходимую область охвата.

Выводы

1. Для организации беспроводных сетей Wi-Fi необходимы соответствующие сетевые карты, точки доступа и антенны.

2. Wi-Fi-точки доступа представляют собой устройства, позволяющие объединять клиентов сети (как проводной, так и беспроводной) в единую сеть. Wi-Fi-точки доступа разделяются по исполнению на два основных варианта – для использования внутри помещений и для внешнего использования.

3. Wi-Fi-антенны служат для усиления сигнала, что позволяет увеличить зону покрытия Wi-Fi-сетей.

4. Выделяют два основных вида организации беспроводных сетей: *Ad-Hoc* и *Infrastructure Mode*. Режим Ad-Hoc является простейшей

структурой локальной сети, при которой узлы сети связываются напрямую друг с другом; в режиме Infrastructure Mode узлы сети связаны друг с другом через точку доступа.

5. Наиболее актуальной проблемой функционирования Wi-Fi-сетей является обеспечение требуемого уровня безопасности.

Контрольные вопросы

1. Каково назначение сетевого адаптера?
2. Какие параметры следует устанавливать у сетевого адаптера?
3. Перечислите функции сетевых адаптеров.
4. Что такое физический адрес адаптера?
5. Как определить физический адрес адаптера?
6. Какие есть типы сетевых адаптеров?
7. На каком из уровней сетевой модели OSI необходим сетевой адаптер?
8. Каково назначение повторителя?
9. В каких случаях ставят сетевой повторитель?
10. Что такое сетевой концентратор и каково его назначение?
11. На каком уровне сетевой модели OSI используется Hub?
12. Каково назначение моста?
13. На каком уровне сетевой модели OSI используется мост?
14. Какие сегменты сети может соединять мост?
15. Объясните назначение коммутатора.
16. На каком уровне сетевой модели OSI необходим коммутатор?
17. Каково различие между мостом и коммутатором?
18. Объясните назначение маршрутизатора.
19. На каком уровне сетевой модели OSI используется маршрутизатор?
20. Каково различие между маршрутизаторами и мостами?
21. Что такое шлюз и каково его назначение?
22. На каком уровне сетевой модели OSI используется шлюз?
23. Оборудование для организации беспроводных сетей Wi-Fi и его назначение.
24. Приведите классификацию Wi-Fi-антенн.
25. Перечислите виды организации беспроводных сетей.

10

СОВРЕМЕННЫЕ И ПЕРСПЕКТИВНЫЕ СЕТЕВЫЕ ТЕХНОЛОГИИ

10.1. Беспроводные сотовые сети

10.1.1. Организация сотовой сети

Сотовой радиосвязью (Cellular Radio) называется технология, разработанная для увеличения пропускной способности мобильных радиотелефонных услуг.

До введения сотовой радиосвязи для предоставления этих услуг требовались передатчики и приемники высокой мощности. Типичная система связи могла обслуживать около 25 каналов и имела *эффективный радиус действия* около 80 км. Для увеличения *пропускной способности* такой системы нужно было использовать оборудование более низкой мощности и, следовательно, меньшего радиуса действия при участии в ней нескольких передатчиков и приемников.

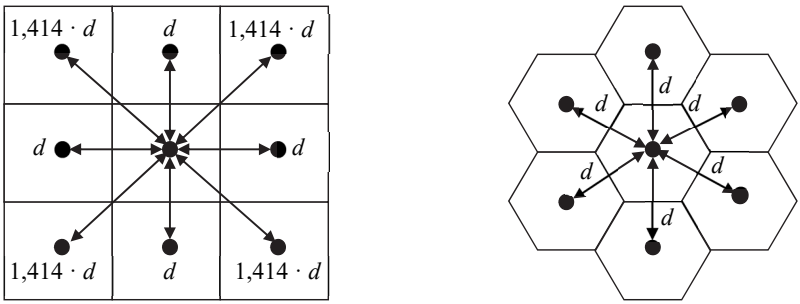
В начале данного раздела кратко рассматривается организация сотовой системы, после чего анализируются некоторые особенности ее внедрения.

Принцип организации сотовой связи состоит в использовании множества маломощных (100 Вт и ниже) передатчиков.

Поскольку диапазон действия таких передатчиков довольно мал, зону обслуживания системы можно разбивать на *ячейки*, каждая из которых будет обслуживаться собственной антенной. Каждая ячейка, которой выделяется своя полоса частот, обслуживается базовой станцией, состоящей из передатчика, приемника и модуля управления. Смежные ячейки используют разные частоты, чтобы избежать появления интерференции или перекрестных помех. В то же время ячейки, находящиеся на довольно большом расстоянии друг от друга, могут использовать одинаковые полосы частот.

При проектировании такой системы первое, что нужно сделать, – это решить, какую форму должны иметь ячейки, на которые будет разбита зона обслуживания. Самым простым решением была бы сетка, состоящая из квадратных ячеек (рис. 10.1, *а*). Однако такая геометрическая форма оказалась не идеальной. Если сторона квадратной ячейки равна *d*, тогда ячейка будет иметь четыре соседа на расстоянии *d* и четыре – на расстоянии $\sqrt{2 \cdot d}$. В то же время, если пользователь мобильных услуг находится в пределах одной ячейки и движется по направлению к ее границе, было бы лучше, чтобы все смежные антенны находились на равных расстояниях друг от друга. В этом случае проще определить момент, в который следует переключать пользователя на другую антенну, а также выбрать новую антенну.

Равное расстояние между смежными антеннами достигается только в шестиугольной схеме (рис. 10.1, *б*). Радиус шестиугольника определяется как радиус окружности, описанной вокруг него (эта величина равна расстоянию от центра фигуры до каждой из ее вершин, а также длине стороны шестиугольника). Для ячейки с радиусом *R* расстояние между центром ячейки и центром любой смежной ячейки равняется $d = \sqrt{3 \cdot R}$.



а *б*

Рис. 10.1. Геометрические структуры сотовых систем:
а – квадратная схема; *б* – гексогональная схема

На практике точная шестиугольная структура не используется. Отклонения от идеальных шестиугольников обусловлены топографическими ограничениями, местными условиями распространения сигнала, а также соображениями целесообразности расположения антенн.

10.1.2. Многократное использование частот и увеличение пропускной способности сети

В каждой ячейке сотовой сети имеется *базовый трансивер* (Base Transceiver). Мощность передаваемых сигналов тщательно регулируется (насколько это возможно для быстро меняющихся условий сред мобильной связи). Как правило, каждой ячейке выделяется 10–50 частот в зависимости от планируемой нагрузки. Кроме того, нужен механизм использования одной и той же частоты в ячейках, расположенных недалеко друг от друга, чтобы одну частоту можно было использовать для нескольких одновременных сеансов связи.

Важным вопросом, разумеется, является определение удаленности двух ячеек, использующих одну частоту, поскольку сигналы этих ячеек не должны интерферировать друг с другом. Были предложены различные модели многократного использования частот, некоторые примеры приведены на рис. 10.2, 10.3 и 10.4.

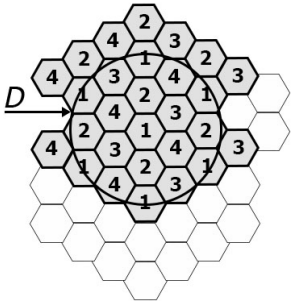


Рис. 10.2. Схема повторного использования частот для $N = 4$

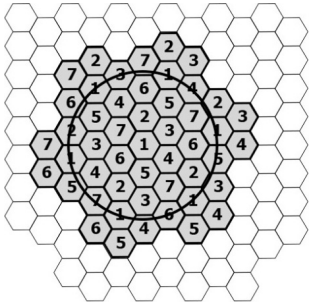


Рис.10.3. Схема повторного использования частот для $N = 7$

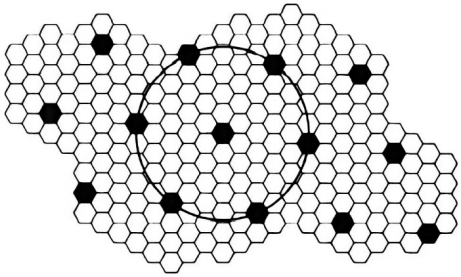


Рис. 10.4. Схема многократного использования частот для $N = 19$

Если схема состоит из N ячеек, для которых выделяется одинаковое количество частот, то каждая ячейка будет иметь K/N частот, где K – общее число частот, выделяемых в системе.

Мобильная телефонная система AMPS (Advanced Mobile Phone Service), где $K = 395$, а $N = 7$, – наименьшая система, в которой можно обеспечить достаточную изоляцию двух сеансов использования одной и той же частоты. Это означает, что в среднем на одну ячейку должно приходиться не более 57 частот. *AMPS* – аналоговый стандарт сотовой связи, относящийся к сетям первого поколения (1G).

Для характеристики повторного использования частоты существуют следующие параметры: D – минимальное расстояние между центрами ячеек (см. рис. 10.2), которые используют одну и ту же полосу частот (называемую группой внутренних каналов); R – радиус ячейки; d – расстояние между центрами смежных ячеек ($d = \sqrt{3} \cdot R$); N – число ячеек в минимальном фрагменте, периодическим повторением которого образуется вся схема (каждая ячейка фрагмента использует уникальную полосу частот). Этот параметр еще называют кратностью использования.

В шестиугольной схеме возможны только следующие значения: $N = I2 + J2 + (I \cdot J)$, где $I, J = 0, 1, 2, 3, \ldots$.

Таким образом, возможными значениями N являются числа 1, 3, 4, 7, 9, 12, 13, 16, 19, 21 и т. д. Верно следующее соотношение:

$$D / R = \sqrt{N}. \qquad (10.1)$$

При определенных условиях последнее соотношение можно записать и по-другому:

$$D / d = \sqrt{N}. \qquad (10.2)$$

Со временем, когда система будет обслуживать все больше клиентов, трафик может распределиться таким образом, что какой-нибудь ячейке для обслуживания звонков не хватит выделенных ей частот. Для выхода из такой ситуации используется несколько подходов.

Добавление новых каналов. Обычно, когда система установлена в определенном регионе, используются не все каналы, и с расширением системы можно просто добавлять новые.

Заимствование частот. В самом простом случае перегруженные ячейки могут использовать частоты смежных ячеек.

Расщепление ячеек. На практике распределение трафика и топография местности неоднородны, что также дает возможность увеличения пропускной способности. Ячейки в областях с повышенным

спросом на услуги мобильной связи можно расщеплять. Как правило, размеры исходных ячеек колеблются от 6,5 до 13 км. Меньшие ячейки также можно разбивать, однако следует помнить, что на практике радиус 1,5 км считается минимальным (см. ниже обсуждение микроячеек). При использовании меньших ячеек нужно уменьшать уровень мощности, чтобы сигнал оставался в пределах ячейки. Кроме того, при движении мобильные устройства переходят из одной ячейки в другую, что требует передачи вызова от одного базового трансивера другому. Этот процесс называется **переключением** (Handoff). Так, по мере уменьшения размера ячейки переключения будут происходить все чаще. При уменьшении радиуса ячейки в F раз размеры покрываемой области уменьшаются в F^2 раз, а требуемое число базовых станций увеличивается в те же F^2 раз.

Разбивка ячеек на секторы. При разбивке на секторы ячейка делится на несколько клиновидных секторов, в каждом из которых остается свой набор каналов. Обычно на ячейку приходится 3–6 секторов. Каждому сектору предоставляется отдельный набор каналов ячейки, а для фокусировки сигнала на отдельных секторах используются направляемые антенны базовой станции.

Микроячейки. По мере уменьшения ячейки антенны перемещаются с крыш высотных зданий и вершин холмов на крыши зданий поменьше или на стены высотных домов и в конце концов оказываются на фонарных столбах, с высоты которых они обслуживают микроячейки. Любое уменьшение размера ячейки сопровождается уменьшением уровня мощности сигналов, излучаемых базовой станцией. Микроячейки полезно располагать на городских улицах в густонаселенных районах, а также внутри больших зданий общественного пользования.

10.1.3. Функционирование сотовой системы

На рис. 10.5 показаны основные элементы сотовой системы.

Примерно в центре каждой ячейки находится ***базовая станция*** (Base Station).

Базовая станция *состоит из антенны*, *контроллера и нескольких трансиверов*, которые служат для связи в каналах, выделенных в этой ячейке.

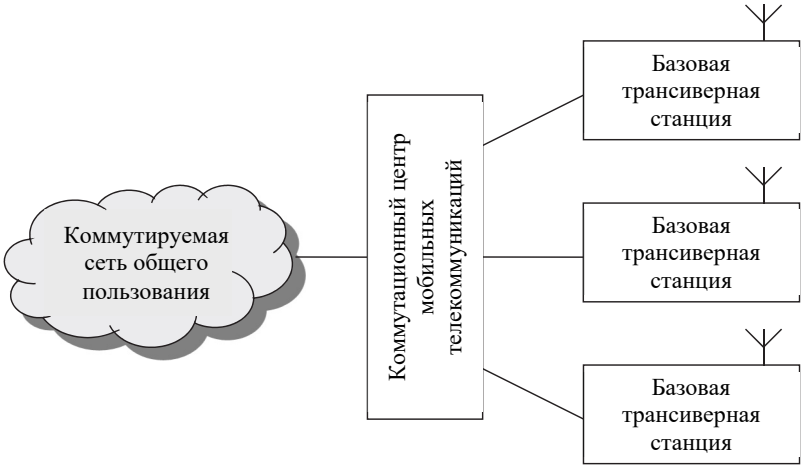


Рис. 10.5. Общая структура системы сотовой связи

Контроллер используется для обработки соединений мобильного устройства с остальной сетью. В любой момент в пределах ячейки могут быть активными и перемещаться несколько пользователей мобильной связи, сообщающихся с базовой станцией.

Каждая *базовая станция* подсоединена к ***коммутатору мобильных телекоммуникаций*** (Mobile Telecommunications Switching Office, MTSO), причем один коммутатор MTSO может обслуживать несколько базовых станций. Обычно связь между коммутатором MTSO и базовой станцией является проводной, хотя возможна также беспроводная связь.

Коммутатор MTSO устанавливает соединение между мобильными устройствами. Кроме того, MTSO соединен с общественной телефонной или телекоммуникационной сетью и может соединять стационарных абонентов с сетью общего пользования и мобильных абонентов с сотовой сетью.

Коммутатор MTSO выделяет для каждого соединения голосовой канал, выполняет переключения и контролирует звонки для передачи информации о счетах.

Работа сотовой системы полностью автоматизирована и не требует от пользователя никаких действий, кроме заказа разговоров и ответа на звонки.

Между мобильным устройством и базовой станцией можно устанавливать каналы связи двух типов: ***каналы управления*** и ***информационные каналы***.

Каналы управления используются для обмена информацией, касающейся заказа и поддержания звонка, а также установления связи между мобильным устройством и ближайшей к нему базовой станцией. Информационные каналы служат для передачи голоса или данных между пользователями.

На рис. 10.6 показаны шаги, которые следует предпринять для обычного соединения двух мобильных пользователей, находящихся в зоне действия одного коммутатора MTSO.

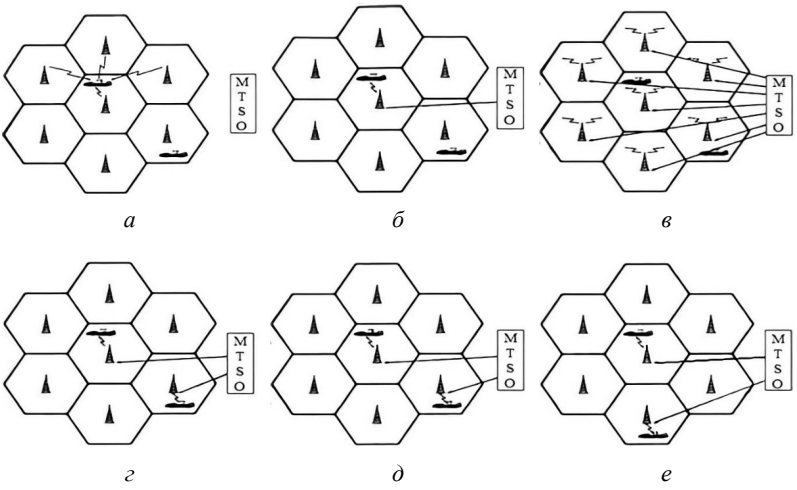


Рис. 10.6. Пример мобильного сотового соединения:
а – поиск сильнейшего сигнала; *б* – запрос на соединение;
в – передача сигнала; *г* – вызов принят; *д* – исходящий вызов;
е – переключение

1. *Инициализация мобильного устройства*. Включенное мобильное устройство проводит *сканирование* и выбирает самый сильный настроечный канал управления, используемый данной системой (рис. 10.6, *а*). Ячейки с различными полосами частот периодически транслируют сигналы в различных настроечных каналах. Приемник

мобильного устройства выбирает самый сильный настроечный канал и начинает его прослушивать.

В результате этой процедуры мобильное устройство автоматически выбирает антенну базовой станции той ячейки, в пределах которой оно будет действовать. Затем выполняется ***квитирование*** (подтверждение установления связи, или *рукопожатие*, – Handshake) между мобильным устройством и коммутатором MTSO, контролирующим данную ячейку, что тоже осуществляется через базовую станцию этой ячейки.

Квитирование используется для опознания пользователя и для регистрации его местоположения.

Все время, пока включено мобильное устройство, эта процедура сканирования периодически повторяется, что позволяет следить за движением устройства. Если устройство входит в новую ячейку, выбирается новая базовая станция. Кроме того, мобильное устройство следит за сигналами избирательного вызова, о чем будет рассказано ниже.

2. *Звонок с мобильного устройства*. Звонок с мобильного устройства начинается с отправки номера вызываемого устройства по предварительно выбранному каналу (рис. 10.6, *б*). Приемник мобильного устройства сначала проверяет, свободен ли настроечный канал, анализируя информацию в прямом (от базовой станции) канале.

Когда обнаруживается, что канал свободен, мобильное устройство может начинать передачу в соответствующем обратном (к базовой станции) канале. Базовая станция в свою очередь отправляет запрос на коммутатор MTSO.

3. *Избирательный вызов*. Далее коммутатор MTSO пытается установить связь с вызываемым устройством. Он отправляет адресное сообщение определенной базовой станции в зависимости от номера вызывающего мобильного устройства (рис. 10.6, *в*). Каждая базовая станция передает сигналы избирательного вызова в собственном выделенном настроечном канале.

4. *Принятие вызова*. Вызываемое мобильное устройство распознает свой номер в настроечном канале, за которым следит в настоящий момент, и отвечает данной базовой станции. Базовая станция отправляет ответ на коммутатор MTSO, который устанавливает канал связи между вызывающей и вызываемой базовыми станциями. В то же самое время коммутатор MTSO выбирает подходящий канал информационного обмена внутри ячейки каждой базовой станции и

уведомляет каждую базовую станцию, которые в свою очередь уведомляют свои мобильные устройства (рис. 10.6, *г*). Оба мобильных устройства настраиваются на выделенные им каналы.

5. *Текущий вызов*. Пока поддерживается соединение, два мобильных устройства обмениваются голосовыми сигналами или данными, проходящими через соответствующие базовые станции и коммутатор MTSO (рис. 10.6, *д*).

6. *Переключение*. Если мобильное устройство во время соединения выходит за пределы одной ячейки и входит в зону действия другой, то старый информационный канал следует заменить каналом, выделенным новой базовой станции в новой ячейке (рис. 10.6, *е*). Система осуществляет это изменение, не прерывая звонка и не беспокоя пользователя.

Система также выполняет некоторые другие функции, не представленные на рис. 10.6.

1. *Блокирование вызова*. Если при звонке с мобильного устройства все информационные каналы, выделенные ближайшей базовой станции, заняты, то мобильное устройство предпринимает предварительно заданное количество последовательных попыток установления связи. После определенного количества неудачных попыток пользователю возвращается сигнал «занято».

2. *Завершение вызова*. Когда один или оба пользователя завершают вызов, об этом узнает коммутатор MTSO и освобождает информационные каналы обеих базовых станций.

3. *Потеря вызова*. Если в определенный период соединения из-за интерференции или слабого сигнала базовая станция не может поддерживать минимально требуемую интенсивность сигнала, то информационный канал связи с пользователем прерывается, о чем уведомляется коммутатор MTSO.

4. *Звонки стационарным и удаленным мобильным абонентам/от стационарных и удаленных мобильных абонентов*. Коммутатор MTSO подключен к коммутатору общественной телефонной сети. Это означает, что коммутатор MTSO может устанавливать соединение между мобильным пользователем из своей зоны и стационарным абонентом через телефонную сеть. Более того, MTSO может соединяться через телефонную сеть либо через выделенные каналы связи с удаленными MTSO и устанавливать соединение между мобильным пользователем из своей зоны и удаленным мобильным пользователем.

10.1.4. Архитектура глобальной системы мобильной связи

До того как была разработана *глобальная система мобильной связи* (GSM – от названия группы Groupe Special Mobile, позже переименован в Global System for Mobile Communications), в странах Европы использовалось множество разных несовместимых сотовых телефонных технологий первого поколения. Стандарт GSM был разработан для внедрения общей технологии второго поколения, чтобы одни и те же абонентские устройства можно было использовать по всему континенту. Эта технология оказалась весьма успешной и стала самым популярным мировым стандартом для систем нового поколения. Впервые стандарт GSM появился в 1990 г. в Европе. Теперь подобные системы внедрены в Северной и Южной Америке, Азии, Северной Африке, а также в Средней Азии и Австралии.

На рис. 10.7 показаны ключевые функциональные элементы системы GSM: линии *Um*, *Abis* и *A* обозначают интерфейсы между функциональными элементами, которые стандартизированы в документации GSM (*AuC – центр аутентификации*; *EIR – регистр идентификации оборудования*; *HLR – регистр исходного положения*; *ME – мобильное оборудование*; *PSTN – общественная коммутируемая телефонная сеть*; *SIM – модуль идентификации абонента*; *VLR – регистр местонахождения посетителей*).

В соответствии с представленной архитектурой в целом можно приобретать оборудование у разных поставщиков и ожидать, что оно будет успешно взаимодействовать. В стандарте GSM определены также дополнительные интерфейсы, но в данном пособии они рассматриваться не будут.

Мобильная станция. Через интерфейс Um, называемый также *радиоинтерфейсом*, мобильная станция сообщается с трансивером базовой станции в той ячейке, в которой находится мобильное устройство. Термином «мобильное оборудование» (Mobile Equipment, ME) обозначается физический терминал, такой как телефон или устройство *персональной службы связи* (Personal Communication Service, PCS), включающее в себя радиотрансивер, процессоры для обработки цифровых сигналов и *модуль идентификации абонента* (Subscriber Identity Module, SIM).

SIM представляет собой портативное устройство, имеющее вид интеллектуальной карточки или встраиваемого модуля, в котором хранятся идентификационный номер абонента, координаты сетей,

которыми разрешено пользоваться абоненту, ключи шифрования и другая информация об абоненте.

Абонентские устройства GSM до вставки модуля SIM абсолютно неотличимы друг от друга. Поэтому путешествующий абонент, захвативший с собой свой модуль SIM, может в разных странах использовать разные устройства, вставляя в них свой модуль. В действительности, за исключением определенных срочных соединений, абонентские устройства не будут работать без вставленного модуля SIM.

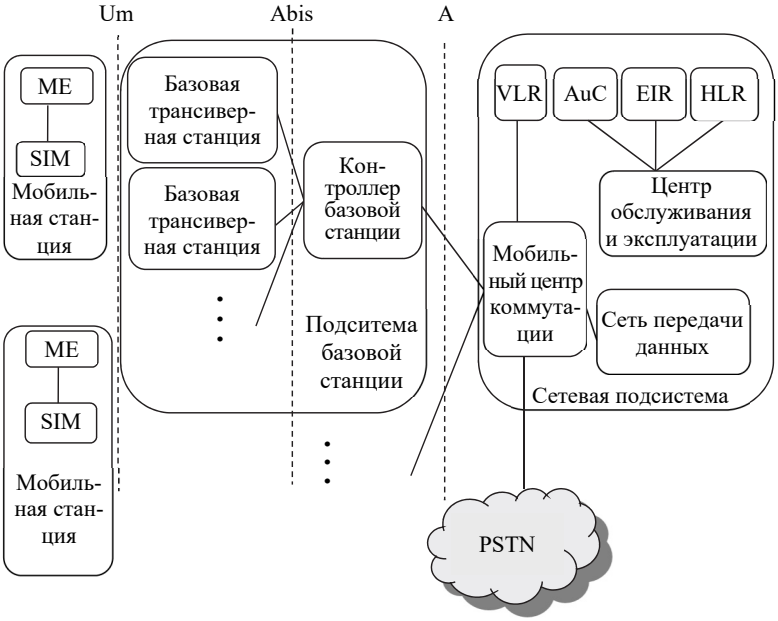


Рис. 10.7. Общая архитектура GSM

Подсистема базовой станции (Base Station Subsystem, BSS) состоит из контроллера базовых станций и одной или нескольких базовых трансиверных станций.

Каждая *базовая трансиверная станция* (Base Transceiver Station, BTS) определяет ячейку, в которую входит радиоантенна, радиотрансивер и канал связи с контроллером базовых станций. Ячейка GSM может иметь радиус от 100 м до 35 км в зависимости от среды. *Контроллер базовой станции* (Base Station Controller, BSC) может

совмещаться с BTS или управлять работой нескольких устройств BTS, а следовательно, нескольких ячеек. Контроллер BSC резервирует радиочастоты, управляет переключениями мобильных устройств с одной ячейки на другую в пределах одной подсистемы BSS и контролирует избирательное обращение.

Сетевая подсистема (Network Subsystem, NS) обеспечивает связь между сотовой сетью и общественными коммутируемыми телекоммуникационными сетями. Подсистема NS управляет переключениями между ячейками, находящимися в различных подсистемах базовых станций, опознает пользователей и подтверждает достоверность их счетов, а также выполняет функции роуминга мобильных пользователей. Центральным элементом подсистемы NS является *мобильный центр коммутации* (Mobile Switching Center, MSC). Он управляет четырьмя базами данных.

1. База данных *регистра исходного положения* (Home Location Register, HLR). В регистре HLR хранится информация, как временная, так и постоянная, о каждом из абонентов, который «принадлежит» системе (т. е. об абонентах, телефонные номера которых связаны с центром коммутации).

2. База данных *регистра местонахождения посетителей* (Visitor Location Register, VLR). Одну из важных частей информации составляет местонахождение абонента. Местонахождение определяется из регистра VLR, в который введен абонент. В регистре местонахождения посетителей хранится информация об абонентах, которые в данный момент физически находятся в районе, обслуживаемом данным центром коммутации. В регистре отмечается, является ли абонент активным, а также фиксируются другие связанные с ним параметры. При поступлении звонка абоненту система использует связанный с ним телефонный номер для опознания исходного для данного абонента центра коммутации. Этот центр коммутации, в свою очередь, в регистре HLR может найти центр коммутации, в зоне действия которого в данный момент физически находится абонент. При поступлении звонка от абонента регистр VLR используется для инициирования звонка. Даже если абонент находится в зоне, принадлежащей его исходному центру коммутации, он может также быть представлен в регистрах VLR других центров коммутации.

3. База данных *центра аутентификации* (Authentication Center, AuC или AC). Эта база данных используется в процессе аутентификации; например, в ней хранятся ключи аутентификации и шифрования

для всех абонентов, представленных как в регистрах исходного положения, так и в регистрах местонахождения посетителей. Центр управляет доступом к данным пользователей, а также процессом аутентификации при присоединении абонента к сети. Данные, передаваемые системами GSM, шифруются, поэтому они конфиденциальны. Для шифровки данных, передаваемых от абонента трансиверу базовой станции, используется поточный шифр A5. В то же время переговоры по сети с наземными линиями связи проходят без шифрования. Другой поточный шифр, A3, используется для аутентификации.

4. База данных *регистра идентификации оборудования* (Equipment Identity Register, EIR). В этой базе данных хранятся записи о типе оборудования, которое имеется на мобильной станции. Эта база данных также важна для безопасности (например, для блокирования звонков с украденных мобильных устройств и предотвращения использования сети станциями, которым не было дано такого разрешения).

10.1.5. Сотовые системы первого и второго поколения

Сотовые системы первого поколения (1G – 1st Generation), подобные AMPS, быстро приобрели широкую популярность. Они строились на основе аналоговых сигналов и опирались на несколько разных стандартов.

Существуют следующие ключевые различия между системами первого и второго поколений.

1. *Цифровые информационные каналы*. Системы первого поколения практически полностью аналоговые, в то время как системы второго поколения являются цифровыми. В частности, системы первого поколения спроектированы для поддержки голосовых каналов с использованием частотной модуляции; цифровые данные можно передавать только с использованием модема, который преобразует цифровые данные в аналоговую форму.

2. *Шифрование*. Системы первого поколения отправляют пользовательские данные в чистом виде, не обеспечивая никакой защиты.

3. *Обнаружение и исправление ошибок*. В результате можно обеспечить довольно чистый прием речи.

4. *Доступ к каналам*. В системах первого поколения каждая ячейка поддерживает несколько каналов. В любой момент времени канал может быть выделен только одному пользователю. В системах второго поколения ячейкам также выделяется по несколько каналов, к тому же каждый канал может совместно использоваться несколькими пользователями посредством *схем множественного доступа с временным разделением* (TDMA) или *множественного доступа с кодовым разделением* (CDMA); см. подглаву 3.2.

Система связи 2G (1991 г.) функционировала на основе двух стандартов: GSM и CDMA, а также включала в себя систему обмена короткими текстовыми сообщениями (SMS), сервис USSD (Unstructured Supplementary Service Data – неструктурированные дополнительные сервисные данные; сервис USSD по формату схож с SMS, однако это сервис моментальных сообщений, которые не хранятся ни на стороне оператора, ни на устройстве абонента) и ряд других протоколов быстрого обмена данными, которые существуют и по сей день.

Начиная с 1990-х годов было внедрено немало различных систем второго поколения (2G). В таблице перечислены некоторые ключевые характеристики трех наиболее важных из них.

Сотовые телефонные системы второго поколения

Характеристика	GSM	IS–136	IS–95
Метод доступа	TDMA	TDMA	CDMA
Полоса частот для передачи сигналов базовой станции, МГц	935–960	869–894	869–894
Полоса частот для передачи сигналов мобильного устройства, МГц	890–915	824–849	824–849

Доступ многих пользователей к сотовой системе первого поколения осуществляется с помощью технологии FDMA (п. 3.2.3).

Технология разделения каналов TDMA уже упоминалась в главе 2. Применение схемы TDMA в сотовой системе можно описать следующим образом. Так же, как и при использовании FDMA, каждой

ячейке выделяется некоторое количество каналов, половина из которых используется для обратной связи, а половина – для прямой. Передача данных осуществляется в виде последовательности кадров с повторяющейся структурой: каждый кадр делится на некоторое число слотов. Положение каждого слота в последовательности кадров определяет отдельный логический канал.

Множественный доступ с кодовым разделением каналов (Code Division Multiple Access, CDMA; см. п. 3.2.2) представляет собой основанную на расширении спектра схему, которая является второй после TDMA альтернативой разделения каналов для сотовых сетей второго поколения.

Использование схемы CDMA для сотовых сетей имеет несколько преимуществ.

1. *Частотное разнесение*. Из-за того что передаваемые сигналы рассеяны по широкой полосе частот, искажение передачи на определенной частоте, например вследствие шума или селективного замирания, меньше влияет на сигнал.

2. *Снижение негативных эффектов многолучевого распространения*.

3. *Конфиденциальность*. Каждый пользователь имеет свой код, такой схеме изначально присуща конфиденциальность.

4. *Постепенное снижение эффективности функционирования системы*. В схемах TDMA или FDMA к системе может одновременно обращаться только фиксированное число пользователей. В CDMA же по мере возрастания количества пользователей, получивших одновременный доступ к системе, уровень шума и, следовательно, *частота появления ошибок* увеличиваются.

Следует упомянуть и о некоторых недостатках сотовой схемы CDMA:

– *проблема расположения*. Сигналы, поступающие от более близких к передатчику объектов, не так затухают, как сигналы, пришедшие издалека. Поэтому в системах CDMA очень важное значение имеют схемы регулирования мощности;

– *мягкое переключение*. Как будет показано далее, для плавности переключения с одной ячейки в следующую требуется, чтобы мобильное устройство вошло в новую ячейку до того, как оно оставит старую. Это называется мягким переключением, которое является более сложным, чем жесткое переключение, используемое в схемах FDMA и TDMA.

10.1.6. Сотовые системы третьего поколения 3G

Системы связи поколения 3G разрабатывались с целью получения высокоскоростных беспроводных средств передачи не только речи, но и данных, мультимедиа и видео.

По инициативе ITU IMT-2000 определено следующее видение союзом ITU возможностей систем третьего поколения. Приведем некоторые их характеристики.

1. Качество речи сравнимо с качеством речи в общественной коммутируемой телефонной сети: доступна скорость передачи данных в 144 Кбит/с.

2. Для низкоскоростных систем доступная скорость передачи данных составляет 384 Кбит/с.

3. Для учреждений поддерживается скорость передачи данных 2,048 Мбит/с.

4. Передачи данных могут быть симметричными и асимметричными.

5. Поддерживается связь как с коммутацией пакетов, так и с коммутацией каналов.

6. Адаптивный интерфейс с Internet позволяет эффективно отражать асимметрию прибывающего и отправляемого трафика.

7. Доступный спектр частот используется более эффективно.

8. Поддерживается разнообразное мобильное оборудование.

9. Система достаточно гибка для введения новых услуг и технологий.

Эти характеристики и концепции глобальной беспроводной связи были названы *персональными службами связи* (Personal Communications Services, PCS) и *персональными сетями связи* (Personal Communications Network, PCN), а их воплощение в жизнь является задачей беспроводных систем третьего поколения.

Телефоны PCS, согласно проекту, должны иметь меньшую мощность и быть относительно маленькими и легкими.

Сети третьего поколения 3G работают на частотах дециметрового диапазона, как правило, в диапазоне около 2 ГГц, передавая данные со скоростью до 14 Мбит/с. Они позволяют организовывать видеотелефонную связь.

Стандарт 3G включает в себя 5 отдельных стандартов семейства IMT-2000 (UMTS/W-CDMA, CDMA2000, TD-CDMA, DECT и UWC-136). Из перечисленных составных частей 3G только первые три представляют собой полноценные стандарты сотовой связи третьего поколения, а DECT и UWC-136 играют вспомогательную роль. DECT (Digital Enhanced Cordless Telecommunications) – это *стандарт беспроводной телефонии домашнего или офисного назначения*, который в рамках мобильных технологий третьего поколения может использоваться только для организации точек горячего подключения к данным сетям. Стандарт UWC-136 (Universal Wireless Communications) – это технология EDGE (Enhanced Data Rates for GSM Evolution), которая относится к предыдущему поколению.

Стандарт UMTS (Universal Mobile Telecommunication System – универсальная система мобильной связи) теоретически обеспечивает обмен информацией на скоростях до 2048 Кбит/с, однако на практике скорость может быть несколько ниже. В сетях W-CDMA (Wideband Code Division Multiple Access) используют разделение сигнала по кодово-частотному принципу, т. е. идентификация пакетов информации, передаваемых абонентами, производится не только по уникальному идентификатору, но и по частоте. Для передачи данных протоколы UMTS используют частоты 1885–2025 МГц для передачи данных в режиме от мобильного терминала к базовой станции и 2110–2200 МГц для передачи данных в режиме от станции к терминалу.

Стандарт TD-CDMA (Time-Division Code Division Multiple Access) близок к рассмотренному выше стандарту W-CDMA, однако его основой является гибридный кодово-временно́й принцип разделения сигнала. В целом считается, что именно стандарт TD-CDMA является наилучшим для передачи данных в сети Интернет.

Технологию CDMA2000 следует рассматривать как эволюцию технологии CDMA, тогда как UMTS радикально отличается от GSM. Стандарт CDMA2000 разделяют на три фазы – 1X (известна также как IS-95C), 1X EV-DO (только данные) и 1X EV-DV (данные и голос). Именно стандарт 1X EV-DV может считаться полноценным 3G-стандартом. Отметим, что изначально не было разделения на 1X EV-DO и 1X EV-DV, а в стандарте CDMA выделяли только две фазы – 1XRTT и 3XRTT. Скорость обмена информацией в сетях CDMA2000 1X может достигать 153,6 Кбит/с, в стандарте CDMA2000 1X EV-DO – 2,4 Мбит/с (ревизия 0) и 3,1 Мбит/с (ревизия A). В отличие от стандарта UMTS, стандарт CDMA2000 не оговаривает, какие частоты

должны использоваться для передачи сигнала, поэтому построение сетей CDMA 2000 возможно во всех частотных диапазонах, используемых операторами сотовой связи – 450, 700, 800, 900, 1700, 1800, 1900, 2100 МГц.

В сетях 3G обеспечивается предоставление двух базовых услуг: передача данных и передача голоса. Сети 3G характеризуются преобладанием трафика *data-cards* (USB-модемы, ExpressCard/PCMCIA-карты для ноутбуков) над трафиком телефонов и смартфонов 3G.

Согласно регламентам ITU (International Telecommunications Union – Международный союз электросвязи) сети 3G должны поддерживать следующие скорости передачи данных:

– для абонентов с высокой мобильностью (до 120 км/ч) – не менее 144 Кбит/с;

– для абонентов с низкой мобильностью (до 3 км/ч) – 384 Кбит/с;

– для неподвижных объектов – 2,048 Мбит/с.

В сетях с кодовым разделением каналов, в том числе и 3G, есть важное преимущество – улучшенная защита от обрывов связи в движении за счет «мягкого» переключения между станциями. По мере удаления от одной базовой станции клиент «подхватывается» другой. Она начинает передавать все больше и больше информации, в то время как первая станция передает все меньше и меньше, пока клиент вообще не покинет зону ее обслуживания. При хорошем покрытии сети вероятность обрыва полностью исключается системой подобных «подхватов». Это отличается от поведения систем с частотным и временным разделением каналов (GSM), в которых переключение между станциями «жесткое» и может приводить к задержкам в передаче и даже обрывам соединения.

10.1.7. Сотовые системы четвертого поколения 4G

Разработка технологий передачи информации в сети четвертого поколения (4G) началась в 2000 г. с исследований японской компании NTT DoCoMo. Позже к ней присоединились такие компании, как Hewlett-Packard, Sprint, Imagine, Google, Intel, Comcast, Bright House,

Time Warner, Ericsson и др. Таким образом, появилось два стандарта: LTE (Long Term Evolutian) и WiMAX (IEEE 802.16e), которые, по мнению IMT-Advanced, и стали новой эрой в развитии сети. Технология LTE возникла в процессе развития организацией 3GPP технологии UMTS, относящейся к сетям 3G.

Параллельно организацией WiMAX Forum создавалась технология WiMAX, основанная на беспроводной технологии Wi-Fi. В 2005 г. в Южной Корее оператор связи КТ представил услуги мобильной связи WiMAX в Пусане, и с 2006 г. началось коммерческое использование данной технологии. Первым запуском LTE считается развертывание сети в городских центрах Стокгольма (Ericsson и Nokia Siemens Networks) и Осло (Huawei) в 2009 г.

Технические характеристики. Система 4G основывается на технологиях LTE и IEEE 802.16e (также известной как WiMAX). Пиковая скорость восходящего потока в технологии LTE – 50 Мбит/с, нисходящего – 100 Мбит/с при использовании канала 20 МГц, однако вместе с MIMO (Multiple Input/Multiple Output – множество входов/множество выходов) в технологии WiMAX можно достигнуть скоростей в 56 и 128 Мбит/с соответственно. В технологиях четвертого поколения используется более 40 диапазонов частот (в каждом регионе – свои). В США – 700 МГц, 1710–1755 МГц (передача) и 2110–2155 МГц (прием), т. е. 1,7 и 2,1 ГГц; в Европе – 1800 и 2600 МГц, в перспективе – 800 МГц; в Японии – 800 и 850 МГц; 1,5, 1,7 и 2,1 ГГц.

> **!** В отличие от предыдущих поколений, а именно от третьего, система 4G не поддерживает традиционные услуги телефонии с коммутацией каналов. В 4G используется IP-телефония, т. е. коммутация пакетов. Для пересылки данных используется IPv4 (планируется перейти на IPv6). Для осуществления услуг телефонии вызовы и сообщения «перенаправляются» через существующие сети 2G и 3G.

Теоретически стандарт 4G позволяет передавать информацию со скоростью в пять-семь раз выше, чем 3G и даже так называемый 3,5G (HSPA, HSPA+), где максимальная скорость равна 150 Мбит/с.

Для увеличения эффективности передачи информации в сетях 4G используются технологии ортогонального частотного уплотнения OFDM и MIMO.

Вопреки ошибочному мнению, первые версии данных технологий не относятся к сетям четвертого поколения в связи со своими техническими возможностями, а именно не удовлетворяют требованию высокой скорости передачи данных. Версии LTE-A (LTE-Advanced) и IEEE 802.16m (WiMAX2) признаны технологиями 4G Международным союзом электросвязи в 2010 г. Пиковая скорость восходящего потока LTE-A составляет 500 Мбит/с, нисходящего – 1 Гбит/с. Технология WiMAX2 обеспечивает скорости 90 и 179 Мбит/с соответственно при использовании частот до 11 ГГц.

Сегодня технология *4G обладает следующими возможностями*:
– высокоскоростной доступ в Интернет;
– IP-телефония;
– мобильное ТВ высокой четкости;
– видеоконференции;
– 3D-телевидение и т. д.

10.1.8. Сотовые системы пятого поколения 5G

Первые исследования данных систем начались в 2008 г. В Великобритании 8 октября 2012 г. был создан исследовательский центр 5G с целью разработки технологии связи для замены 4G (ожидаемо в течение десяти лет). В этом же году организован проект METIS для определения стандартов 5G, который добивается значимых результатов путем обсуждения на ведущих мировых форумах.

Уже в мае 2013 г. компания Samsung Electronics заявила об изобретении 5G-системы. Технология поддерживала скорость передачи данных в десятки гигабитов в секунду. При тестировании был показан результат в 1,056 Гбит/с при передаче данных на расстояние 2 км. В октябре 2013 г. компания Huawei заявила о вложении 600 млн долл. США в развитие технологии, которая увеличит скорость передачи данных в 100 раз по сравнению со скоростью LTE-сетей. Тестирования технологий в 2016 г. показали скорость в 5 Гбит/с, что позволило передавать видео в разрешении 8K Ultra HD. В октябре 2016 г. компания Qualcomm представила первый мобильный модем, поддерживающий 5G. Тем временем множество компаний по всему миру заявляют о развертывании 5G-сетей.

5G – это системы пятого поколения беспроводной телефонной связи, от которых ожидается значительный рост скорости передачи данных (от 10 Гбит/с), увеличение емкости сети, уменьшение задержек. Одной из задач, предписываемой для 5G-системы, является создание базы для интернета вещей (IoT).

Технические характеристики 5G-сетей. Так как на сегодняшний день общепринятых норм и характеристик технологии 5G не существует, компании-разработчики заключили соглашение об обсуждении стандартов для данного вида связи раз в полгода.

В настоящее время компания NGMN Alliance определяет следующие требования к 5G:

1) скорость передачи данных от 10 Мбит/с для нескольких десятков тысяч подключений;

2) скорость передачи данных от 100 Мбит/с для крупных городов с пригородами;

3) скорость передачи данных от 1 Гбит/с для пользователей, находящихся на одном этаже здания;

4) поддержка подключения 10–100 тыс. устройств в беспроводных сенсорных сетях;

5) задержка сигнала до 10 мс;

6) усиление эффективности использования полосы пропускания по сравнению с 4G;

7) увеличение охвата связи;

8) усиление сигнальной эффективности сети.

Для технологии 5G частотный диапазон может быть выше 3,5 ГГц (теоретически до 100 ГГц).

В целом предполагается использование сетей пятого поколения во всех сферах жизни информационного общества. Новое поколение мобильной связи может стать базовым для интернета вещей (IoT). Множество устройств с элементарными датчиками смогут обмениваться данными между собой, и для этого потребуется высокая скорость обмена данными, а самое главное – поддержка огромного количества соединений.

Выполнение этих требований ждут от технологий 5G. В концепции нового поколения связи разрабатывается и отдельная сеть для умных устройств − 5G-IoT.

Выводы

1. Принцип организации сотовой связи состоит в использовании множества маломощных передатчиков. Поскольку диапазон действия таких передатчиков довольно мал, зону обслуживания системы можно разбивать на ячейки, каждая из которых будет обслуживаться собственной антенной (на практике используется гексагональная схема). Каждая ячейка, которой выделяется своя полоса частот, обслуживается базовой станцией, состоящей из передатчика, приемника и модуля управления.

2. В каждой ячейке сотовой сети имеется базовый трансивер. Мощность передаваемых сигналов тщательно регулируется. Как правило, каждой ячейке выделяется 10–50 частот в зависимости от планируемой нагрузки.

3. Предусмотрено несколько вариантов расширения числа обслуживаемых клиентов одной ячейкой: добавление новых каналов, заимствование частот, расщепление ячеек, разбивка ячеек на секторы, микроячейки.

4. Сотовые системы первого поколения являлись аналоговыми, а системы второго поколения – цифровыми, что позволило использовать шифрование, а также средства исправления ошибок для обеспечения качества передачи. В системах первого поколения каждая ячейка поддерживает несколько каналов. В любой момент времени канал может быть выделен только одному пользователю. В системах второго поколения ячейкам также выделяется по несколько каналов, однако каждый канал может совместно эксплуатироваться несколькими пользователями посредством схем множественного доступа с временным разделением. Дальнейшее развитие сотовых систем второго поколения привело к использованию множественного доступа с кодовым разделением (CDMA).

5. Стандарт GSM был разработан для внедрения в Европе общей технологии второго поколения, чтобы одни и те же абонентские устройства можно было использовать по всему континенту (ранее это было невозможно).

6. Беспроводные системы связи третьего поколения разрабатываются с целью получения высокоскоростных беспроводных средств передачи не только речи, но и данных, мультимедиа и видео.

10.2. Сети Bluetooth

Технология была представлена шведским производителем мобильных средств связи Ericsson в 1994 г. как средство, которое позволяет портативным компьютерам совершать звонки по мобильным телефонам. С тех пор несколько тысяч компаний работают над тем, чтобы технология *Bluetooth* стала стандартом множества маломощных, действующих на близком расстоянии беспроводных устройств.

Название *Bluetooth* («голубой зуб») произошло от прозвища датского короля Гарольда Блаатанда (Harald Blaatand), жившего в X в. Стандарты Bluetooth публикуются промышленным консорциумом Bluetooth SIG (Special Interest Group – специальная группа).

Bluetooth – это внедренное в микрочип невыключающееся радиоустройство ближнего действия. *Цель разработки стандартов Bluetooth* – унифицировать возможности ближней радиосвязи. Технология Bluetooth предназначена для работы в среде со многими пользователями.

В диапазоне 2,4 ГГц (общедоступные нелицензируемые частоты для маломощных устройств) два аппарата Bluetooth, находящиеся на расстоянии до 10 м, могут совместно использовать пропускную способность до 720 Кбит/с. Технология Bluetooth предназначена для поддержки многих приложений (полный список достаточно объемен и продолжает пополняться: передача данных, аудио, графики, видео и т. д.). Например, чип Bluetooth может внедряться в такие аудиоустройства, как наушники, беспроводные и обычные телефоны, домашние стереопроигрыватели и цифровые MP3-плееры. С помощью Bluetooth потребители могут делать следующее:

– звонить с беспроводного головного телефона, удаленно связанного с телефоном ячейки сотовой связи;

– соединять компьютеры с периферией (принтеры, клавиатуры, мыши), не используя кабель;

– подключать без проводов MP3-плееры к другим аппаратам с целью загрузки музыкальных файлов;

– организовывать домашние сети.

10.2.1. Топология, адресация и особенности эксплуатации сети Bluetooth

Все устройства сети Bluetooth делятся на ведущие (Master) и подчиненные (Slave). Обмен информацией может осуществляться только между ведущим и подчиненным устройствами, при этом каждое устройство может быть и ведущим, и подчиненным.

Основным элементом организации сетей Bluetooth является ***пикосеть***, состоящая из одного ведущего устройства и 1–7 активных подчиненных устройств.

Кроме того, в одну пикосеть может входить неограниченное количество устройств, находящихся в неактивном режиме. Подчиненное устройство может сообщаться только с ведущим, причем только тогда, когда это разрешает ведущее устройство. В каждый момент времени обмен данными может идти только между двумя устройствами в одном направлении.

Любое устройство одной пикосети может также входить в другую пикосеть в качестве как подчиненного, так и ведущего. Данная схема с перекрытием пикосетей называется ***рассеянной сетью***.

На рис. 10.8 слева приведен пример пикосетей, а справа – рассеянной сети, состоящей из трех перекрывающихся пикосетей. На этом рисунке буквой M (Master) обозначены ведущие устройства, буквой S (Slave) – подчиненные, а M/S означает, что устройство является ведущим в одной пикосети и подчиненным – в другой.

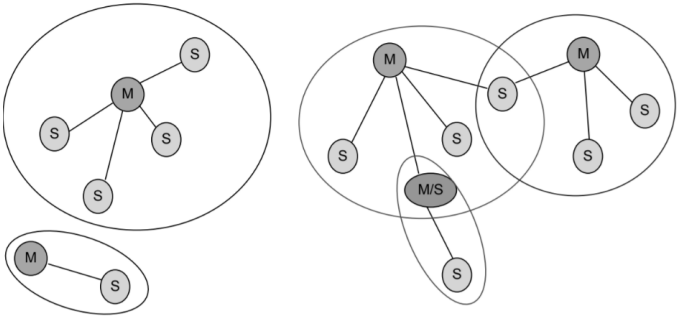


Рис. 10.8. Топология сетей Bluetooth

С целью однозначной идентификации каждое устройство Bluetooth имеет уникальный 48-битный адрес (выдается регистрирующим органом IEEE). Адрес состоит из следующих частей (см. рис. 10.9):

– LAP (Lower Address Part) – нижняя часть адреса;

– UAP (Upper Address Part) – верхняя часть адреса;

– NAP (Non-Significant Part) – несущественная часть адреса.

Назначаемый						ID производителя					
LAP						UAP		NAP			
0000	0001	0000	0000	0000	0000	0001	0010	0111	1011	0011	0101

Рис. 10.9. Структура и пример адреса устройства Bluetooth

64 значения (0x9E8B00–0x9E8B3F) LAP-части зарезервированы для кодов доступа процедуры опроса и не могут являться частью адреса устройства. LAP и UAP вместе участвуют в выборе псевдослучайной последовательности перестройки частоты, кроме того, LAP формирует синхрослово в коде доступа, а UAP участвует в процессе проверки ошибок.

В ходе установления и эксплуатации канала устройство Bluetooth может находиться в различных состояниях (трех основных и семи промежуточных).

Основные состояния:

1) *холостое состояние* – низкое энергопотребление, работают только часы устройства;

2) *состояние соединения* – устройство подключено к пикосети;

3) *состояние парковки* – состояние подчиненного устройства, от которого не требуется участия в работе пикосети, но которое должно оставаться ее частью.

Промежуточные состояния (для подключения к пикосети новых подчиненных устройств):

1) *опрос* – определение устройством наличия других устройств в пределах его досягаемости;

2) *поиск опроса* – ожидание устройством опроса;

3) *ответ на опрос* – устройство, получившее опрос, отвечает на него;

4) *запрос* – посылается одним устройством другому для установления с ним соединения (запрашивающее устройство становится ведущим, запрашиваемое – подчиненным);

5) *поиск запроса* – устройство ожидает запрос;

6) *ответ подчиненного устройства* – подчиненное устройство отвечает на запрос ведущего;

7) *ответ ведущего устройства* – ведущее устройство отвечает подчиненному после получения от него ответа на запрос.

На рис. 10.10 приведена диаграмма возможных переходов между состояниями устройства Bluetooth.

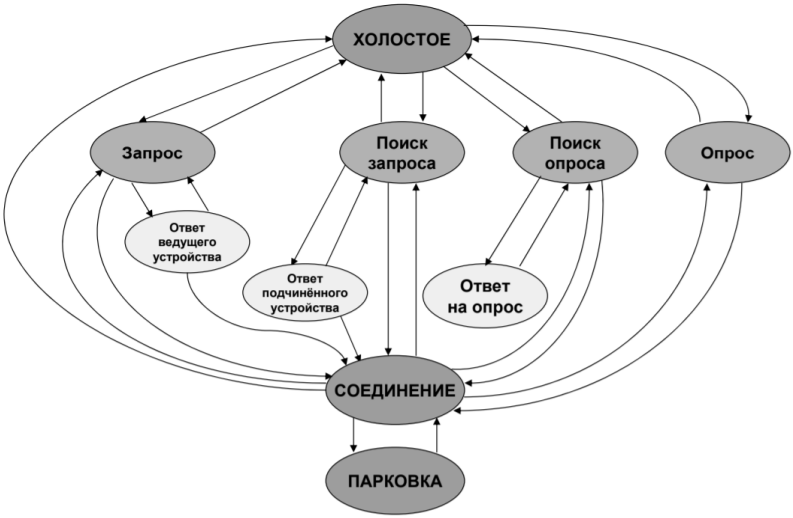


Рис. 10.10. Диаграмма переходов между состояниями Bluetooth

10.2.2. Области применения Bluetooth

В маленькую сеть, называемую *пикосетью* (Piconet), могут объединяться до восьми устройств. Десять таких пикосетей могут сосуществовать в одном радиодиапазоне Bluetooth. Для обеспечения безопасности каждый канал связи кодируется и защищается от подслушивания и интерференции.

Bluetooth предусматривает поддержку трех основных областей применения с использованием *беспроводной связи ближнего действия*. Кратко охарактеризуем их.

1. *Точки доступа для ввода данных* (в том числе – посредством голоса). Bluetooth способствует передаче в реальном времени данных

и речи, обеспечивая удобную беспроводную связь портативных и стационарных аппаратов связи.

2. *Замена кабеля*. При наличии Bluetooth отпадает необходимость в многочисленных кабельных проводах, которые сопровождают практически все устройства связи. Соединение Bluetooth устанавливается мгновенно, причем связываемые устройства не обязательно должны находиться в пределах прямой видимости. Радиус охвата порядка 10 м, а если использовать усилитель, эту величину можно довести до 100 м.

3. *Организация эпизодических сетей*. Устройство, оснащенное чипом Bluetooth, может мгновенно устанавливать связь с другим устройством, находящимся в пределах области охвата.

Приведем несколько примеров того, как можно использовать технологию Bluetooth.

1. *Телефон «три в одном».* Если вы в офисе – телефон работает как *интерком* (не нужно платить за услуги телефонии), дома это беспроводной телефон (оплачивается как стационарное устройство), а если вы перемещаетесь, его можно использовать как обычный мобильный телефон (это уже сотовая связь).

2. *Мост Internet.* Портативный ПК можно связать с Internet откуда угодно либо посредством мобильного телефона (беспроводное соединение), либо с помощью кабеля (PSTN, ISDN, JIBC, xDSL).

3. *Интерактивная конференция.* На встречах и конференциях информацию можно мгновенно распространять между всеми участниками. Кроме того, проектором можно управлять и не имея проводного соединения.

4. *Удаленный головной телефон.* Соединить головной телефон с мобильным ПК или любым проводным соединением.

5. *Громкоговоритель портативного ПК.* Соединить беспроводной головной микрофон с портативным ПК и использовать его как микрофон.

6. *Почта из портфеля.* Получить доступ к электронной почте, не вынимая портативный ПК из портфеля. Как только ПК получит почту, вам об этом сообщит мобильный телефон. Используя тот же мобильный телефон, можно просмотреть входящую почту и прочитать сообщения.

7. *Задержанные сообщения.* Электронную почту можно занести в ПК. Как только появится возможность воспользоваться мобильным телефоном, сообщения будут посланы немедленно.

8. *Автоматическая синхронизация.* Автоматически синхронизируется настольный компьютер, портативный ПК, ноутбук и мобильный телефон.

9. *Мгновенная цифровая открытка.* Подключить (без проводов) камеру к мобильному телефону или к любому выходу проводной связи, ввести комментарий с мобильного телефона, ноутбука или портативного ПК – и готовую открытку можно отправлять немедленно.

10. *Беспроводной настольный компьютер.* Компьютеру не нужны провода, чтобы соединиться с принтером, сканером, клавиатурой, мышью или локальной сетью.

10.2.3. Стандарты Bluetooth и структура протоколов

Стандарты Bluetooth содержат *технологическую (внутреннюю) спецификацию Bluetooth* и *спецификацию профиля*. *Внутренние спецификации* описывают детали разнообразных уровней протокольной архитектуры Bluetooth (от радиоинтерфейса до управления каналом связи).

В *спецификациях профиля* описано использование технологии Bluetooth для поддержки различных приложений. В каждой спецификации рассматривается применение технологии, определенной во внутренней спецификации, для реализации конкретной модели использования. В спецификации профиля указывается, какие аспекты внутренних спецификаций Bluetooth являются обязательными, необязательными и неприменимыми. Bluetooth определяется как *многоуровневая протокольная архитектура* (рис. 10.11).

Архитектура Bluetooth состоит из *внутренних протоколов*, *протоколов замены* кабеля, управления телефонией и *адаптированных протоколов*.

К последним относятся: *AT* – сигнальная последовательность (префикс модема); *IP* – сетевой протокол (Internet); *OBEX* – протокол объектного обмена; *PPP* – протокол двухточечного соединения; *RFCOMM* – связь на радиочастотах; *SDP* – протокол обнаружения службы; *TCP* – транспортный протокол управления передачей; *UDP* – протокол пользовательских дейтаграмм; *TCS BIN* – спецификация

управления телефонией; *vCal* – виртуальный календарь; *vCard* – виртуальная карта; *WAE* – среда беспроводных приложений; *WAP* – протокол беспроводных приложений). Внутренние протоколы формируют пятиуровневый стек.

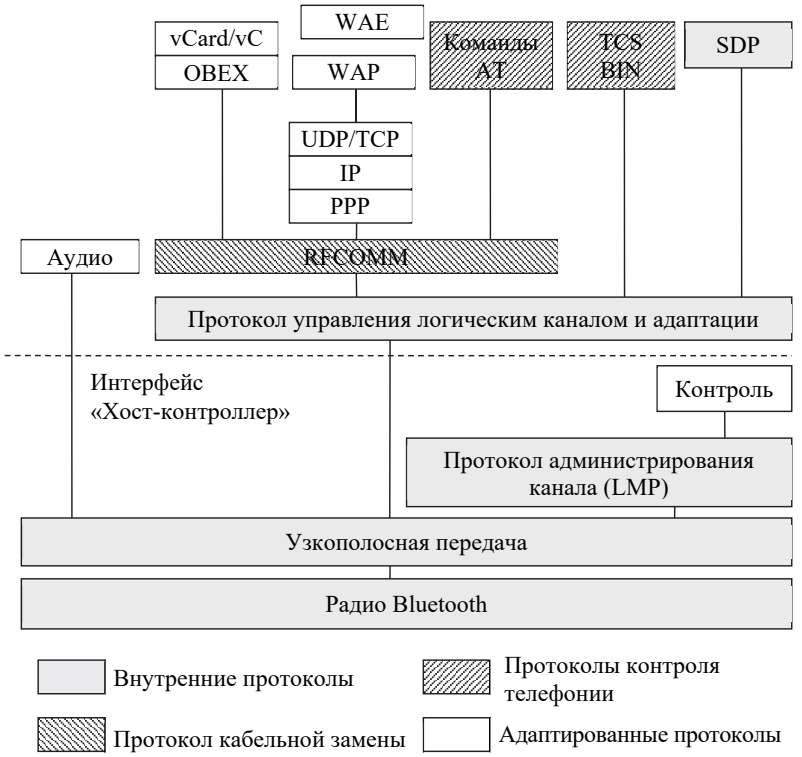


Рис. 10.11. Стек протоколов Bluetooth

В Bluetooth определен *протокол управления телефонией*. Протокол TCS BIN (Telephony Control Specification Binary – спецификация управления телефонией – бинарная) – это протокол с битовой структурой, который определяет передачу сигналов управления вызовами с целью установления сеансов передачи речи и данных между устройствами Bluetooth. Кроме того, он определяет процедуры управления мобильностью для управления группами устройств Bluetooth TCS.

Адаптированный протокол определяется в спецификациях, выпускаемых другими организациями по стандартизации, и вводится в общую архитектуру Bluetooth. Стратегия Bluetooth заключается в создании только необходимых протоколов при максимально возможном использовании имеющихся стандартов.

10.2.4. Модели использования Bluetooth

В документах по профилям Bluetooth определено несколько моделей использования.

Модель использования – это набор протоколов, которые реализуют конкретное приложение на основе Bluetooth. Каждый профиль определяет протоколы и свойства протоколов, поддерживающие конкретную модель использования. Перечислим модели использования с наивысшим приоритетом.

1. *Передача файлов.* Модель поддерживает передачу каталогов, файлов, документов, изображений и потоковую информацию. Данная модель использования также содержит возможность просмотра папки с удаленного устройства.

2. *Мост Internet*. Используя данную модель, ПК связывается без проводов с мобильным телефоном или беспроводным модемом для удаленного телефонного доступа к сети или факсу.

3. *Доступ к локальной сети*. Данная модель использования позволяет устройствам пикосети получить доступ к локальной сети. После соединения работа устройства та же, что и при проводном подключении.

4. *Синхронизация*. Данная модель обеспечивает синхронизацию содержащейся на устройствах персональной информации, такой как записи в телефонной книге, календаре, сообщения и заметки. Здесь следует упомянуть IrMC (Ir Mobile Communications – мобильная связь в инфракрасном диапазоне), протокол IrDA, который позволяет передавать между устройствами обновленную персональную информацию (по схеме «клиент-сервер»).

5. *Телефон «три в одном».* Телефонные трубки, которые реализуют данную модель использования, могут работать как беспроводной телефон, подсоединенный к голосовой базовой станции, как интерком, связанный с другими телефонами, и как сотовый телефон.

6. *Головной телефон*. Головной телефон может использоваться как устройство аудио ввода-вывода удаленного устройства.

Выводы

1. Bluetooth – это внедренное в микрочип невыключающееся радиоустройство ближнего действия. Технология Bluetooth предназначена для работы в среде со многими пользователями.

2. Целью разработки стандартов Bluetooth явилась необходимость унификации возможностей ближней радиосвязи. Bluetooth определяется как многоуровневая протокольная архитектура.

3. При разработке стандарта Bluetooth упор был сделан на создание только необходимых протоколов при максимально возможном использовании имеющихся, например TCP/IP, IPX/SPX и т. д.

4. Все устройства сети делятся на ведущие (Master) и подчиненные (Slave). Обмен информацией может осуществляться только между ведущим и подчиненным устройствами, при этом каждое из них может быть и ведущим, и подчиненным.

5. Внутренние протоколы сетей Bluetooth формируют пятиуровневый стек.

6. В документах стандарта Bluetooth определено несколько моделей использования, под которыми понимают набор протоколов, реализованных в виде конкретных приложений на основе Bluetooth. Каждый профиль определяет протоколы и свойства протоколов.

7. К моделям с наивысшим приорететом относятся: передача файлов, мост Internet, доступ к локальной сети, синхронизация.

8. Структура протоколов формально не следует ни модели OSI, ни TCP/IP, ни IEEE 802, ни какой-либо другой известной модели. В самом низу находится физический (радиотехнический) уровень, который вполне соответствует моделям OSI и IEEE 802. На нем описывается радиосвязь и применяемые методы модуляции. Уровень управления каналом связи (прямой передачи) чем-то напоминает подуровень MAC, однако включает в себя и некоторые элементы физического уровня. Здесь описывается то, как главный узел управляет временными интервалами и как эти интервалы группируются в кадры. Далее следуют два протокола, которые используют протокол управления каналом связи. Протокол управления соединением устанавливает логические каналы между устройствами, управляет режимами энергопотребления, сопряжением и шифрованием, а также качеством обслуживания. Он находится ниже линии

интерфейса хостконтроллера. Этот интерфейс – удобство для реализации: как правило, протоколы ниже линии реализуются на чипе Bluetooth, а протоколы выше линии – на устройстве Bluetooth, где чип размещен.

10.3. Сверхвысокоскоростные сети

10.3.1. Общая характеристика стандарта Gigabit Ethernet

В 1996 г. было объявлено о создании группы 802.3z для разработки протокола, максимально подобного Ethernet, но с битовой скоростью 1000 Мбит/с.

Первая версия стандарта была рассмотрена в 1997 г., а окончательно стандарт 802.3z был принят 29 июня 1998 г. на заседании комитета IEEE 802.3.

Основная идея разработчиков *стандарта Gigabit Ethernet* состоит в максимальном сохранении идей классической технологии Ethernet при достижении битовой скорости в 1000 Мбит/с.

Отметим, что избыточные связи и тестирование оборудования не поддерживаются технологией Gigabit Ethernet.

Сходство технологий Gigabit Ethernet, Ethernet и Fast Ethernet состоит в следующем.

Сохраняются все форматы кадров Ethernet. По-прежнему будут существовать полудуплексная версия протокола, поддерживающая метод доступа CSMA/CD, и полнодуплексная версия, работающая с коммутаторами.

Поддерживаются все основные виды кабелей, используемые в Ethernet и Fast Ethernet: волоконно-оптический, витая пара категории 5, коаксиал.

Для расширения максимального диаметра сети Gigabit Ethernet в полудуплексном режиме до 200 м разработчики технологии предприняли достаточно естественные меры, основывающиеся на известном

соотношении времени передачи кадра минимальной длины и времени двойного оборота.

Минимальный размер кадра был увеличен (без учета преамбулы) с 64 до 512 байтов или до 4096 битов. Станция может передать подряд несколько кадров с общей длиной не более 65 536 битов или 8192 байтов. Если станции нужно передать несколько небольших кадров, то она может не дополнять их до размера в 512 байтов, а передавать подряд до исчерпания предела в 8192 байта (в этот предел входят все байты кадра, в том числе преамбула, заголовок, данные и контрольная сумма). Предел в 8192 байта называется BurstLength.

10.3.2. Спецификации физической среды стандарта 802.3z

В стандарте 802.3z определены следующие типы физической среды:

– одномодовый волоконно-оптический кабель;
– многомодовый волоконно-оптический кабель 62,5/125;
– многомодовый волоконно-оптический кабель 50/125;
– двойной коаксиал с волновым сопротивлением 75 Ом.

Многомодовый кабель. Для передачи данных по традиционному для компьютерных сетей многомодовому волоконно-оптическому кабелю стандарт определяет применение излучателей, работающих на двух длинах волн: 1300 и 850 нм. Для *многомодового оптоволокна* стандарт 802.3z определил спецификации l000Base-SX и l000Base-LX.

В первом случае используется длина волны 850 нм (S означает Short Wavelength, короткая волна), а во втором – 1300 нм (L – Long Wavelength, длинная волна).

Для спецификации l000Base-SX предельная длина оптоволоконного сегмента для кабеля 62,5/125 составляет 220 м, а для кабеля 50/125 – 500 м.

Одномодовый кабель. Для спецификации l000Base-LX в качестве источника излучения всегда применяется полупроводниковый лазер с длиной волны 1300 нм.

Максимальная длина кабеля для *одномодового волокна* равна 5000 м.

Для присоединения оптоволоконного трансивера к многомодовому кабелю необходимо использовать специальный адаптер.

Твинаксиальный кабель. В качестве среды передачи данных используется высококачественный твинаксиальный кабель (Twinax) с волновым сопротивлением 150 Ом (2×75 Ом). Максимальная длина *твинаксиального сегмента* составляет всего 25 м, поэтому это решение подходит для оборудования, расположенного в одной комнате.

10.3.3. Gigabit Ethernet

Спецификация Gigabit Ethernet изначально предусматривала три среды передачи: одномодовый и многомодовый оптический кабель с длинноволновыми лазерами 1000BaseLX для длинных магистралей для зданий и комплексов зданий, многомодовый оптический кабель с коротковолновыми лазерами 1000BaseSX для недорогих коротких магистралей, симметричный экранированный короткий 150-омный медный кабель 1000BaseCX для межсоединения оборудования в аппаратных и серверных.

Однако в настоящее время четырехпарная 100-омная проводка категории *cat5* и *cat6* является наиболее распространенной кабельной системой во всем мире. Как известно, каждая пара кабеля категории 5 имеет гарантированную полосу пропускания до 100 МГц. Для передачи по такому кабелю данных со скоростью 1000 Мбит/с было решено организовать параллельную передачу одновременно по всем четырем парам кабеля (так же, как и в технологии 100VG-AnyLAN).

Один из ключевых вопросов для Gigabit Ethernet – это максимальный размер сети. При переходе от Ethernet к Fast Ethernet сохранение минимального размера кадра привело к уменьшению диаметра сети с 2 км для 10BaseT до 200 м для 100BaseT.

Однако перенос без изменения всех отличительных составляющих Ethernet – минимального размера кадра, времени обнаружения коллизии (или кванта времени – Time Slot) и CSMA/CD – на Gigabit Ethernet обернулся бы сокращением диаметра сети до 20 м, поэтому в стандарте 802.3z было реализовано увеличение времени обнаружения коллизии с тем, чтобы сохранить прежний диаметр сети в 200 м. Такое переопределение подуровня MAC необходимо для Gigabit Ethernet, иначе отстоящие друг от друга на расстоянии 200 м станции

не смогут обнаружить конфликт, когда они обе одновременно передают кадр длиной 64 байта. Предложенное решение было названо расширением несущей (carrier extension). Суть его в следующем. Если сетевой адаптер или порт Gigabit Ethernet передает кадр длиной менее 512 байтов, то он посылает вслед за ним биты расширения несущей, т. е. время обнаружения конфликта увеличивается. Если за время передачи кадра и расширения несущей отправитель зафиксирует коллизию, то он реагирует традиционным образом: подается специальный сигнал и применяется механизм разрешения коллизий.

Очевидно, однако, что если все станции (узлы) передают кадры минимальной длины (64 байта), то производительность реально повышается всего на 12,5% (125 Мбит/с вместо 100 Мбит/с). Но даже с учетом того, что средняя длина кадра составляет на практике 200–500 байтов, пропускная способность возрастает всего лишь до 300–400 Мбит/с.

Поэтому с целью повышения эффективности Gigabit Ethernet комитет предложил метод пакетной передачи кадров (он подразумевает передачу серии кадров подряд). В соответствии с этим методом короткие кадры накапливаются и передаются вместе. Передающая станция заполняет интервал между кадрами битов расширения несущей, поэтому другие станции будут воздерживаться от передачи, пока она не освободит линию.

Для распознавания коллизий и организации *полнодуплексного режима* разработчики спецификации 802.3z применили технику, используемую при организации дуплексного режима на одной паре проводов в современных модемах и аппаратуре передачи данных абонентских окончаний ISDN (рис. 10.12).

Одним из способов обойти ограничения, связанные с расширением несущей, является использование так называемых *буферных распределителей*. Этот новый класс устройств (иногда их еще называют *полнодуплексными повторителями*) представляет собой нечто среднее между повторителем и коммутатором.

Все порты гигабитного буферного распределителя работают в полнодуплексном режиме и задействуют механизмы контроля потоков, определенные стандартом IEEE 802.3x. Как обычный повторитель Ethernet, он передает поступивший кадр на все свои порты; как и коммутатор Ethernet, способен принимать кадры на нескольких портах одновременно, при этом поступившие кадры помещаются в буферы. При заполнении буферов распределитель задействует механизмы

управления потоками для информирования передающего узла о необходимости приостановить передачу. Такой подход позволяет достичь близкой к номинальной пропускной способности в разделяемом сегменте Gigabit Ethernet.

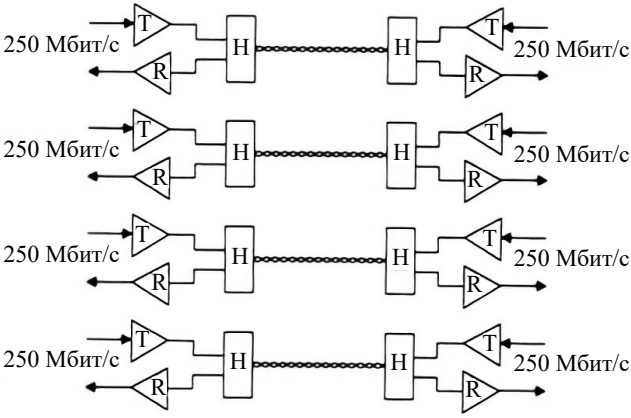


Рис. 10.12. Двунаправленная передача по четырем парам UTP 5

Также были реализованы механизмы контроля потоков на основе стандарта 802.3x. Хотя использование их не обязательно, но суть заключается в следующем. Если принимающая станция (узел) на одном конце прямого соединения оказывается перегружена, то она отправляет передающей станции так называемый *кадр приостановки передачи* (Pause Frame) с «просьбой» отказаться от передачи кадров на определенный промежуток времени. В результате передающая станция останавливает передачу данных на указанный промежуток времени. Однако принимающая станция может отправить кадр с нулевым временем ожидания с тем, чтобы отправитель возобновил передачу.

Первоначально Gigabit Ethernet использовался для увеличения пропускной способности каналов между коммутаторами и соединений между коммутаторами и серверами. Соединение коммутаторов Fast Ethernet по Gigabit Ethernet позволяло резко поднять пропускную способность магистрали локальной сети и поддерживать в результате большее число как коммутируемых, так и разделяемых сегментов Fast Ethernet. Установка сетевой платы Gigabit Ethernet на сервер давала возможность расширить канал с сервером и таким образом увеличить производительность пользователей мощных рабочих станций.

10.3.4. 10-Gigabit Ethernet

Стандарт, прежде всего, обеспечивается сохранением стандартного протокола MAC, формата Ethernet-пакетов 802.3 и диапазона допустимых размеров пакетов. Поэтому устройства, соответствующие этому стандарту, смогут взаимодействовать с оборудованием предыдущих разновидностей Ethernet, а значит сделанные в него инвестиции не обесценятся.

10-гигабитные сети отличаются от своих предшественников не только возросшей скоростью передачи. Во-первых, транспортировка трафика в них осуществляется только в дуплексном режиме. Это позволяет отказаться от протокола CSMA/CD, который сильно сдерживает производительность сетей Ethernet. Во-вторых, в сетях 10GE можно использовать преимущественно волоконно-оптические соединения.

Ориентация на оптоволоконные системы привела к необходимости прямо указывать в стандарте допустимые виды оптического волокна и алгоритмы передачи. Несмотря на то, что формально протокол Ethernet относится ко второму уровню эталонной модели OSI, ни один из принятых стандартов Ethernet не обходит стороной физическую среду передачи.

В архитектуре Ethernet физический уровень (PHY) разделяется на подуровень, зависящий от среды (PMD), и кодирующий подуровень PCS. Спецификации 802.3ae определяют два физических интерфейса: для локальных (LAN PHY) и глобальных (WAN PHY) сетей, причем второй фактически является расширением первого. Оба они используют один и тот же подуровень PMD и, следовательно, поддерживают одинаковые длины соединений.

Существенной особенностью нового стандарта является способность оборудования 10GE взаимодействовать с сетями SONET/SDH, которая означает возможность передавать пакеты Ethernet по каналам SONET/SDH с высокой эффективностью.

В результате экспансия технологии Ethernet на распределенные городские сети, которая началась с появлением стандарта Gigabit Ethernet (802.3z), теперь распространяется на сети глобальные. Именно этой цели служит интерфейс WAN PHY. Поддержка упрощенного формата кадров SONET/SDH – главное, что отличает два физических интерфейса друг от друга. Поскольку полоса пропускания интерфейсов SONET OC-192/SDH STM-64 мало отличается от таковой для 10GE (9,29 и 10 Гбит/с соответственно), был разработан единый протокол MAC.

Для кодирования данных на физическом уровне выбран алгоритм 64/66b вместо 8/10b, используемого в сетях Gigabit Ethernet. Это позволило повысить эффективность использования полосы пропускания: если в сетях 1000Base-T для достижения информационной пропускной способности 1 Гбит/с требовалась физическая полоса 1,25 Гбит/с, то в 10-гигабитных сетях вполне достаточно 3%-ного превышения (10,3 Гбит/с).

Платой за возросшую эффективность являются ограниченные возможности исправления ошибок передачи. Только внедрение алгоритма упреждающей коррекции ошибок (Forward Error-Correction, FEC) и усовершенствованных методов восстановления сигналов позволяет удерживать надежность сетей Ethernet на прежнем уровне.

Что же касается интерфейсов, зависящих от среды передачи, то из исходного многообразия потенциальных вариантов в окончательной версии стандарта останутся пять – два для многомодового и три для одномодового волокна. На применение в локальных сетях в первую очередь ориентирован интерфейс спектрального мультиплексирования (WWDM) на длине волны 1310 нм. Используемое при этом многомодовое волокно (62,5/125 мкм) является в настоящее время наиболее распространенным, а длина соединений 300 м – довольно высока по меркам локальных сетей Ethernet. В другом интерфейсе для локальных сетей задействовано менее популярное многомодовое волокно 50/125 мкм. При этом рабочая длина волны составляет 850 мкм. Данный интерфейс оставлен в окончательной версии стандарта в основном по экономическим соображениям: он предлагает недорогой вариант организации высокоскоростных соединений между серверами.

Остальные интерфейсы рассчитаны на применение в магистральных каналах городских и глобальных сетей. Максимальная длина соединения доведена до 40 км (в стандарте 802.3z она равнялась 5 км).

10.3.5. Сети на основе технологии ATM

Альтернативой технологии Ethernet является технология *асинхронного режима передачи* (Asynchronous Transfer Mode, ATM), разработанная как единый универсальный транспорт для нового поколения сетей с интеграцией услуг, которые называются ***широкополосными сетями ISDN***. Термин «асинхронный» означает, что реализуется асинхронное взаимодействие между тактовой частотой передатчика и приемника.

Технология ATM разрабатывалась как технология, способная обслужить все виды трафика (мы хотим одновременно быть подключены к интернету, смотреть IPTV и разговаривать по телефону) в соответствии с их требованиями. Последнее обстоятельство связано с тем, что практически все виды трафика создают *неравномерную по интенсивности нагрузку*. К примеру, во время обычного двухстороннего телефонного разговора уровень речевого сигнала каждого из абонентов непостоянен и имеют место различные паузы, промежутки молчания на время прослушивания собеседника. В целом передача речевого сигнала от одного из собеседников осуществляется примерно в течение 40% общего времени разговора.

Традиционным способом передачи неравномерной нагрузки является тот или иной вид коммутации сообщений (пакетов).

Пакеты ATM называются **ячейками** (Cell), так как все они имеют фиксированную длину.

Длина ячеек ATM равна 53 байтам (октетам), из которых 48 байтов отводится для передачи информации (нагрузки) и 5 байтов – для заголовка. Информация, содержащаяся в 5 байтах заголовка, достаточна для доставки сетью каждой ячейке по назначению.

Ячейки имеют два важных преимущества перед кадрами:

– требуют меньшей буферизации;

– имеют одинаковую длину, поэтому предсказуемы – их заголовки всегда находятся на одном и том же месте.

Технология ATM используется как в локальных, так и в глобальных сетях. Основная ее идея – передача цифровых, голосовых и мультимедийных данных по одним и тем же каналам.

Если передается речь, данные и видео, то необходимо преобразовать эти сигналы в ячейки и обратно. Ячейки мультиплексируются в один поток, который по линии связи поступает в облако сети ATM. Сеть ATM коммутирует и доставляет ячейки по назначению.

Используя ***инфраструктуру коммутации ячеек***, можно добавлять новые типы нагрузки без изменения самой инфраструктуры. Поскольку пользователь взаимодействует только с пограничными устройствами, то для изменения (введения) нового типа нагрузки достаточно модифицировать данные пограничные устройства. Это одна из положительных сторон технологии ATM. При необходимости без затруднений можно производить изменение или расширение сети.

Сеть ATM сохраняет порядок передачи ячеек. Иногда ячейки могут теряться, но порядок их сохраняется. Принимающее устройство воспринимает поток ячеек и преобразует его в исходный битовый поток.

В сетях ATM используются два вида соединений. Кратко рассмотрим их.

Постоянные виртуальные соединения в сетях ATM. Постоянные виртуальные соединения (Permanent Virtual Connection, PVC) устанавливаются по определенным правилам обслуживания сети.

В общем случае в сети имеется некоторая система управления (Network Management System, NMS), которая сообщает устройствам сети значения VPI/VCI для необходимых соединений:

VPI – Virtual Path Identifier – *идентификатор виртуального пути* в технологии ATM: он используется для указания, какому виртуальному маршруту принадлежит виртуальный канал, и состоит из 12 или 8 битов данных, которые записываются в заголовок при подключении к виртуальному каналу.

VCI – Virtual Circuit Identifier – идентификатор виртуального звена или канала, состоит уже из 16 битов и отвечает за конкретный канал.

VPI вместе с VCI используется для определения следующего места назначения ячейки при прохождении нескольких ATM-свитчей (рис. 10.13).

Функция VCI сходна с функцией DLCI (Data-Link Connection Identifier – *идентификатор канала передачи данных* в технологии Frame Relay.

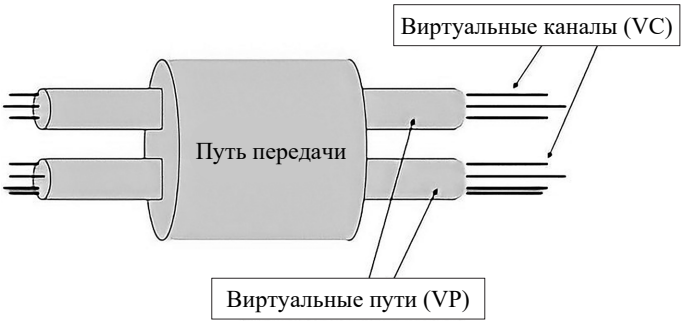


Рис. 10.13. Виртуальные пути и виртуальные каналы

Коммутаторам ATM передаются *таблицы соединений маршрутизации* (соединений между оборудованием провайдера сети и абонента), т. е. ATM-коммутаторы используют поля VPI/VCI для идентификации виртуального канала (Virtual Channel Link, VCL) следующей сети, которую должна пройти ячейка, чтобы достичь места назначения.

В публичных сетях ATM-таблицы маршрутизации могут строиться администраторами вручную.

Такие соединения наиболее целесообразно устанавливать на сети ATM с малым числом подключенных устройств со стабильным местоположением (аналогом может служить телефонная сеть, построенная на основе выделенных линий). Также эти соединения целесообразно устанавливать, если имеется значительное информационное тяготение между двумя пунктами сети. В этом случае нет необходимости часто устанавливать и отменять соединения. Именно поэтому данные соединения получили название *постоянных*.

Коммутируемые виртуальные соединения ATM. Другим видом устанавливаемых на сети ATM соединений являются коммутируемые виртуальные соединения (Switched Virtual Connection, SVC). Эта техника позволяет терминалу устанавливать и отменять соединения в динамике.

Для возможности установления коммутируемых соединений используется *протокол сигнализации* (Signaling Protocol).

Рассмотрим процесс установления SVC-коммутации более подробно.

Предположим, что абонент *A* хочет связаться с абонентом *B*. Сначала по сигнальному каналу передается сообщение *установки*

(Setup). Это сообщение содержит множество информации, в том числе: адреса обоих абонентов, характеристики трафика, требуемое качество обслуживания. Кроме того, в сообщении установки содержится *идентификатор* (указатель) *вызова*, поскольку с помощью канала сигнализации VPI/VCI устанавливается множество соединений и они должны быть различимы для сети.

На этой стадии ближний к абоненту *A* коммутатор просто подтверждает соединение и определяет для него значение VPI/VCI. Соединение еще реально не установлено – оно не может использоваться для передачи информации. Однако терминал уже имеет значение VPI/VCI.

Далее сеть проверяет ресурсы, ищет пути и т. д., стараясь найти путь к абоненту *B* с соответствующими характеристиками.

Следующий шаг является эквивалентом вызова (звонка). Сообщение, поступающее в терминал вызываемого абонента, идентифицирует входящий вызов, включает информацию о характеристиках трафика и пр.

Вызываемый терминал подтверждает получение данной информации. Если у абонента *B* достаточно ресурсов, тогда он посылает сети сообщение о соединении со смыслом «Я готов!».

И наконец, сообщение о готовности проходит обратно через сеть к абоненту *A*. В нем говорится: «Соединение установлено. Можно начать передачу информации».

Проблемы совмещения технологий ATM и существующих сетей решаются организацией *ATM Forum* и рядом промышленных фирм. Разрабатываются коммутаторы и концентраторы, обеспечивающие совместную работу ATM-магистралей, сетей, работающих по протоколам TCP/IP, и локальных сетей, таких как Ethernet, Fast Ethernet, FDDI. В частности, разработаны спецификации IP-over-ATM и более современные MPOA (Multi-Protocol-Over-ATM), а также реализующие их средства для передачи IP-дейтаграм и пакетов, сформированных по другим протоколам, через ATM-сети.

При реализации TCP/IP поверх ATM-протоколов необходимо сохранить высокую скорость ATM-сети. Однако этому препятствуют возможные потери при передаче некоторых 53-байтных ячеек, на которые разбивается TCP-сегмент. Такая потеря вызывает необходимость повторной передачи всех ячеек сегмента, поскольку в ATM контроль правильности передачи ведется по отношению ко всему сообщению (в данном случае – сегменту).

Выводы

1. Технология Gigabit Ethernet добавляет новую, 1000 Мбит/с, ступень в иерархии скоростей семейства Ethernet. Эта ступень позволяет эффективно строить крупные локальные сети, в которых мощные серверы и магистрали нижних уровней сети работают на скорости 100 Мбит/с, а магистраль Gigabit Ethernet объединяет их, обеспечивая достаточно большой запас пропускной способности.

2. Разработчики технологии Gigabit Ethernet сохранили большую степень преемственности с технологиями Ethernet и Fast Ethernet. Gigabit Ethernet использует те же форматы кадров, что и предыдущие версии Ethernet, работает в полнодуплексном и полудуплексном режимах, поддерживая на разделяемой среде тот же метод доступа CSMA/CD с минимальными изменениями.

3. Для обеспечения приемлемого максимального диаметра сети в 200 м в полудуплексном режиме разработчики технологии пошли на увеличение минимального размера кадра с 64 до 512 байтов. Разрешается также передавать несколько кадров подряд, не освобождая среду, на интервале 8096 байтов, при этом кадры не обязательно дополнять до 512 байтов. Остальные параметры метода доступа и максимального размера кадра остались неизменными.

4. В 1998 г. был принят стандарт 802.3z, который определяет использование в качестве физической среды трех типов кабеля: многомодового оптоволоконного (расстояние до 500 м), одномодового оптоволоконного (расстояние до 5000 м) и двойного коаксиального (twinax), по которому данные передаются одновременно по двум медным экранированным проводникам на расстояние до 25 м.

5. Реализация Gigabit Ethernet на UTP категории 5 или 6 требует использования всех четырех пар медного кабеля данной категории – является на сегодняшний день самым распространенным вариантом локальных сетей.

6. В целом 10-Gigabit Ethernet является наиболее перспективным направлением развития кабельных сетей. Очевидно, что 10-гигабитные сети будут отличаться от своих предшественников как возросшей скоростью передачи, так и возможностью отказаться от протокола CSMA/CD, который сильно сдерживал производительность сетей Ethernet.

7. Альтернативой технологии Ethernet является технология *асинхронного режима передачи* (Asynchronous Transfer Mode, ATM), разработанная как единый универсальный транспорт для нового поколения сетей с интеграцией услуг, которые называются широкополосными сетями ISDN. Термин *асинхронный* означает, что реализуется асинхронное взаимодействие между тактовой частотой передатчика и приемника.

Технология *ATM* используется как в локальных, так и в глобальных сетях. Основная ее идея – передача цифровых, голосовых и мультимедийных данных по одним и тем же каналам.

Пакеты ATM называются *ячейками*, так как все они имеют фиксированную длину – 53 байта.

10.4. Удаленный доступ и виртуальные частные сети

Возможность использования удаленными пользователями ресурсов локальной сети называется **удаленным доступом** (Remote Access).

В настоящее время основным видом удаленного доступа является *соединение с использованием виртуальных частных сетей* (Virtual Private Networks, VPN), построенное по модели *клиент-сервер*.

Клиент удаленного доступа (Remote Access Client, RAC) – это компьютер, который имеет возможность подключаться к удаленному компьютеру и работать с его ресурсами или с ресурсами удаленной сети так же, как с ресурсами своей локальной сети. Единственное отличие удаленной работы от локальной с точки зрения клиента – более низкая скорость соединения.

Сервер удаленного доступа (Remote Access Server, RAS) – это компьютер, способный принимать входящие запросы от клиентов удаленного доступа и предоставлять им собственные ресурсы или ресурсы своей локальной сети.

Компьютер с установленной операционной системой Windows Server 2003 может исполнять роль как клиента удаленного доступа, так и сервера. В последнем случае на нем должна быть запущена *Служба маршрутизации и удаленного доступа* (Routing and Remote Access Service, RRAS).

10.4.1. Виды коммутируемых линий

Соединения по коммутируемым линиям могут осуществляться с использованием следующих средств связи.

1. *Телефонные сети* – наиболее распространенный и дешевый вариант, хотя и самый медленный (максимальная скорость передачи данных 56,6 Кбит/с). Предполагает установку модемов на клиенте и сервере.

2. *Сети ISDN* (Integrated Services Digital Network – цифровая сеть с комплексными услугами) обеспечивают скорость передачи данных 128 Кбит/с, но их использование дороже, чем использование обычных телефонных сетей.

3. *ATM поверх ADSL* – передача трафика ATM (Asynchronous Transfer Mode – асинхронный режим передачи) посредством линий ADSL (Asymmetric Digital Subscriber Line – асимметричная цифровая абонентская линия).

Для соединения посредством виртуальных частных сетей клиентский компьютер и сервер должны быть подключены к Интернету.

10.4.2. Протоколы удаленного доступа

Подключение клиента к серверу удаленного доступа по коммутируемым линиям состоит из следующих основных этапов:

– *установка соединения*;

– *аутентификация и авторизация клиента* удаленного доступа;

– *сервер удаленного доступа, который выступает в роли маршрутизатора*, предоставляя доступ клиенту к ресурсам локальной сети, серверам баз данных, электронной почте, файловым серверам, принтерам и т. д.

Схема подключения представлена на рис. 10.14.

Для соединений удаленного доступа по коммутируемым линиям было разработано несколько специальных протоколов. Например, Windows Server 2003 поддерживает два протокола удаленного доступа:

– *протокол SLIP* (Serial Line Internet Protocol – межсетевой протокол для последовательного канала);

– *протокол PPP* (Point-to-Point Protocol – протокол соединения «точка-точка»).

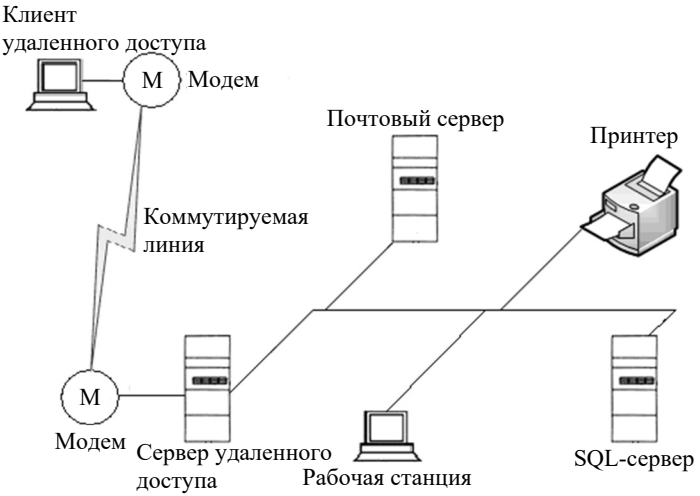


Рис. 10.14. Схема подключения удаленного доступа по коммутируемым линиям

! Протокол SLIP является одним из старейших протоколов удаленного доступа и предлагает передачу TCP/IP-пакетов без обеспечения безопасности данных и контроля целостности.

Протокол описан в RFC 1055. В Windows Server 2003 поддержка протокола SLIP реализована только на уровне клиента.

! Протокол PPP предназначен для коммутируемых соединений типа «точка-точка». Это означает, что в протоколе отсутствуют средства адресации, поэтому в процессе связи могут принимать участие только два компьютера – клиент и сервер.

Протокол PPP, в отличие от SLIP, обеспечивает функции безопасности и контроля ошибок на основе корректирующих кодов.

Описание протокола PPP содержится в RFC 1332, 1661 и 1662.

Соединение «точка-точка» устанавливается в четыре этапа.

1. *Настройка параметров канального уровня*. Клиент и сервер согласовывают максимальный размер кадра, возможность сжатия, протокол аутентификации и некоторые другие параметры.

2. *Аутентификация клиента*. Сервер осуществляет аутентификацию и авторизацию клиента на основе протокола, выбранного на предыдущем этапе.

3. *Обратный вызов* (Callback). В целях безопасности может использоваться процедура обратного вызова, когда сервер разрывает соединение с клиентом и сам вызывает его по определенному телефонному номеру.

4. *Настройка протоколов верхних уровней*. Сервер отправляет клиенту список протоколов верхних уровней, отвечающих за передачу данных, шифрование и сжатие. Клиент выбирает один из подходящих протоколов списка.

10.4.3. Протоколы аутентификации удаленных клиентов

Разработано несколько протоколов, используемых для *аутентификации удаленных клиентов*.

1. *PAP* (Password Authentication Protocol) – протокол аутентификации по паролю (описан в RFC 1334). Самый простой протокол аутентификации, в котором имя пользователя и пароль передаются открытым, незашифрованным способом. В Windows Server 2003 протокол PAP применяется только в том случае, если клиент удаленного доступа не поддерживает больше никаких протоколов.

2. *CHAP* (Challenge Handshake Authentication Protocol) – протокол аутентификации с предварительным согласованием вызова (описан в RFC 1994). В этом протоколе клиент посылает серверу пароль в виде специальной хеш-последовательности, созданной с использованием *алгоритма MD-5*. Сервер принимает *хеш (свертку) пароля клиента*, вычисляет хеш по хранимому у себя паролю и сравнивает обе последовательности. В случае совпадения соединение устанавливается, иначе происходит разрыв. Недостатком является отсутствие взаимной аутентификации, т. е. сервер аутентифицирует клиента, а клиент не получает информации о подлинности сервера.

3. *MS-CHAP* (Microsoft Challenge Handshake Authentication Protocol) – реализация протокола CHAP, разработанного Microsoft (описан в RFC 2433). Действует по принципу протокола CHAP за исключением того, что для хеширования используется *алгоритм MD-4*, а не MD-5.

4. *MS-CHAPv2* – вторая версия протокола MS-CHAP (описан в RFC 2759), где также, как и в MS-CHAP, применяется *алгоритм хеширования MD-4*, но отличием является требование взаимной аутентификации. Между клиентом и сервером происходит обмен следующими сообщениями:

– сервер отправляет клиенту сообщение, содержащее некоторую последовательность символов, называемую ***строкой вызова***;

– клиент отправляет серверу хеш-последовательность, полученную на основе строки вызова и пароля пользователя, а также свою строку вызова для сервера;

– сервер вычисляет хеш по своей строке вызова и пользовательскому паролю, сравнивает его с полученным хешем от клиента и в случае успеха отправляет хеш, вычисленный на основе своей строки вызова, строки вызова от клиента, имени и пароля пользователя;

– клиент, получая сообщение сервера, вычисляет хеш на основе тех же данных, и в случае совпадения найденного хеша с полученным от сервера процесс взаимной аутентификации считается законченным успешно.

5. *EAP* (Extensible Authentication Protocol) – *расширяемый протокол аутентификации* (описан в RFC 2284). Отличается от вышеописанных протоколов тем, что выбор типа аутентификации EAP происходит в процессе соединения.

6. В ОС Windows Server применяются следующие типы аутентификации EAP: EAP-MD5, CHAP, EAP-TLS (Transport Level Security, безопасность на транспортном уровне), PEAP (Protected EAP, защищенный EAP).

10.4.4. Общая характеристика виртуальных сетей. Сети VLAN и VPN

Виртуализация сетей. В последние годы стоимость использования каналов связи сети Интернет уменьшилась и стала ниже, чем цена использования коммутируемых линий. Однако при установлении соединения через Интернет серьезной проблемой является обеспечение безопасности, так как сеть является открытой и злоумышленники могут перехватывать пакеты с конфиденциальной информацией. Решением этой проблемы стала ***технология виртуальных сетей*** (Virtual Networking, Virtual LAN, VLAN).

В наиболее общем смысле ***концепция виртуальной локальной сети состоит в следующем: локальная сеть включает все устройства в одном широковещательном домене***.

Широковещательный домен – это устройства, подключенные к локальной сети таким образом, что при отправлении широковещательного кадра одним из устройств остальные получают копию этого кадра.

Коммутатор с настройками по умолчанию «считает», что все его интерфейсы находятся в одном широковещательном домене: когда широковещательный кадр приходит на один конкретный порт коммутатора, устройство пересылает этот кадр на все остальные свои порты. В связи с таким принципом работы коммутатора, чтобы создать два разных широковещательных домена, придется купить два разных коммутатора для локальной сети Ethernet.

Таким образом, понятия *локальной сети* и *широковещательного домена* являются практически одинаковыми.

Подобно широковещательным доменам на базе маршрутизаторов в *виртуальной ЛВС* широковещательные пакеты и пакеты с неизвестными адресами получают все устройства, если такие пакеты исходят из того же домена (виртуальной сети). Здесь нет ничего нового, такие же методы используются в традиционных сетях на базе концентраторов и маршрутизаторов.

Виртуальная означает логическая, т. е. работающая в рамках общедоступной сети, в отличие от *частной сети*, которая создается на базе специально арендованных для этой цели линий.

В традиционных сетях трафик является широковещательным внутри образующего сегмент концентратора и маршрутизируется между концентраторами. При использовании виртуальных сетей кадры становятся широковещательными внутри VLAN и маршрутизируются между ними. Таким образом, виртуальные сети представляют собой не что иное, как более гибкий вариант традиционных ЛВС с несколько большими возможностями.

Виртуальная ЛВС (и связанные с ней коммутаторы) должна поддерживать различные типы физических сред. В коммутируемых сетях возможна работа централизованных ресурсов (магистралей) с более высокими скоростями, нежели скорость рабочих станций. Например, рабочие станции Ethernet (10 Мбит/с) могут работать с серверами Fast Ethernet, Gigabit Ethernet или ATM.

Существует достаточно много вариантов реализации VLAN. Простые варианты VLAN представляют собой набор портов коммутатора, более сложные реализации позволяют создавать группы на основе других критериев. В общем случае возможности организации VLAN тесно связаны с возможностями коммутаторов.

Для создания системы, построенной на виртуальных ЛВС, сетевому администратору нужно решить, сколько будет виртуальных сетей, какие компьютеры будут в них входить и как они будут называться (иногда их обозначают разными цветовыми оттенками; см. рис. 10.15).

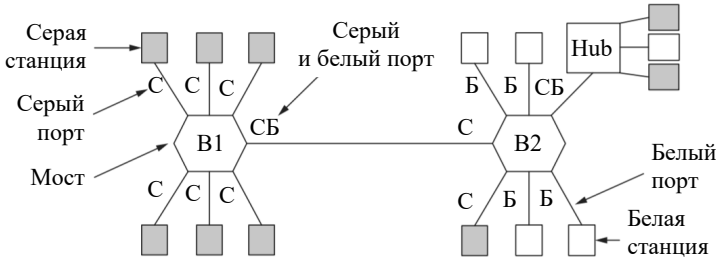


Рис. 10.15. Две виртуальные сети, серая и белая, с мостом

На рис. 10.15 девять машин входят в виртуальную сеть С (серая), а еще пять машин – в виртуальную сеть Б (белая). Машины «серой» сети распределены между двумя коммутаторами, в том числе две машины подключены к коммутатору через концентратор.

Чтобы виртуальные сети функционировали корректно, необходимо наличие *конфигурационных таблиц* в мостах. Эти таблицы сообщают о том, через какие порты производится доступ к тем или иным виртуальным сетям.

Сформулируем основные причины для разделения хостов на разные VLAN:

1) широковещательная передача, отправленная одним узлом во VLAN, будет приниматься и обрабатываться всеми другими узлами этой VLAN, но не узлами из другой VLAN. Чем меньше посторонних

узлов в сети получают широковещательные кадры, тем выше *безопасность локальной сети*;

2) чтобы уменьшить нагрузку на процессор на каждом устройстве, повысить производительность узла путем уменьшения числа устройств, которые принимают каждый широковещательный кадр;

3) чтобы повысить безопасность хостов за счет применения различной политики безопасности для каждой VLAN;

4) для создания подразделений, группирующих пользователей по отделам или группам, которые работают вместе, а не по физическому местоположению;

5) уменьшение нагрузки для *протокола связующего дерева* (Spanning Tree Protocol, STP) путем ограничения VLAN одним коммутатором доступа; STP – канальный протокол, основной задачей которого является устранение петель в топологии произвольной сети Ethernet, в которой есть один или более сетевых мостов, связанных избыточными соединениями; STP решает эту задачу, автоматически блокируя соединения, которые в данный момент для полной связности коммутаторов являются избыточными.

Виртуальные частные сети (Virtual Private Network, VPN) – это защищенное соединение двух узлов через открытые сети. При этом организуется виртуальный канал, обеспечивающий безопасную передачу информации, а узлы, связанные VPN, могут работать так, как будто соединены напрямую.

Частная – значит не публичная, т. е. такая, в которой находятся только «дозволенные» узлы. Именно эта составляющая VPN и является самой главной, так как она определяет ряд требований к этой самой «частности».

VPN следует использовать в корпоративных сетях компаний, где удаленные пользователи работают на незащищенных соединениях.

Компьютер, инициирующий VPN-соединение, называется **VPN-клиентом**, а тот, с которым устанавливается соединение, – **VPN-сервером**.

VPN-магистраль – это последовательность каналов связи открытой сети, через которые проходят пакеты виртуальной частной сети.

На рис. 10.16 продемонстрирована работа VPN-сети. Компьютеры с IP-адресами 1.1.1.1–1.1.1.100 подключаются через сетевой шлюз, который также выполняет функцию VPN-сервера.

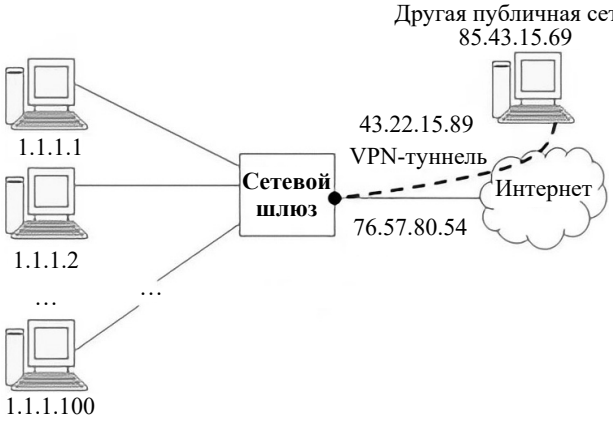


Рис. 10.16. Принцип работы VPN-сети

Когда происходит подключение через VPN, в заголовке сообщения передается информация об IP-адресе VPN-сервера и удаленном маршруте. Исходные данные, проходящие по общей или публичной сети, невозможно перехватить, так как вся информация зашифрована.

Этап VPN-шифрования реализуется на стороне отправителя, а расшифрования – на стороне получателя по заголовку сообщения (при наличии общего ключа шифрования).

После корректного расшифрования сообщения между двумя сетями устанавливается *VPN-подключение*, которое позволяет также работать в публичной сети (например, обмен данными с клиентом 85.43.15.69).

Существует два типа VPN-соединений:

– *соединение с удаленными пользователями* (Remote Access VPN Connection), или *шлюз защищенного удаленного доступа к VPN*;

– *соединение маршрутизаторов* (Router-to-Router VPN Connection) или *VPN типа «сеть-сеть».*

Соединение с удаленными пользователями осуществляется в том случае, если одиночный клиент подключается к локальной сети организации через VPN (рис. 10.17). Другие компьютеры, подключенные к VPN-клиенту, не могут получить доступ к ресурсам локальной сети. Этот тип соединения позволяет пользователям подключиться к другой сети (к интернету или внутренней системе своей компании) по *частному зашифрованному туннелю.*

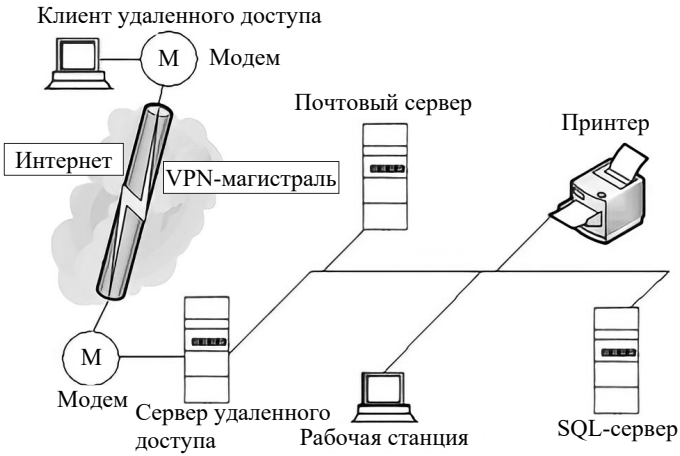


Рис. 10.17. Схема VPN-соединения с удаленным пользователем

Соединение маршрутизаторов устанавливается между двумя локальными сетями, если узлы обеих сетей нуждаются в доступе к ресурсам друг друга (рис. 10.18). При этом один из маршрутизаторов играет роль VPN-сервера, а другой – VPN-клиента.

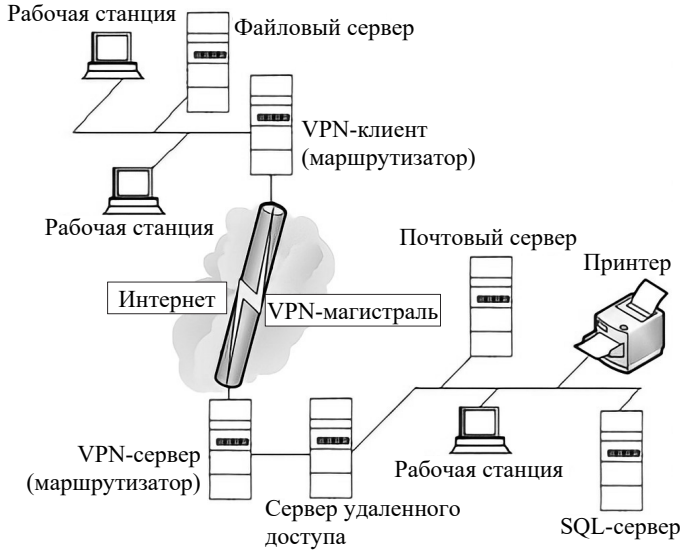


Рис. 10.18. Схема VPN-соединения между маршрутизаторами

VPN типа «сеть – сеть» используется для создания закрытой внутренней сети, где все офисы могут подключаться друг к другу. Эта технология известна как *интранет.*

> Одно из ***основных преимуществ VPN*** – между двумя сторонами необходима подключенная платформа, которая не только быстро *масштабируется*, но и (в первую очередь) *обеспечивает секретность данных, целостность этих данных и аутентификацию.*

10.4.5. Протоколы виртуальных частных сетей

Безопасность передачи IP-пакетов через Интернет в VPN реализуется с помощью *туннелирования*.

> **Туннелирование** (Tunneling) – это процесс включения IP-пакетов в пакеты другого формата, позволяющий передавать зашифрованные данные через открытые сети.

VPN-туннелирование используется с целью:

– построения и реализации видеоконференций;

– обслуживания цифровой телефонии с большим набором телекоммуникационных услуг;

– объединения удаленных офисов, а также мелких отделений;

– организации безопасной корпоративной сети.

Сценарии использования VPN могут быть разными, самые популярные их них:

– построение *защищенного канала* между двумя или более удаленными сегментами сети (например, между офисами в Минске и Гродно);

– *подключение удаленного работника* к корпоративной сети (теперь об этом знает почти каждый офисный сотрудник);

– *виртуальное изменение местоположения* с помощью услуг *VPN Providers* (весь трафик будет проходить через чужой сервер).

Для реализации этих сценариев существуют различные виды *VPN-протоколов* – для связи, шифрования трафика и др.

В системах Windows Server поддерживаются следующие протоколы туннелирования.

1. *PPTP* (Point-to-Point Tunneling Protocol) – *протокол туннелирования соединений «точка-точка»*, один из старейших протоколов, основан на протоколе PPP (описан в RFC 2637).

PPTP использует два соединения: одно для управления, другое для инкапсуляции данных. Первое работает с использованием TCP, в котором порт сервера – 1723. Второе – с помощью протокола GRE (Generic Routing Encapsulation – общая инкапсуляция маршрутов, протокол туннелирования сетевых пакетов, разработанный компанией Cisco Systems), который является транспортным протоколом (т. е. заменой TCP/UDP). Этот факт может затруднять подключение с сервером, так как для них установлено подключение «точка-точка». Данная технология поддерживается большим количеством современного клиентского сетевого оборудования.

PPTP поддерживается нативно на всех версиях Windows и большинстве других операционных систем. Несмотря на относительно высокую скорость, PPTP не слишком надежен: после обрыва соединения он не восстанавливается так же быстро, как, например, OpenVPN.

Поддерживает все возможности, предоставляемые PPP, в частности аутентификацию по протоколам PAP, CHAP, MS-CHAP, MS-CHAPv2, EAP. Шифрование данных обеспечивается методом MPPE (Microsoft Point-to-Point Encryption), который применяет алгоритм RSA/RC4. Сжатие данных происходит по протоколу MPPC (Microsoft Point-to-Point Compression), описанному в RFC 2118.

Недостатком протокола является относительно низкая скорость передачи данных.

2. *L2TP* (Layer Two Tunneling Protocol – *туннельный протокол канального уровня*) – протокол туннелирования, основанный на протоколе L2F (Layer Two Forwarding), разработанном компанией Cisco, и протоколе PPTP (описан в RFC 2661). Поддерживает те же протоколы аутентификации, что и PPP. Для шифрования данных используется протокол IPsec. Также поддерживает сжатие данных. Имеет более высокую скорость передачи данных, чем PPTP.

Протокол PPTP остается единственным протоколом, который поддерживают старые версии Windows (Windows NT 4.0, Windows 98, Windows Me). Однако существует бесплатный VPN-клиент Microsoft L2TP/IPSec, который позволяет старым операционным системам Windows устанавливать соединение VPN по протоколу L2TP.

3. *IPSec*. Internet Protocol Security (IPSec) – это набор протоколов для обеспечения защиты данных, передаваемых по IP-сети.

В отличие от SSL, который работает на прикладном уровне, IPSec работает на сетевом уровне и может использоваться нативно со многими операционными системами, что позволяет пользоваться им без сторонних приложений (в отличие от OpenVPN).

IPSec стал очень популярным протоколом для использования в паре с L2TP или IKEv2 (о чем мы поговорим ниже).

IPSec шифрует весь IP-пакет, используя:

– *Authentication Header* (AH) – *аутентификационный заголовок*, который ставит *цифровую подпись* на каждом пакете;

– протокол *Encapsulating Security Payload* (ESP), который обеспечивает конфиденциальность, целостность и аутентификацию пакета при передаче.

Кроме того, ESP позволяет идентифицировать отправителя данных, а также обеспечить защиту от воспроизведения информации.

Отличие протокола ESP от протокола Authentication Header (AH) состоит в том, что ESP выполняет шифрование данных. При этом оба протокола обеспечивают идентификацию, проверку целостности и защиту от воспроизведения информации. При работе с ESP для шифрования и расшифровки данных обе конечные системы применяют общий ключ.

Если одновременно применяются средства шифрования и идентификации данных, то отвечающая система вначале идентифицирует пакет, а если идентификация выполнена успешно, то расшифровывает пакет. Такой способ обработки пакетов снижает нагрузку на систему и уменьшает риск взлома защиты с помощью атаки типа «*отказ в обслуживании*, DoS».

4. *IKEv2/IPSec*. Internet Key Exchange (*протокол согласования ключей*) version 2 (IKEv2) является протоколом IPSec, используемым для выполнения *взаимной аутентификации*, создания и обслуживания Security Associations (SA), стандартизован в RFC 7296. Так же как и L2TP, защищен IPSec, что может говорить об их одинаковом уровне безопасности. Хотя IKEv2 был разработан Microsoft совместно с Cisco, существуют реализации протокола с открытым исходным кодом (например, OpenIKEv2, Openswan и Strongswan).

Благодаря поддержке Mobility and Multi-homing Protocol (MOBIKE) IKEv2 очень устойчив к смене сетей. Это делает IKEv2 отличным выбором для пользователей смартфонов, которые регулярно переключаются между домашним Wi-Fi и мобильным соединением или перемещаются между точками доступа.

Во многих случаях IKEv2 быстрее OpenVPN, так как он менее ресурсоемкий. С точки зрения производительности IKEv2 может быть лучшим вариантом для мобильных пользователей, потому как он хорошо переустанавливает соединения. IKEv2 нативно поддерживается на Windows 7+, Mac OS 10.11+, iOS, а также на некоторых Android-устройствах.

IKEv2/IPSec может использовать ряд различных криптографических алгоритмов, включая AES, Blowfish и Camellia, в том числе с 256-битными ключами.

5. *OpenVPN*. OpenVPN – это универсальный протокол VPN с открытым исходным кодом, разработанный компанией OpenVPN Technologies.

Сейчас *OpenVPN* – это, пожалуй, самый популярный протокол VPN. Будучи открытым стандартом, он прошел не одну независимую экспертизу безопасности.

В большинстве ситуаций, когда нужно подключение через VPN, скорее всего подойдет протокол OpenVPN. Он стабилен и предлагает хорошую скорость передачи данных. К тому же OpenVPN использует стандартные протоколы TCP и UDP, что позволяет ему стать альтернативой IPSec тогда, когда провайдер блокирует некоторые протоколы VPN.

Для работы OpenVPN нужно специальное клиентское программное обеспечение. Большинство VPN-сервисов создают свои приложения для работы с OpenVPN, которые можно использовать в разных операционных системах и устройствах. Протокол может работать на любом из портов TCP и UPD и может использоваться на всех основных платформах через сторонние клиенты: Windows, Mac OS, Linux, Apple iOS, Android.

С точки зрения пользователя, VPN – простая технология. Для ее использования нужно загрузить и установить приложение на свое устройство, выбрать нужный сервер и нажать кнопку подключения.

Если выбран качественный VPN-сервис, то можно быть уверенным, что после этих действий онлайн-активность пользователя будет надежно защищена.

Последнее обусловлено тем, что хороший VPN-сервис использует самое современное шифрование для защиты данных; хороший VPN-сервис также гарантирует, что сайты, обрабатывающие данные клиента, не видят, откуда они пришли, обеспечивая высокую степень анонимности. Следует искать и применять сервис, обеспечивающий *шифрование* AES-256 (является отраслевым стандартом в некоторых странах СНГ; более подробно анализ *криптографических алгоритмов* в сетевых технологиях будет рассмотрен в главе 13) и использующий надежные протоколы, такие как, например, OpenVPN и IKEv2.

При использовании виртуальной частной сети никто не увидит настоящий *IP-адрес любого пользователя*, потому что вместо него теперь будет распознаваться адрес VPN.

Проанализируем ***организацию туннеля IPSec VPN***. Для этого понадобятся два *интернет*-центра на основе беспроводных роутеров. Для построения туннеля можно использовать различные сочетания моделей, например Ultra II – Ultra II, Giga III – Giga III и Ultra II – Giga III.

Рассмотрим пример объединения двух сетей через IPSec VPN с помощью интернет-центров Keenetic Ultra II и Giga III. Установить туннель IPSec можно как в локальной сети (когда на внешнем интерфейсе WAN-роутера используются частные/внутренние IP-адреса (IPv4), их называют еще «серые»), так и через глобальную сеть Интернет (когда на внешнем интерфейсе WAN-роутера используются публичные/внешние IP-адреса, их называют также «белые»). Желательно иметь статический/постоянный IP-адрес на WAN-интерфейсе или можно воспользоваться сервисом *динамических доменных имен*, DyDNS.

В примере организация туннеля IPSec VPN осуществляется в рамках локальной сети, т. е. используются частные/внутренние IP-адреса.

! Наличие публичного IP-адреса на роутере или компьютере позволит организовать собственный сервер (VPN, FTP, WEB и др.), удаленный доступ к компьютеру, камерам видеонаблюдения и даст возможность получить к ним доступ из любой точки глобальной сети. С «белым» IP-адресом можно организовать любой собственный домашний сервер для публикации его в сети Интернет: веб (HTTP), VPN (L2TP/IPSec, PPTP, IPSec, OpenVPN, WireGuard), медиа (аудио/видео), FTP, сетевой накопитель NAS, игровой сервер и т. д.

Для построения IPSec VPN-туннеля через глобальную сеть Интернет на внешнем интерфейсе WAN-роутера должен использоваться публичный/внешний IP-адрес. Адреса локальной и удаленной сети должны быть из разных подсетей. Например, в данном случае объединяемые сети имеют разные адресные пространства: 192.168.1.0/24 и 192.168.2.0/24 (рис. 10.19.)

Рассмотрим сценарий, в котором Keenetic Ultra II будет выступать в роли ожидающего подключение IPSec VPN (условно назовем его *сервером*), а Keenetic Giga III – в роли инициатора подключения IPSec VPN (условно назовем его *клиентом*).

В нашем примере роутер Keenetic Ultra II работает в домашней сети 192.168.1.0 (клиенты этой сети получают IP-адреса в диапазоне от 192.168.1.2 и выше) и имеет IP-адрес для управления 192.168.1.1. На внешнем интерфейсе WAN этого роутера установлен вручную статический IP-адрес из другой подсети: 10.10.1.1 (в вашем случае это может быть внешний/публичный IP-адрес).

Роутер Keenetic Giga III работает в другой сети: 192.168.2.0 (клиенты этой сети получают IP-адреса в диапазоне от 192.168.2.2 и выше) и имеет IP-адрес управления 192.168.2.1. На внешнем интерфейсе WAN этого роутера установлен вручную статический IP-адрес из подсети 1.

Для проверки работоспособности туннеля с компьютера локальной сети Giga III и имеющего IP-адрес 192.168.2.35 выполнили *пинг* компьютера удаленной сети 192.168.1.33, который находится за туннелем IPSec в локальной сети роутера Ultra II.

Обратим внимание, что широковещательные broadcast-запросы (например, NetBIOS) не будут проходить через VPN-туннель, поэтому в сетевом окружении имена удаленных хостов не будут отображаться (доступ к ним возможен по IP-адресу, например 192.168.1.33).

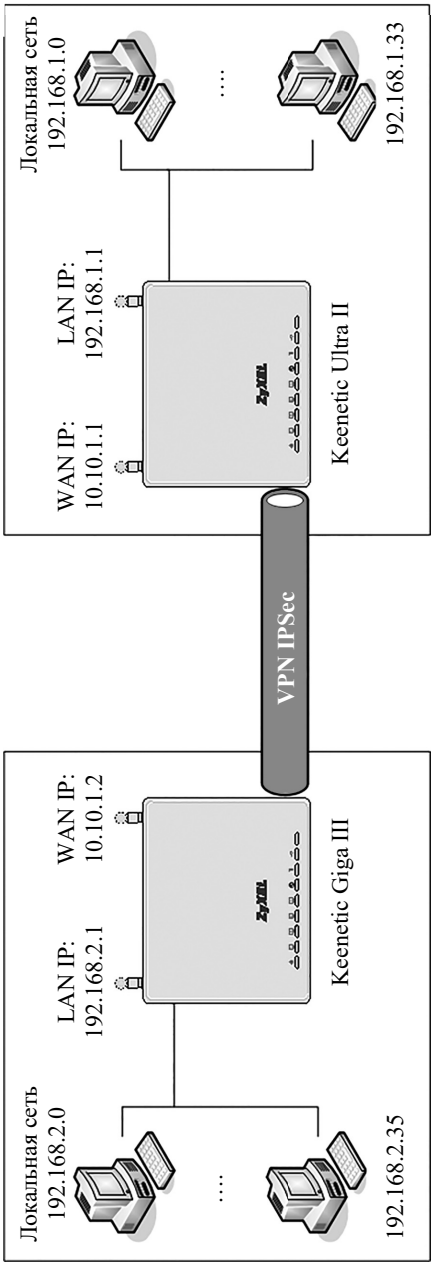


Рис. 10.19. Пример организация IPSec VPN-туннеля

Выводы

1. Использование удаленными пользователями ресурсов локальной сети называется удаленным доступом.

2. Различают два основных вида удаленного доступа: соединение по коммутируемой линии; соединение с использованием виртуальных частных сетей. Оба вида соединений работают по модели «клиент-сервер».

3. Для соединений удаленного доступа по коммутируемым линиям было разработано несколько специальных протоколов, но в Windows Server 2003 поддерживается только два: протокол SLIP; протокол PPP, отличающийся обеспечением функций безопасности и контроля ошибок.

4. В современных операционных системах серверного сегмента применяются следующие протоколы аутентификации: PAP, CHAP, MS-CHAP, MS-CHAPv2, EAP.

5. Под виртуальными частными сетями понимают защищенное соединение двух узлов через открытые сети, при этом организуется виртуальный канал, обеспечивающий безопасную передачу информации, а узлы, связанные с VPN, могут работать так, как будто соединены напрямую.

6. Безопасность передачи IP-пакетов через Интернет в VPN реализуется с помощью туннелирования, которое в свою очередь обеспечивается протоколами PPTP и L2TP, IPSec, IKEv2/IPSec, OpenVPN.

7. Для использования VPN нужно загрузить и установить на свое устройство приложение, выбрать нужный сервер и нажать кнопку подключения. Если выбран качественный VPN-сервис, то можно быть уверенным, что после этих действий онлайн-активность пользователя будет надежно защищена.

8. Повышенный уровень безопасности сетей VPN обусловлен следующим:

– история просмотров становится анонимной – VPN скрывает историю просмотров и историю поиска от вашего интернет-провайдера (ISP); провайдер сможет видеть только ваш зашифрованный трафик, который отправляется на VPN-сервер;

– можно изменить виртуальное местонахождение;

– интернет-активность становится анонимной – безлоговый VPN гарантирует, что никто не узнает о том, что вы делаете в интернете;

это особенно полезно людям, которые загружают торренты (Torrent – это интернет-протокол, позволяющий быстро скачивать большие файлы и обмениваться данными между разными сетями) и пользуются сетями Peer-to-Peer (P2P).

Контрольные вопросы

1. Что называют сотовой радиосвязью?
2. Опишите принципы организации сотовой сети.
3. Какие существуют структуры сотовых систем?
4. Опишите функционирование сотовой сети.
5. Охарактеризуйте сотовые сети первого поколения.
6. Опишите сотовые сети второго поколения.
7. Приведите архитектуру глобальной системы мобильной связи.
8. Опишите сотовые сети третьего поколения.
9. Представьте основные характеристики сотовых систем третьего поколения.
10. Опишите сотовые сети четвертого поколения.
11. Дайте характеристику сотовых сетей пятого поколения.
12. Что такое сети Bluetooth?
13. Какова область применения сетей Bluetooth?
14. Опишите стек протоколов Bluetooth.
15. Какие существуют модели использования сетей Bluetooth?
16. Приведите общую характеристику стандарта Gigabit Ethernet.
17. Какие типы кабеля могут использоваться в сетях на базе технологии Gigabit Ethernet?
18. Что такое твинаксиальный кабель?
19. Какой метод доступа используется в сетях Gigabit Ethernet?
20. Опишите технологию ATM.
21. Дайте определение удаленного доступа.
22. Приведите определение виртуальных частных сетей.
23. Перечислите основные протоколы виртуальных частных сетей.
24. Приведите перечень основных протоколов удаленного доступа.
25. Перечислите протоколы аутентификации, используемые в технологиях удаленного доступа.

11

ОСНОВНЫЕ ЭЛЕМЕНТЫ СОВРЕМЕННЫХ КОМПЬЮТЕРНЫХ СЕТЕЙ И СЕТЕВЫХ ТЕХНОЛОГИЙ

11.1. Сетевые экосистемы

Выше уже упоминалось о пяти ключевых технологиях, которые преобразовали сеть:

– *программно-определяемые* (или *программно-реконфигурируемые*) *сети* (Software-Defined Networking, SDN);

– *облачные сервисы* (Cloud Computing Services, CCS);

– *интернет вещей* (Internet of Things, IoT);

– *виртуализация сетевых функций* (Network Functions Virtualization, NFV);

– *качество взаимодействия, восприятия* (Quality of Experience, QoE).

Далее проанализируем более подробно важнейшие особенности и элементы этих новшеств.

Согласно прогнозам известной консалтинговой компании, Gartner, одним из трендов развития цифровых технологий становятся *сетевые (цифровые) экосистемы* (СЭС). Главный элемент любой цифровой экосистемы – *технология единого входа* (Single Sign-On), т. е. работа под единой учетной записью во множестве цифровых сервисов.

Определение СЭС является достаточно общим и размытым.

В наиболее общем виде **сетевая экосистема** – это веб-соединение между пользователями, предприятиями и вещами, совместно использующими цифровую платформу.

Современная *сетевая экосистема* в наиболее общем виде приведена на рис. 11.1.

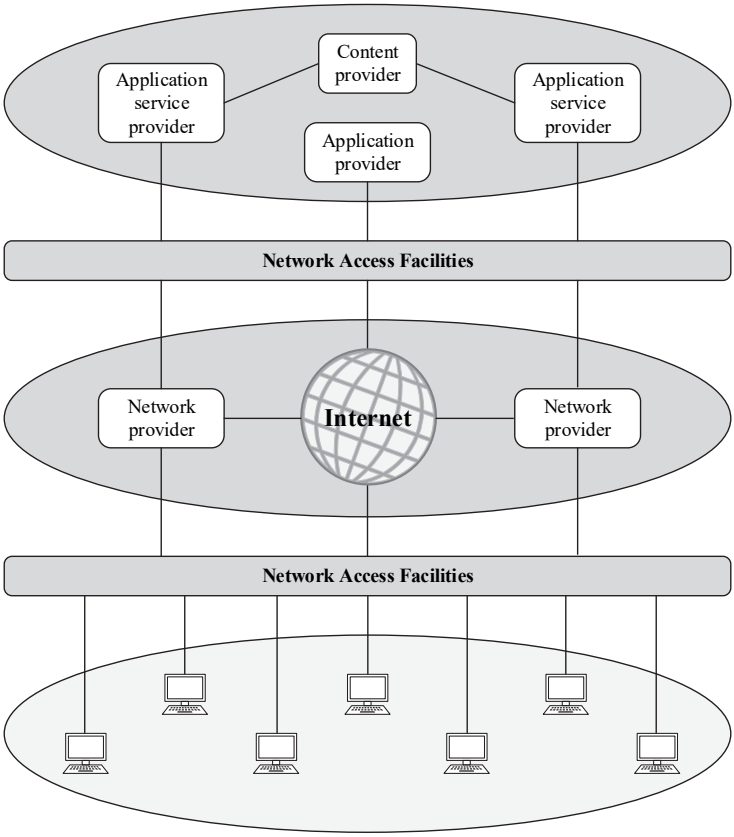


Рис. 11.1. Современная сетевая экосистема

Вся экосистема существует для предоставления услуг конечным пользователям. Термин «конечный пользователь» или «пользователь» используется как очень общий термин, охватывающий пользователей, работающих на предприятии, в общественных местах или дома.

Платформа пользователя (Network Access Facilities, NAF) может быть *стационарной* (ПК или рабочая станция), *портативной* (ноутбук) или *мобильной* (планшет или смартфон).

На рис. 11.1 показаны три категории, представляющие интерес для пользователей.

Поставщики приложений (Application Providers, AP) предоставляют приложения, работающие на платформе пользователя, которая

обычно представлена мобильной платформой. В последнее время концепция магазина приложений стала доступна для стационарных и мобильных платформ.

Отдельной категорией провайдеров является *провайдер прикладных услуг* (Application Service Provider, ASP).

В то время как поставщик приложения загружает программное обеспечение на платформы, поставщик прикладных услуг действует как сервер или хост, прикладное программное обеспечение которого работает на платформах провайдера.

Традиционные примеры такого программного обеспечения включают веб-серверы, электронную почту и серверы баз данных. Самый яркий пример сейчас – поставщик облачных вычислений.

Последний (самый верхний) элемент, показанный на рис. 11.1, – это *поставщик контента* (content provider). Он предоставляет данные, которые будут использоваться на устройстве пользователя (например, электронная почта, музыка, видео). Эти данные могут быть коммерчески предоставленной интеллектуальной собственностью. В некоторых случаях предприятие может быть поставщиком приложений или контента. Примерами поставщиков контента являются звукозаписывающие компании и киностудии.

Еще раз отметим, что *сетевая и коммуникационная инфраструктура* предприятия состоит из следующих компонентов:

– *каналы связи*;
– *локальные и глобальные сети*;
– *подключения к Интернету*, доступные предприятию.

> **!** ***Инфраструктура корпоративной сети*** все чаще включает частные/общедоступные *облачные подключения* к *центрам обработки данных* (ЦОД), в которых размещаются *хранилища больших объемов данных* и *веб-сервисы*.

Ключевым аспектом конвергенции на этом уровне является возможность передачи голоса, изображения и видео по сетям, которые изначально были разработаны для передачи трафика данных. Конвергенция инфраструктуры также произошла в отношении сетей, которые были разработаны для голосового трафика. Например, видео, изображения, текст и данные обычно доставляются пользователям смартфонов по *сетям сотовой связи*.

Здесь стоит выделить два основных элемента современных сетей, в явном виде не изображенных на рис. 11.1:

– *сеть центров обработки данных*: как центры обработки данных крупных предприятий, так и центры обработки данных облачных провайдеров состоят из очень большого количества взаимосвязанных серверов. Обычно до 80% трафика данных находится в сети центра обработки данных, и только 20% полагаются на внешние сети для достижения пользователей;

– *IoT* (Internet of Things – интернет вещей), или *облачная сеть*: интернет вещей, развернутый на предприятии, может состоять из сотен, тысяч и даже миллионов современных устройств. Подавляющая часть трафика данных к этим устройствам и от них передается от машины к машине, а не от пользователя к машине.

На рис. 11.2 показаны некоторые типичные элементы связи и сети, используемые в контексте архитектуры, которая может представлять корпоративную сеть национального или глобального масштаба или часть Интернета с некоторыми из связанных с ней сетями.

В центре рисунка находится магистраль IP, или ядро, – сеть, которая может представлять собой часть Интернета или корпоративную IP-сеть. Обычно магистраль состоит из высокопроизводительных маршрутизаторов, называемых *базовыми маршрутизаторами*, соединенных между собой оптическими каналами с высокой пропускной способностью.

В оптических каналах часто используется так называемое *мультиплексирование с разделением по длине волны* (Wave Length-Division Multiplexing, WDM), так что каждый канал имеет несколько логических каналов, занимающих разные части полосы пропускания в оптическом диапазоне.

На периферии магистрали IP находятся маршрутизаторы, обеспечивающие подключение к внешним сетям и пользователям. Эти маршрутизаторы иногда называют *граничными маршрутизаторами* или *маршрутизаторами агрегации*.

Маршрутизаторы агрегации также используются в корпоративной сети для подключения нескольких маршрутизаторов и коммутаторов к внешним ресурсам, таким как магистраль IP или высокоскоростная глобальная сеть. Анализ показывает, что требования к маршрутизаторам агрегации сейчас находятся в диапазоне от 200 до 400 Гбит/с на оптический канал и от 400 Гбит/с до 1 Тбит/с на оптический канал – для базовых маршрутизаторов.

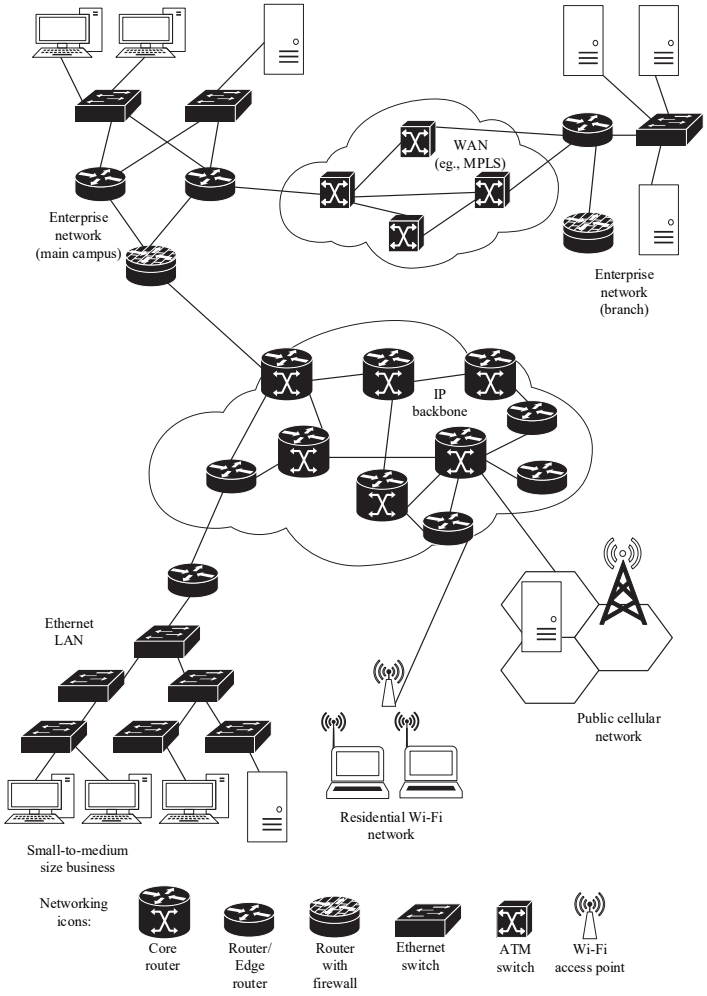


Рис. 11.2. Архитектура современной глобальной сети

Верхняя часть рис. 11.2 изображает часть того, что может быть большой корпоративной сетью. На рисунке показаны два участка сети, подключенные через частную высокоскоростную глобальную сеть с коммутаторами, соединенными оптическими линиями.

Многопротокольная коммутация по меткам (Multiprotocol Label Switching, MPLS) с использованием IP является распространенным

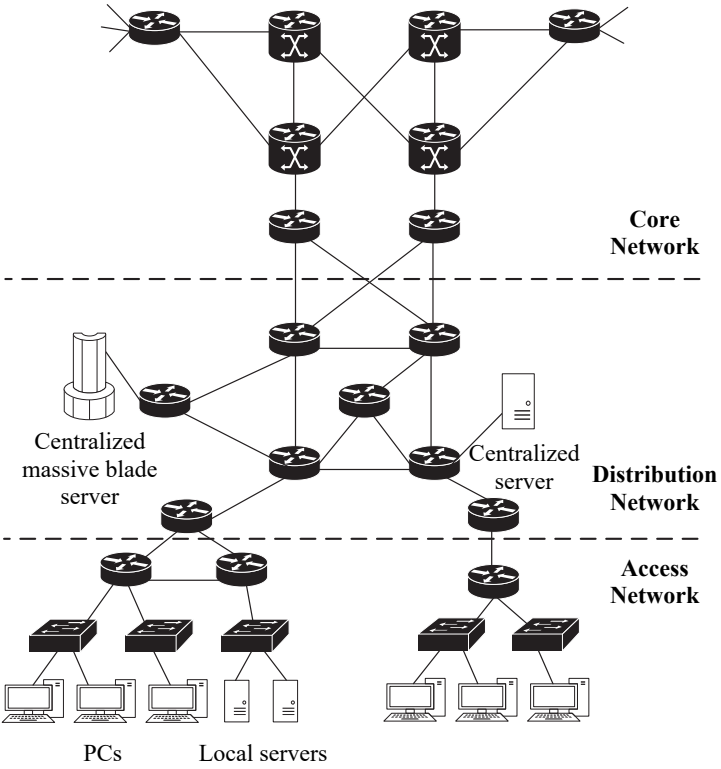
протоколом коммутации для таких глобальных сетей. Корпоративные активы подключаются к магистральной IP-сети или Интернету и защищаются от них через маршрутизаторы с возможностью межсетевого экрана, что не является редкостью для реализации межсетевого экрана. В нижнем левом углу рисунка изображена компоновка для малого или среднего бизнеса, использующего локальную сеть Ethernet. Подключение к Интернету через маршрутизатор может осуществляться через DSL, кабельное соединение или выделенный высокоскоростной канал.

В нижней части рис. 11.2 также показан индивидуальный домашний пользователь, подключенный к *провайдеру интернет-услуг* (Internet Service Provider, ISP) через какое-то абонентское соединение. Распространенными примерами такого подключения являются *цифровые абонентские линии* (Digital Subscriber Line, DSL), обеспечивающие высокоскоростное соединение по телефонным линиям и требующие наличие специального модема DSL и средств кабельного телевидения, для которых необходимы кабельный модем или какое-либо беспроводное соединение. В каждом случае существуют отдельные вопросы, касающиеся кодирования сигналов, контроля ошибок и внутренней структуры абонентской сети. Наконец, мобильные устройства, такие как смартфоны и планшеты, могут подключаться к Интернету через общедоступную сотовую сеть, которая имеет высокоскоростное соединение, обычно оптическое, с Интернетом.

Предприятия часто проектируют свои сетевые объекты в виде ***трехуровневой иерархии*** (рис. 11.3):
− сеть доступа (Access Network);
− сеть распределения, или распределительная сеть (Distribution Network);
− базовая сеть (Core Network).

Ближе всего к конечному пользователю находится *сеть доступа* (Access Network). Как правило, сеть доступа представляет собой локальную сеть (LAN) или сеть в масштабе кампуса, состоящую из коммутаторов LAN (обычно коммутаторов Ethernet) и, в более крупных LAN, IP-маршрутизаторов, обеспечивающих связь между коммутаторами. Коммутаторы этого уровня (не показаны) также обычно используются в локальной сети. Сеть доступа поддерживает оборудование

конечных пользователей, такое как настольные и портативные компьютеры и мобильные устройства. Сеть доступа также поддерживает локальные серверы, которые в основном или исключительно обслуживают пользователей в локальной сети доступа.



Источник: https://mu.ac.in/wp-content/uploads/2021/01/Modern-Networking.pdf.

Рис. 11.3. Иерархия современной типовой сети

Один или несколько *маршрутизаторов доступа* соединяют сети доступа со следующим более высоким уровнем иерархии – *распределительной сетью* (Distribution Network). Это соединение может быть реализовано через Интернет или другое общедоступное, или частное средство связи. Таким образом, эти маршрутизаторы доступа функционируют как *граничные маршрутизаторы*, которые перенаправляют трафик в сеть доступа и из нее. Для большого локального объекта

могут использоваться дополнительные маршрутизаторы доступа, которые обеспечивают внутреннюю маршрутизацию, но не функционируют как граничные маршрутизаторы (не показаны на рис. 11.3).

> ***Распределительная сеть*** соединяет сети доступа друг с другом и с базовой сетью. Граничный маршрутизатор в распределительной сети подключается к граничному маршрутизатору в сети доступа для обеспечения возможности подключения.

Два маршрутизатора сконфигурированы для распознавания друг друга и обычно обмениваются информацией о маршрутизации и подключени, а также некоторой информацией, связанной с трафиком. Это взаимодействие между маршрутизаторами называется **пирингом**.

Распределительная сеть также служит для агрегирования трафика, предназначенного для основного маршрутизатора, который защищает ядро от пиринга с высокой плотностью. Это значит, что использование распределительной сети ограничивает количество маршрутизаторов, которые устанавливают одноранговые отношения с граничными маршрутизаторами в ядре, экономя память, обработку и пропускную способность. Сеть распространения может также напрямую соединять серверы, которые используются в сетях с множественным доступом, такие как *серверы баз данных и серверы управления сетью*.

Опять же, как и в случае с сетями доступа, некоторые из маршрутизаторов распределения могут быть чисто внутренними и не обеспечивать функцию граничного маршрутизатора.

> **Базовая сеть** (Core Network), также называемая *магистральной сетью* (Backbone Network), соединяет географически разнесенные распределительные сети, а также обеспечивает доступ к другим сетям, которые не являются частью корпоративной сети.

Как правило, базовая сеть будет использовать высокопроизводительные маршрутизаторы, линии передачи с большой пропускной способностью и несколько взаимосвязанных маршрутизаторов для увеличения избыточности и пропускной способности. Базовая сеть может также подключаться к высокопроизводительным серверам с

большой емкостью, таким как большие серверы баз данных и объекты частного облака.

Иерархическая сетевая архитектура – пример хорошей модульной конструкции. Благодаря такой конструкции емкость, характеристики и функциональность сетевого оборудования (маршрутизаторов, коммутаторов, серверов управления сетью) могут быть оптимизированы в соответствии с их положением в иерархии и требованиями на данном иерархическом уровне.

11.2. Ethernet, Wi-Fi и сотовые сети 4G/5G

Можно отметить, что такие технологии, как Ethernet, Wi-Fi и сотовые сети 4G/5G, описанные в главе 10, в современном обществе эволюционируют для обеспечения очень высоких скоростей передачи данных, поддерживающих множество мультимедийных приложений, необходимых для предприятий и потребителей, и в то же время предъявляют высокие требования к сетевому коммутационному оборудованию и средствам управления сетью. В данной главе обратимся еще раз к упомянутым технологиям, рассматривая их под углом зрения современной структуры и специфики аппаратной реализации соответствующих сетей.

11.2.1. Особенности современных сетей Ethernet

Ethernet давно используется в домашних условиях для создания локальной сети компьютеров с доступом в Интернет через широкополосный модем/маршрутизатор. С увеличением доступности высокоскоростного и недорогого Wi-Fi на компьютерах, планшетах, смартфонах, модемах/маршрутизаторах и других устройствах зависимость таких сетей от Ethernet снизилась. Тем не менее почти все анализируемые сетевые решения включают использование Ethernet.

Два недавних расширения технологии Ethernet улучшили возможности использования Ethernet в домашних условиях: *связь через ЛЭП* (Power Line Carrier, PLC) и *передача электроэнергии через Ethernet* (Power over Ethernet, PoE).

Модемы Powerline используют преимущества существующих линий электропередач и силовой провод в качестве канала связи для передачи пакетов Ethernet поверх сигнала питания. Это упрощает включение устройств с поддержкой Ethernet по всему дому. PoE действует комплементарно, распределяя мощность по кабелю передачи данных Ethernet. PoE использует существующие кабели Ethernet для распределения питания устройствам в сети, что упрощает разводку таких устройств, как компьютеры и телевизоры. Со всеми этими вариантами Ethernet сохранит прочное присутствие в домашних сетях, дополняя преимущества Wi-Fi.

На рис. 11.4 представлен упрощенный пример архитектуры корпоративной локальной сети. LAN подключается к Интернету/WAN через брандмауэр.

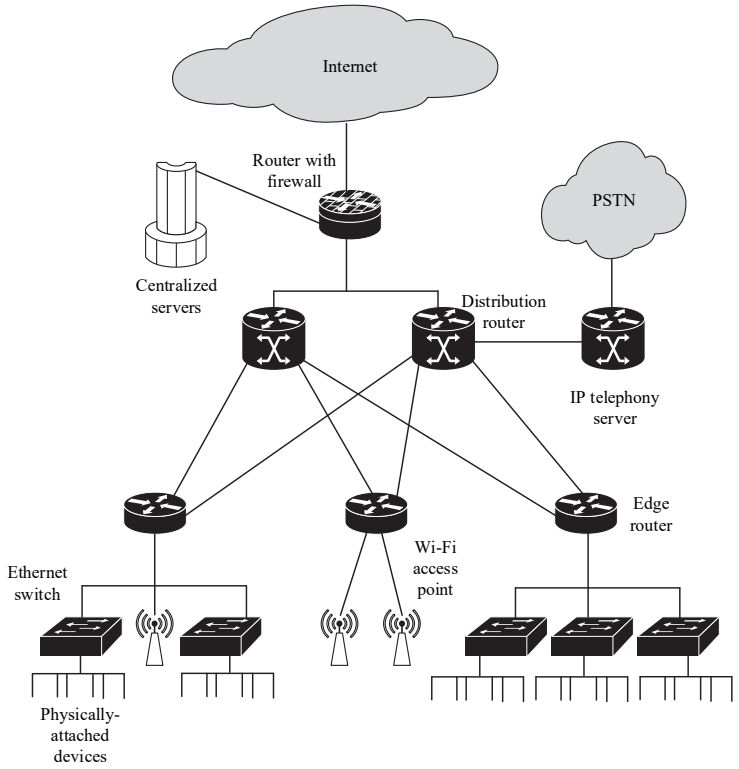


Рис. 11.4. Пример архитектуры корпоративной локальной сети

Иерархическое расположение маршрутизаторов и коммутаторов обеспечивает взаимосвязь серверов, пользовательских и беспроводных устройств. Обычно беспроводные устройства подключаются только на краю или внизу иерархической архитектуры; остальная часть инфраструктуры организации – это сеть Ethernet. Также может использоваться сервер IP-телефонии, обеспечивающий функции управления вызовами (коммутация голоса) для телефонных операций в корпоративной сети с возможностью подключения к *коммутируемой телефонной сети общего пользования* (Public Switched Telephone Network, PTSN).

Предприятие может легко реализовать сеть Ethernet между несколькими зданиями, находящимися на некотором расстоянии друг от друга, используя каналы от 10 Мбит/с до 100 Гбит/с, различные типы кабелей и оборудование Ethernet. Поскольку все оборудование и коммуникационное программное обеспечение соответствуют одному стандарту, то можно легко комбинировать оборудование от разных поставщиков.

В последнее время стали говорить об особенностях применения Ethernet в различных областях, в связи с чем выделяют *стандартный* и *промышленный Ethernet*.

Стандартный Ethernet больше подходит для офисных применений, нежели для использования в промышленности. Он предназначен для повседневного использования, в то время как промышленный Ethernet предусматривает различные уровни и может применяться в более сложных условиях эксплуатации (в том числе в зашумленных производственных помещениях).

Промышленный Ethernet, как следует из названия, используется для подключения промышленного оборудования: когда требуются более надежные разъемы, кабели и, что самое важное, высокий уровень детерминизма. Для достижения высокого уровня детерминизма в промышленном Ethernet, помимо стандартного протокола Ethernet, применяются специализированные протоколы. Наиболее популярными из них являются PROFINET, EtherNet/IP, EtherCAT, SERCOS III и POWERLINK.

Как и в других областях, Ethernet стал доминировать в *центрах обработки данных, ЦОД* (Data Center, DC), где требуется очень высокая скорость передачи данных для обработки огромных объемов данных между *сетевыми серверами* и *устройствами хранения*.

Исторически сложилось так, что ЦОД использовали различные технологии для передачи больших объемов данных на короткие

расстояния, включая Infini Band и Fibre Channel. Но теперь, когда Ethernet может масштабироваться до 100 Гбит/с, а в перспективе – до 400 Гбит/с, аргументы в пользу единого протокола для всего предприятия становятся более весомыми.

Используется также термин *городской Ethernet* или Ethernet городской сети (MAN). Здесь преимущество Ethernet состоит в том, что он легко встраивается в корпоративную сеть, для которой обеспечивает глобальный доступ.

> ***Операторский Ethernet*** (Carrier Ethernet, CE – расширений Ethernet для поставщиков услуг связи, которые используют технологию Ethernet в своих сетях) – одна из самых быстрорастущих технологий Ethernet, которой суждено стать доминирующим средством, с помощью которого предприятия получают доступ к обширным сетям и средствам Интернета.

Но более важным преимуществом является то, что операторский Ethernet обеспечивает гораздо большую гибкость с точки зрения используемой скорости передачи данных по сравнению с традиционными глобальными альтернативами.

Следует отметить две другие особенности нового подхода к Ethernet. Во-первых, для расположенных рядом серверов и устройств хранения необходимую сетевую инфраструктуру обеспечивают высокоскоростные оптоволоконные каналы и коммутаторы Ethernet. Вторая особенность связана с использованием так называемой *объединительной платы Ethernet* (Backplane Ethernet) – печатной платы, которая обеспечивает параллельное соединение нескольких контактов друг с другом, формируя компьютерную шину. Работа Ethernet через *электрические объединительные платы* объединяет подуровни IEEE 802.3 Media Access Control (MAC) и MAC Control с семейством физических уровней, определенных для поддержки работы через объединительную плату модульного шасси.

Объединительная плата Ethernet обеспечивает скорость до 100 Гбит/с на очень коротких расстояниях. Эта технология идеально подходит для ***блейд-серверов*** (Blade Server), представляющих собой модульную электронную плату, содержащую один, два или более микропроцессоров и память, которая предназначена для одного специального приложения и может быть легко вставлена в ***блейд-шасси***,

представляющее собой компактную конструкцию для размещения многих аналогичных серверов.

За выпуск стандартов для локальных сетей Ethernet в рамках комитета по стандартам IEEE 802 LAN отвечает группа 802.3.

11.2.2. Эволюция Wi-Fi

Подобно тому как Ethernet стал доминирующей технологией для проводных локальных сетей, так и Wi-Fi, стандартизированный комитетом IEEE 802.11, доминирует в беспроводных локальных сетях.

Wi-Fi – преобладающая технология беспроводного доступа в Интернет, используемая в домах, офисах, общественных местах.

Wi-Fi в доме теперь соединяет компьютеры, планшеты, смартфоны и множество электронных устройств, таких как видеокамеры, телевизоры и термостаты. Wi-Fi на предприятии стал важным средством повышения производительности труда и эффективности сети. А общественные точки доступа Wi-Fi значительно расширились, чтобы обеспечить бесплатный доступ в Интернет в обязательных общественных местах.

Сегодня важность Wi-Fi в доме значительно возросла. Wi-Fi остается схемой по умолчанию для соединения домашней компьютерной сети. Первое важное применение Wi-Fi в домашних условиях связано с возможностью убрать кабели Ethernet для соединения настольных и портативных компьютеров друг с другом и с Интернетом. Типичная структура домашней сети обычно представляет собой настольный компьютер с подключенным маршрутизатором/модемом, который обеспечивает интерфейс с Интернетом. Другие настольные и портативные компьютеры подключаются к центральному маршрутизатору через Ethernet или Wi-Fi. Таким образом, все домашние гаджеты могут связываться друг с другом и с Интернетом.

Основное качество беспроводных соединений в том, что они значительно упростили подключение. Нет необходимости в физическом использовании кабеля, и портативные средства можно легко перемещать из комнаты в комнату или даже за пределы дома. Wi-Fi значительно упростил подключение.

Поскольку и *Wi-Fi, и сотовая связь теперь являются стандартом как для смартфонов, так и для планшетов*, домашний Wi-Fi обеспечивает достаточно качественный выход в Интернет. Смартфон или планшет будет автоматически использовать соединение Wi-Fi с Интернетом, если оно доступно, и переключаться на более дорогостоящее сотовое соединение, только если соединение Wi-Fi недоступно.

В последние годы все больше и больше объектов предоставляют точку доступа Wi-Fi: кафе, рестораны, общественный транспорт, вокзалы, аэропорты, библиотеки, отели и др. Даже достаточно географически удаленные объекты используют Wi-Fi. Первой компанией, разработавшей такой инфраструктурный продукт, стала компания спутниковой связи *Иридиум*.

Вместе с тем использование Wi-Fi также резко расширилось на предприятиях. Сейчас примерно половина всего корпоративного сетевого трафика проходит через Wi-Fi, а не через традиционный Ethernet. Этому имеются следующие объяснения. Во-первых, все больше и больше сотрудников предпочитают использовать ноутбуки, планшеты и смартфоны для подключения к корпоративной сети, а не настольные компьютеры. Во-вторых, с появлением Gigabit Ethernet (IEEE 802.11ac) расширились возможности поддерживать высокоскоростные подключения ко многим мобильным устройствам одновременно.

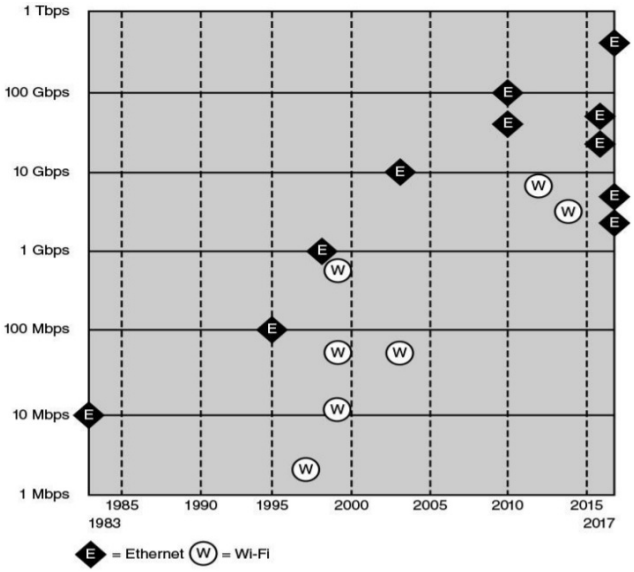
Благодаря гигабитным скоростям передачи данных, доступным в офисной локальной сети, необходим гигабитный Wi-Fi, чтобы мобильные пользователи могли эффективно использовать офисные ресурсы. Упомянутый стандарт IEEE 802.11ac как раз направлен на решение данной задачи.

По мере совершенствования антенного оборудования, методов беспроводной передачи и проектирования беспроводных протоколов комитет IEEE 802.11 смог ввести стандарты для новых версий Wi-Fi на все более высоких скоростях. Характер изменения скоростных возможностей можно проследить на основании нижепредставленной информации (в скобках указаны годы адаптации стандартов), а также на рис. 11.5:

− 802.11 (1997): 2 Мбит/с;

− 802.11a (1999): 54 Мбит/с;

– 802.11b (1999): 11 Мбит/с;
– 802.11g (2003): 54 Мбит/с;
– 802.11n (2009): 600 Мбит/с;
– 802.11ac (2014): 1,3 Гбит/с (максимум 3,5 Гбит/с);
– 802.11ad (2016): 6,76 Гбит/с;
– 802.11ax (2020): до 9,6 Гбит/с.



Источник: https://mu.ac.in/wp-content/uploads/2021/01/Modern-Networking.pdf.

Рис. 11.5. Эволюция Ethernet и Wi-Fi (до 2018 г.)

Стандарт IEEE 802.11ac работает в диапазоне 5 ГГц, как и более старые и более медленные стандарты 802.11a и 802.11n. Он разработан, чтобы обеспечить плавный переход от стандарта 802.11n. Стандарт 802.11ac использует передовые технологии в конструкции антенн и обработке сигналов для достижения гораздо большей скорости передачи данных при меньшем расходе энергии батареи.

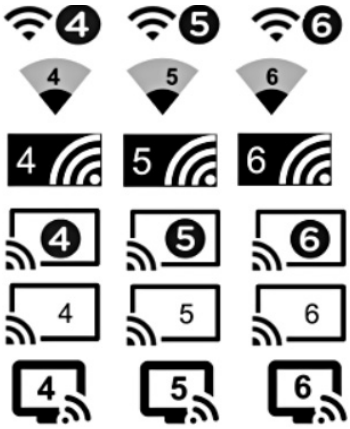
Стандарт IEEE 802.11ax отличается увеличенной скоростью передачи данных – до 9,6 Гбит/с. Кроме того, новый стандарт предусматривает более совершенную систему шифрования WPA3 (Wi-Fi Protected Access III). Технология работает в диапазонах частот 2,4 и 5 ГГц, что обеспечивает большую пропускную способность.

IEEE 802.11ad – это версия 802.11, ориентированная для работы в диапазоне частот 60 ГГц, что дает возможность для использования более широкой полосы пропускания канала, обеспечивая высокие скорости передачи данных с относительно простым кодированием сигнала и использованием относительно простых антенн. Немногие устройства работают в диапазоне 60 ГГц, а это означает, что при обмене данными возникает меньше помех.

Wi-Fi необходим также для реализации последней эволюции Интернета: интернета вещей (об этом – ниже).

! Наборы стандартов 802.11 переименованы на более простые и понятные имена:
– 802.11b → Wi-Fi 1;
– 802.11a → Wi-Fi 2;
– 802.11g → Wi-Fi 3;
– 802.11n → Wi-Fi 4;
– 802.11ac → Wi-Fi 5;
– 802.11ax → Wi-Fi 6.

В связи с этим на сетевых устройствах может появиться маркировка вида, показанного на рис. 11.6.



Источник: https://www.wi-fi.org/news-events/newsroom/wi-fi-alliance-introduces-wi-fi-6.

Рис. 11.6. Варианты маркировки сетевых устройств в соответствии со стандартом IEEE 802.11ax

В сентябре 2019 г. началась программа сертификации Wi-Fi 6 и в сентябре 2020 г. ассоциация стандартов IEEE официально его приняла (ратифицировала). В 2021 г. компания *Keenetic* (см. п. 10.4.5) выпустила первую модель роутера с поддержкой класса Wi-Fi AX.

В целом IEEE 802.11ax базируется на современном, актуальном стандарте 802.11ac (Wi-Fi 5, AC) и использует уже существующие технологии. По словам разработчиков стандарта, некоторые новые технологии будут полезны при развертывании Wi-Fi-сетей с высокой плотностью. Отдельные решения улучшат качество связи в местах с высокой нагрузкой на сеть и в условиях высокой заполненности радиоэфира (например, в общественном транспорте, торговых центрах, отелях, на стадионах или в корпоративных сетях). Ощутимый результат перехода на 802.11ax окажется заметен только в том случае, если все устройства сети будут поддерживать новый стандарт.

Преимущества стандарта Wi-Fi 6 не скажутся кардинально для домашних пользователей. Сеть Wi-Fi-роутера с поддержкой 802.11ax не станет мощнее в сравнении с 802.11ac, зона покрытия сигнала не увеличится, скорости подключения устройств, не имеющих поддержки AX, не возрастут.

Один роутер стандарта Wi-Fi 6 по-прежнему не способен заменить Wi-Fi-систему и без существенных потерь скорости покрыть многокомнатное или разноуровневое помещение с капитальными стенами и перекрытиями.

> **!** В Wi-Fi 6 добавлен режим OFDMA (Orthogonal Frequency Division Multiple Access – *множественный доступ с ортогональным частотным разделением каналов*) для улучшения спектральной эффективности. Технология OFDMA была позаимствована из сотовой индустрии 4G LTE и похожа на многопользовательскую версию OFDM, которая применяется в сети Wi-Fi 5.

OFDMA обеспечивает возможность установления соединений между точкой доступа и несколькими клиентами одновременно за счет деления сигнала на поднесущие частоты (дополнительные более мелкие подканалы) и выделять их в группы для обработки отдельных потоков данных, называемых *ресурсными единицами* (Resource Units, RU). Она позволит одновременно транслировать данные сразу

нескольким клиентам Wi-Fi 6 с усредненной скоростью и использовать один и тот же канал вместо ожидания.

Wi-Fi 6 обеспечивает для технологии MU-MIMO (Multi-User Multiple-Input, Multiple-Output – *многопользовательский многоканальный вход/выход*) поддержку *восходящего направления* (UL MU-MIMO). Ранее в стандарте Wi-Fi 5 технология MU-MIMO работала только в *нисходящем направлении*: от роутера к клиентам (DL MIMO).

В отсутствие поддержки клиентами новых технологий OFDMA и UL MU-MIMO намного больший эффект даст использование нескольких точек доступа.

Если уже есть клиентские устройства с поддержкой 802.11ax или планируется приобретение самого современного роутера с поддержкой последнего стандарта Wi-Fi 6, что называется «на будущее», с учетом темпов развития новых технологий и гаджетов, с перспективой обновления Wi-Fi-устройств сети, выбор интернет-центра Keenetic с поддержкой стандарта AX будет хорошим решением. Но и сейчас во многих случаях практичным и оптимальным вариантом для подключения к Интернету и организации собственной сети Wi-Fi остается использование *двухдиапазонного Wi-Fi-роутера* стандарта 802.11ac.

IEEE планирует к 2024 г. завершить работу над стандартом 802.11be, который, вероятно, получит название Wi-Fi 7. Он станет развитием 802.11ax, и в его основу ляжет уже знакомая технология OFDMA. Это – вариация решения для параллельной передачи данных, разделяющего канал связи на поднесущие с помощью быстрого преобразования Фурье. Ориентировочная пропускная способность стандарта 802.11be составит 40 Гбит/с.

11.2.3. Технологии 4G/5G

Стандартизацию сетей мобильной связи 2–5-го поколений выполняет *партнерский проект для стандартизации систем 3-го поколения* (3rd Generation Partnership Project, 3GPP).

Отметим, что техническое описание мобильной связи 2–5-го поколений было представлено в главе 10, поэтому далее обратим внимание только на наиболее значимые аспекты использования мобильных сетей на основе технологий 4G и 5G.

Развитие смартфонов и сотовых сетей привело к появлению нового поколения возможностей и стандартов, которые в совокупности

называются 4G (точное название: 4G LTE – Long Term Evolution – долговременное развитие).

Системы 4G обеспечивают *сверхширокополосный доступ* в Интернет для различных мобильных устройств, включая ноутбуки, смартфоны и планшеты. Эти требования привели к разработке мобильной беспроводной технологии 4G, которая предназначена для максимального увеличения полосы пропускания и пропускной способности, а также максимальной спектральной эффективности (см. п. 10.1.7 и 10.1.8).

> **!** **Сети 4G** поддерживают мобильный доступ в Интернет и приложения с высокой пропускной способностью, такие как мобильное телевидение высокой четкости, мобильные видеоконференции и игровые сервисы.

В целом можно выделить наиболее значимые для функционирования современных сетей характеристики систем 4G:

– функционируют на базе сети с коммутацией пакетов, полностью основанной на IP;

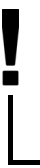
– обеспечивают поддержку пиковых скоростей передачи данных примерно до 100 Мбит/с для доступа с высокой мобильностью (мобильный доступ) и примерно до 1 Гбит/с – для доступа с низкой мобильностью, такого как локальный беспроводной доступ;

– предоставляют возможность динамического разделения и использования сетевых ресурсов для поддержки большого числа одновременных пользователей;

– обеспечивают поддержку высокого качества обслуживания для мультимедийных приложений следующего поколения.

В отличие от предыдущих поколений системы 4G не поддерживают традиционные телефонные услуги с коммутацией каналов, предоставляя только услуги IP-телефонии.

В 2017 г. организация 3GPP официально сообщила, что 5G станет официальным названием следующего поколения мобильной связи и представила новый официальный логотип стандарта связи. Технология телекоммуникаций **5G** – это технологический стандарт пятого поколения для широкополосных сотовых сетей, который сотовые компании начали развертывать во всем мире в 2019 г. Он является планируемым преемником сетей 4G.



Основное *преимущество сетей* ***5G*** заключается в том, что они будут иметь большую пропускную способность, обеспечивая более высокую скорость загрузки, в конечном итоге до 10 Гбит/с.

Основные услуги в сетях 5G:

– *сверхширокополосная мобильная связь* (Extreme Mobile Broadband, eMBB) – реализация ультраширокополосной связи с целью передачи «тяжелого» контента;

– *массовая межмашинная связь* (Massive Machine-Type Communications, mMTC) – поддержка интернета вещей (ультраузкополосная связь);

– *сверхнадежная межмашинная связь с низкими задержками* (Ultra-Reliable Low Latency Communication, URLLC) – обеспечение особого класса услуг с очень низкими задержками.

Основные требования к сетям 5G:

– *пропускная* способность сети до 20 Гбит/с по линии «вниз» (т. е. к абоненту); и до 10 Гбит/с в обратном направлении;

– *поддержка* одновременного подключения до 1 млн устройств на 1 км^2;

– *сокращение временной задержки* на радиоинтерфейсе до 0,5 мс – для сервисов сверхнадежной межмашинной связи URLLC и до 4 мс – для сервисов сверхширокополосной мобильной связи eMBB.

Ожидается, что из-за увеличения пропускной способности новые сети будут не только обслуживать мобильные телефоны, как существующие сотовые сети, но и сами будут использоваться в качестве общих интернет-провайдеров для ноутбуков и настольных компьютеров, конкурируя с существующими интернет-провайдерами (кабельный Интернет), а также будут делать возможными новые приложения в интернете вещей (IoT) и межмашинных областях. Текущие мобильные телефоны 4G не смогут использовать новые сети, для чего потребуются новые беспроводные устройства с поддержкой 5G.

Повышенная скорость достигается за счет использования радиоволн более высокой частоты, чем в существующих сотовых сетях. Однако более высокочастотные радиоволны имеют более короткий диапазон, чем частоты, используемые предыдущими вышками сотовой связи, что требует меньших ячеек. Таким образом, для

обеспечения широкого спектра услуг сети 5G работают в трех диапазонах частот: низком, среднем и высоком. Сеть 5G будет состоять из сетей, содержащих до 3 различных типов ячеек, каждая из которых требует разных антенн, причем каждый тип дает различный компромисс между скоростью загрузки, расстоянием и зоной обслуживания. Мобильные телефоны и беспроводные устройства 5G будут подключаться к сети через самую высокоскоростную антенну в пределах досягаемости в том месте, где они находятся.

Узкополосный 5G использует тот же частотный диапазон, что и современные сотовые телефоны 4G, 600–700 МГц, что дает скорость загрузки немного выше, чем 4G: 30–250 Мбит/с. Вышки сотовой связи с низким диапазоном частот будут иметь радиус действия и зону покрытия, как современные вышки 4G. Среднечастотный 5G использует микроволны 2,5–3,7 ГГц, что в настоящее время обеспечивает скорость 100–900 Мбит/с, при этом каждая вышка сотовой связи обеспечивает обслуживание в радиусе до нескольких километров.

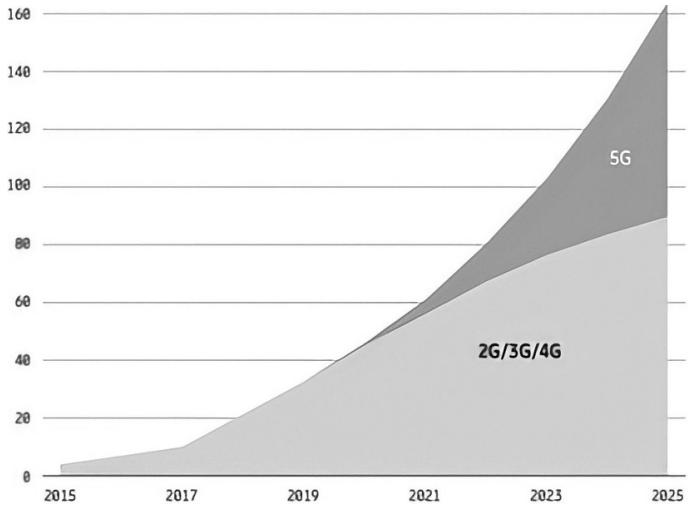
Этот уровень обслуживания является наиболее широко применяемым и доступен в большинстве крупных городов с 2020 г. Некоторые страны не внедряют низкочастотный диапазон, что делает его минимальным уровнем обслуживания.

В высокочастотном диапазоне 5G в настоящее время используются частоты 25–39 ГГц, близкие к нижней границе диапазона миллиметровых волн, хотя в будущем могут использоваться более высокие частоты. Дело в том, что при повышении частоты, на которой передается информация, уменьшается дальность связи. Это закон физики, обойти его можно лишь повышая мощность передатчика, которая ограничена санитарными нормами. Однако считается, что базовые станции сетей пятого поколения будут располагаться плотнее, чем сейчас, что вызвано необходимостью создать гораздо бóльшую емкость сети. Преимуществом диапазонов десятков гигагерц является наличие большого количества свободного спектра.

Есть еще одна особенность. Скорость загрузки гигабита в секунду сравнима с кабельным Интернетом. Однако миллиметровые волны имеют более ограниченный диапазон, требующий множества маленьких ячеек. Здесь имеются проблемы с прохождением некоторых типов стен и окон. Из-за их более высокой стоимости текущие планы заключаются в развертывании этих ячеек только в плотной городской среде и в местах скопления людей, таких как спортивные

стадионы и конференц-центры. Вышеуказанные скорости были достигнуты в реальных тестах в 2020 г. Ожидается, что скорости будут увеличиваться во время развертывания.

К концу 2019 г. общий объем трафика мобильных данных в мире достиг примерно 33 Эбайт (1 эксабайт (Эбайт) = $1 \cdot 10^{18}$ байтов) в месяц и, по прогнозам, к 2025 г. вырастет примерно в 5 раз, достигнув 164 Эбайт в месяц. На рис. 11.7 представлены мобильные данные, которые будут использовать более 6 млрд человек.



Источник: https://mu.ac.in/wp-content/uploads/2021/01/Modern-Networking.pdf.

Рис. 11.7. Глобальный трафик мобильных данных (Эбайт/мес)

11.3. Сетевые технологии и облачные вычисления

Общие концепции облачных вычислений (ОВ) восходят к 1950-м гг. Однако реальные услуги ОВ впервые стали доступны в начале 2000-х гг. С тех пор ОВ распространились на малый и средний бизнес, а в последнее время – и на потребителей. Evernote, облачный сервис для создания заметок и архивирования, запущенный в 2008 г., охватил 100 млн пользователей менее чем за 6 лет. Облако

от Apple iCloud было запущено в 2012 г. В конце 2014 г. Google объявил, что у облачного сервиса Google Drive почти четверть миллиарда активных пользователей. Кратко охарактеризуем ключевые элементы облаков, включая облачные вычисления, облачные сети и облачные хранилища.

Облачная сеть (Cloud-Based Network, CBN) – это корпоративная сеть, которую можно расширить до облака, показанного на рис. 11.8. Облачная сеть позволяет предприятию распространять свою сеть по всему миру. Облако значительно упрощает разработку сетевой системы предприятия. В облаке базовая сеть создается поставщиком облачных услуг. Все, что нужно сделать предприятию, – это подключить свою локальную сеть к сети, построенной в облаке, чтобы сформировать глобальную сетевую систему корпоративного класса.

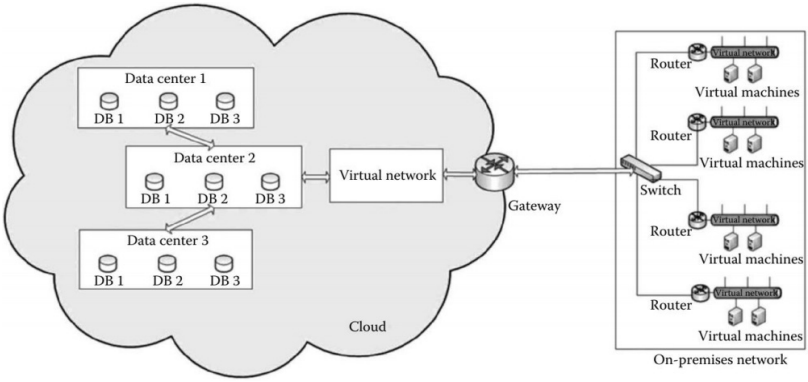


Рис. 11.8. Сеть, основанная на облачной технологии

Облачная сеть нацелена на организации взаимодействия с множеством сайтов по всему миру.

На нескольких объектах, таких как филиалы, школы, клиники, производственные предприятия или магазины розничной торговли, могут работать от пары сотен до десятков тысяч сотрудников. С помощью инструментов управления, развернутых в облаке, сетевые администраторы могут управлять распределенными корпоративными сетями в любом месте и в любое время. Инструменты управления можно использовать для руководства виртуальными машинами и мобильными службами, размещенными в облаке. Они применяются

для выполнения таких задач, как централизованное управление, удаленный мониторинг, удаленная установка программного обеспечения и приложений, удаленная очистка и аудит безопасности.

Существует три основные реализации облачных вычислений (Cloud Computing): технологии использования облачных вычислений в деталях могут отличаться, но общая идея должна относиться к одному из трех решений, реализованных, в частности, в следующих системах:
– Amazon's EC2;
– Google App Engine;
– Berkeley Open Infrastructure for Network Computing (BOINC).

Облачные вычисления являются наиболее гибкими в своем предложении и могут использоваться для самых разных целей; это зависит от приложения, к которому пользователь хочет получить доступ. Для работы в облаке необходимо создать учетную запись. Указанные приложения подходят для любых организаций, но крупные организации могут оказаться в невыгодном положении, поскольку эти приложения не предлагают стандартных возможностей управления, мониторинга и управления, к которым они привыкли.

Облачные приложения отличаются от вычислительных облаков тем, что в них используются программные средства, основанные на облачной инфраструктуре. Облачные приложения и средства представляют собой версии *программного обеспечения как услугу* (SaaS – Software as a Service), *инфраструктуру* (IaaS – Infrastructure as a Service) и *платформу* (PaaS – Platform as a Service).

SaaS включают в себя такие вещи, как веб-приложения, которые доставляются пользователям через браузер или приложение, такое как Microsoft Online Services. Эти приложения переносят хостинг и управление ИТ в облако.

Облачные приложения часто избавляют от необходимости устанавливать и запускать приложение на собственном компьютере клиента, тем самым облегчая задачу обслуживания программного обеспечения.

Облачные приложения могут состоять:
– из *одноранговых сетей* (Peer-to-Peer Computing, P2P; например, Skype);
– из *веб-приложений* (например, MySpace или YouTube);
– из *SaaS* (например, Google Apps);
– из *программных средств и услуг* (например, Microsoft Online Services).

Национальный институт стандартов и технологий США (NIST) определяет основные характеристики облачных вычислений следующим образом:

– *широкий сетевой доступ* (Broad Network Access): возможности доступны по сети и через стандартные механизмы, которые используются разнородными платформами «тонких» или «толстых» клиентов (например, мобильные телефоны, ноутбуки и персональные цифровые помощники) и другие традиционные или облачные программные услуги;

– *быстрая* (*мгновенная*) *эластичность* (Rapid Elasticity): облачные вычисления позволяют расширять и сокращать ресурсы в соответствии с конкретными требованиями к услугам; например, если потребуется много ресурсов сервера на время выполнения определенной задачи, можно освободить эти ресурсы по завершении задачи;

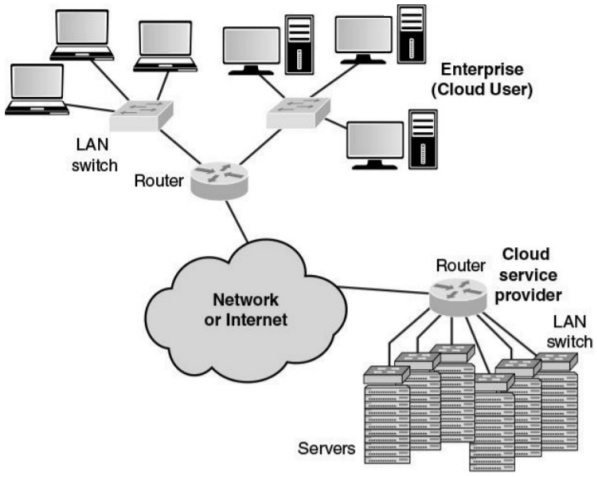
– *измеримость сервиса* (Measured Service): облачные системы автоматически контролируют и оптимизируют использование ресурсов, реализуя возможность измерения или оценки на некотором уровне абстракции, соответствующем типу сервиса (например, хранение, обработка, пропускная способность и активные учетные записи пользователей); использование ресурсов можно отслеживать, контролировать и составлять отчеты, обеспечивая прозрачность как для поставщика, так и для потребителя используемой услуги;

– *самообслуживание по запросу* (On-Demand Self-Service): потребитель может в одностороннем порядке автоматически предоставлять вычислительные возможности, такие как время сервера и сетевое хранилище, без необходимости взаимодействия человека с каждым поставщиком услуг; поскольку услуга предоставляется по запросу, ресурсы не являются постоянными частями ИТ-инфраструктуры;

– *объединение ресурсов* (Resource Pooling): вычислительные ресурсы провайдера объединяются для обслуживания нескольких

потребителей с использованием многопользовательской модели; при этом различные физические и виртуальные ресурсы динамически назначаются и переназначаются в соответствии с потребительским спросом. Существует определенная степень независимости от местоположения, поскольку заказчик обычно не может контролировать или знать точное местоположение предоставленных ресурсов, но может указать местоположение на более высоком уровне абстракции (например, страна, область или центр обработки данных). Примерами ресурсов являются: хранилище, обработка данных, память, пропускная способность сети, виртуальные машины. Даже частные облака имеют тенденцию объединять ресурсы между разными частями одной организации.

На рис. 11.9 показана типичная структура облачной системы. Предприятие поддерживает рабочие станции в корпоративной локальной сети или в наборе локальных сетей, которые подключены маршрутизатором через сеть или Интернет к поставщику облачных услуг. Поставщик облачных услуг поддерживает огромное количество серверов, которыми он управляет с помощью различных инструментов управления сетью, резервирования и безопасности. На этом рисунке облачная инфраструктура показана в виде набора *блейд-серверов* (Blade-Server).



Источник: https://mu.ac.in/wp-content/uploads/2021/01/Modern-Networking.pdf.

Рис. 11.9. Общий состав инфраструктуры системы облачных вычислений

11.4. Интернет вещей

Интернет вещей (Internet of Things, IoT) – одно из последних достижений революции в области вычислений и коммуникаций. Повсеместное распространение IoT и его влияние на повседневную жизнь людей, бизнес и деятельность правительственных структур многих стран едва ли затмевают все предыдущие достижения в области ИТ.

> **!** **Интернет вещей** – это термин, который относится к расширенному взаимодействию интеллектуальных устройств от бытовой техники до крошечных датчиков. Доминирующей темой является встраивание мобильных трансиверов ближнего действия в широкий спектр гаджетов и предметов повседневного обихода, что позволяет создавать новые формы связи между людьми и вещами, а также между самими вещами.

Интернет прошел примерно четыре стадии эволюции, достигнув современной – IoT:

1) *информационные технологии* (ИТ): ПК, серверы, маршрутизаторы, межсетевые экраны и т. д., приобретаемые и используемые как ИКТ-устройства ИТ-специалистами предприятия, в основном – с использованием проводной связи;

2) *операционные технологии* (ОТ): устройства со встроенной ИТ-инфраструктурой для прямого наблюдения и/или управления промышленным оборудованием, активами, процессами и событиями (медицинское оборудование, SCADA – диспетчерский контроль и сбор данных, управление процессами и подобные, приобретаемые для корпоративного применения с использованием проводной связи);

3) *персональные технологии*: смартфоны, планшеты и устройства для чтения электронных книг, приобретаемые в качестве ИТ-устройств потребителями (сотрудниками) исключительно с использованием беспроводной связи, а зачастую – и нескольких форм беспроводной связи);

4) *технология датчиков/исполнительных механизмов*: одноцелевые устройства, приобретаемые потребителями, ИТ-специалистами

и специалистами OT, исключительно использующими беспроводную связь, как часть более крупных систем.

Это четвертое поколение, которое обычно называют IoT, и в нем используется огромное количество встроенных устройств.

Интернет теперь поддерживает соединение миллиардов промышленных и личных объектов, обычно через облачные системы. Объекты доставляют сенсорную информацию, воздействуют на окружающую среду и в некоторых случаях изменяют себя, чтобы создать общее управление более крупной системой, такой как фабрика или город, обеспечивая необходимый уровень защиты данных.

> ***В основе IoT лежат глубоко встроенные устройства.*** Эти устройства представляют собой элементы со сравнительно низкой пропускной способностью, которые обмениваются данными друг с другом и предоставляют данные через пользовательские интерфейсы.

Вместе с тем, встроенные устройства, такие, например, как камеры видеонаблюдения с высоким разрешением, телефоны для передачи видео по IP (VoIP) и некоторые другие требуют обеспечения возможностей потоковой передачи с высокой пропускной способностью.

> ***IoT-системы работают в режиме реального времени*** и обычно состоят из сети smart-устройств и облачной платформы, к которой они подключены с помощью Wi-Fi, Bluetooth или других видов связи.

Интернет вещей обычно разделяют на группы по отраслям: в медицине, телекомуникациях, ЖКХ, армии, электроэнергетике, строительстве, логистике, сельском хозяйстве (IoTAg) и др.

Промышленный интернет вещей (Industrial Internet of Things) – многоуровневая система, включающая в себя датчики и контроллеры, установленные на узлах и агрегатах промышленного объекта, средства передачи собираемых данных и их визуализации, мощные аналитические инструменты интерпретации получаемой информации и многие другие компоненты.

Техническая литература по анализируемому вопросу часто акцентирует внимание на двух элементах IoT: «вещах», которые связаны

между собой, и Интернете, который их связывает. Будем далее рассматривать интернет вещей как систему, состоящую из пяти уровней. Кратко их охарактеризуем.

1. *Датчики и исполнительные механизмы*: датчики «наблюдают» за окружающей средой (в общем случае) и сообщают о количественных измерениях таких переменных, как температура, влажность, наличие или отсутствие некоторых наблюдаемых и т. д. Механизмы (приводы) воздействуют на окружающую среду, например, изменяя настройку термостата или управляя клапаном.

2. *Связь*: устройство может подключаться к сети через беспроводное или проводное соединение для отправки собранных данных в соответствующий центр обработки данных (датчик) или получения рабочих команд от узла контроллера (исполнительного механизма).

3. *Емкость*: сеть, поддерживающая устройства, должна быть способна обрабатывать потенциально огромный поток данных.

4. *Хранение*: необходимо большое хранилище для аккумуляции данных и поддержки резервных копий всех собранных данных. Обычно это реализуется на основе облака.

5. *Анализ данных*: для больших коллекций устройств создаются системы *больших данных*, требующие реализации возможности анализа и обработки потока данных.

В Рекомендации ITU-T/Y.4416 (06/2018) Международного союза электросвязи в области телекоммуникаций, МСЭ-Т (International Telecommunication Union – Telecommunication sector, ITU-T) представлено описание архитектуры IoT на основе развития *сетей последующих поколений* (Next Generation Network, NGNe) с учетом эталонной модели IoT, определенной в Рекомендации МСЭ-Т Y.4000/Y.2060, общих требований к IoT, определенных в Рекомендации МСЭ-Т Y.4100/Y.2066, а также функциональной структуры и возможностей IoT, определенных в Рекомендации МСЭ-Т Y.4401/Y.2068.

В Рекомендации ITU-T/Y.4416 описаны расширения функциональных объектов, контрольных точек и функциональных компонентов NGNe, а также расширения возможностей NGNe согласно описанию, приведенному в Рекомендации МСЭ-Т Y.2012 и других соответствующих рекомендациях, в целях поддержки IoT.

На рис. 11.10 приведена структура эталонной модели IoT согласно ITU-T/Y.2060.

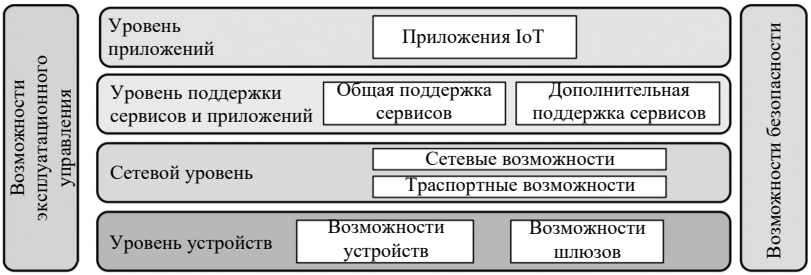


Рис. 11.10. Структура эталонной модели IoT
согласно ITU-T/Y.2060

Для *экосистем IoT*, которые не используют ручной труд непосредственно при исполнении производственных процессов и система управления которых автоматически обращается напрямую к необходимым исполнительным устройствам и сенсорам, базовым ресурсом является информация и автоматические средства ее обработки.

! ***Концепция IoT на основе облачных технологий***, показанная на рис. 11.11, представляет собой объединение устройств в своеобразные пулы ресурсов и виртуализацию функций управления ими.

Рис. 11.11. Концепция IoT на основе облачных технологий

11.5. Большие данные

Объемы накапливаемых данных продолжают расти, поскольку их все больше собирают удаленные датчики, мобильные устройства, камеры, микрофоны, считыватели радиочастотной идентификации (Radio Frequency Identification, RFID) и др. Исследование, проведенное несколько лет назад, показало, что ежедневно создается 2,5 Эбайта ($2{,}5 \cdot 10^{18}$ байтов) данных. Причем 90% всех данных в мире были созданы за последние несколько лет. Сегодня эти цифры, вероятно, выше.

> **!** **«Большие данные»** (здесь начальная буква обычно прописная; Big Data) – *это данные, которые не помещаются в оперативную память компьютера*. Это означает, что свойство больших для определенного объема данных зависит, прежде всего, от структуры и характеристики системы, применяемой для обработки этих данных.

Технологии Big Data могут быть полезными при решении следующих задач:

– прогнозирование рыночной ситуации;
– маркетинг и оптимизация продаж;
– совершенствование продукции;
– принятие управленческих решений;
– повышение производительности труда;
– эффективная логистика;
– мониторинг состояния основных фондов и среды;
– криптовалютные операции и др.

Для работы с Большими данными используются сложные системы, в которых можно выделить несколько компонентов или слоев (Layers). Обычно выделяют четыре уровня компонентов таких систем:

– *прием данных* (Data Ingestion);
– *сбор данных* (Data Staging);
– *анализ данных* (Data Analysis; Analysis Layer);
– *представление результатов* (Consumption Layer).

Это деление является в значительной мере условным так как, с одной стороны, каждый компонент, в свою очередь, может быть разделен на подкомпоненты, а с другой – некоторые функции компонентов

могут перераспределяться в зависимости от решаемой задачи и используемого программного обеспечения, например, выделяют хранение данных в отдельный слой.

Прием данных от источников заключается в их начальной подготовке с целью приведения этих данных к общему формату представления. Этот единый формат выбирается в соответствии с принятой моделью данных. Выполняются преобразования систем измерения, типов (*типизация*), *верификация*. Обработка данных содержательно не затрагивает имеющуюся в данных информацию, но может изменять ее представление (например, приводить координаты к единой системе координат, а значения – к единой размерности).

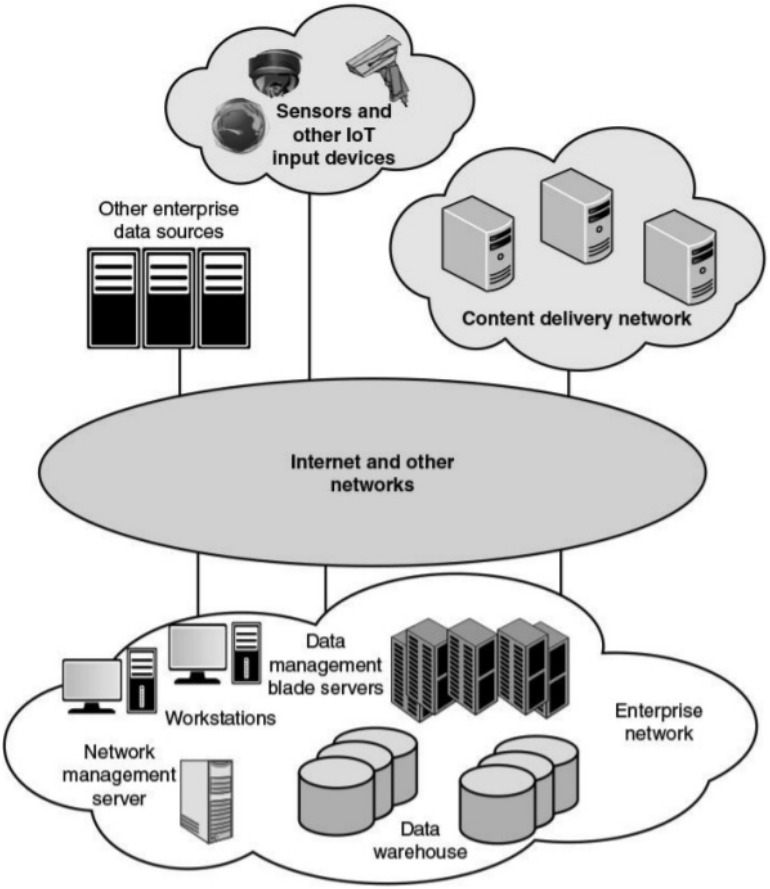
Этап *сбора данных* характеризуется непосредственным взаимодействием с системами их хранения. Устанавливается точка сбора, в которой собранные данные снабжаются локальными метаданными и помещаются в хранилище либо передаются для последующей обработки. Данные, по каким-либо причинам не прошедшие точку сбора, игнорируются.

Анализ данных, в отличии от сбора, использует информацию, содержащуюся в самих данных. Анализ может проводиться как в реальном времени, так и в пакетном режиме. Анализ данных составляет основную по трудоемкости задачу при работе с Большими данными. Методики выполнения анализа используют определенные алгоритмы в зависимости от поставленных целей.

Результаты анализа данных *предоставляются* на уровне потребления. Имеется несколько механизмов, позволяющих использовать результаты анализа Больших данных.

Системы обработки Больших данных являются фреймворками, т. е. каркасами, для использования которых необходимо состыковать их с другими фреймворками, прикладным программным обеспечением пользователя и системой хранения данных.

На рис. 11.12 приведена *экосистема сети Больших данных*. На нем не показаны некоторые важные сетевые устройства, включающие *межсетевые экраны*, *системы обнаружения/предотвращения вторжений* (Intrusion Detection/Prevention Systems, IDS/IPS), коммутаторы LAN и маршрутизаторы.



Источник: https://mu.ac.in/wp-content/uploads/2021/01/Modern-Networking.pdf.

Рис. 11.12. Экосистема сети Больших данных предприятия

Ключевые элементы сети Больших данных предприятия включают:

– *хранилище данных* (Data Warehouse, DW): содержит интегрированные данные из нескольких источников данных, используемые для отчетности и анализа;

– *серверы управления данными* (Data Management Servers, DMS): большие группы серверов выполняют несколько функций в отношении Больших данных: на серверах работают приложения для анализа

данных, такие как инструменты интеграции данных и инструменты аналитики. Другие приложения интегрируют и структурируют данные о деятельности предприятия, такие, например, как финансовые данные, данные о производственной деятельности и электронной коммерции;

– *рабочие станции/системы обработки данных* (Workstations/ Data Processing Systems, DPS) – системы, участвующие в использовании приложений для Больших данных и в создании входных данных для хранилищ Больших данных;

– *сервер управления сетью* (Network Management Server, NMS): один или несколько серверов, отвечающих за управление, контроль и мониторинг сети.

Корпоративная сеть может включать несколько сайтов, распределенных на региональном, национальном или глобальном уровнях. Кроме того, в зависимости от характера системы Больших данных, предприятие может получать данные с других корпоративных серверов, с рассредоточенных датчиков и других устройств в интернете вещей в дополнение к мультимедийному контенту из сетей доставки контента.

Как видно, сетевая среда для больших данных сложна.

Влияние больших данных на сетевую инфраструктуру предприятия определяется так называемыми тремя V:
– *объем* (Volume, растущий объем данных);
– *скорость* (Velocity, увеличение скорости операций записи и чтения данных);
– *изменчивость* (Variability, растущее количество типов данных и источников их происхождения).

Оборудование для обработки данных размещается в *центре обработки данных* ЦОД или *дата-центрах* ДЦ (Data Center, DC).

Телекоммуникационные пространства ЦОД и соответствующие топологии можно найти в стандарте TIA/EIA-942 (Telecommunications Infrustructure Standard for Data Centers). Помимо локальной вычислительной сети (ЛВС) основными компонентами ЦОД являются система управления, вычислительные ресурсы, система хранения данных.

На рис. 11.13 показан пример распределенной топологии ДЦ.

Источник: документ SP–3-0092 (стандарт TIA/EIA-942).

Рис. 11.13. Пример распределенной топологии ДЦ

И крупные корпоративные ЦОД, и ЦОД облачных провайдеров состоят из очень большого количества взаимосвязанных серверов. Обычно до 80% трафика данных приходится на сеть ЦОД, и только 20% – на внешние сети для связи с пользователями.

Основное требование к локальной сети – низкие задержки.

При большом количестве узлов используются сетевые коммутаторы, начинающие передачу пакета данных сразу после обработки заголовка пакета данных. При малом количестве узлов распространено прямое соединение компьютеров по схеме гиперкуб, где размерность куба соответствует числу портов на интерфейсных платах. Ранее массово использовались сети на базе различных вариантов интерфейса Infiniband, сейчас, в основном, используется Gigabit Ethernet.

В контексте рассмотрения вопроса, относящегося к большим данным, отметим также следующее.

Традиционные технологии хранения и управления бизнес-данными включают:
– *системы управления реляционными базами данных* (Relational Database Management Systems, RDBMS);
– *сетевой накопитель* (Network-Attached Storage, NAS);
– *сеть хранения данных* (Storage-Area Networks, SAN);
– *хранилища данных* (Data Warehouses, DW);
– *систему бизнес-аналитики* (*Business Intelligence*, BI).

Традиционные хранилища данных и системы бизнес-аналитики обычно сильно централизованы в корпоративной инфраструктуре. Они часто включают центральное хранилище данных с РСУБД, высокопроизводительное хранилище и аналитическое программное обеспечение, такое как инструменты *онлайн-аналитической обработки* (Online Analytical Processing, OLAP) для интеллектуального анализа и визуализации данных.

Для примера на рис. 11.14–11.17 приведены иллюстрации, показывающие работу с Большими данными в корпорации Oracle*.

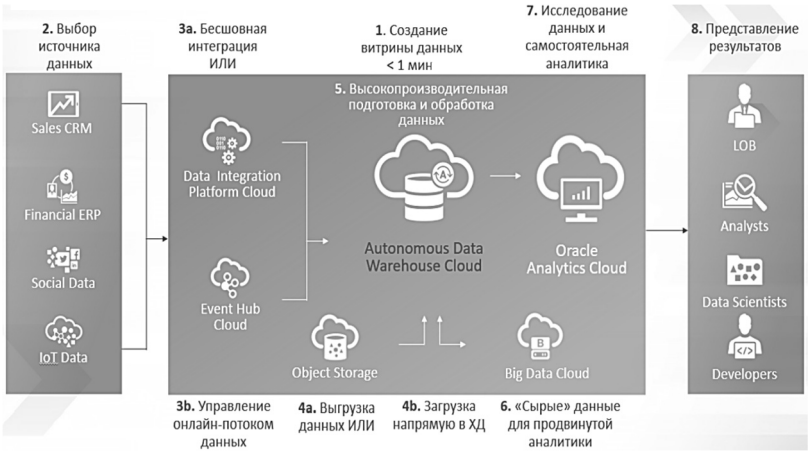


Рис. 11.14. Концептуальная архитектура управления данными корпорации Oracle

* BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня: сб. материалов V Междунар. науч.-практ. конф., Минск, 13–14 марта 2019 г.: в 2 ч. / Белорус. гос. ун-т информатики и радиоэлектроники; редкол.: В. А. Богуш [и др.]. Минск, 2019. Ч. 1. 370 с.

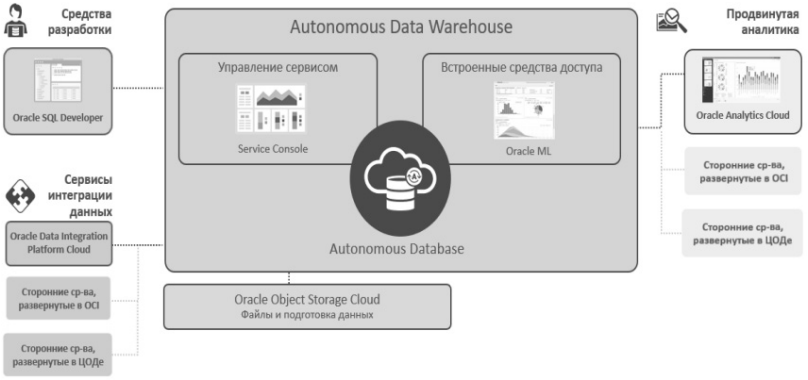


Рис. 11.15. Архитектура современного хранилища данных в корпорации Oracle

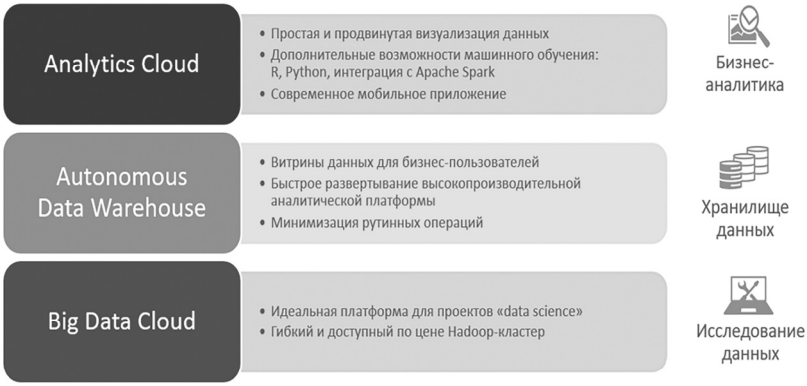


Рис. 11.16. Схема Analytics & Big Data, применяемая в корпорации Oracle

Управление онлайн-потоком данных базируется на *Data Integration Platform Cloud* (DIPC), а также на *Event Hub Cloud Service* (EHCS). *Облачная платформа интеграции данных* (DIPC) – это унифицированная платформа для репликации данных в реальном времени, преобразования данных, слежения за качеством данных и управления данными. *Event Hub Cloud Service* EHCS – это управляемый облачный сервис, который обеспечивает высокодоступную и масштабируемую платформу обмена сообщениями для работы с потоковыми данными.

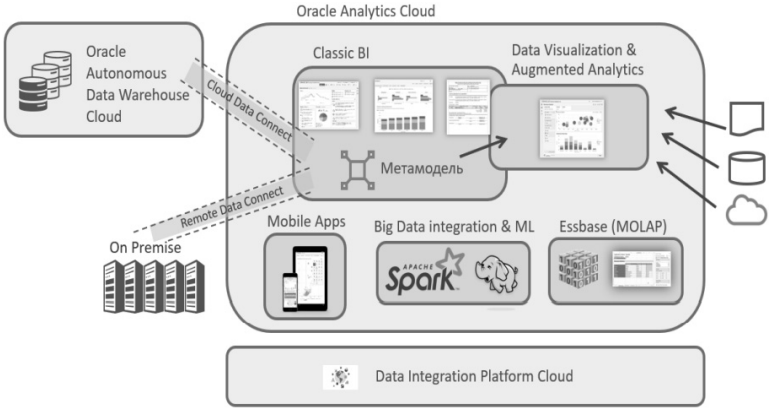


Рис. 11.17. Схема Analytics Cloud

Analytics Cloud характеризуется следующими особенностями: простая и продвинутая визуализация данных; дополнительные возможности машинного обучения: R, Python, интеграция с Apache Spark; современное мобильное приложение.

Приложения для работы с большими данными все чаще становятся источником конкурентной ценности для предприятий. Есть все признаки того, что использование данных будет становиться все более важным направлением, поскольку все больше и больше предприятий будут пользоваться преимуществами приложений для работы с Большими данными.

11.6. Виртуализация сетевых функций

Как уже неоднократно отмечалось выше, классические сети создавались и функционировали на основе использования специализированных аппаратных средств (маршрутизаторов, Ethernet-коммутаторов, оборудования EPC, межсетевых экранов, балансировщиков нагрузки и пр.). Эти устройства создавались на базе специализированных аппаратных и программных платформ отдельных производителей.

Архитектура опорной сети, EPC (Evolved Packet Core, буквально: «эволюционировавшая пакетная опорная сеть») состоит из множества

базовых (Core) функций, взаимодействующих через инфраструктуру IP-сети. Эти функции обеспечивают передачу пакетов данных в сеть радиодоступа, RAN (Radio Access Network) сети 4G LTE (Long Term Evolution). В последнее же время сети телекоммуникационных операторов состоят в основном из «монолитных» сетевых элементов, где функции управления, администрирования и пересылки данных (трафик данных пользователя) выполняются на основе физических, «железных» устройств.

Развертывание новых услуг, модификации оборудования или услуг делаются поочередно на каждом сетевом элементе и требуют тесной координации внутренних и внешних ресурсов оператора. Такая монолитная организация делает операторскую сеть негибкой, затрудняет ввод новых услуг и функций, а также увеличивает зависимость оператора от специфических (*проприетарных*) решений конкретных вендоров. Поэтому в настоящее время многие операторы выбрали путь цифровой трансформации на базе технологий SDN/NFV.

> ***Цифровые трансформации*** обычно определяются термином ***программно-определяемая сеть*** или ***программно-конфигурируемая сеть*** (Software-Defined Networking, SDN), и являются одной из форм *виртуализации сетевых функций* (Network Functions Virtualization, NFV).

Технология SDN обеспечивает абстрагирование топологии сети и моделей данных для вышестоящих систем. Это дает возможность быстрого введения новых приложений, основанных на свойстве про-граммируемости сети.

Еще раз отметим, что до появления SDN функции передачи пакетов данных выполнялись интегрировано на каждом сетевом устройстве (маршрутизатор, мост, коммутатор пакетов и т. д.). Управление в такой традиционной сети осуществлялось посредством сетевого протокола маршрутизации и управления, который был реализован в каждом сетевом узле. Этот подход относительно негибкий и требует, чтобы все сетевые узлы реализовывали одни и те же протоколы.

> В SDN *центральный контроллер* (или *SDN-контроллер*) выполняет все сложные функции, включая маршрутизацию, присвоение имен, объявление политик и контроля безопасности.

Контроллер состоит из нескольких модулей или уровней, каждый модуль отвечает за ряд необходимых функциональных возможностей.

Кроме классического управления сетью прямыми командами системного администратора к контроллеру, SDN-контроллер поддерживает запуск на себе *приложений управления сетью*.

Контроллер SDN выступает единой централизованной точкой управления и взаимодействует с уровнем приложений посредством открытого интерфейса API, а также выполняет мониторинг и управление физическими устройствами сети посредством открытого интерфейса – ***протокола OpenFlow***.

OpenFlow является первым стандартизированным открытым интерфейсом, который отвечает за взаимодействие между уровнем управления и уровнем передачи данных. OpenFlow обеспечивает доступ, обмен информацией и доставку управляющих команд элементам сетевой инфраструктуры.

Каждое *SDN-приложение*, по сути, являет собой интерфейс оптимизации сети под конкретное бизнес приложение и его основная роль – изменение сети в реальном времени под текущие нужды обслуживаемой программы.

В приложении *Microsoft Lynk* это может быть, к примеру, изменение QoS (см. подглаву 11.8) сети между двумя телефонными абонентами для передачи HD-видеозвонка в реальном времени без задержек или создание VPN-туннеля между двумя абонентами.

Вспомним, что приложение *Microsoft Lync*, которое известно также как *MS Office Communicator*, является коммуникационной программой-клиентом, позволяющей пользователям общаться друг с другом в реальном времени с использованием различных видов коммуникаций, таких как мгновенные сообщения, видео- и голосовая связь, общий доступ к рабочему столу, конференции, передача файлов.

SDN-приложения предоставляют конечным пользователям желаемые сервисы. Приложения содержат ряд требований к состоянию и поведению сетевой инфраструктуры.

Технология *виртуальных машин* на уровне Интернета или корпоративной сети используется для реализации функций сервера на уровне приложений, таких как серверы баз данных, облачные серверы, веб-серверы, серверы электронной почты и т. д.

Виртуализация сетевых функций разделяет такие сетевые функции, как *маршрутизация*, *межсетевое экранирование*, *обнаружение вторжений*, *преобразование сетевых адресов*, и реализует их с помощью программного обеспечения.

Здесь используются стандартные технологии виртуализации, которые реализуются на высокопроизводительном оборудовании для виртуализации сетевых функций. Этот подход применим к любой обработке уровня данных или уровня управления как в проводной, так и в беспроводной сетевой инфраструктуре. NFV имеет несколько общих черт с SDN.

Отметим также, что ***NFV и SDN – независимые, но дополняющие друг друга подходы***. SDN разделяет уровни данных и управления при операциях с сетевым трафиком, делая управление и маршрутизацию сетевого трафика более гибкими и эффективными. NFV отделяет сетевые функции в зависимости от конкретных аппаратных платформ посредством виртуализации, чтобы сделать предоставление этих функций более эффективным и гибким.

В конечном итоге поставщик сетевых услуг заботится о наборе сетевых устройств (таких как маршрутизаторы), а также о контроле и управлении функциями, которые они выполняют (например, пересылка пакетов). Если используется NFV, эти сетевые функции реализуются программно и выполняются на виртуальных машинах. Если вместо этого сетевые функции реализуются на выделенных машинах и используется SDN, функции управления реализуются на центральных контроллерах SDN, которые взаимодействуют с сетевыми устройствами. Если же для сети реализованы и SDN, и NFV (NFV/SDN), то выполняются следующие отношения:

– *функционал уровня сетевых данных* – реализован на виртуальных машинах;

– *функционал уровня управления* – может быть реализован на выделенной платформе SDN или на виртуальной машине SDN. В любом случае контроллер SDN взаимодействует с функциями уровня данных, выполняемыми на виртуальных машинах.

> **!** Таким образом, ***как в SDN реализуется отделение уровня управления от уровня передачи и продвижения данных, так и в NFV происходит разделение программного обеспечения, реализующего функции и услуги, – от оборудования***. В NFV это делается при помощи *виртуальных функций сети* (или *виртуальных сетевых функций*), VNF (Virtual Network Functions), представляющих функции соответствующих физических сетевых элементов PNF (Physical Network Functions).

Здесь следует обратить внимание на одно обстоятельство. Иногда используют термины ***виртуальные сетевые функции*** (VNF) и ***виртуализация сетевых функций*** (NFV) в качестве взаимозаменяемых. На самом деле эти два понятия взаимосвязаны, но имеют разные значения. Термин *сетевая функция* обычно относится к какой-либо компоненте сетевой инфраструктуры, которая обеспечивает «точно определенное функциональное поведение» (в соответствии со спецификацией Европейского института телекоммуникационных стандартов, ETSI), такое как обнаружение вторжений, предотвращение вторжений или их маршрутизация. Мы разворачиваем сетевые функции, подобные физическим приборам, где программное обеспечение тесно связано с конкретной, определенной аппаратурой. VNF в тоже время относится к реализации сетевой функции с использованием программного обеспечения, которое отделено от используемого оборудования. Это может привести к большей гибкости сети со значительными показателями экономии капитальных и операционных затрат.

В отличие от этого, NFV обычно относится к общему принципу или концепции работы программно-определенных сетевых функций, независимо от какой-либо конкретной аппаратной платформы, а также официальных инициатив виртуализации сети, возглавляемых некоторыми из крупнейших мировых телекоммуникационных операторов сетей. Во взаимодействии с ETSI эти компании нацелены на создание и стандартизацию всеобъемлющей, комплексной основы *NFV*.

NFV является всеобъемлющей концепцией, в то время как VNF – это строительство блока в текущих NFV фреймворках ETSI.

Подводя итог, отметим: *витруализация* – процесс, эмулирующий компьютер. В случае полной *виртуализации* устройства виртуальной машины полностью эмулируют работу оборудования, вплоть до регистров, памяти и т. д.

При *полной виртуализации* используются немодифицированные экземпляры гостевых операционных систем, а для поддержки работы этих ОС служит общий слой эмуляции их исполнения поверх хостовой ОС.

К достоинствам данного подхода можно причислить:

– относительную простоту реализации;

– универсальность и надежность решения;

– все функции управления берет на себя хост-ОС.

Недостатки:

– высокие дополнительные накладные расходы на используемые аппаратные ресурсы;

– отсутствие учета особенностей гостевых ОС;

– меньшая, чем нужно, гибкость в использовании аппаратных средств.

Исследователи из компьютерной лаборатории Кембриджского университета в проектах Denali и Xen несколько изменили подход в направлении виртуализации: модификация ядра гостевой ОС выполняется таким образом, что в нее включается новый набор API, через который она может напрямую работать с аппаратурой, не конфликтуя с другими виртуальными машинами.

Такой процесс они назвали ***паравиртуализацией*** (Paravirtualization). В этом случае некоторые компоненты, например сетевые или дисковые (и другие) устройства могут быть доступны напрямую через вызовы снаружу виртуальной машины (с некоторыми оговорками). Для работы с паравиртуализированным оборудованием нужны отдельные драйверы и программные вызовы.

Паравиртуализация, таким образом, требует, чтобы гостевая операционная система была изменена для гипервизора, и это является недостатком метода, так как подобное изменение возможно лишь в случае, если гостевые ОС имеют открытые исходные коды, которые можно модифицировать согласно лицензии.

Но зато паравиртуализация предлагает производительность почти как у реальной не виртуализированной системы. Как и при полной виртуализации, одновременно могут поддерживаться многочисленные различные операционные системы.

Метод паравиртуализации позволяет добиться более высокой производительности, чем метод динамической трансляции. Паравиртуализация предоставляет специально установленные обработчики прерываний, чтобы позволить гостю (гостям) и хосту принимать и опознавать эти задачи, которые иначе были бы выполнены в виртуальном домене (где производительность меньше). Таким образом, успешная паравиртуализированная платформа может позволить монитору виртуальных машин (VMM) быть проще (путем перевода выполнения критически важных задач с виртуального домена к его хосту) и/или уменьшить общие потери производительности.

11.7. Система сигнализации ОКС-7

В настоящее время на *сетях связи общего пользования* во многих развитых странах мира в эксплуатации находятся:

– *телефонные сети общего пользования*, ТСОП (Public Switched Telephone Network, PSTN), построенные на базе технологии *цифровых сетей связи с интеграцией служб* ЦСИС (ISDN, Integrated Services Digital Network); их часто называют *сетями ТфОП* (*ISDN)*;

– *сотовые сети* подвижной связи стандарта GSM;

– *интеллектуальные сети* (Intelligent Network, IN) – способ организации сети связи, ориентированный на введение в сеть услуг и управление ими.

Для реализации соответствующих услуг в этих сетях необходимо обеспечить передачу сигнальных и служебных сообщений между узлами сети. Для этих целей используется в составе оборудования *общеканальная система сигнализации № 7* (ОКС-7 или ОКС № 7 – SS7, Signaling System № 7; в Северной Америке ее называют CCSS7 – Common Channel Signaling System 7).

ОКС-7 – набор сигнальных телефонных протоколов, используемых для настройки большинства телефонных станций (PSTN) по всему миру на основе *сетей с канальным разделением по времени*.

В основе ОКС-7 лежит использование аналоговых или цифровых каналов для передачи данных и соответствующей управляющей информации.

На рис. 11.18 приведена схема стека протоколов ОКС-7 сети ТфОП/ISDN в сопоставлении уровней модели OSI и уровней модели ОКС-7.

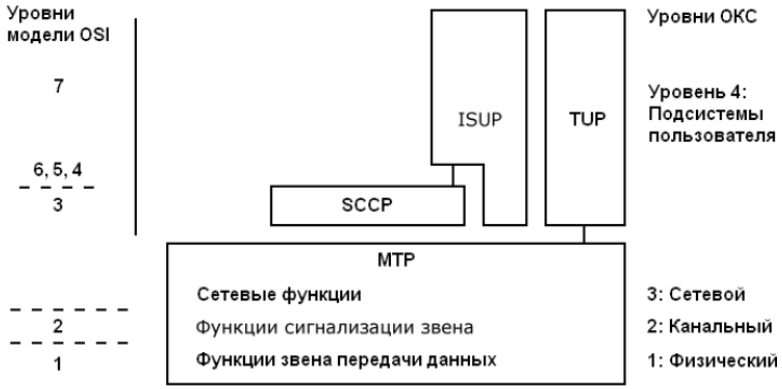


Рис. 11.18. Схема стека протоколов ОКС-7 сети ТфОП/ISDN

Подсистема передачи сообщений (Message Transfer Part, MTP) включает три уровня ОКС-7: физический, канальный и сетевой. Два нижних уровня (физический MTP1 и канальный MTP2) так же, как и в сети передачи данных X.25, полностью совпадает с моделью OSI. Уровень 1 в ОКС-7 называется *звеном передачи данных*, а уровень 2 – *звеном сигнализации*. К подсистеме сетевого уровня ОКС-7 относится не только сетевой уровень MTP3, но и SCCP (Signaling Connection Control Part – подсистема управления соединениями сигнализации). Заметим, что функции уровней 4–6 модели OSI в модели ОКС-7 не требуют реализации. Подсистема пользователей ISDN, которая функционирует по протоколу ISUP (ISDN User Part – прикладная часть ISDN) поддерживает сигнализацию телефонной сети и сети ISDN. Ранее для поддержания функций телефонного соединения использовался специальный протокол TUP (Telephone User Part), но, например, в российской сети его функции выполняются с помощью ISUP.

Как видим, стек протоколов ОКС-7 отталкивается от модели OSI и имеет только четыре уровня. Уровни совпадают с уровнями OSI 1 (физический), 2 (канальный) и 3 (сетевой). Уровень 4 ОКС-7 соответствует уровню 7 OSI. Уровни называются MTP1, MTP2 и MTP3.

Уровень 4 ОКС-7 содержит несколько различных пользовательских уровней, например Telephone User Part (TUP), ISDN User Part (ISUP), Transaction Capabilities Application Part (TCAP) и Signaling Connection and Control Part (SCCP). SCCP – это подуровень из других протоколов уровня 4, и вместе с MTP3 может быть назван Network Service Part (NSP). NSP обеспечивает адресацию и маршрутизацию сообщений без установления соединения (UDT) и сервис управления для других частей 4-го уровня. TUP – это система сигнализации *точка-точка* для соединения звонков. ISUP – это ключевой протокол, который предоставляет *канально-ориентированный протокол* для установки, подключения и завершения соединения при звонке. TCAP используется для создания запросов к базе данных и при расширенной функциональности сети или как связующий протокол с интеллектуальными сетями (INAP), мобильными службами (MAP) и т. д.

11.8. Качество взаимодействия (восприятия)

Понятие и характеристика ***качество обслуживания*** (Quality of Service, QoS) служило одним из главных элементов исследований и анализа в сетях связи более десяти лет. При этом акцент, как правило, делался на техническую составляющую качества обслуживания (оказание высококачественных услуг в реальном времени): например, передача голоса по IP или потоковое видео с целью повышения конкурентоспособности сетей TCP/IP на основе пакетной технологии.

Разъяснение концепции QoS дается в Рекомендации ITU-T E.800 [i.31], где указано, что QoS – это «совокупность характеристик услуги электросвязи, которые влияют на ее способность удовлетворять заявленные и подразумеваемые потребности пользователя услуги» («Totality of characteristics of a telecommunications service that bear on its ability to satisfy stated and implied needs of the user of the service»).

При этом свойства QoS включают:

– *пропускную способность* (Throughput): минимальная или средняя пропускная способность в байт/с или бит/с для данного логического соединения или транспортного потока;

– *задержку* (Delay): средняя или максимальная задержка;

– *джиттер пакета* (Packet Jitter): обычно максимально допустимый джиттер (напомним, что джиттер или дисперсия задежки пакета – термин, определяемый для обозначения колебаний задержки при передаче пакетов по сети);

– *частоту ошибок* (Error Rate): обычно максимальная частота ошибок, выраженная в доле битов, переданных с ошибкой;

– *потерю пакетов* (Packet Loss): доля потерянных пакетов;

– *приоритет* (Priority): сеть может предлагать заданное количество уровней приоритета; присвоенный уровень для различных потоков трафика влияет на способ, которым различные потоки обрабатываются сетью;

– *доступность* (Availability): выражается в процентах от общего времени доступа;

– *безопасность* (Security): могут быть определены различные уровни или типы безопасности.

Для примера: в Рекомендации ITU-T Q.706 приведены следующие показатели QoS подсистемы MTP (высокая степень централизации функций сигнализации является причиной высоких требований к количественным значениям этих показателей):

– время неготовности *пучка маршрутов сигнализации* не должно превышать в сумме 10 мин в год; под ***пучком сигнальных маршрутов*** понимается совокупность всех маршрутов между исходящим пунктом и пунктом назначения;

– общая потеря сигнальных сообщений из-за отказа подсистемы MTP должна быть ниже 10^{-7}, т. е. не более одной ошибки на 10 млн сообщений;

– вероятность передачи сигнального сообщения в неправильной последовательности должна быть ниже 10^{-10};

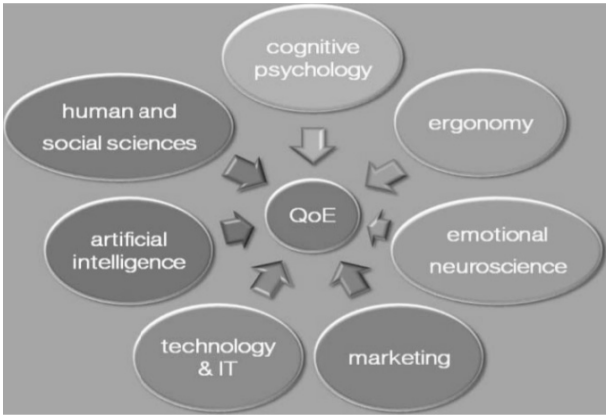
– вероятность приема сигнальной единицы с необнаруженной ошибкой должна быть ниже 10^{-10}, т. е. не более одной ошибки на 10^{10} всех ошибок в сигнальных единицах, не обнаруженной MTP.

В последнее время обычно используется термин ***качество восприятия*** (QoE, Quality of Experience), который перенаправляет внимание на конечного пользователя: на количественную оценку субъективного опыта пользователя, полученного от использования услуги.

QoE стало одним из наиболее важных факторов для пользователей при выборе оператора. Мобильные операторы постоянно анализируют данные из разных источников, что в итоге создает так называемое ***Network View*** (*представление сети*).

Существуют разные определения QoE в документах ITU, ETSI (Европейский институт телекоммуникационных стандартов) и в другой литературе. Следствием этого является то, что, несмотря на растущую исследовательскую деятельность, связанную с опытом конечных пользователей, концепция качества опыта по-прежнему остается неоднозначной, не имеющей последовательной теоретической основы и общепринятого определения. Вместе с тем отметим, что в документе ETSI (ETSI TS 103 294) качество опыта (QoE) определяется как «a measure of user performance based on both objective and subjective psychological measures of using an ICT service or product» («мера производительности пользователя, основанная как на объективных, так и на субъективных психологических показателях использования услуги или продукта ИКТ»).

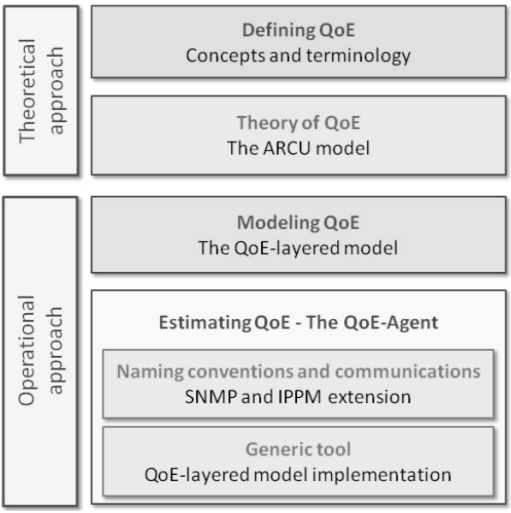
В этом же документе ETSI подчеркивается, что QoE – это в высшей степени междисциплинарная концепция. В той или иной форме концепция опыта пользователя или восприятия качества применима для отдельных областей: гуманитарных и социальных наук, нейро- и когнитивных наук, маркетинга и бизнеса, эргономики и т. д. (рис. 11.19).



Источник: https://www.etsi.org/deliver/etsi_ts/103200_103299/103294/01.01.01_60/ts_103294v010101p.pdf.

Рис. 11.19. Мультидисциплинарная природа понятия и концепции QoE

Концепция модели QoE представлена на рис. 11.20.



Источник: https://www.etsi.org/deliver/etsi_ts/103200_103299/103294/01.01.01_60/ts_103294v010101p.pdf.

Рис. 11.20. Концепция QoE-модели

Модель формально состоит из элементов, разделенных на 2 уровня:

1) теоретический,

2) операционный, или расчетный.

Первый уровень содержит:

– *описание концепции*, положенной в основу модели, и описание используемой терминологии (Defining QoE);

– содержание *абстрактной теоретической модели QoE*: ARCU-модели.

Второй уровень QoE-модели состоит:

– из формального описания и содержания *уровней модели QoE* (the QoE Layered Model);

– спецификации *программного агента*, реализующего многоуровневую модель QoE (the QoE-Agent).

Теоретическая модель QoE: модель ARCU – Application – Resource – Context – User (приложение – ресурс – контекст – пользователь), которая классифицирует факторы влияния по четырем многомерным пространствам. Модель также отображает точки из этих пространств

в многомерное пространство QoE, представляя как качественные, так и количественные *факторы* или *показатели QoE*. Например, в модели ARCU для сетей на основе технологии 4G (4G LTE – Long-Term Evolution) можно идентифицировать и классифицировать *факторы влияния* QoE (Influence Factors) в четырех многомерных пространствах:

1) *пространство ресурсов* (Resource Space) – включает измерения, связанные с техническими свойствами системы и сетевыми ресурсами, используемыми для услуги;

2) *пространство приложения* (Application Space) – включает измерения, связанные с факторами конфигурации сервиса/приложения, например разрешение;

3) *контекстное пространство* (Context Space) – включает измерения, относящиеся к ситуации, в которой используется сервис/приложение, например время использования;

4) *пользовательское пространство* (User Space) – включает факторы, относящиеся к конкретному пользователю.

Уровни модели QoE-Layered и связанные с ними факторы можно охарактеризовать следующим образом:

1) *уровень ресурсов* (Resource Layer) – факторы связаны с характеристиками и производительностью сетевых ресурсов и технической системы, которые используются для предоставления услуги; сетевые ресурсы связаны с сетевым QoS (например, задержкой), а системные ресурсы – с возможностями сервера и возможностями устройств конечных пользователей;

2) *уровень приложения* (Application Layer) – учитываются факторы конфигурации, относящиеся к приложению или услуге, и факторы, относящиеся к контенту (частота кадров и т. д.); на этом уровне проявляется ряд подходов, таких как сотрудничество с поставщиком контента (например, YouTube);

3) *уровень интерфейса* (Interface Layer) – охватывает физическое оборудование пользователя, например тип устройства и интерфейс, с которым пользователь взаимодействует с приложением/сервисом; к этому уровню сетевой провайдер не имеет никакого отношения, поэтому здесь необходимо использовать соответствующие подходы;

4) *уровень контекста* (Context Layer) – включает в себя все факторы, которые связаны с физическим контекстом (географические аспекты), контекстом использования (мобильность/отсутствие мобильности) и экономическим контекстом (стоимость услуг); что касается

физического контекста, есть некоторые факторы, о которых поставщик услуги не может знать (освещение помещения, место нахождения пользователя или шум в помещении); существуют способы получения таких сведений;

5) *человеческий*, или *гуманитарный*, *уровень* (Human Layer) – включает психофизические факторы, связанные с характеристиками восприятия информации пользователем; такие факторы практически невозможно измерить, они связаны с пользователем как с человеком; эти данные могут быть известны сетевому провайдеру только в том случае, если они предоставлены самим пользователем;

6) *пользовательский уровень* (User Layer) – касается точки зрения (взгляда) пользователя на сервис/приложение и включает такие факторы, как уровень знаний, шаблоны, история, социальные характеристики и т. д.

Оценка QoE – агент QoE (Estimating QoE – the QoE Agent). Как описано выше, предлагаемый подход оценки восприятия идет от концептуальной модели для QoE (модели ARCU) к оперативной (многоуровневой модели QoE). Для фактической реализации многоуровневой модели используется архитектура на основе *агентов*, допускающая интеграцию в расширения системы управления. Такой подход обеспечивает гибкий способ развертывания оценщиков QoE в крупномасштабной распределенной среде. Здесь основная цель состоит в том, чтобы позволить приложениям с поддержкой QoE (управление сетью на основе QoE, SLA (Service Level Agreement) – соглашение об уровне обслуживания на основе QoE, мониторинг и т. д.) получать необходимую информацию о любой интересующей услуге. Предлагаемый агент представляет собой прямую реализацию многоуровневой модели QoE.

При выполнении оценочных действий, например при обеспечении качественной мультимедийной трансляции для пользователей, обладающих разными устройствами, необходимо учесть достаточно много факторов. Одним из основных факторов является пропускная способность сети, к которой подключен конкретный пользователь (как видим, здесь есть некоторая общность между QoS и QoE). Необходимо создавать мультимедийные потоки с разными битовыми скоростями.

Говоря о провайдерах широкополосного доступа, мы понимаем, что опыт абонентов строится исключительно на качестве оказываемых услуг. А именно: доступ в Интернет, IP-телефония, IPTV (Internet

Protocol Television – телевидение по протоколу интернета), а также деятельность организации в целом. При этом клиенту совершенно нет дела до того, по чьей вине снизилось качество услуг: из-за действий оператора связи или в результате неисправности домашнего маршрутизатора. На качество восприятия могут повлиять также следующие факторы:

1) *Интернет и сеть*:

– общее время ответа и/или прохождения пакета;

– наличие и величина провалов;

2) *VoIP* (Voice over Internet Protocol или IP-телефония):

– шумы и эхо;

– уровень громкости;

– «заикания»;

3) *IPTV*:

– качество картинки и звука;

– целостность картинки (наличие артефактов);

4) *организация*:

– субъективная оценка обслуживания провайдера, в частности службы поддержки, монтажников связи и пр.;

– пакет каналов в IPTV;

– общее мнение об операторе связи среди населения.

В серии Рекомендаций МСЭ-Т P.1203 определены модули для набора алгоритмов моделей, предназначенные для мониторинга общего качества медиасеанса при потоковой передаче видео на основе протокола управления передачей данных (TCP). В Рекомендации МСЭ-Т P.1203.1 определены модули оценки качества краткосрочного представления видео для МСЭ-Т P.1203 (модуль *Pv*). В состав также входят модули, выполняющие оценку качества изображения и звука за короткий отрезок времени. Посекундные выходные значения этих кратковременных модулей суммируются в оценки аудиовизуального качества и вместе с информацией о задержке, обусловливаемой начальной загрузкой, а также о событиях остановки воспроизведения суммируются далее в окончательное выходное значение модели – *оценку общего качества восприятия*.

Для модуля *Pv* могут использоваться четыре разных режима, определенные Рекомендацией. Эти режимы, называемые режимами от 0 до 3, используют входную информацию различной сложности и объема и представляют четыре алгоритма реализации модели, каждый из которых характеризуется разным уровнем сложности. В состав

модуля *Pv* входят компоненты, отражающие воздействие сжатия видео, увеличения размера отображаемого контента и низкой частоты кадров. Эти четыре разных режима используют ту же общую архитектуру модели и индивидуальные коэффициенты, а также все они имеют те же компоненты для наращивания объема и для частоты кадров.

Пример 11.1. Динамическая адаптивная потоковая передача через HTTP обеспечивает совместное решение для преодоления нестабильных сетевых условий, но ее сложная характеристика создает новые проблемы для объективного измерения качества видео (QoE). Чтобы проверить возможность обобщения и облегчить широкое использование методов измерения QoE в реальных приложениях, создаются специализированные базы данных (известна такая БД под названием Waterloo Streaming QoE Database III или SQoE-III). Такие БД состоят из большого числа потоковых видео (SQoE-III – из 450), созданных из разнообразного исходного контента и различных шаблонов искажения, имеющих различные алгоритмы адаптации и характеристики в репрезентативных сетевых условиях. Все потоковые видео оцениваются некоторым числом субъектов, проводится всесторонняя оценка результативности объективных моделей QoE с точки зрения их эффективности в прогнозировании субъективного QoE.

Пример 11.2. В Рекомендации МСЭ-Т G.107 приведен алгоритм для так называемой *E-модели* в качестве общей модели МСЭ-Т для оценки передачи речи. Эта вычислительная модель может быть полезна проектировщикам систем передачи, чтобы убедиться в том, что пользователи будут удовлетворены сквозным качеством передачи. Основным результатом применения этой модели является скалярная оценка качества передачи. Главной особенностью модели является использование *коэффициентов снижения качества*, отражающих влияние современных устройств обработки сигналов.

Следует подчеркнуть, что первоначально выходом этой модели был «*коэффициент рейтинга*» (Rating Factor) *R*, но этот коэффициент может быть преобразован, чтобы дать оценку мнению пользователя. Такие оценки сделаны только для целей планирования передачи, а не для предсказания мнения (для которого не существует согласованной модели, рекомендованной МСЭ-Т) конкретного потребителя. E-модель может использоваться для многих комбинаций, имеющих важное значение для проектировщиков передачи.

Как суммирующий результат нашего анализа, на рис. 11.21 приведена общая схема взаимодействия компонентов современных сетей.

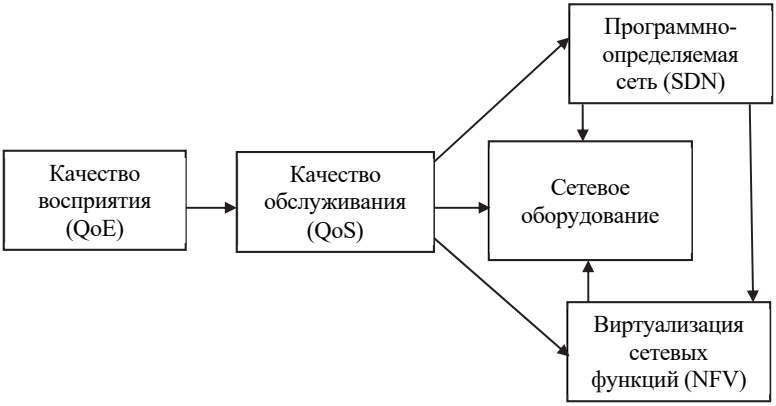


Рис. 11.21. Общая схема взаимодействия компонентов современных сетей

Следует понимать, что ***QoS является объективной оценкой, а QoE – эмпирической***, т. е. основывается на личном опыте. На субъективное мнение клиента может влиять мнение окружающих его людей.

Выводы

1. *Сетевая экосистема* – это веб-соединение между пользователями, предприятиями и вещами, совместно использующими цифровую платформу.

2. *Инфраструктура корпоративной сети* включает частные/общедоступные облачные подключения к центрам обработки данных, в которых размещаются хранилища больших объемов данных и веб-сервисы.

3. Предприятия часто проектируют свои сетевые объекты в виде *трехуровневой иерархии*: сеть доступа (Access Network); сеть распределения, или распределительная сеть (Distribution Network); базовая сеть (Core Network).

4. Подобно тому как Ethernet стал доминирующей технологией для проводных локальных сетей, так и Wi-Fi, стандартизированный комитетом IEEE 802.11, преобладает в беспроводных локальных сетях.

Wi-Fi – это преобладающая технология беспроводного доступа в Интернет, используемая в домах, офисах и общественных местах.

5. *Сети 4G* поддерживают мобильный доступ в Интернет и приложения с высокой пропускной способностью, такие как мобильное телевидение высокой четкости, мобильные видеоконференции и игровые сервисы. Основное преимущество *сетей 5G* заключается в том, что они будут иметь большую пропускную способность, обеспечивая более высокую скорость загрузки, в конечном итоге до 10 Гбит/с.

6. *IoT-системы* работают в режиме реального времени и обычно состоят из сети smart-устройств и облачной платформы, к которой они подключены с помощью Wi-Fi, Bluetooth или других видов связи.

7. *Цифровые трансформации* обычно определяются термином *программно-определяемая сеть* или *программно-конфигурируемая сеть* (Software-Defined Networking, SDN) и являются одной из форм *виртуализации сетевых функций.*

8. Используются два похожих стандарта во взаимоотношениях производителей и потребителей ИТ-услуг: *качество обслуживания* (QoS) и *качество взаимодействия* (QoE). Из-за ключевого различия в структуре используемых при этом сетей (частное и общедоступное подключение) обе эти услуги предлагают свой собственный уникальный стандарт качества. Соглашение о качестве обслуживания (QoS) долгое время было «стандартом качества» в многосайтовых глобальных сетях, соединенных MPLS (Multiprotocol Label Switching), в то время как новый термин – QoE – определяет стандарты качества для соединений программно-определяемых распределенных сетей (SD-WAN).

Контрольные вопросы

1. Назовите пять ключевых технологий, которые преобразовали сети к современному состоянию.
2. Дайте определение сетевой экосистемы.
3. Опишите структуру сетевой экосистемы.
4. Дайте определение термину «облачная сеть».
5. Опишите три основные реализации облачных вычислений.
6. Приведите основные характеристики облачных вычислений.

7. Дайте определение термина «интернет вещей».

8. На пути к достижению IoT Интернет прошел несколько стадии эволюции. Опишите их.

9. Интернет вещей обычно разделяют на группы по отраслям. Приведите и опишите их.

10. Приведите структуру эталонной модели IoT.

11. Опишите концепцию IoT на основе облачных технологий.

12. Дайте понятие Больших данных (Big Data).

13. Опишите области применения (решаемые задачи) при помощи технологии Big Data.

14. Опишите уровни компонентов систем для работы с технологией Big Data.

15. Опишите экосистему сети Больших данных.

16. Что такое виртуализация сетевых функций?

17. Расскажите о системе сигнализации ОКС-7.

18. Что подразумевается под качеством обслуживания (QoS) и качеством взаимодействия (QoE)? Приведите примеры использования.

12

НАДЕЖНОСТЬ И БЕЗОПАСНОСТЬ КОМПЬЮТЕРНЫХ СЕТЕЙ И СИСТЕМ

12.1. Основные понятия и определения из предметной области

Ключевые проблемы функционирования сетей – *контроль ошибок* и *управление потоком данных*.

Контроль ошибок отвечает за то, чтобы данные передавались с необходимой степенью *надежности*. Это означает, что *данные должны доставляться без ошибок*.

Управление потоком данных состоит в согласовании скорости передатчика и приемника. Оба этих аспекта имеют отношение к нескольким уровням модели OSI, о чем речь шла выше (см. главы 5 и 6).

В настоящей главе проанализируем более подробно общие принципы использования избыточных корретирующих кодов (CRC) для контроля ошибок в передаваемых сообщениях.

Сначала рассмотрим некоторые базовые понятия, относящиеся к изучаемой предметной области (в дополнение к приведенным в подглаве 1.2).

Теория защиты информации (Information Security Theory) – система основных идей, относящихся к защите информации, дающая целостное представление о сущности проблемы защиты, закономерностях ее развития и существенных связях с другими отраслями знаний, формирующаяся на основе опыта практического решения задач защиты и определяющая основные ориентиры в направлении совершенствования практики защиты информации.

Следует заметить, что наряду с термином «*защита информации*» применительно к компьютерным сетям широко используется, как правило в близком значении, термин «*компьютерная безопасность*».

Компьютерная безопасность (Computer Security) – одна из основных задач, решаемых любой компьютерной сетью. Проблему безопасности можно рассматривать с разных сторон – злонамеренная порча данных, конфиденциальность информации, несанкционированный доступ, хищения и т. п.

Теория надежности (Reliability Theory; в том числе – теория надежности компьютерных сетей и систем) в большинстве случаев оперирует случайными величинами, поэтому большая часть понятий и определений связана с понятийным аппаратом *теории вероятностей*.

Достоверность (Validity) **работы системы** (устройства) – свойство, характеризующее истинность конечного (выходного) результата работы (выполнения программы), определяемое способностью средств контроля фиксировать правильность или ошибочность работы.

Ошибка (Error) **устройства** – неправильное значение сигнала (бита – в цифровом устройстве) на внешних выходах устройства или отдельного его узла, вызванное технической неисправностью или воздействующими на него помехами (преднамеренными либо непреднамеренными).

Ошибка программы – проявляется в несоответствии промежуточного или конечного значения (результата) реальному (требуемому) вследствие неправильно запрограммированного алгоритма или неправильно составленной программы.

Надежность компьютерной сети – характеристика способности ее аппаратного, программного и программно-аппаратного обеспечения выполнить при определенных условиях требуемые функции в течение определенного периода времени. Повышение надежности основано на принципе предотвращения неисправностей путем снижения интенсивности отказов и сбоев за счет применения электронных схем и компонентов с высокой и сверхвысокой степенью интеграции, снижения уровня помех, облегченных режимов работы схем, обеспечения тепловых режимов их работы, а также за счет совершенствования методов сборки аппаратуры.

Отказ (Failure) – событие, заключающееся в том, что система полностью или частично теряет свойство работоспособности.

Аппаратный отказ – событие, при котором изделие утрачивает работоспособность и для его восстановления требуется проведение ремонта аппаратуры или замена отказавшего изделия на работоспособное.

Отказоустойчивость (Fault Tolerance) – это такое свойство вычислительной системы и компьютерной сети, которое обеспечивает ей возможность продолжения действий, заданных программой, после возникновения неисправностей.

Главной целью повышения надежности систем является обеспечение целостности хранимых, перерабатываемых и передаваемых в них данных.

Секретность (конфиденциальность) информации – свойство информации быть известной только допущенным и прошедшим авторизацию субъектам системы (пользователям, программам, процессам и др.); статус, предоставленный информации и определяющий требуемую степень ее защиты.

Субъект – активный компонент системы, который может инициировать поток информации или изменить состояние системы.

Объект – пассивный компонент системы, хранящий, перерабатывающий, передающий или принимающий информацию (например, страницы, файлы, папки, директории, компьютерные программы, устройства и т. д.).

Доступ – специальный тип взаимодействия между объектом и субъектом, в результате которого создается поток информации от одного к другому.

Санкционированный доступ (СД) к информации – это доступ к информации, не нарушающий установленные правила разграничения доступа, реализуемый на основе следующих процедур:

– ***идентификация*** – процедура распознавания пользователя по его идентификатору (по ID, т. е. по логину), в результате выполнения которого для субъекта идентификации выявляется его идентификатор, однозначно определяющий этого субъекта в информационной системе;

– ***аутентификация*** – процедура проверки идентификации пользователя (например, путем сравнения введенного им *пароля* с паролем, хранящимся в соответствующей базе данных), устройства

или другого компонента в системе (обычно для принятия решения о разрешении доступа к ресурсам системы); частным вариантом аутентификации является установление принадлежности сообщения конкретному автору;

– ***авторизация*** – предоставление субъектам доступа к объектам системы; доступ к объекту означает доступ к содержащейся в нем информации.

Таким образом, при входе пользователя в систему последовательно (идентификация – аутентификация – авторизация) совершаются 3 (может быть и меньше, если какая-либо из операций дает отрицательный результат) указанных процедуры.

Например, пользователь хочет войти в свой аккаунт Google. Google подходит лучше всего, потому что там процедура входа явным образом разбита на несколько простейших этапов. Вот что при этом происходит.

Для начала система запрашивает логин, пользователь его указывает, система распознает его как существующий – это *идентификация*. После этого Google просит ввести пароль, пользователь его вводит, и система соглашается, что пользователь, похоже, действительно настоящий, раз пароль совпал, – это *аутентификация*. Скорее всего, Google дополнительно спросит еще и одноразовый код из SMS или приложения. Если пользователь и его правильно введет, то система окончательно согласится с тем, что он настоящий владелец аккаунта, – это *двухфакторная аутентификация*.

После этого система предоставит пользователю право читать письма в его почтовом ящике – это *авторизация*.

Безопасная (***защищенная*** – protected) **система** – система со средствами защиты, которые успешно и эффективно противостоят **угрозам безопасности** (возможным действиям, которые прямо или косвенно могут нанести ущерб системе).

Скрытый канал (Covert Channel) – любой канал связи, который может быть использован процессом для передачи информации способом, нарушающим политику безопасности системы.

Несанкционированный доступ (НСД, Unauthorized Access) **к информации** – наиболее распространенный вид компьютерных нарушений, который характеризуется пренебрежением установленными

правилами разграничения доступа. НСД использует любую ошибку в системе защиты и возможен при нерациональном выборе средств защиты, их некорректной установке и настройке.

НСД обычно связывают с *атаками*.

Интруз (Intruz) – физическое лицо или процесс, которые реализуют неразрешенный, или несанкционированный, доступ к информации, т. е. ***атаку*** на систему.

Атака (Attack) **на информационную систему** – это совокупность преднамеренных действий злоумышленника, направленных на нарушение одного из трех свойств информации – *конфиденциальности* (Confidentiality), *целостности* (Integrity) или *доступности* (Availability); см. подглаву 1.2.

Удаленная атака (Remote Attack) – информационное разрушающее воздействие на распределенную компьютерную сеть, программно осуществленное по каналам связи.

12.2. Методы обеспечения надежности компьютерных сетей

12.2.1. Численные характеристики надежности

В наиболее общем случае *надежность компьютерной сети* зависит от безотказного и безошибочного функционирования всех составляющих ее структуру аппаратурных блоков и каналов связи, а также сетевого программного обеспечения.

Основной количественной характеристикой надежности является *вероятность безотказной работы*, определяемая как вероятность $P(t)$ нахождения системы в исправном состоянии в течение времени $T \geq t$, где T – случайная величина продолжительности работы системы до отказа; t – детерминированная величина текущего времени или его конкретное значение.

Характеристикой, противоположной $P(t)$, является *вероятность отказа* $Q(t)$ как вероятность того, что устройство или техническая система откажет в течение времени $T < t$ / $Q(t) = 1 – P(t)$.

Наряду с аналитическими методами определения различных параметров надежности широко используются статистические методы,

с помощью которых определяются так называемые *статистические характеристики надежности систем*. Эти характеристики представляют собой результаты обработки экспериментальных данных или данных прямых наблюдений.

Пример 12.1. Если из 1000 компьютеров, объединенных сетью, за 1 год работы становились неисправными 2, то P (t = 1 год) = = 2 / 1000 = 0,002.

Пример 12.2. В течение фиксированного времени (например, t = 1 ч) по каналам связи осуществлялась передача двоичной информации между двумя компьютерами со скоростью S = 10 Кбит/с. За время передачи 1000 символов приняты с ошибками. Определить вероятность того, что произвольный двоичный символ при передаче по тому же каналу будет принят безошибочно или с ошибкой.

Решение. Если принять, что за 1 ч было передано $N_1 = St$ = = 10 000 бит/с · 3600 с = 36 · 10^6 битов, то при числе ошибочных битов N_2 = 1000 искомая вероятность вычисляется как $P() = (N_1 - N_2) / N_1$ = = (36 · 10^6 – 1000) / (36 · 10^6) = 35 999 000 / 36 000 000 = 0,99997.

Соответственно, вероятность появления ошибки (информационной) составляет 3 · 10^{-5}.

Еще одна важнейшая характеристика надежности: *интенсивность отказов* $\lambda(t)$ – условная плотность вероятности возникновения отказа невосстанавливаемого объекта; показывает, какая часть элементов выходит из строя в единицу времени по отношению к среднему числу исправно работающих элементов.

Статистической интерпретацией интенсивности отказов является отношение числа однотипных устройств, вышедших из строя в некотором интервале времени Δt, к числу устройств, поставленных на испытания, продолжающих к моменту времени t оставаться исправными, умноженному на длину интервала. Таким образом, размерностью интенсивности является обычно 1/с.

При расчете вероятности безотказной (или безошибочной) работы электронного устройства или системы обычно используется *экспоненциальная модель* и принимается, что $\lambda(t)$ = const, т. е. $\lambda(t) = \lambda$:

$$P(t) = \exp(-\lambda t). \qquad (12.1)$$

Пример 12.3. Определим вероятность безотказной работы сети и вероятность наступления отказа в ней за 1 год эксплуатации, если $\lambda = 3{,}17 \cdot 10^{-12}$ $с^{-1}$. Интенсивность отказов дана в секундах. Один

год – это 31 536 000 с (за 365 дней). Тогда в соответствии с формулой (12.1) имеем:

$$P(t) = \exp(-\lambda t) = \exp(3{,}17 \cdot 10^{-12} \cdot 3\ 153\ 600) = 0{,}99999$$

(говорят: пять девяток после запятой); тогда $Q(t) = 1 - P(t) = 0{,}00001$.

Для полноты приведем еще одну численную характеристику надежности – *время наработки на отказ*: $t_н / t_н = 1/\lambda$.

В англоязычной литературе этот параметр обозначается MTBF (Mean Time Between Failure – *среднее время между отказами*); единица размерности – час.

12.2.2. Основные методы повышения надежности ИВС

В некоторых каналах (оптоволокно, например) вероятность ошибки очень низкая, поэтому потеря данных происходит исключительно редко. Но количество ошибок, например, в беспроводных сетях или старых местных сетях в десятки раз больше. Повышение надежности таких каналов – необходимое условие качественного функционирования всей сети или системы.

Естественное желание повысить функциональную надежность информационной системы или сети напрямую связано с увеличением объема как программного обеспечения, так и аппаратного. Однако известно, что всякое усложнение уже само по себе снижает надежность. Следовательно, усложняя систему для повышения ее надежности, разработчики понимают, что какая-то часть этой надежности будет потеряна.

Специфика ИВС такова, что невыполнение какой-либо прикладной задачи не приводит к отказу системы в целом. Однако это классифицируется как *частичный отказ*, что означает снижение надежности системы в допустимых пределах. Такие частичные отказы могут возникать, например, из-за воздействия внешних факторов, которые были охарактеризованы выше.

В компьютерных сетях обеспечение надежности – это общая цель, для достижения которой должны слаженно работать все уровни модели OSI. На самом деле, во многих сетях эти функции являются прерогативой верхних уровней и не только относятся к канальному уровню, одной из функций которого является обработка ошибок передачи данных, но и решаются на транспортном уровне.

Существующие методы повышения надежности ИВС и каналов связи основаны на использовании *избыточности* (Redundancy).

Избыточность – это наличие в техническом объекте возможностей сверх тех, которые минимально необходимы для обеспечения его нормального функционирования.

Назначение механизмов и средств избыточности – оперативный контроль за функциональным состоянием устройства или системы, блокирование или нейтрализация последствий появления отказов.

С диалектической точки зрения избыточность является одним из условий перехода количества в качество.

Большинство микрочипов, интеллектуальных устройств, начиная от смартфонов и заканчивая большими компьютерами, мозг человека и многих животных заведомо обладают избыточной памятью и избыточными «вычислительными мощностями».

С позиции обеспечения надежности информационных систем и сетей различают следующие основные виды избыточности:
– *структурная*;
– *временна́я*;
– *информационная*.

Иногда также рассматривают дополнительные виды избыточности: *функциональную*, *алгоритмическую* и *программную*.

Структурная избыточность (Structure Redundancy или Hardware Redundancy) – наличие у объекта избыточных элементов, узлов, устройств, с помощью которых можно своевременно заменить отказавшие основные узлы или устройства, для предотвращения выхода из строя всего изделия, системы.

Избыточные узлы или устройства могут работать параллельно с основными или могут быть подключены и использоваться в режиме ожидания или применяться с помощью иного способа.

Структурная избыточность может реализовываться на основе методов *резервирования* (параллельной работы нескольких однотипных устройств или каналов). При этом могут использоваться средства

принятия решения «на основе голосования» – мажоритарные элементы. Использование источников бесперебойного питания – пример структурной избыточности.

Временна́я избыточность (Time Redundancy) заключается в использовании некоторой части производительности компьютера или сети для контроля за исполнением программ и восстановления (рестарта) информационного процесса.

Примером информационной системы с временной избыточностью являются системы с *повтором передачи сообщений*. К этому типу избыточных методов можно отнести *автоматический запрос повторной передачи* (Automatic Repeat reQuest, ARQ), используемый в протоколах семейства TCP и других протоколах (беспроводной связи).

Остановимся в контексте рассматриваемого вида избыточности на особенностях передачи данных в модели OSI. Канальный уровень может предоставлять различные сервисы. Их набор может быть разным в разных протоколах. Обычно возможны следующие варианты:

1) *сервис без подтверждений, без установки соединения* (Unacknowledged Connectionless Service);

2) *сервис с подтверждениями, без установки соединения* (Acknowledged Connectionless Service);

3) *сервис с подтверждениями, ориентированный на соединение* (Acknowledged Connection-Oriented Service).

Сервис без подтверждений и без установки соединения заключается в том, что передающая машина посылает независимые кадры принимающей машине, и принимающая машина не посылает подтверждений о приеме кадров. Пример предоставления сервиса такого класса – Ethernet. Никакие соединения заранее не устанавливаются и не разрываются после передачи кадров. Если какой-либо кадр теряется из-за шума в канале, то на канальном уровне не предпринимается никаких попыток восстановить его. Данный класс сервисов приемлем при очень низком уровне ошибок. В этом случае вопросы, связанные с восстановлением потерянных при передаче данных, могут быть оставлены верхним уровнем. Он также применяется в линиях связи реального времени, таких как передача речи, в которых лучше получить искаженные данные, чем получить их с большой задержкой.

Шагом в сторону повышения надежности является сервис с подтверждениями, без установки соединения. При его использовании

соединение также не устанавливается, но получение каждого кадра подтверждается. Таким образом, отправитель знает, дошел ли кадр до пункта назначения в целости или потерялся. Если в течение установленного интервала времени подтверждения не поступает, кадр посылается снова. Такой сервис полезен в случае использования каналов с большой вероятностью ошибок, например, в беспроводных системах. Среди сервисов такого класса можно назвать, например, 802.11 (Wi-Fi).

Ориентированный на соединение сервис предоставляет процессам сетевого уровня эквивалент надежного потока битов. Он подходит для длинных ненадежных связей, таких как спутниковый канал или междугородное телефонное соединение.

В общем случае значение временной избыточности зависит от требований к надежности функционирования системы. Этот параметр составляет, например, от 5–10% производительности компьютера в системе до трех- и четырехкратного дублирования производительности в мажоритарных вычислительных комплексах.

Информационная избыточность (Information Redundancy) основана на специальных методах дублирования некоторой части информации, позволяющих восстанавливать исходные данные в случае нарушений в работе системы.

Этот вид избыточности используется для обеспечения соответствия между входными, промежуточными и выходными данными, которые в наибольшей степени влияют на нормальное функционирование системы и требуют определенного времени восстановления.

Информационная избыточность способствует обнаружению искажения данных (ошибок) и устранению ошибок на определенном уровне контроля функционирования системы.

Информационная избыточность обычно реализуется на основе алгоритмов *обнаружения и коррекции ошибок* (Error Detection and Correction), в основе которых лежит *помехоустойчивое кодирование данных* или методы ECC (Error-Correcting Code).

Один из стандартов, введенных IBM, предусматривает хранение информации в банках памяти фрагментами по 9 битов, причем 8 из них (составляющих один байт) предназначены собственно для данных, а девятый является *битом четности* (Parity Bit). Использование

девятого бита позволяет схемам управления памятью на аппаратном уровне контролировать целостность каждого байта данных. Если обнаруживается ошибка, работа компьютера останавливается и на экран выводится сообщение о неисправности.

Метод простого контроля четности является простейшей реализацией ECC. Этот метод не позволяет исправлять ошибки, однако дает возможность их обнаружить пользователю компьютера.

Метод простой четности (или *простой нечетности*) требует присоединения к сообщению лишь одного избыточного бита, вне зависимости от длины сообщения.

Значение этого бита равно свертке по модулю 2 символов передаваемого сообщения, т. е. число единичных символов в закодированном (избыточном) сообщении должно быть четным – для метода простой четности; для метода простой нечетности – нечетным: к результату свертки добавляется 1.

Отметим, что рассмотренный метод простой четности используется для контроля целостности ключа в *симметричном криптографическом алгоритме DES.*

Практически все существующие системы и сети, предназначенные для работы с данными, содержат встроенные средства на основе ECC. Эти алгоритмы реализованы в модулях оперативной памяти (RAM) компьютера, в контроллерах шин на материнской плате компьютера, обеспечивающих в том числе кодирование/декодирование информации при ее передаче от процессора к памяти и обратно; ECC является частью протокола TCP. Можно привести множество подобных примеров.

12.3. Методы помехоустойчивого кодирования информации

Как известно, методы помехоустойчивого кодирования относятся к классу *избыточных*. Определение рассматриваемых методов как «помехоустойчивых» означает, прежде всего, что они являются

противодействием помехам, влияющим на систему и приводящим к ошибкам в данных.

Суть этих методов состоит в преобразовании исходного *информационного сообщения* X_k (k – длина сообщения), называемого также *информационным словом*. К слову X_k дополнительно присоединяют (наиболее часто по принципу конкатенации) избыточные символы длиной r битов, составляющие *избыточное слово* X_r. Таким образом, формируют *кодовое слово* X_n длиной $n = k + r$ двоичных символов: $X_n = X_k X_r$. Информацию содержит только информационное слово. Назначение слова X_r – обнаружение и исправление ошибок на стороне получателя сообщения (ПС).

В зависимости от принципа вычисления дополнительных символов и их значения реализуются различные алгоритмы помехоустойчивого кодирования. Общим является то, что избыточное слово X_r генерируется на передающей стороне (источником сообщения, ИС) и используется принимающей для обнаружения и исправления ошибок.

С учетом избыточных блоков обобщенная структурная схема системы передачи информации примет вид, показанный на рис. 12.1.

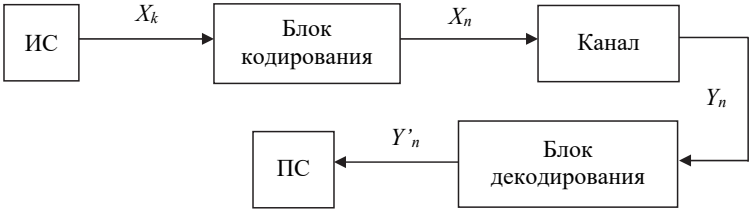


Рис. 12.1. Обобщенная структурная схема системы передачи информации с помехоустойчивым кодированием

Для дальнейшего рассмотрения необходимо упомянуть о некоторых базовых понятиях теории помехоустойчивого кодирования.

Вес по Хеммингу произвольного двоичного слова $X(w(X))$ равен количеству ненулевых символов в слове.

Пример 12.4. Пусть $X = 1101011$. Тогда $w(X = 1101011) = 5$.

Расстояние по Хеммингу, или кодовое расстояние (d), между двумя произвольными двоичными словами (X, Y) одинаковой длины равно количеству позиций, в которых X и Y отличаются между собой.

Пример 12.5. $X = 101$, $Y = 110$. Очевидно, что $d(X, Y) = 2$.

Кодовое расстояние можно вычислить как вес от суммы по модулю 2 этих двух слов: $d(X, Y) = w(X \oplus Y)$. Обратим внимание: знак «⊕» означает побитное сложение по модулю 2.

Пример 12.6. $X = 1011$, $Y = 0000$; $d(X, Y) = 3$:

$$\begin{array}{r} 1011 \\ 0000 \\ \hline w(1011) = 3 \end{array}.$$

Пример 12.7. $X = 11111$, $Y = 11111$; $d(X, Y) = 0$.

Длина слова и *расстояние Хемминга – основополагающие понятия* в теории помехоустойчивого кодирования информации.

Все многообразие существующих кодов для обнаружения и исправления ошибок можно разделить на два больших класса: линейные и нелинейные коды.

Линейные коды базируются на использовании линейных операций над данными (как правило, умножение и сложение по модулю 2 соответствующих символов), **нелинейные коды**, следовательно, основаны на нелинейных операциях.

12.4. Линейные блочные коды

12.4.1. Теоретические основы линейных блочных кодов

Линейные блочные коды – это класс кодов с контролем четности, которые можно описать парой чисел (n, k).

Первое из чисел определяет длину кодового слова X_n, второе – длину информационного слова X_k. Отношение числа битов данных к общему числу битов данных k/n именуется *степенью кодирования* (Code Rate) – доля кода, приходящаяся на полезную информацию.

Для формирования проверочных символов (кодирования) используется **порождающая матрица** – совокупность базисных векторов, которая будет далее записываться в виде матрицы G размерностью $k \times n$ с единичной подматрицей I в первых k строках и столбцах:

$$G = \left[P \mid I \right]. \qquad (12.2)$$

Матрица *G* *называется порождающей матрицей линейного корректирующего кода в приведенно-ступенчатой форме.*

Кодовые слова являются линейными комбинациями строк матрицы G (кроме слова, состоящего из нулевых символов). Кодирование, результатом которого является кодовое слово X_n, заключается в умножении вектора сообщения длиной k (X_k) на порождающую матрицу по правилам матричного умножения (все операции выполняются по модулю 2). Очевидно, что при этом первые k символов кодового слова равны соответствующим символам сообщения, а последние r символов (X_r) образуются как линейные комбинации первых.

Для всякой порождающей матрицы G существует матрица H размерности $r \times n$, задающая базис нулевого пространства кода и удовлетворяющая равенству

$$G \cdot H^T = 0. \tag{12.3}$$

Матрица *H*, называемая *проверочной* (H^T – транспонированная проверочная матрица), может быть представлена так:

$$H = \left[-P^T \mid I \right]. \tag{12.4}$$

В последнем выражении I – единичная матрица порядка r, P^T – транспонированная матрица P.

Кодовое слово X_n может быть получено на основе следующего тождества:

$$H \cdot (X_n)^T = 0, \tag{12.5}$$

где $X_n = x_1, x_2, \ldots, x_n = x_1, x_2, \ldots, x_k, x_{k+1}, \ldots, x_{k+r}$.

Результат умножения сообщения (Y_n) на *транспонированную* проверочную матрицу (H) называется **синдромом** S:

$$S = (Y_n)^T \cdot H, \tag{12.6}$$

где $Y_n = y_1, y_2, \ldots, y_n = y_1, y_2, \ldots, y_k, y_{k+1}, \ldots, y_{k+r}$. Слово Y_n обычно представляют в следующем виде:

$$Y_n = X_n \oplus E, \tag{12.7}$$

где $E = e_1, e_2, \ldots, e_n$; n – разрядный *вектор ошибки*, $\oplus$ – сложение по модулю 2.

Если все r символов синдрома нулевые ($S = 0$), то принимается решение об отсутствии ошибок в принятом сообщении Y_n, в противном случае – об их наличии.

Код, характеризующийся *минимальным кодовым расстоянием* $d_{\min}$ между двумя произвольными кодовыми словами, позволяет обнаружить $t_о$ ошибок, где $t_о = \frac{d}{2}$, если d – четное, и $t_о = \frac{d-1}{2}$, если d – нечетное. Количество исправляемых кодом ошибок ($t_и$) определяется так:

$$t_и = \begin{cases} \frac{d-1}{2}, & d - \text{нечетное}, \\ \frac{d-2}{2}, & d - \text{четное}. \end{cases} \qquad (12.8)$$

12.4.2. Избыточный код простой четности

Простейший избыточный код основан на контроле четности (либо нечетности) единичных символов в сообщении X_n. Количество избыточных символов r всегда равно 1 и не зависит от k. Значение этого символа будет нулевым, если сумма всех символов кодового слова по модулю 2 равна нулю.

Назначение X_r в алгоритме – обнаружение ошибки. Код простой четности позволяет обнаружить все нечетные ошибки (нечетное число ошибок), но не позволяет их исправить. Легко убедиться, что данный код характеризуется минимальным кодовым расстоянием, равным 2.

Пример 12.8. Пусть X_k = 10101, тогда

$$X_r = \sum_i^n X_i = 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 1.$$

Проверочная матрица для данного случая будет состоять из одной строки и шести столбцов и примет следующий вид:

$$H = 11111\ 1.$$

Кодовое слово X_n, вычисленное в соответствии с формулой (12.5), будет равно 10101 1. Как видим, $w(X_n)$ имеет четное значение.

Слово X_n будет передаваться от источника сообщения к приемнику сообщения (например, от одного компьютера к другому). Пусть на приемной стороне имеем Y_n = 1<u>1</u>101 1: Y_k = 1<u>1</u>101 и $Y_r = Y_{k+1} = 1$ (ошибочный символ подчеркнут).

Для вычисления синдрома ошибки в соответствии с (12.6) достаточно выполнить следующие простые действия:

а) вычисляется дополнительное слово (в данном случае – символ) Y'_r, которое является сверткой по модулю 2 слова Y_k:

$$Y'_r = 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 0;$$

б) вычисляется синдром $S = Y_r \oplus Y'_r = 1 \oplus 0 = 1$. Неравенство синдрома нулю означает, что получено сообщение с ошибкой (ошибками).

12.4.3. Код Хемминга

Рассмотрим более подробно методы помехоустойчивого кодирования на примере широко известного и часто используемого кода Хемминга. Данный код характеризуется минимальным кодовым расстоянием $d_{\min} = 3$. При его использовании кодирование сообщения также должно удовлетворять соотношению (12.5). Причем *вес столбцов подматрицы A должен быть больше либо равен 2*. Второй особенностью данного кода является то, что используется *расширенный контроль четности групп символов информационного слова*, т. е. $r > 1$. Для упрощенного вычисления r можно воспользоваться следующим простым соотношением:

$$r = \log_2 k + 1. \tag{12.9}$$

В сравнении с предыдущим кодом данный позволяет не только обнаруживать, но и исправлять одиночную ошибку в кодовом слове (см. формулу (12.8)).

В нашем рассмотрении подматрицу A можно определить так:

$$A = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1k} \\ h_{21} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ h_{r1} & \cdots & \cdots & h_{rk} \end{bmatrix}. \tag{12.10}$$

Элемент этой подматрицы (0 или 1) h_{ij} относится к i-й строке и j-му столбцу ($i = \overline{1, r}; j = \overline{1, k}$).

Вычисляем проверочные символы в соответствии с выражением (12.5):

$$x_{k+i} = \sum_{ij}^{rk} h_{ij} \cdot x_j \pmod 2. \tag{12.11}$$

Определяем синдром:

$$y'_{k+i} = \sum_{ij}^{rk} h_{ij} y_k \mod 2; \; S = y_{k+i} \oplus y'_{k+i}. \qquad (12.12)$$

Пример 12.9. Имеется информационное слово $X_k = 1001$. Проанализируем использование рассматриваемого кода.

Для начала отмечаем, что $k = 4$. В соответствии с выражением (12.9) подсчитываем длину избыточного слова: $r \geq \log_2 4 + 1 = 3$, тогда $n = k + r = 7$.

Создаем проверочную матрицу $H_{7,4}$:

$$\underbrace{H_{7,4}}_{n \times k} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}, \quad \underbrace{\quad\quad\quad\quad}_{A} \quad \underbrace{\quad\quad\quad}_{I}$$

Вычисляем проверочные символы, используя соотношение (12.11).

В соответствии с этим первый проверочный символ x_{k+1} будет равен 1, остальные – нулю:

$$x_{k+1} = h_{11} \cdot x_1 \oplus h_{12} \cdot x_2 \oplus \ldots \oplus h_{14} \cdot x_4 = 0 \cdot 1 \oplus 1 \cdot 0 \oplus 1 \cdot 0 \oplus 1 \cdot 1 = 1;$$
$$x_{k+2} = h_{21} \cdot x_1 \oplus h_{22} \cdot x_2 \oplus \ldots \oplus h_{24} \cdot x_4 = 1 \cdot 1 \oplus 0 \cdot 0 \oplus 1 \cdot 0 \oplus 1 \cdot 1 = 0;$$
$$x_{k+3} = h_{31} \cdot x_1 \oplus h_{32} \cdot x_2 \oplus \ldots \oplus h_{34} \cdot x_4 = 1 \cdot 1 \oplus 1 \cdot 0 \oplus 0 \cdot 0 \oplus 1 \cdot 1 = 0.$$

Таким образом, избыточное слово будет таким: $X_r = 100$, а кодовое слово – $X_n = 1001\ 100$.

Рассмотрим ситуацию, когда ошибок в переданной информации нет ($t = 0$), т. е. $X_n = Y_n = 1001\ 100$.

Вычислим новый набор проверочных символов в соответствии с выражением (12.12) и синдром:

$$Y'_r = 100,$$

$$S = Y_r \oplus Y'_r = 100 \oplus 100 = 000 \equiv 0.$$

Нулевой синдром означает безошибочную передачу (или прием) информации.

Рассмотрим ситуацию, когда возникает одиночная ошибка ($t = 1$).

Пусть ошибка произошла в служебных символах $Y_n = 1001\underline{0}00$ (ошибочный символ подчеркнут).

Синдром вычисляем по методике, приведенной для случая отсутствия ошибок. Получаем $S = 100$. Вес синдрома равен 1 и это означает, что произошла ошибка. Местоположение ошибки выявляется анализом (декодированием) синдрома. Декодирование опирается на вышеприведенное соотношение (12.5), в соответствии с которым, принимая во внимание выражение (12.6), можем записать:

$$H \cdot (Y_n)^T = H \cdot (X_n \oplus E_n)^T = H \cdot (X_n)^T \oplus H \cdot (E_n)^T = 0 \oplus h_5, \quad (12.13)$$

где h_5 – пятый столбец матрицы, номер которого соответствует номеру ошибочного символа в принятом кодовом слове. Действительно, $h_5 = S = 100$.

В результате декодирования синдрома получается вектор ошибки (*унарный вектор*, имеющий единичный вес): $E_n = 0000100$. Исправление ошибочного бита достигается простым сложением по модулю 2 вектора E_n и кодового слова Y_n:

$$Y_n = E_n \oplus Y_n = 0000100 \oplus 1001000 = 1001100.$$

Пусть ошибка произошла в бите информационного слова: $Y_n = \underline{0}001100$.

Вычислим дополнительные проверочные символы и синдром:

$$Y'_r = 111;$$
$$S = Y_r \oplus Y'_r = 011.$$

Убедимся, что синдром соответствует первому столбцу используемой проверочной матрицы. Это означает, что декодирование синдрома однозначно укажет на местоположение ошибочного бита:

$$E_n = 1000000;$$
$$Y'_n = E_n \oplus Y_n = 0001100 \oplus 1000000 = 1001100 \equiv X_n.$$

При возникновении ошибок кратности 2 (например, на позициях l и m) данный код не позволяет *однозначно* идентифицировать ошибки, поскольку с учетом формулы (12.13) имеем:

$$S = h_l \oplus h_m. \quad (12.14)$$

Таким образом, код Хемминга с $d_{\min} = 3$ *гарантированно обнаруживает и исправляет* одиночную ошибку в любом разряде кодового слова.

Порядок следования вектор-столбцов в матрице A не имеет особого значения, однако важно, чтобы на передающей и на принимающей сторонах используемые матрицы были бы абсолютно идентичными.

Матрица кода Хемминга с $d_{\min} = 3$ (H) может быть трансформирована в матрицу с $d_{\min} = 4$ (H') путем добавления в нее одной строки и одного столбца (столбец добавятся в подматрицу I):

$$H' = \left|\begin{array}{ccc:c} & & & 0 \\ & H & & 0 \\ & & & \vdots \\ & & & 0 \\ \hdashline 1 & 1 & \cdots\ 1 & 1 \end{array}\right|.$$

Приведение такой матрицы к канонической форме (в которой единичная матрица содержит 1 только на главной диагонали) производится путем выполнения суммирования строк матрицы H с единичной дополнительной строкой таким образом, чтобы вес каждого столбца единичной матрицы был равен 1. После таких линейных операций каждый столбец матрицы A будет иметь нечетный вес, а сумма двух любых столбцов – четный. Это обеспечивает коду на основе модифицированной матрицы (при неизменном k) минимальное кодовое расстояние $d_{\min} = 4$. Значит такой код обеспечит обнаружение двух ошибок в принятом сообщении, однако исправит (см. формулу (12.8)) лишь одиночные.

12.4.4. Циклический код

Циклические избыточные коды, ЦК (Cyclic Redundancy Codes, CRC; использование, например, см. на рис. 6.2 и 6.12) относятся к *линейным систематическим*.

Важнейшие свойства кодов:

– каждое кодовое слово, получаемое из исходного кодового слова путем циклической перестановки его символов, также является разрешенным кодовым словом;

– при циклической перестановке символы кодового слова перемещаются слева направо на одну позицию.

Следует запомнить: при циклическом сдвиге вправо на один разряд необходимо исходную кодовую комбинацию поделить на X, а при сдвиге влево на один символ – умножить на X.

Пример 12.10. Если кодовое слово имеет вид: 1101100, то разрешенной кодовой комбинацией будет следующая: 0110110.

Принято описывать ЦК при помощи *порождающих полиномов* $G(X)$ степени $r = n - k$, где r – число проверочных символов в кодовом слове.

Порождающие полиномы циклических кодов. Формирование *разрешенных кодовых комбинаций* ЦК $B_j(X)$ (обозначение кодовой комбинации здесь соответствует обозначению кодового слова X_n для кода Хемминга) основано на предварительном выборе *порождающего (образующего) полинома* $G(X)$, обладающего важным отличительным признаком: все комбинации $B_j(X)$ делятся на порождающий полином $G(X)$ без остатка; в результате деления получаем информационное слово:

$$B_j(X) / G(X) = A_j(X), \qquad (12.15)$$

здесь $B_j(X)$ – кодовое слово; $A_j(X)$ – информационное слово (соответствует X_k – для кода Хемминга).

Пример 12.11. При $G(X) = X^4 + X^2 + X$ и $B_j(X) = X^5 + X^3 + X^2$ имеем

$$\begin{array}{l|l} X^5 + X^3 + X^2 & \underline{X^4 + X^2 + X} \\ \underline{X^5 + X^3 + X^2} & X \\ 0 \quad 0 \quad 0 & \end{array}$$

остаток от деления – 000 или просто 0 ($R(X) = 0$).

Степень *порождающего полинома* r определяет число проверочных символов: $r = n - k$. Например, CRC-32 означает, что используется полином 32-й степени, а CRC-16 – полином 16-й степени.

Из этого свойства следует простой способ формирования разрешенных кодовых слов ЦК – умножение информационного слова на порождающий полином $G(X)$:

$$B(X) = A(X) \cdot G(X). \qquad (12.16)$$

Порождающими могут быть только такие полиномы, которые являются делителями двучлена (бинома) $X^n + 1$:

$$(X^n + 1) / G(X) = H(X) \qquad (12.17)$$

при нулевом остатке: $R(X) = 0$.

С увеличением максимальной степени порождающих полиномов r резко увеличивается их количество: при $r = 3$ имеется всего два полинома, а при $r = 10$ их уже несколько десятков. В таблице приведены некоторые полиномы и соответствующие им коды.

Некоторые из известных кодов, описываемые с помощью полиномов

Степень полинома r	Полином $G(X)$	Двоичное представление полинома	n	k	Примечание
1	$X + 1$	11	3	2	Код с проверкой на четность (3, 2)
2	$X^2 + X + 1$	111	3	1	Код с повторением (3, 1)
3	$X^3 + X^2 + 1$ $X^3 + X + 1$	1101 1011	7	4	Классический код Хемминга (7, 4)
4	$X^4 + X^3 + 1$ $X^4 + X + 1$ $X^4 + X^2 + X + 1$ $X^4 + X^3 + X^2 + 1$	11001 10011 10111 11101	15 15 7 7	11 11 3 3	Классический код Хемминга (15, 11) Классический код Хемминга (15, 11) Коды Файра – Абрамсона (7, 3) Коды Файра – Абрамсона (7, 3)
5	$X^5 + X^2 + 1$ $X^5 + X^3 + 1$	100101 101001	31	26	Классический код Хемминга (31, 26)

В Bluetooth, например, используется полином CRC-16 вида $X^{16} + X^{12} + X^5 + 1$ или в двоичной форме: 10001000000100001, в некоторых протоколах на основе стандарта IEEE 802.3 – CRC-32: $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$.

Другой из возможных вариантов кода CRC-32 задается полиномом $X^{32} + X^{22} + X^2 + X + 1$, которому соответствует двоичное представление: 10000000001000000000000000000111.

Два варианта порождающих полиномов кода Хемминга (7, 4) с записью по модулю 2 в виде 1101 и 1011 представляют собой так называемые *двойственные многочлены* (полиномы): весовые коэффициенты одного полинома, зачитываемые слева направо, становятся

весовыми коэффициентами двойственного полинома при считывании их справа налево.

Порождающие полиномы кода Хемминга (7, 4) являются не только двойственными, но и неприводимыми.

Неприводимые полиномы не делятся ни на какой другой полином степени меньше r, поэтому их называют еще **неразложимыми**, **простыми** и **примитивными**.

Пример 12.12. Порождающий полином $G(X) = X^7 + 1$ раскладывается на три неприводимых полинома:

$$X^7 + 1 = (X + 1) \cdot (X^3 + X^2 + 1) \cdot (X^3 + X + 1) = G_1(X) \cdot G_2(X) \cdot G_3(X),$$

каждый из которых является порождающим для следующих кодов:

$G_1(X) = X + 1$ – код с проверкой на четность, КПЧ (7, 6);

$G_2(X) = X^3 + X^2 + 1$ – первый вариант кода Хемминга (7, 4);

$G_3(X) = X^3 + X + 1$ – двойственный $G_2(X)$, второй вариант кода Хемминга.

Различные вариации произведений $G_{1,2,3}(X)$ дают возможность получить остальные порождающие полиномы.

Порождающая матрица G циклического кода имеет в качестве строк векторы $G(X)$, $XG(X)$, ..., $X^{k-1}G(X)$:

$$G = \begin{matrix} G(X) \\ XG(X) \\ \\ X^{k-1}G(X) \end{matrix} = \begin{vmatrix} g_0 & g_1 & g_2 & \dots & g_r & 0 & \dots & 0 \\ 0 & g_0 & g_1 & \dots & g_r & 0 & \dots & 0 \\ & & & \dots & & & & \\ 0 & \dots & 0 & g_0 & g_1 & g_2 & \dots & g_r \end{vmatrix}, \qquad (12.18)$$

где g_0, ..., g_r – коэффициенты генераторного полинома.

Проверочная матрица H кода строится на основе полинома (см. выражение (12.17)):

$$H(X) = (X^n + 1) / G(X) \qquad (12.19)$$

или

$$H = \begin{matrix} H(X) \\ XH(X) \\ \\ X^{r-1}H(X) \end{matrix} = \begin{vmatrix} 0 & \dots & 0 & h_k & \dots & h_2 & h_1 & h_0 \\ 0 & \dots & h_k & h_{k-1} & \dots & h_1 & h_0 & 0 \\ & & & \dots & & & & \\ h_k & \dots & h_2 & h_1 & h_0 & 0 & \dots & 0 \end{vmatrix}, \qquad (12.20)$$

где h_j – коэффициенты полинома $H(X)$.

Справедливо,

$$G(X) \cdot H(X)^T = 0 \text{ или } H(X) \cdot G(X)^T = 0,$$

здесь индекс «T» означает транспонирование матрицы.

Пример 12.13. Задан ЦК (7, 4) дуальными порождающими полиномами $G(X) = X^3 + X + 1$ и $\underline{G}(X) = X^3 + X^2 + 1$.

Составить порождающие матрицы кодов.

Первой строкой в матрице записывается порождающий полином (в двоичном представлении) с умножением его на оператор сдвига X_r для резервирования места под запись трех ($r = 3$) проверочных символов. Следующие $k - 1$ строк матриц получаются путем последовательного циклического сдвига базового кодового слова матриц G и $\underline{G}$ на одну позицию вправо:

$$G(X) = \begin{matrix} 1011000 \\ 0101100 \\ 0010110 \\ 0001011 \end{matrix} \qquad \underline{G}(X) = \begin{matrix} 1101000 \\ 0110100 \\ 0011010 \\ 0001101 \end{matrix}$$

Для построения порождающей матрицы, формирующей разделимый блочный код, необходимо преобразовать ее к каноническому виду путем линейных операций над строками.

Каноническая матрица должна в левой части порождающей матрицы ЦК содержать единичную диагональную квадратную подматрицу порядка k для получения в итоге блочного ЦК.

Пример 12.14. Привести к каноническому виду матрицы из примера 12.13.

С этой целью для получения первой строки канонической матрицы $G_к(X)$ необходимо сложить по модулю 2 строки с номерами 1, 3 и 4 матрицы $G(X)$, а для матрицы $\underline{G}_к(X)$ – строки с номерами 1, 2 и 3, вторая строка канонической матрицы образуется сложением строк 2 и 4; оставшиеся строки – без изменений. В итоге имеем следующий вид первой из *дуальных канонических матриц*:

$$G_к(X) = \begin{matrix} 1000 & 101 & (1 + 3 + 4) \\ 0100 & 111 & (2 + 4) \\ 0010 & 110 & (3 = 3) \\ 0001 & 011 & (4 = 4). \end{matrix}$$

Запись (1 + 3 + 4) означает, что данная строка матрицы получена в результате суммы по модулю 2 первой, третьей и четвертой строк матрицы $G(X)$.

Вторую матрицу ($G_к(X)$) построить самостоятельно.

Проверочная матрица $H_{7,4}$ размерностью $n \times r$ может быть получена из порождающей матрицы канонического вида путем дополнения проверочной подматрицы единичной матрицей размерности $r \times r$:

$$H_{7,4} = \begin{matrix} 101 \\ 111 \\ 110 \\ \underline{011} \\ 100 \\ 010 \\ 001, \end{matrix}$$

или в каноническом виде:

$$H_{7,4} = \begin{matrix} 1110 & 100 \\ 0111 & 010 \\ 1101 & 001. \end{matrix}$$

Вычисление проверочных символов (X_r) кодового слова (X_n) чаще всего основывается на методе деления полиномов. Метод позволяет представить разрешенные к передаче кодовые комбинации в виде разделенных информационных X_k и проверочных X_r символов, т. е. получить *блочный* код.

Поскольку число проверочных символов равно r, то для компактной их записи в последние младшие разряды кодового слова надо предварительно к X_k (соответствует $A_j(X)$ в формуле (12.15)) справа приписать r нулей, что эквивалентно умножению X_k на оператор сдвига X^r. При этом имеется возможность представить кодовую комбинацию в виде последовательности информационных и проверочных символов:

$$X_n = X_k \cdot X^r \,||\, R(X), \qquad (12.21)$$

где $R(X)$ – остаток от деления $(X_k \cdot X^r) / G(X)$ (см. формулу (12.15)).

В алгоритме на основе (12.21) можно выделить три этапа формирования разрешенных кодовых комбинаций в кодере:

1) к комбинации слова X_k дописывается справа r нулей, что эквивалентно умножению X_k на X^r;

2) произведение $X_k \cdot X_r$ делится на соответствующий порождающий полином $G(X)$ и определяется остаток $R(X)$, степень которого не превышает $r-1$, этот остаток и дает группу проверочных символов (X_r);

3) вычисленный остаток присоединяется справа к X_k.

Пример 12.15. Рассмотрим процедуру кодирования при $X_k = 1001$, т. е. сформируем кодовое слово циклического кода (7, 4).

В заданном ЦК ($n = 7$, $k = 4$, $r = 3$) выберем порождающий полином $G(X) = X^3 + X + 1$ (код Хемминга).

$X_k = 1001 \sim X^3 + 1$ (знак «~» – *тильда*, означает соответствие).

1) $X_k X_r = (X^3 + 1) \cdot X^3 = X^6 + X^3 \sim 1001000$ ($n = 7$).

2) $(X_k X_r) / G(X) = X^6 + X^3 \,|\, X^3 + X + 1$

$$\underline{X^6 + X^4 + X^3} \,\big|\, X^3 + X$$

$$X^4$$

$$\underline{X^4 + X^2 + X}$$

$$X^2 + X,$$

$X^2 + X$ – остаток; $R(X) = X^2 + X \sim 110$.

3) $X_n = X_k X_r \,||\, R(X) = 1001110$ – итоговая комбинация ЦК (кодовое слово).

Синдромный метод декодирования ЦК. Основная операция: принятое кодовое слово (X_n) нужно поделить на порождающий полином, используемый в операции кодирования. Если значение X_n принадлежит коду, т. е. не искажено помехами, то остаток от деления (синдром) будет нулевым. Ненулевой остаток свидетельствует об ошибке в принятой кодовой комбинации. Для ее исправления нужно определить вектор (полином) ошибки (обозначим его E_n, см. формулу (12.13)).

После передачи по каналу с помехами принимается кодовое слово

$$Y_n = X_n \oplus E_n, \tag{12.22}$$

здесь также сумма по модулю 2.

При декодировании принятое кодовое слово делится на $G(X)$:

$$Y_n / G(X) = U, S_r, \tag{12.23}$$

где U – результат деления; S_r (остаток от деления $Y_n / G(X)$) – синдром.

Всякому ненулевому синдрому соответствует определенное расположение (конфигурация) ошибок: синдром для ЦК имеет те же свойства, что и для кода Хемминга и используется при декодировании.

Пример 12.16. Рассмотрим процедуру декодирования сообщения, сформированного в примере 12.15. Пусть $Y_n = 10\underline{1}1110$ (ошибочным является третий бит – подчеркнут).

Вспомним, что порождающая матрица имеет вид, показанный в примере 12.14:

$$H_{7,4} = \begin{matrix} 1110 & 100 \\ 0111 & 010 \\ 1101 & 001. \end{matrix}$$

Для решения задачи последовательно выполняем следующие операции:

1) деление в соответствии с формулой (12.15):

$$Y_n / G(X) = \begin{array}{l} X^6 + X^4 + X^3 + X^2 + X \,\big|\, \underline{X^3 + X + 1} \\ \underline{X^6 + X^4 + X^3 \qquad\quad} \,\big|\, X^3 \\ X^2 + X \end{array};$$

$X^2 + X$ – остаток, т. е. синдром $S_r = X^2 + X \sim 110$;

2) декодирование синдрома, позволяющее определить местоположение ошибки: по полученному синдрому 110 в анализаторе синдрома (дешифраторе синдрома) определяем вид вектора $E_n = 0010000$. Здесь обратим внимание на важнейшую деталь: *синдром равен третьему вектор-столбцу в матрице* $H_{7,4}$, *поэтому единичный символ будет в третьем разряде вектора* E_n;

3) исправление ошибки: $Y_n \oplus E_n = 10\underline{1}1110 \oplus 0010000 = 10\underline{\underline{0}}1110$. После исправления (исправленный бит подчеркнут двойной линией) получим то же слово, что и было сформировано в источнике сообщения.

12.5. Основы информационной безопасности компьютерных сетей

12.5.1. Характеристика основных угроз информационной безопасности сетей

Как мы уже неоднократно подчеркивали (например, в пункте 2.3.6), защита данных в компьютерных сетях стала едва ли не самой главной проблемой в сфере информационных технологий.

Сетевая атака – это вторжение в операционную систему удаленного компьютера. Целями сетевой атаки являются: захват управления над операционной системой, приведение ее к состоянию *отказа в обслуживании* (Denial-of-Service, DoS attack) или получение доступа к защищенной информации.

Существуют следующие основные *типы DoS-атак*:

– отправка на удаленный компьютер специально сформированных сетевых пакетов, не ожидаемых этим компьютером, которые вызывают сбои в работе операционной системы или ее остановку;

– отправка на удаленный компьютер большого количества сетевых пакетов за короткий период времени. Все ресурсы атакуемого компьютера используются для обработки отправленных злоумышленником сетевых пакетов, из-за чего компьютер перестает выполнять свои функции.

Сетевые атаки-вторжения (Network Intrusion Attack или Network Infiltration Attack). Это сетевые атаки, целью которых является «захват» операционной системы атакуемого компьютера. Это самый опасный вид сетевых атак, поскольку в случае ее успешного завершения операционная система полностью переходит под контроль злоумышленника. Этот вид сетевых атак применяется в случаях, когда злоумышленнику требуется получить конфиденциальные данные с удаленного компьютера (например, номера банковских карт или пароли) либо использовать удаленный компьютер в своих целях (например, атаковать с этого компьютера другие компьютеры) без ведома пользователя.

Термин «сетевая атака» иногда заменяется термином «удаленная атака» при анализе угроз информационным ресурсам и защите от таких угроз.

Угроза – это возможные или реализуемые попытки завладеть информационными ресурсами без согласия владельца этих ресурсов.

Источниками угроз информационной безопасности являются:

– компании-конкуренты;

– злоумышленники (хакеры);

– тестировщики на проникновение – это высококвалифицированные специалисты (их называют также *пенсестерами*), которые, используя специальные методы (так называемые *Black/Gray/White-box* по полному стеку: социальная инженерия, внешний периметр, мобильные приложения, веб-приложения, внутренняя инфраструктура компаний-заказчиков, написание отчетной документации), проводят углубленный анализ потенциальных уязвимостей информационных систем и сетей; OSINT (Open Source Intelligence) – разведка на основе открытых источников;

– сотрудники компании (как правило, обиженные) или *инсайдеры*;

– организованные преступные группы;

– государственные органы (подтверждением являются откровения Э. Сноудена*).

Виды сетевых атак и их последствия имеют значительные отличия друг от друга. Современная классификация (условная) угроз проводится по следующим параметрам:

– характер воздействия, оказываемого на сеть;

– цель оказываемого воздействия;

– наличие обратной связи с сетью, подвергнутой атаке;

– условие начала атаки;

– расположение субъекта по отношению к объекту атаки;

– уровень эталонной модели ISO.

Рассмотрим подробнее основные виды сетевых атак по этим категориям.

По характеру воздействия на сеть атаки можно разделить на *активные* и *пассивные*.

Активная атака проводится с непосредственным воздействием на сеть, которое может предусматривать ограничение ее работоспособности, модификацию настроек. Воздействие такого рода обязательно оставляет следы, поэтому при его планировании изначально предусматривается обнаружение.

Пассивная атака проводится без непосредственного влияния на работу сети. Однако в результате ее нарушается сетевая безопасность. Обнаружить пассивную атаку намного сложнее именно из-за отсутствия прямого воздействия. Примером таких угроз можно назвать постановку наблюдения или прослушки.

---

*Edward Snowden – американский технический специалист и спецагент, бывший сотрудник ЦРУ и АНБ, раскрыл факт всеобъемлющего слежения в 60 странах за более чем миллиардом человек и правительствами 35 стран.

По цели различают виды сетевых атак, направленных на нарушение:

– функционирования;

– конфиденциальности;

– целостности атакуемой сети.

Основной целью таких атак является, как правило, несанкционированный доступ к закрытой информации методом ее искажения или перехвата. В первом случае сведения могут быть изменены, во втором – доступ производится без изменения данных.

По наличию обратной связи с атакуемой сетью атака может проводиться *с обратной связью* или без нее (*однонаправленная атака*).

В первом случае атакующим субъектом устанавливается обмен данными с атакуемым объектом. В результате злоумышленники получают актуальные данные о состоянии сети.

Однонаправленная атака не предусматривает установления обратной связи. Ее проводят в ситуации, когда для реализации целей злоумышленников не требуется оперативной реакции на изменения состояния объекта.

По условию начала атаки можно выделить типы сетевых атак по следующему критерию:

– по запросу от объекта;

– по выполнению на стороне объекта определенного действия;

– безусловные атаки.

Первые два типа атак начинаются после соответствующего события, а безусловные – в любой момент.

В зависимости от ***расположения субъекта по отношению к объекту атаки*** различают сетевые *атаки межсегментного* и *внутрисегментного типа*. Особенностью категории первого типа является расположение субъекта и объекта в разных сегментах сети. Второй тип характеризуется их расположением в одном сегменте.

Сегментом сети называю хосты (компьютеры), физически объединенные между собой.

Воздействие на атакуемую сеть может осуществляться на разных ***уровнях эталонной модели ISO/OSI.***

Как видно, сетевые атаки столь же разнообразны, как и системы, против которых они направлены. Некоторые атаки отличаются

большой сложностью. Другие может осуществить обычный оператор, даже не предполагающий, какие последствия будет иметь его деятельность.

Для характеристики специфических типов атак необходимо знать некоторые ограничения, изначально присущие протоколу TPC/IP.

Ввиду того что изначально средства защиты для протокола IP не разрабатывались, все его реализации стали дополняться разнообразными сетевыми процедурами, услугами и продуктами, снижающими риски, присущие этому протоколу.

Кратко рассмотрим основные типы атак, обычно применяемых против сетей IP.

Сниффинг пакетов (Sniffer – в данном случае в смысле фильтрация) – атака на основе прикладной программы (снифера), которая использует сетевую карту, работающую в режиме (promiscuous (не делающий различия) mode), в котором все пакеты, полученные по физическим каналам, сетевой адаптер отправляет приложению для обработки.

Сниффер перехватывает все сетевые пакеты, которые передаются через атакуемый домен. В настоящее время снифферы работают в сетях на вполне законном основании. Они используются для диагностики неисправностей и анализа трафика. Однако ввиду того, что некоторые сетевые приложения передают данные в текстовом формате (telnet, FTP, SMTP, POP3 и т. д.), с помощью сниффера можно узнать полезную, а иногда и конфиденциальную информацию (например, имена пользователей и пароли).

Перехват имен и паролей создает большую опасность, так как пользователи часто применяют один и тот же логин и пароль для множества приложений и систем. Многие пользователи вообще имеют один пароль для доступа ко всем ресурсам и приложениям. Если приложение работает в режиме «клиент-сервер», а аутентификационные данные передаются по сети в читаемом текстовом формате, эту информацию с большой вероятностью можно использовать для доступа к другим корпоративным или внешним ресурсам. Таким образом, человек, конечный пользователь, оказывается самым слабым звеном системы информационной безопасности, и хакеры, зная это, умело применяют методы социальной инженерии.

Помимо Ethernet сниффер умеет перехватывать трафик беспроводных сетей (стандарты 802.11 и протокол Bluetooth).

Спуфинг (Spoofing) – тип атаки, основанной на фальсификации передаваемых данных.

Спуфинг может быть нацелен на получение расширенных привилегий и основан на обходе механизма верификации при помощи формирования запроса, аналогичного настоящему. Одним из вариантов такой подмены является подделка HTTP-заголовка для получения доступа к скрытому контенту. Целью спуфинга может также быть обман пользователя – классическим примером подобной атаки может служить подмена адреса отправителя в письмах электронной почты.

Вот несколько примеров различных видов спуфинга:

– *спуфинг с подменой номера вызывающего абонента* (Caller ID Spoofing) – идентификатор вызывающего абонента (Caller ID) позволяет получателю телефонного звонка определить личность того, кто звонит. Такой вид спуфинга происходит в тех случаях, когда мошенник использует ложную информацию для изменения идентификатора вызывающего абонента (т. е. мошенник звонит якобы с другого телефона – например, с телефона вашего друга). Большинство спуфинг-атак с подменой идентификатора вызывающего абонента *происходит с помощью VoIP* (*Voice over Internet Protocol*), который позволяет мошенникам создавать номер телефона и имя идентификатора вызывающего абонента по своему выбору. Как только получатель звонка ответит на звонок, мошенник попытается убедить его предоставить ему требуемую важную информацию;

– *спуфинг с подменой сайта* (Web-site Spoofing) – это тип спуфинг-атаки, в рамках которой мошенник пытается создать опасный (вредоносный) сайт похожим на надежный безопасный сайт (например, известного банка), используя его шрифты, цвета и логотипы; такой спуфинг *проводится путем репликации оригинального надежного сайта* с целью привлечения пользователей на специально созданный поддельный фишинговый или вредоносный сайт;

– *спуфинг с подменой адреса электронной почты* (E-mail Spoofing) – это тип спуфинг-атаки, в рамках которой мошенник рассылает электронные письма с поддельными адресами отправителей с

намерением заразить атакуемый компьютер вредоносными программами, заполучить деньги или украсть информацию; в качестве адресов электронной почты отправителей зачастую подставляются те адреса, которым доверяет атакуемая сторона;

– *спуфинг с подменой IP-адреса* (IP Spoofing) – мошенник стремится скрыть реальное местоположение в Интернете того места, откуда запрашиваются или куда отправляются данные пользователя/жертвы; цель IP-спуфинга – заставить компьютер жертвы «думать», что информация, отправляемая мошенником пользователю, исходит из надежного источника, что позволяет вредоносному контенту доходить до пользователя. IP-спуфинг часто является отправной точкой для прочих атак. Классический пример – атака DoS, которая начинается с чужого адреса, скрывающего истинную личность хакера. Если же хакеру удается поменять таблицы маршрутизации и направить трафик на ложный IP-адрес, он получит все пакеты и сможет отвечать на них так, будто он является санкционированным пользователем;

– *спуфинг с подменой DNS-сервера* (DNS Server Spoofing) – используется для перенаправления трафика пользователя на поддельные IP-адреса (на вредоносные сайты); в рамках такой атаки мошенник меняет IP-адреса DNS-серверов, указанных на компьютере жертвы, на поддельные IP-адреса, которые мошенник хочет использовать для обмана жертвы;

– *ARP-спуфинг* (Address Resolution Protocol Spoofing) – используется для изменения или кражи данных, а также для взлома компьютера жертвы внутри его сессии (подключения); для этого злоумышленник свяжет себя с IP-адресом жертвы, чтобы иметь возможность получить доступ к тем данным, которые изначально предназначались для владельца этого IP-адреса (т. е. жертвы);

– *SMS-спуфинг* (Text Message Spoofing) – спуфинг с подменой текстовых сообщений: мошенник отправляет текстовое или SMS-сообщение, используя номер телефона другого человека. Мошенники делают это, скрывая свою личность за буквенно-цифровым идентификатором отправителя, и обычно в свои сообщения включают ссылки для загрузки вредоносных программ или для перехода на фишинговые сайты;

– *GPS-спуфинг* (GPS Spoofing) – атака происходит для «обмана» GPS-приемника, когда передаются поддельные сигналы, которые напоминают настоящие: мошенник притворяется, что находится в одном месте, а на самом деле находится в другом.

Мошенники могут использовать *GPS-спуфинг*, чтобы, например, взломать GPS в автомобиле и отправить вас по ложному адресу или даже вмешаться в GPS-сигналы кораблей, самолетов и т. д. Любое мобильное приложение, которое полагается на данные о местоположении смартфона, может стать мишенью для такого типа атаки.

– *атака посредника или «человека посередине»* (Man-in-the-Middle Attack, MitM) происходит в тех случаях, когда мошенник взламывает Wi-Fi-сеть или создает дублирующую поддельную Wi-Fi-сеть в том же месте для перехвата web-трафика между двумя сторонами подключения (отправитель и получатель трафика). Если пользователь подключен к публичной Wi-Fi-сети, злоумышленник может легко перенаправить весь трафик между устройством пользователя и роутером через свой ноутбук, тем самым он получит доступ ко всем данным, которые пользователь передает или загружает. С помощью такой атаки мошенники могут перенаправлять к себе используемую жертвой конфиденциальную информацию, такую как логины, пароли или номера банковских карт.

Чтобы спуфинг-атака была успешной, она должна включать в себя определенный уровень *социальной инженерии*. Это означает, что методы, которые используют мошенники, способны эффективно обмануть своих жертв и заставить их предоставить свою личную информацию.

Социальная инженерия (Sicial Ingeneering) – это использование хакером психологических приемов «работы» с пользователем. Базируется на психологических особенностях личности и закономерностях человеческого мышления.

В самом худшем случае хакер, перехватив пароль, получает доступ к пользовательскому ресурсу на системном уровне и с его помощью создает нового пользователя, которого можно в любой момент использовать для доступа в сеть и к ее ресурсам.

Одним из самых известных социальных инженеров в истории является Кевин Митник (Kevin David Mitnick). Авторы данного пособия настоятельно рекомендуют прочитать или хотя бы просмотреть

ставшие бестселлерами книги К. Митника по проблемам информационной безопасности (некоторые из книг приведены в библиографии данного пособия).

В самостоятельный класс методов и средств, представляющих угрозу сетевой безопасности, выделяют ***компьютерные вирусы***, предназначенные для реализаций деструктивных действий на сетевом уровне – *сетевые вирусы*.

Сетевые вирусы, или **черви** (Worms), – это программы с *вредоносным кодом*, которые атакуют компьютеры в сети и распространяются через нее.

Сетевой червь Морриса известен как *первый компьютерный червь*, который распространялся в Интернете и заразил тысячи компьютеров. Он был написан в 1988 г. Робертом Таппаном Моррисом (Robert Tappan Morris), 23-летним доктором Корнельского университета. Этот вирус парализовал 2 ноября 1988 г. работу шести тысяч двухсот компьютеров в США.

Большинство известных компьютерных червей распространяется следующими способами:

– в виде файла, отправленного во вложении в электронном письме;

– в виде ссылки на интернет- или FTP-ресурс;

– в виде ссылки, переданной через сообщение ICQ или IR;

– через пиринговые сети обмена данными – P2P (Peer-to-Peer).

Некоторые черви распространяются как сетевые пакеты. Они проникают прямо в компьютерную память, после чего активируют код червя.

Специфическим типом деструктивных программ являются *руткиты*.

Руткит (Rootkit) – программа или набор программ, который позволяет скрыть присутствие в системе вредоносного ПО.

Как правило, руткит представляет собой набор утилит, который злоумышленник устанавливает на скомпрометированную систему сразу после получения root-доступа. В набор входят утилиты для

«заметания следов» вторжения в систему, которые делают незаметными снифферы, сканеры, кейлоггеры, троянские программы.

Руткит может содержать различные вредоносные инструменты, такие как *клавиатурный шпион*, *вор сохраненных паролей*, *сканер данных о банковских карточках*, *дистанционно управляемый бот для осуществления DDoS-атак*, а *также функции для отключения антивирусов*. Руткит обычно имеет функции *бэкдора*, т. е. он позволяет атакующему дистанционно подключаться к зараженному компьютеру, устанавливать или удалять дополнительные модули. Некоторые примеры руткитов для Windows: *TDSS*, *ZeroAccess*, *Alureon and Necurs*.

Мобильные устройства также подвержены традиционным атакам (например, DNS Hijacking, E-mail Phishing), так как используют те же базовые пользовательские сервисы, что и персональные ПК.

Наиболее известной формой хакерских атак является *DoS-атака.*

Атака **отказ в обслуживании** (Denial of Service, DoS) и **распределенная атака** типа **отказ в обслуживании** (Distributed DoS, DDoS) – это самая настоящая бомбардировка центрального сервера одновременными запросами данных. Задача – «забить» канал сервера и привести к его недоступности для легитимных клиентов.

Во втором случае (DDoS) злоумышленник отправляет такие запросы из нескольких взломанных систем. Атаки делают сеть недоступной для обычного использования за счет превышения допустимых пределов функционирования сети, операционной системы или приложения.

Против атак такого типа труднее всего создать стопроцентную защиту. Атаки считаются тривиальными, а от хакера для своей организации они требуют минимум знаний и умений: все необходимое программное обеспечение вместе с описаниями самой технологии находятся в свободном доступе в Интернете. Именно простота реализации и огромный причиняемый вред привлекают к DoS пристальное внимание администраторов, отвечающих за сетевую безопасность.

О DoS-атаках широко заговорили после того, как в декабре 1999 г. при помощи этой технологии были успешно атакованы web-узлы таких известных корпораций, как Amazon, Yahoo, CNN, eBay и E-Trade.

В случае использования некоторых серверных приложений (таких, например, как web-сервер или FTP-сервер) атаки DoS могут заключаться в том, чтобы занять все соединения, доступные для этих приложений, и держать их в занятом состоянии, не допуская обслуживания обычных пользователей. В ходе атак DoS могут использоваться обычные Интернет-протоколы, такие как TCP и ICMP (Internet Control Message Protocol). Большинство атак DoS опирается не на программные ошибки или бреши в системе безопасности, а на общие слабости системной архитектуры. Некоторые атаки сводят к нулю производительность сети, переполняя ее нежелательными и ненужными пакетами или сообщая ложную информацию о текущем состоянии сетевых ресурсов. Этот тип атак трудно предотвратить, так как требуется координация действий с провайдером. Когда атака такого типа проводится одновременно через множество устройств, речь идет о распределенной атаке DDoS.

Наиболее известными разновидностями атак DoS являются: *TCP SYN Flood, Ping of Death, Tribe Flood Network (TFN)* и *Tribe Flood Network 2000 (TFN2K), Trinco, Stacheldracht, Trinity, Smurf, ICMP Flood, UDP Flood, TCP Flood*.

Рассмотрим некоторые из них более подробно.

Smurf-атаки – это *ping-запросы* ICMP (Internet Control Message Protocol) по адресу направленной широковещательной рассылки. Фальшивый адрес источника, который используется в пакетах этого запроса, в результате оказывается мишенью атаки. Системы, получившие направленный широковещательный ping-запрос, отвечают на него и «затапливают» сеть, в которой находится сервер-мишень.

ICMP Flood – атака, аналогичная Smurf, только без усиления, создаваемого запросами по направленному широковещательному адресу.

UDP Flood – отправка на адрес системы-мишени множества пакетов UDP, что приводит к «связыванию» сетевых ресурсов.

TCP Flood – отправка на адрес системы-мишени множества TCP-пакетов, что также приводит к «связыванию» сетевых ресурсов.

TCP SYN Flood – при проведении такого рода атаки выдается большое количество запросов на инициализацию TCP-соединений с узлом-мишенью, которому в результате приходится расходовать все свои ресурсы на то, чтобы отслеживать эти частично открытые соединения.

Парольная атака – попытка подбора пароля легального пользователя для входа в сеть.

Хакеры могут проводить парольные атаки с помощью целого ряда методов, таких как *простой перебор* (Brute Force Attack), *троянский конь*, *IP-спуфинг* и *сниффинг пакетов*. Хотя *логин* и *пароль* можно получить при помощи IP-спуфинга и сниффинга пакетов, хакеры часто пытаются подобрать пароль и логин, используя для этого многочисленные попытки доступа. Такой подход носит название простого перебора. Нередко для такой атаки используется специальная программа, которая пытается получить доступ к ресурсу общего пользования (например, к серверу).

Еще одна проблема возникает, когда пользователи применяют один и тот же пароль для доступа ко многим системам: корпоративной, персональной и системам Интернет.

Парольных атак можно избежать, если не пользоваться паролями в текстовой форме. Одноразовые пароли и/или криптографическая аутентификация могут практически свести на нет угрозу таких атак. К сожалению, не все приложения, хосты и устройства поддерживают указанные выше методы аутентификации.

Атаки на уровне приложений. Могут проводиться несколькими способами. Самый распространенный из них состоит в использовании хорошо известных слабостей серверного программного обеспечения (sendmail, HTTP, FTP). Используя эти слабости, хакеры могут получить доступ к компьютеру от имени пользователя, работающего с приложением (обычно это бывает не простой пользователь, а привилегированный администратор с правами системного доступа). Сведения об атаках на уровне приложений широко публикуются, чтобы дать возможность администраторам исправить проблему с помощью коррекционных модулей (патчей). К сожалению, многие хакеры также имеют доступ к этим сведениям, что позволяет им учиться.

Главная проблема с атаками на уровне приложений состоит в том, что они часто «пользуются» портами, которым разрешен проход через *межсетевой экран*. К примеру, хакер, эксплуатирующий известную слабость web-сервера, часто использует в ходе атаки TCP порт 80. Поскольку web-сервер предоставляет пользователям web-страницы, межсетевой экран должен предоставлять доступ к этому порту. С точки зрения межсетевого экрана, атака рассматривается как стандартный трафик для порта 80.

> ***Сетевая разведка*** – сбор информации о сети с помощью общедоступных данных и приложений.

При подготовке атаки против какой-либо сети хакер, как правило, пытается получить о ней как можно больше информации. Сетевая разведка проводится в форме *запросов DNS*, *эхо-тестирования* (Ping Sweep) и *сканирования портов*.

Запросы DNS помогают понять, кто владеет тем или иным доменом и какие адреса этому домену присвоены. Эхо-тестирование адресов, раскрытых с помощью DNS, позволяет увидеть, какие хосты реально работают в данной среде.

Полностью избавиться от сетевой разведки невозможно.

Системы IDS на уровне сети и хостов обычно хорошо справляются с задачей уведомления администратора о ведущейся сетевой разведке, что позволяет лучше подготовиться к предстоящей атаке и оповестить провайдера, в сети которого установлена система, проявляющая чрезмерное любопытство.

> **Злоупотребление доверием** – злонамеренное использование отношений доверия, существующих в сети.

Классическим примером такого злоупотребления является ситуация в периферийной части корпоративной сети. В этом сегменте часто располагаются серверы DNS, SMTP и HTTP. Поскольку все они принадлежат к одному и тому же сегменту, взлом одного из них приводит ко взлому и всех остальных, так как эти серверы доверяют другим системам своей сети.

Другим примером является система, установленная с внешней стороны *межсетевого экрана*, имеющая отношения доверия с системой, установленной с его внутренней стороны. В случае взлома внешней системы хакер может использовать отношения доверия для проникновения в систему, которая защищена межсетевым экраном.

Риск злоупотребления доверием можно снизить за счет более жесткого контроля уровней доверия в пределах своей сети. Системы, которые расположены с внешней стороны межсетевого экрана, никогда не должны пользоваться абсолютным доверием со стороны защищенных экраном систем. Отношения доверия должны

ограничиваться определенными протоколами и по возможности аутентифицироваться не только по IP-адресам, но и по другим параметрам.

> **Переадресация портов** представляет собой разновидность злоупотребления доверием, когда взломанный хост используется для передачи через межсетевой экран трафика, который в противном случае был бы обязательно отбракован.

Представим себе межсетевой экран с тремя интерфейсами, к каждому из которых подключен определенный хост. Внешний хост может подключаться к *хосту общего доступа*, но не к хосту, установленному с внутренней стороны межсетевого экрана. Хост общего доступа может подключаться и к внутреннему, и к внешнему хосту. Если хакер захватит хост общего доступа, он сможет установить на нем программное средство, перенаправляющее трафик с внешнего хоста прямо на внутренний хост. Примером приложения, которое может предоставить такой доступ, является *netcat*.

Как итог рассмотрению существующих основных рисков безопасности компьютеров и сетей приведем краткий свод постулатов, разработанный Microsoft Security Response, которые выглядят следующим образом (приводятся здесь без комментариев):

1. Если злоумышленник (Bad Fellow) убедит вас в том, что его программа должна выполняться на вашем компьютере, этот компьютер перестанет принадлежать вам.

2. Если злоумышленник сможет изменить операционную систему на вашем компьютере, этот компьютер перестанет вам принадлежать.

3. Если злоумышленник имеет физический доступ к вашему компьютеру, этот компьютер перестанет вам принадлежать.

4. Если злоумышленник загрузит свои программы на ваш web-сайт, этот сайт перестанет вам принадлежать.

5. Использование простых паролей ослабляет строгие меры безопасности.

6. Компьютер безопасен только в том случае, когда администратор (безопасности) добросовестно относится к своим обязанностям.

7. Шифрование данных является безопасным в той мере, в какой соблюдены правила безопасного хранения ключа.

8. Устаревший сканер вирусов лишь немногим лучше отсутствующего сканера вирусов.

9. Абсолютная анонимность лишена практического смысла как в реальной жизни, так и при работе в Интернете.

10. Любая технология не является панацеей от всех бед.

12.5.2. Основные методы и средства нейтрализации угроз сетевой безопасности

Все многообразие методов и средств противодействия несанкционированному доступу к сетевым ресурсам условно подразделяется на следующие группы.

1. ***Организационные методы и средства*** подразумевают разработку, и исполнение в любой организации/лаборатории правил, регламентирующих и регулирующих доступ физических лиц к информации, хранящейся на носителях либо передаваемой внутри сети данного предприятия. В организациях, осуществляющих операции над критической информацией (правительственные, государственные, банковские, коммерческие и иные структуры), назначается специальное ответственное лицо – администратор безопасности (АБ), следящий за реализацей и соблюдением правил на основе реализуемой *политики безопасности* и необходимых *организационных правил.*

Критическое управление безопасностью на основе разработанной политики информационной безопасности и защиты данных основывается на следующих принципах:

– инвентаризация авторизированных и неавторизированных устройств;

– инвентаризация авторизированных и неавторизированных программных средств;

– безопасные конфигурации аппаратно-программных средств мобильных устройств, ноутбуков, рабочих станций и серверов;

– непрерывная оценка и нейтрализация уязвимостей;

– защита от вредоносного программного обеспечения (Malware);

– применение лицензионного программного обеспечения;

– контроль беспроводного доступа;

– возможность восстановления данных;

– квалифицированная оценка уровня безопасности системы на основе надлежащего обучения персонала;

– безопасная конфигурация сетевых устройств, таких как брандмауэры, маршрутизаторы, коммутаторы и др.;
– ограничение и контроль сетевых портов, протоколов и служб;
– управление использованием административных привилегий;
– определение периметра защиты (границы обороны);
– техническое обслуживание, мониторинг и анализ аудита;
– контролируемый доступ пользователей к разрешенным ресурсам;
– мониторинг и контроль учетных записей;
– защита данных на всех уровнях;
– реагирование на инциденты и управление их разрешением;
– инженерное обеспечение сетевой безопасности;
– выполнение тестов на проникновение и их анализ.

2. ***Правовые методы.*** Гражданский правовой кодекс предусматривает наказание за *компьютерные преступления*. В 1983 г. Организация экономического сотрудничества и развития определила это понятие.

> ***Компьютерная преступность*** (или «связанная с компьютерами преступность») – любые незаконные, неэтичные или неправомерные действия, связанные с автоматической обработкой или передачей информации.

Практически во всех странах с развитой информационной инфраструктурой предусматривается уголовно-правовая защита от компьютерных преступлений.

Перечень преступлений против информационной безопасности, зафиксированный, например, в законодательных и нормативных актах Республики Беларусь, соответствует положениям Будапештской конвенции (2001 г.)[*]. Конвенция охватывает широкий круг вопросов, в том числе все аспекты *киберпреступности*, включая незаконный доступ к компьютерным системам и перехват данных, воздействие на данные, воздействие на работу системы, противозаконное использование устройств, подлог и мошенничество с использованием компьютерных технологий, правонарушения, связанные с детской порнографией, и правонарушения, связанные с авторским правом и смежными правами. При подготовке конвенции преследовались

[*]Режим доступа: http://conventions.coe.int/treaty/rus/treaties/html/185.htm (дата доступа: 10.02.2015).

цели формирования общей правоохранительной системы для борьбы с киберпреступностью и создания условий для обмена информацией между всеми странами, подписавшими конвенцию. В Будапештской конвенции также устанавливаются требования защиты прав и свободы каждого в сети Интернет.

Международная уголовная полиция «Интерпол» пользуется *классификацией компьютерных преступлений* по кодификатору международной уголовной полиции генерального секретариата Интерпола. В 1991 г. данный кодификатор был интегрирован в автоматизированную систему поиска и в настоящее время доступен подразделениям Национальных центральных бюро Международной уголовной полиции «Интерпол» более чем 120 стран мира.

Все коды, характеризующие компьютерные преступления, имеют идентификатор, начинающийся с буквы ***Q***. Для характеристики преступления могут использоваться до пяти кодов, расположенных в порядке убывания значимости совершенного. Например:

QA – несанкционированный доступ и перехват:
- QAH – компьютерный абордаж;
- QAI – перехват;
- QAT – кража времени;
- QAZ – прочие виды несанкционированного доступа и перехвата;

QD – использование деструктивных программных средств:
- QDL – логическая бомба;
- QDT – троянский конь;
- QDV – компьютерный вирус;
- QDW – компьютерный червь;
- QDZ – прочие виды;

QF – компьютерное мошенничество:
- QFC – мошенничество с банкоматами;
- QFF – компьютерная подделка;
- QFG – мошенничество с игровыми автоматами;
- QFM – манипуляции с программами ввода-вывода;
- QFP – мошенничество с платежными средствами;
- QFT – телефонное мошенничество;
- QFZ – прочие компьютерные мошенничества;

QR – незаконное копирование (пиратство):
- QRG – компьютерные игры;
- QRS – программное обеспечение;

QRT – топография полупроводниковых изделий;
QRZ – прочее незаконное копирование;
QS – компьютерный саботаж:
QSH – с аппаратным обеспечением;
QSS – с программным обеспечением;
QSZ – прочие виды саботажа.

3. ***Физические (технические) методы.*** Объединяют методы ограничения физического доступа лиц к каналам передачи информации, устройствам ее хранения и обработки. Основаны на использовании простых замков, магнитных карт, чипов, таблеток, бесперебойных источников питания, а также на анализе антропометрических и биологических параметров человека (сетчатка глаза, отпечатки пальцев и др.).

4. ***Программно-технические методы.*** Основаны на использовании аппаратных и/или программных средств, позволяющих идентифицировать пользователя (либо техническое средство), а также оценить происхождение программного средства, поступающего в информационную сеть. Наиболее известными из указанных средств являются: использование пароля, антивирусных программ, брандмауэров или «огненных стен» (Firewalls) на входе сети, криптографического преобразования информации на основе методов шифрования, а также помехоустойчивого кодирования.

Далее в пособии в основном будем анализировать методы именно этой группы.

Подчеркнем простой и неоспоримый факт: ***абсолютно защищенные персональный компьютер или компьютерная сеть, операционная система, прикладная программа – такая же иллюзия, как и абсолютно надежно охраняемый дом.***

Также следует принять во внимание следующие обстоятельства общего плана.

Внедрение систем на основе технологии MDM (Mobile Device Management – управление мобильными устройствами) является составной частью стратегии обеспечения *безопасности конфиденциальной информации при использовании мобильных устройств*.

Технология управления жизненным циклом платформы включает функционал учета используемых устройств, управления конфигурациями ОС, управления мобильными приложениями (в том

числе – их инициализация и деинициализация, удаленная очистка, удаленный мониторинг и контроль за сбойными ситуациями). Обычно эти задачи реализуются через установку на мобильное устройство соответствующих MDM-профилей.

Такие системы позволяют обеспечивать контроль над мобильными устройствами, имеющими доступ к корпоративным сервисам организации, и способствуют снижению рисков, связанных с утечкой конфиденциальной информации с данных устройств.

Эта технология реализована, например, в приложении *AirWatch.*

Защита данных и каналов связи с Интернетом от воздействия *интрузов* (нежелательных программ или физических лиц). Является по определению результатом известного компромисса. Этот компромисс базируется на важнейшем и универсальном подходе к разработке и реализации политики защиты: защита информации только тогда является оправданной и разумной, когда стоимость реализации политики безопасности, по крайней мере, не меньше стоимости потерь, вызванных несанкционированным ее использованием.

12.6. Программно-аппаратные методы и средства обеспечения сетевой безопасности

Для защиты сетей от внешних угроз могут применяться следующие основные методы и средства:

– *порты высокой надежности*, *криптографическое шифрование данных*;

– эффективные *антивирусы и сканеры сигнатур вирусов*;

– программный или аппаратный сетевой экран (*брандмауэр*);

– *блокираторы руткитов и снифферов*.

Основная ***сложность борьбы с руткитами*** в том, что они активно противодействуют своему обнаружению, пряча свои файлы и ключи реестра от сканирующих программ, а также применяя другие методики. Руткиты – наиболее сложная в обнаружении и удалении разновидность Malware.

Существуют утилиты, специально созданные для поиска известных и неизвестных руткитов разными узкоспециальными методами, а также с помощью сигнатурного и поведенческого анализа. Удаление руткита – тоже сложный и многоэтапный процесс, который редко сводится к удалению пары файлов. Обычно приходится применять специальную программу, такую как *TDSSkiller*, созданную для борьбы с руткитом *TDSS*. В некоторых случаях жертве даже приходится переустанавливать операционную систему, если в результате заражения компьютерные файлы повреждены слишком глубоко.

Для менее сложных и вредоносных руткитов удаление может быть осуществлено с помощью обычной функции лечения в *Kaspersky Internet Security*.

Смягчить угрозу сниффинга пакетов можно с помощью следующих средств.

1. *Аутентификация*. Сильные средства аутентификации являются первым способом защиты от сниффинга пакетов. Под «сильным» понимается такой метод аутентификации, который трудно обойти. Примером такой аутентификации являются *однократные пароли* (One-Time Passwords, OTP).

OTP – это технология ***двухфакторной аутентификации***. Типичным примером двухфакторной аутентификации является работа обычного банкомата, который опознает клиента, во-первых, по пластиковой карточке и, во-вторых, по вводимому ПИН-коду. Для аутентификации в системе OTP также требуется ПИН-код и личная карточка. Снифферы, перехватывающие другую информацию (например, сообщения электронной почты), не теряют своей эффективности.

2. *Коммутируемая инфраструктура*. Еще одним способом борьбы со сниффингом пакетов в сетевой среде является создание коммутируемой инфраструктуры. Если, к примеру, во всей организации используется коммутируемый Ethernet, хакеры могут получить доступ только к трафику, поступающему на тот порт, к которому они подключены. Коммутируемая инфраструктура не ликвидирует угрозу сниффинга, но заметно снижает ее остроту.

3. *Антиснифферы*. Третий способ борьбы со сниффингом заключается в установке аппаратных или программных средств, распознающих снифферы, работающие в сети. Эти средства не могут полностью ликвидировать угрозу, но, как и многие другие средства

сетевой безопасности, включаются в общую систему защиты. Так называемые «антиснифферы» измеряют время реагирования хостов и определяют, не приходится ли хостам обрабатывать «лишний» трафик. Подобного рода средства не могут полностью ликвидировать угрозу сниффинга, но крайне необходимы при построении комплексной системы защиты.

4. *Криптография*. Самый эффективный способ борьбы со сниффингом пакетов не предотвращает перехвата и не распознает работу снифферов, однако делает эту работу бесполезной. Если канал связи является криптографически защищенным, это значит, что хакер перехватывает не сообщение, а зашифрованный текст. Например, криптография Cisco на сетевом уровне базируется на протоколе *IPSec*. Данный протокол представляет собой стандартный метод защищенной связи между устройствами с помощью протокола IP. К прочим криптографическим протоколам сетевого управления относятся протоколы *SSH* (Secure Shell) и *SSL* (Secure Socket Layer).

Полностью устранить угрозу спуфинга практически невозможно, но ее можно ослабить с помощью следующих мер.

1. *Контроль доступа*. Самый простой способ предотвращения IP-спуфинга состоит в правильной настройке управления доступом. Чтобы снизить эффективность IP-спуфинга, необходимо настроить контроль доступа на отсечение любого трафика, поступающего из внешней сети с исходным адресом, который должен располагаться внутри защищаемой сети. Заметим, что это помогает бороться с IP-спуфингом, когда санкционированными являются только внутренние адреса. Если санкционированными являются и некоторые адреса внешней сети, данный метод становится неэффективным.

2. *Фильтрация RFC 2827*. Можно пресечь попытки спуфинга чужих сетей пользователями некоторой сети. Для этого необходимо отбраковывать любой исходящий трафик, исходный адрес которого не является одним из IP-адресов данной организации. В результате отбраковывается весь трафик, который не имеет исходного адреса, ожидаемого на определенном интерфейсе.

3. *Криптография*. Наиболее эффективный метод борьбы с IP-спуфингом тот же, что и в случае со сниффингом пакетов: необходимо сделать атаку абсолютно неэффективной. IP-спуфинг может функционировать только при условии, что аутентификация происходит на базе IP-адресов.

Угроза атак типа DoS может снижаться тремя способами.

1. *Функции антиспуфинга*. Правильная конфигурация функций антиспуфинга на маршрутизаторах и межсетевых экранах помогает снизить риск DoS-атак. Эти функции, как минимум, должны включать фильтрацию RFC 2827.

2. *Функции антиDoS*. Правильная конфигурация функций антиDoS на маршрутизаторах и межсетевых экранах может ограничить эффективность атак за счет снижения числа полуоткрытых каналов в любой момент времени.

3. *Ограничение объема трафика* (Traffic Rate Limiting). Организация может попросить провайдера ограничить объем трафика. Это происходит в результате уменьшения объема некритического трафика, проходящего по сети. Обычным примером является ограничение объемов трафика ICMP, который используется только для диагностических целей. Атаки DDoS часто используют ICMP.

Парольных атак можно избежать, если не пользоваться паролями в текстовой форме. Одноразовые пароли и/или криптографическая аутентификация могут практически свести на нет угрозу таких атак. К сожалению, не все приложения, хосты и устройства поддерживают указанные выше методы аутентификации.

С точки зрения администратора, существует несколько методов борьбы с подбором паролей. Один из них заключается в использовании средства *L0phtCrack*, которое часто применяют хакеры для подбора паролей в среде Windows NT. Данное средство быстро показывает, легко ли подобрать пароль, выбранный пользователем.

Эффективно бороться с атаками типа Man-in-the-Middle можно только с помощью криптографии.

Исключить полностью атаки на уровне приложений невозможно. Хакеры постоянно открывают и публикуют в Интернете все новые уязвимые места прикладных программ. Самое главное здесь – хорошее системное администрирование. Вот некоторые меры, которые можно предпринять, чтобы снизить уязвимость для атак этого типа:

– чтение *лог-файлов* операционных систем и сетевых лог-файлов и/или их анализ с помощью специальных аналитических приложений;

– подписка на услуги по рассылке данных о слабых местах прикладных программ;

– использование последних версий операционных систем и приложений и самых последних коррекционных модулей (патчей);

– кроме системного администрирования необходимо использование *систем распознавания вторжений* или *атак* (Intrusion Detection System, IDS); существуют две взаимно дополняющие друг друга технологии IDS: первая – *сетевая система IDS* (NIDS), которая отслеживает все пакеты, проходящие через определенный домен; когда система NIDS видит пакет или серию пакетов, совпадающих с сигнатурой известной или вероятной атаки, она генерирует сигнал тревоги и/или прекращает сессию; вторая – *хост-система IDS* (HIDS), защищающая хост с помощью программных агентов; эта система борется только с атаками против одного хоста;

– в своей работе системы IDS пользуются *сигнатурами атак*, которые представляют собой профили конкретных атак или типов атак; сигнатуры определяют условия, при которых трафик считается хакерским.

Основным ***способом борьбы с переадресацией портов*** является использование надежных моделей доверия.

При рассмотрении условий обеспечения безопасности компьютерных сетей часто возникает вопрос о том, к какому уровню стека протоколов относится система сетевой безопасности? Ответ является очевидным.

На каждом из уровней протоколов могут осуществляться мероприятия по защите компьютерной сети, но каждый из этих уровней, что вполне естественно, использует свои специфические методы.

На *уровне передачи данных* пакеты, которые передаются по двухточечной линии, могут кодироваться при передаче в канал и декодироваться при приеме. Все детали этих преобразований могут быть известны только уровню передачи данных, причем более высокие уровни могут даже не «догадываться» о том, что именно происходит. Такой метод защиты называется ***шифрованием в канале связи***.

На *сетевом уровне* могут быть установлены *брэндмауэры*, позволяющие отвергать подозрительные пакеты. На этом же уровне может быть использована IP-защита.

На *транспортном уровне* можно зашифровать соединения целиком, от одного конца до другого, поскольку максимальную защиту может обеспечить только сквозное шифрование. Проблемы аутентификации и обеспечения строгого выполнения обязательств могут решаться только на *прикладном уровне*.

На всех уровнях стека протоколов, за исключением физического, решение проблем защиты информации базируется на принципах криптографии.

Выводы

1. Безопасность компьютерных сетей затрагивает широкий спектр вопросов, касающихся надежной передачи информации, обеспечения конфиденциальности информации, защиты от различных атак с целью повреждения или хищения информации.

2. Абсолютно защищенные персональный компьютер или компьютерная сеть, операционная система, прикладная программа считаются такой же иллюзией, как и абсолютно надежно охраняемый дом.

3. *Компьютерная преступность* (или «связанная с компьютерами преступность») – любые незаконные, неэтичные или неправомерные действия, связанные с автоматической обработкой или передачей информации.

4. *Сетевая атака* – это вторжение в операционную систему удаленного компьютера. Целями сетевой атаки являются: захват управления над операционной системой, приведение ее к состоянию отказа в обслуживании (Denial-of-Service, DoS Attack) или получение доступа к защищенной информации.

5. Для проверки целостности информации, а также исправления ошибок в сетях используют *методы помехоустойчивого кодирования*, базирующиеся на введении в информацию дополнительных избыточных символов. В зависимости от принципа вычисления дополнительных символов и их числа реализуются различные алгоритмы помехоустойчивого кодирования.

Контрольные вопросы

1. В чем состоит суть и как обеспечивается информационная безопасность компьютерных сетей и сетевых технологий?

2. Дайте определение основных понятий в предметной области, анализируемой в данной главе.

3. Охарактеризуйте основные виды атак на копьютерные сети.

4. Как можно оценить надежность компьютерной сети? Приведите примеры.

5. В чем заключаются методы помехоустойчивого кодирования информации?

6. Как задаются и как связаны между собой проверочная и порождающая матрицы избыточного кода?

7. Объясните различие между кодами Хемминга и циклическими кодами. Приведите примеры их использования в сетевых протоколах?

8. Покажите на примере обнаружение и исправление ошибки кодом Хемминга при длине информационного слова 1 байт. То же – циклическим кодом.

9. Постройте проверочную матрицу кода Хемминга для $k = 4$.

10. Дайте понятие синдрома и поясните его суть на примере.

11. Покажите пример кодирования сообщения $X_k = 10100011$ кодом Хемминга. Как будет исправлена ошибка на приемной стороне в 7-м бите?

12. Покажите пример кодирования сообщения $X_k = 1010$ циклическим кодом (порождающий полином выбрать самостоятельно). Как будет исправлена ошибка на приемной стороне в 3-м бите?

13. Опишите сниффинг как метод атаки.

14. Перечислите методы защиты от сниффинга пакетов.

15. Что такое IP-спуфинг? Опишите методы снижения угроз IP-спуфинга.

16. Опишите понятие DoS-атаки. В чем ее сущность?

17. Приведите разновидности DoS-атак.

18. Поясните суть парольных атак.

13

КРИПТОГРАФИЧЕСКИЕ И СТЕГАНОГРАФИЧЕСКИЕ МЕТОДЫ В СЕТЕВЫХ ТЕХНОЛОГИЯХ

13.1. Принципы криптографической защиты информации

Основной и практически непреодолимый «рубеж» защиты от несанкционированного доступа в компьютерных системах и в сетях образует шифрование.

Шифрование (Encryption) – один из способов преобразования данных на основе криптографии, обеспечивающий конфиденциальность и аутентичность информации.

В проблематике современной криптографии можно выделить следующие три типа основных задач:

– обеспечение *конфиденциальности*;

– создание условий для *анонимности* (неотслеживаемости);

– обеспечение *аутентификации* информации и источника сообщения.

Первый тип задач относится к защите информации от несанкционированного доступа по секретному ключу. Доступ к информации (информационным ресурсам) имеют только обладатели ключа. Второй и третий типы задач обязаны своей постановкой массовому применению электронных способов обработки и передачи информации (банковская сфера, электронная коммерция, каналы межличностной коммуникации и др.).

Криптографическое преобразование состоит из двух этапов: прямого и обратного. Прямое преобразование называют *зашифрованием* (в соответствии со стандартом ISO 7492-2, *шифрованием*, *encrypt*), обратное – *расшифрованием* (*дешифрованием*, *decrypt*).

С точки зрения криптографии **шифр**, или **криптографическая система** (Cryptographic System), – совокупность обратимых преобразований множества открытых данных на множество зашифрованных данных, задаваемых *ключом* и *алгоритмом криптографического преобразования*.

Следует различать понятия *ключ* и *пароль*.

Пароль (Password) является секретной последовательностью букв алфавита, однако используется не для шифрования (как ключ), а для аутентификации субъектов.

В *симметричных криптосистемах* для зашифрования и расшифрования используется один и тот же ключ.

В *ассиметричных криптосистемах* используются два ключа – открытый (публичный) и закрытый (секретный, тайный), которые математически связаны друг с другом и принадлежат одному физическому или юридическому лицу (в редких исключениях – группе лиц, например, реализующих один проект).

Электронной цифровой подписью (Electronic Digital Signature или E-Signature) называется присоединяемое к тексту его криптографическое преобразование, которое позволяет при получении текста другим пользователем проверить авторство и целостность сообщения.

Криптостойкостью (Crypto Resistance) называется характеристика шифра, определяющая его стойкость к расшифрованию без знания ключа, т. е. к криптоатаке.

Удачную (успешную) *криптоатаку* называют **взломом**.

13.1.1. Симметричные криптосистемы

Стандартные методы шифрования информации, передаваемой по сетям для повышения степени устойчивости к несанкционированному использованию, реализуют несколько этапов (шагов) шифрования, на каждом из которых используются различные «классические» методы шифрования.

К числу известных *симметричных криптосистем* можно отнести стандарт шифрования США DES, алгоритм IDEA, отечественный ГОСТ28147–89 и др.

Достаточно надежным считается алгоритм *IDEA* (International Data Encryption Algorithm), разработанный в Швейцарии и считающийся блочным шифром. Алгоритм также оперирует 64-битными блоками открытого текста. Несомненным достоинством IDEA является то, что его ключ имеет длину 128 битов. Один и тот же алгоритм используется и для зашифрования, и для расшифрования.

В алгоритме IDEA используются следующие математические операции:

– поразрядное сложение по модулю 2 («исключающее ИЛИ»);

– сложение беззнаковых целых по модулю 2^{16} (модуль 65536);

– умножение целых по модулю $2^{16} + 1$ (модуль 65537), рассматриваемых как беззнаковые целые, за исключением того, что блок из 16 нулей рассматривается как 2^{16}.

Все перечисленные операции выполняются над 16-битовыми субблоками. Комбинирование этих операций обеспечивает комплексное преобразование входа, существенно затрудняя криптоанализ IDEA по сравнению с DES, который базируется только на операции «исключающее ИЛИ».

К достоинствам симметричных методов шифрования относится высокая скорость шифрования и дешифрования, к недостаткам – малая степень защиты в случае, если ключ стал доступен третьему лицу.

13.1.2. Ассиметричные криптосистемы

13.1.2.1. Алгоритм RSA. *Криптосистема с открытым ключом* определяется тремя алгоритмами: *генерации ключей*, *зашифрования* и *расшифрования*. Алгоритм генерации ключей открыт, и каждый может дать ему на вход строку надлежащей длины и получить пару ключей.

Рассмотрим ассиметричные криптосистемы на примере *алгоритма RSA*. Названный в честь трех изобретателей Рона Ривеста (Ron Rivest), Ади Шамира (Adi Shamir) и Леонарда Адлемана (Leonard Adleman), этот алгоритм многие годы противостоит многочисленным попыткам криптоаналитического вскрытия.

Безопасность алгоритма основана на трудоемкости разложения на множители больших чисел. Открытый и закрытый ключи являются

функциями двух больших простых чисел разрядностью 100–200 десятичных цифр. Предполагается, что *восстановление открытого текста по шифртексту и открытому ключу равносильно разложению числа на два больших простых множителя*.

Ключ состоит из тройки больших целых чисел: e, d, n. Пара чисел (*e* и *n*) является не секретной и образует *публичный* (*открытый*) *ключ*. Число *d* является секретным, и пара (*d* и *n*) образует *тайный ключ*, известный только данному пользователю. Проблема верификации пользователей на основе их открытых ключей является одной из важных.

Основные операции алгоритма RSA.

1. Для генерации двух ключей применяются два больших случайных простых числа: *p* и *q*. Для большей криптостойкости алгоритма эти числа должны иметь равную длину.

2. Рассчитывается произведение $n = p \cdot q$ и вычисляется функция $\varphi(n) = (p - 1) \cdot (q - 1)$, которая называется функцией Эйлера и указывает количество положительных целых чисел в интервале от 1 до *N*, которые взаимно просты с *N*.

3. Случайным образом выбирается такое число *e*, которое вместе с $\varphi(n)$ являются взаимно простыми числами.

4. С помощью *расширенного алгоритма Евклида* вычисляется такое число *d*, что $e \cdot d = 1 \bmod \varphi(n)$, другими словами $d = e^{-1} \bmod \varphi(n)$.

Подразумевается, что эти шаги выполняет лицо, которое генерирует для себя (или для другого лица по его просьбе) соответствующие ключи.

Отметим, что числа *d* и *n* также являются взаимно простыми. Открытый и закрытый ключи составляют вышеуказанные пары чисел: *e* и *n*, *d* и *n* соответственно.

Зашифрование сообщения (открытым ключом получателя) $M = m_1 m_2 \ldots m_k$ (сообщение делится на *k* блоков) выглядит просто:

$$c_i = (m_i)^e \bmod n.$$

При расшифровании каждого блока c_i сообщения *C* (тайным ключом получателя) производится следующая операция:

$$m_i = (c_i)^d \bmod n.$$

Точно также сообщение может быть зашифровано с помощью пары *d* и *n* и расшифровано с помощью чисел *e* и *n*. Именно такой подход используется в системах электронной цифровой подписи.

Пример 13.1. Пусть сообщение M представляется числом 688 232. Предполагаем, что длина ключа составляет три десятичных цифры. Проиллюстрируем использование алгоритма RSA для передачи зашифрованного сообщения.

Выбираем числа $p = 47$ и $q = 71$. Имеем $n = p \cdot q = 47 \cdot 71 = 3337$. Вычисляем $\varphi(n) = (p - 1) \cdot (q - 1) = 46 \cdot 70 = 3220$.

Число e не должно иметь общих сомножителей с числом 3220; выбираем (случайным образом) e равным 79.

Вычисляем d: $d = 79^{-1} \bmod 3220 = 1019$.

Имеем открытый ключ – числа 79 и 3337 (его можно разместить в общедоступных источниках, например в *Центре сертификации*), и закрытый ключ – числа 1019 и 3337 (как видим, секретным является только число d; в нашем случае – это число 1019).

Для зашифрования сообщения M разбиваем его на блоки длиной, равной длине ключа, т. е. по 3 символа: $m_1 = 688$, $m_2 = 232$. Первый блок шифруется как $688^{79} \bmod 3337 = 1570 = c_1$; второй блок – $232^{79} \bmod 3337 = 2756 = c_2$.

Отправитель зашифрованного сообщения выполняет процедуру *зашифрования*, используя публичный ключ получателя.

Шифртекст C сообщения M выглядит следующим образом: $C = 15\ 702\ 756$.

Для обратного преобразования нужно выполнить похожие операции, однако с использованием числа 1019 в качестве степени:

$$m_1 = 1570^{1019} \bmod 3337 = 688;$$

$$m_2 = 2756^{1019} \bmod 3337 = 232.$$

Получатель зашифрованного сообщения выполняет процедуру *расшифрования*, используя собственный тайный ключ.

Несимметричные методы шифрования имеют преимущества и недостатки, противоположные тем, которыми обладают *симметричные методы*. В отличие от *симметричных методов шифрования*, проблема рассылки ключей в *несимметричных методах* решается проще – пары ключей (открытый и закрытый) генерируются «на месте» с помощью специальных программ. Для рассылки открытых

ключей используются такие технологии как *LDAP* (Light-weight Directory Access Protocol, протокол облегченного доступа к справочнику). Рассылаемые ключи могут быть предварительно зашифрованы с помощью одного из *симметричных методов шифрования*.

13.1.2.2. Управление ключевой информацией. Отметим также, что помимо выбора подходящей для конкретной информации системы средств криптографической защиты информации, важной проблемой является *управление ключами*. Как бы ни была сложна и надежна сама криптосистема, она основана на использовании ключей.

Управление ключами – информационный процесс, включающий в себя три элемента:
– генерацию ключей;
– накопление ключей;
– распределение ключей.

Для получения надежных криптографических ключей используются специальные аппаратные и программные методы ***генерации случайных значений ключей***.

Как правило, применяют *датчики псевдослучайных чисел* (ПСЧ). Однако степень случайности генерации чисел должна быть достаточно высокой. Идеальными генераторами являются устройства на основе «натуральных» случайных процессоров, например на основе *белого радиошума*.

Под **накоплением ключей** понимается организация их хранения, учета и удаления.

Поскольку ключ является самым привлекательным для злоумышленника объектом, открывающим ему путь к конфиденциальной информации, то вопросам накопления ключей следует уделять особое внимание. *Секретные ключи никогда не должны записываться в явном виде на носителе, который может быть считан или скопирован*.

Таким образом, вся информация о ключах должна храниться в зашифрованном виде. Ключи, зашифровывающие ключевую информацию, называются *мастер-ключами*.

Распределение ключей – самый ответственный процесс в управлении ключами. К нему предъявляются следующие требования: оперативность и точность распределения, скрытность распределяемых ключей.

Распределение ключей между пользователями компьютерной сети реализуется двумя способами:

1) использованием одного или нескольких центров распределения ключей;

2) прямым обменом *сеансовыми ключами* между пользователями сети.

Задача распределения ключей сводится к построению *протокола распределения ключей*, обеспечивающего:

– взаимное *подтверждение подлинности участников сеанса*;

– подтверждение *достоверности сеанса* механизмом запроса-ответа или отметки времени;

– использование минимального числа сообщений при обмене ключами;

– возможность исключения злоупотреблений со стороны *центра распределения ключей*.

Криптография с использованием открытых ключей позволяет передавать секретные данные, не обладая общим ключом заранее, а также создавать ***электронные подписи*** (***или электронные цифровые подписи***, ЭЦП) сообщений без необходимости привлечения третьей, *доверительной стороны*. Отрытый ключ можно хранить в открытом, общедоступном источнике. Однако при этом нужно иметь в виду возможность подделки ключа злоумышленником. Чтобы исключить эту возможность, в процесс обмена ключевой информацией вовлекается *центр распределения ключей* или *центр сертификации.*

Обязанность сертификации открытых ключей, принадлежащих как физическим, так и юридическим лицам выполняют организации, называемые **управлением сертификации** (CA – Certification Authority).

CA выдает (обычно с подписанием соответствующего двустороннего договора) физическим и юридическим лицам документ –

сертификат, который связывает открытый и закрытый ключи владельца с указанием данных этого владельца.

Для упрощения процедур получения сертификатов был разработан и утвержден известной организацией ITU специальный стандарт. Он называется **X.509** и широко применяется в Интернете. Начиная с 1988 г. вышло несколько версий этого стандарта. Данный документ X.509 связан с OSI.

Понятно, что один или даже несколько CA на весь мир не в состоянии обслужить всех клиентов. В связи с этим разработан альтернативный способ сертификации открытых ключей. Он известен под общим названием **PKI** (Public Key Infrastructure – *инфраструктура систем с открытыми ключами*).

PKI состоит из множества компонентов: *пользователи*, *управления сертификации*, сами *сертификаты*, а также *каталоги*. Инфраструктура PKI предоставляет возможность структурной организации этих компонентов и определяет стандарты, касающиеся различных документов и протоколов. Одним из простейших видов PKI является *иерархия управлений сертификаций*.

Управление сертификации верхнего уровня (root) называют ***центральным управлением*** (ЦУ). Центральное управление сертифицирует управления второго уровня – *региональные отделы, РО* (Regional Authorities, RA), так как они могут обслуживать некоторый географический регион, например страну или континент. Региональные отделы, в свою очередь, занимаются легализацией *низовых управлений сертификации* (НУС), эмитирующих сертификаты стандарта X.509 для физических и юридических лиц.

Поскольку все сертификаты подписаны (сверху донизу), на любом уровне можно выявить любые попытки подлога.

Цепочка сертификатов, восходящая к ЦУ, называется ***доверительной цепочкой*** (Chain of Trust) или ***путем сертификата*** (Certification Path). Описанный метод широко применяется на практике. В том числе – в криптовалютных технологиях.

13.1.2.3. Протокол Kerberos. Рассмотрим в качестве примера *протокол аутентификации и распределения ключей Kerberos* (по-русски – Цербер).

Протокол Kerberos спроектирован для работы в сетях TCP/IP и предполагает участие в аутентификации и распределении ключей третьей доверенной стороны. Kerberos обеспечивает надежную аутентификацию в сети, разрешая законному пользователю доступ к различным машинам в сети. Протокол основывается на симметричной криптографии (реализован алгоритм DES, хотя возможно применение и других симметричных криптоалгоритмов). Kerberos разделяет отдельный секретный ключ с каждым субъектом сети. Знание такого секретного ключа равносильно доказательству подлинности субъекта сети.

Основной протокол Kerberos является вариантом протокола аутентификации и распределения ключей Нидхема – Шрёдера.

В основном протоколе Kerberos (версия 5) участвуют две взаимодействующие стороны A и B и доверенный сервер KS (Kerberos Server). Стороны A и B, каждая по отдельности, разделяют свой секретный ключ с сервером KS. Доверенный сервер KS выполняет роль *центра распределения ключей* (*ЦРК*).

Пусть сторона A хочет получить сеансовый ключ для информационного обмена со стороной B.

Сторона A инициирует фазу распределения ключей, посылая по сети серверу KS идентификаторы участников сеанса Id_A и Id_B: $A \rightarrow KS$: Id_A, Id_B. Сервер KS генерирует сообщение с временной отметкой T, сроком действия L, случайным сеансовым ключом K и идентификатором Id_A. Он шифрует это сообщение секретным ключом, который разделяет со стороной B.

Затем сервер KS берет временную отметку T, срок действия L, сеансовый ключ K, идентификатор Id_B стороны B и шифрует все это секретным ключом, который разделяет со стороной A. Оба эти зашифрованные сообщения он отправляет стороне A: $KS \rightarrow A$: $E_A(T, L, K, Id_B)$, E_B (T, L, K, Id_A).

Сторона A расшифровывает первое сообщение своим секретным ключом, проверяет отметку времени T, чтобы убедиться, что это сообщение не является повторением предыдущей процедуры распределения ключей.

Затем сторона A генерирует сообщение со своим идентификатором Id_A и отметкой времени T, шифрует его сеансовым ключом K и отправляет стороне B. Кроме того, A отправляет для B сообщение

от *KS*, зашифрованное ключом стороны *B*: $A \rightarrow B$: $E_K(Id_A, T)$, $E_B(T, L, K, Id_A)$.

Только сторона *B* может расшифровать сообщения. Сторона *B* получает отметку времени *T*, срок действия *L*, сеансовый ключ *K*, идентификатор Id_A. Затем сторона *B* расшифровывает сеансовым ключом *K* вторую часть сообщения. Совпадение значений *T* и Id_A в двух частях сообщения подтверждают подлинность *A* по отношению к *B*.

Для взаимного подтверждения подлинности сторона *B* создает сообщение, состоящее из отметки времени *T* плюс 1, шифрует его ключом *K* и отправляет стороне *A*: $B \rightarrow A$: $E_K(T + 1)$.

Если после расшифрования сообщения сторона *A* получает ожидаемый результат, то она знает, что на другом конце линии связи находится действительно *B*.

Этот протокол успешно работает, если часы каждого участника синхронизированы с часами сервера *KS*. В этом протоколе необходим обмен с *KS* для получения сеансового ключа каждый раз, когда *A* желает установить связь с *B*. Протокол обеспечивает надежное соединение абонентов *A* и *B* при условии, что ни один из ключей не скомпрометирован и сервер *KS* защищен.

Система Kerberos обеспечивает защиту сети от несанкционированного доступа, базируясь исключительно на программных реализациях, и предполагает многократное шифрование передаваемой по сети управляющей информации.

Система Kerberos имеет структуру типа «клиент-сервер» и состоит из клиентских частей *C*, установленных на все машины сети, и Kerberos-сервера (*KS*), расположенного на каком-либо компьютере.

Область действия системы Kerberos распространяется на тот участок сети, все пользователи которого зарегистрированы под своими именами и паролями в базе данных Kerberos-сервера.

13.2. Эффективность использования пароля для защиты информации

Наиболее часто применяемыми методами идентификации и аутентификации пользователей являются методы, основанные на использовании паролей.

Пароль представляет собой некоторую последовательность символов, сохраняемую в секрете и предъявляемую при обращении к компьютерной или иной информационной системе.

Для ввода пароля, как правило, используется штатная клавиатура компьютерных систем (КС); при этом в процессе ввода пароль не должен отображаться на экране монитора, а чтобы пользователь мог ориентироваться в количестве введенных символов, на экран выдаются специальные символы.

При составлении и хранении пароля пользователи должны придерживаться следующих рекомендаций:

– пароль должен запоминаться субъектом доступа;

– запись пароля (на бумажном или электронном носителе) значительно повышает вероятность его компрометации (нарушения конфиденциальности);

– легко запоминаемый пароль должен быть в то же время сложным для отгадывания; не рекомендуется использовать для этой цели имена, фамилии, даты рождения и т. п.; известно, что наиболее часто используются пароли типа «12345», «qwerty» и подобные;

– желательным является наличие в пароле парадоксального сочетания букв, слов и т. п., полученного, например, путем набора русских букв пароля на латинском регистре.

Рекомендуется составлять пароль из символов, входящих минимум в три множества: букв на нижнем регистре (*a*, *b* и т. д.), букв на верхнем регистре (*A*, *B* и т. д.), цифр и специальных символов (*!*, */*, *?*, *;* и т. д.).

Вероятность подбора пароля уменьшается также при увеличении его длины и времени задержки между разрешенными попытками повторного ввода неправильно введенного пароля. Ожидаемое время раскрытия пароля T_P можно вычислить по следующей приближенной формуле:

$$T_P = \frac{A^s \cdot t}{2}, \tag{13.1}$$

где A – число символов в алфавите, из которых составляется пароль (например, 26 символов латинского алфавита); s – длина пароля; $t = E / R$ – время, необходимое на попытку введения пароля; E –

число символов в сообщении, передаваемом в систему при попытке получить к ней доступ (включая пароль и служебные символы); R – скорость передачи символов пароля по сети, символов/мин.

Если пароль состоит только из заглавных и строчных букв, то A = 52, а если из всех возможных символов, доступных на клавиатуре, то A = 95 · (26 + 26 + 10 + 33).

Использование формулы (13.1) предполагает, что злоумышленник имеет возможность непрерывно осуществлять подбор пароля – на основе *лобовой атаки*. Например, если $A = 26$, $t = 2$ с и $s = 6$ символов, то ожидаемое время раскрытия T_P пароля приблизительно равно одному году. Если в данном примере после каждой неудачной попытки ввода пароля предусмотреть временную задержку в 10 с, то ожидаемое время раскрытия пароля увеличится в 5 раз.

Следует также отметить, что на безопасное время раскрытия пароля оказывает существенное влияние длина пароля S (в степенной зависимости). Так, если для трехсимвольного пароля, выбранного из 26-символьного алфавита, время T_P составит 3 месяца, то для четырехсимвольного – 65 лет.

Чтобы сократить время выполнения атакующих действий до получения положительного (для атакующей стороны) результата, используется *распараллеливание вычислений* (точно такие же подходы можно использовать и при взломе криптографических систем). При этом применяются два способа распараллеливания.

1. Работают параллельно k процессоров, при этом i-й ($1 \le i \le k$) процессор выполняет три одинаковые по времени операции:

– получение данных от $(i - 1)$-го процессора;

– выполнение операции анализа;

– передача данных $(i + 1)$-му процессору.

2. Основан на разбиении множества $\mathbf{\Pi}$ всех возможных паролей (ключей) на z непересекающихся подмножеств: $\mathbf{\Pi}_1$, $\mathbf{\Pi}_2$, …, $\mathbf{\Pi}_z$. Система, с k процессорами, созданная атакующей стороной, перебирает, например, варианты так, что i-я ($1 \le i \le k$) машина анализирует пароли из подмножества $\mathbf{\Pi}_i$. Здесь, вероятно, определенную сложность представляет разделение множества на подмножества.

Выбор необходимой длины пароля S можно производить исходя из заданной вероятности P того, что данный пароль может быть раскрыт посторонним лицом за время M. Если необходимо построить систему, в которой вероятность отгадывания правильного пароля незаконным пользователем будет меньше, чем заданная вероятность P,

то следует выбрать такое значение *s*, которое удовлетворяло бы *формуле Андерсена*:

$$A^s \geq \frac{4{,}32 \cdot 10^4 \cdot R \cdot M}{E \cdot P}, \qquad (13.2)$$

где M – период времени, в течение которого предпринимаются попытки раскрытия пароля (в месяцах при ежедневном 24-часовом тестировании).

Пример 13.2. Допустим, требуется, используя стандартный латинский алфавит, установить пароль такой длины, чтобы вероятность его отгадывания не превысила 0,001 после трехмесячного ($M = 3$ мес) систематического тестирования.

Если за одну попытку доступа посылается 20 символов ($E = 20$), а скорость их передачи $R = 600$ символов/мин, то по формуле Андерсена получаем:

$$\frac{4{,}32 \cdot 10^4 \cdot R \cdot M}{E \cdot P} = \frac{4{,}32 \cdot 10^4 \cdot 3 \cdot 10^3 \cdot 600}{20} = 3{,}888 \cdot 10^9.$$

Для $S = 6$: $26^S = 3{,}089 \cdot 10^8$, т. е. $<3{,}888 \cdot 10^9$.

Для $S = 7$: $26^S = 8{,}03 \cdot 10^9$, т. е. $>3{,}888 \cdot 10^9$.

Таким образом, при данных обстоятельствах следует выбрать длину пароля $S = 7$.

При существенном увеличении длины пароля он может быть разбит на две части: запоминаемую пользователем и вводимую вручную, а также размещенную в зашифрованном виде на специальном носителе (например, дискете, магнитной карте и т. д.) и считываемую специальным устройством.

Повышение стойкости системы защиты на этапе аутентификации можно достигнуть и увеличением числа символов алфавита, используемого при вводе пароля. Для этого при наборе символов пароля можно использовать несколько регистров клавиатуры, соответствующих, например, строчным и прописным латинским символам, а также строчным и прописным символам кириллицы.

На степень информационной безопасности при использовании простого парольного метода проверки подлинности пользователей большое влияние оказывают ограничения на минимальное и максимальное время действия каждого пароля. Чем чаще меняется пароль, тем выше его безопасность. Администратор службы безопасности

должен постоянно контролировать своевременность смены паролей пользователей.

Таким образом, для повышения надежности аутентификации пользователей следует, по возможности, использовать нетривиальные (уникальные) пароли и, кроме того, обеспечивать более частую их смену.

С этой точки зрения являются достаточно эффективными методы, основанные на использовании *динамически изменяющихся паролей*. При смене пароля осуществляется его функциональное преобразование, зависящее от динамически изменяющихся параметров, например суточного времени в часах, номера дня недели, месячной даты и т. д. Такая смена пароля производится либо периодически (ежедневно, каждые три дня или каждую неделю), либо при очередном обращении пользователя.

Об особенностях и безопасности парольной защиты много полезной информации можно найти в упоминавшихся книгах К. Митника.

13.3. Методы и средства защиты от удаленных атак через сеть Интернет

13.3.1. Межсетевые экраны

Помимо опасности утечки информации за пределы сети, имеется опасность проникновения вредной информации, такой как вирусы, черви. Часто эту инфекцию заносят беззаботные сотрудники, желающие поиграть в новую модную компьютерную игру.

> ***Шифрование не спасает от вирусов и хакеров***, способных проникнуть в локальную сеть. Помочь защитить сеть от нежелательного проникновения снаружи может установка межсетевых экранов.

Межсетевые экраны (или *брэндмауэры*, Firewall) способны решать ряд задач по отражению наиболее вероятных угроз для внутренних сетей.

Межсетевой экран призван обеспечить безопасный доступ к внешней сети и ограничить доступ внешних пользователей к внутренней сети (рис. 13.1).

Межсетевой экран (МЭ) – система межсетевой защиты, позволяющая разделить сеть на две или более частей и реализовать набор правил, определяющих условия прохождения пакетов с данными через границу из одной общей части сети в другую.

МЭ пропускает через себя весь трафик, принимая для каждого проходящего пакета решение – пропускать его или отбросить. Для того чтобы МЭ мог осуществить это, ему необходимо определить набор правил фильтрации.

Однако следует отметить, что ни один межсетевой экран не может гарантировать полной защиты внутренней сети при всех возможных обстоятельствах.

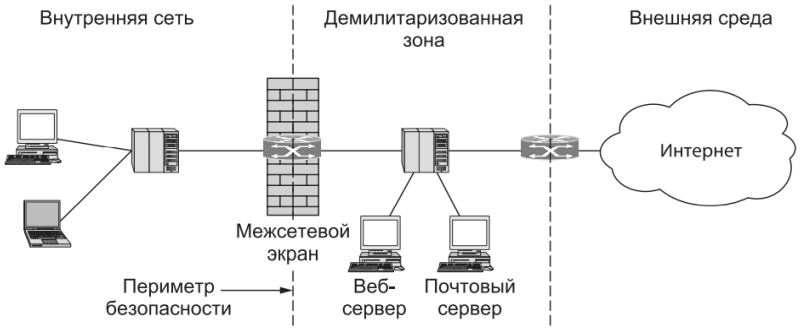


Рис. 13.1. Защита брандмауэром внутренней сети

Межсетевые экраны являются устройствами сетевого уровня, но также контролируют транспортный уровень и уровень приложений, чтобы производить фильтрацию.

Основными компонентами межсетевых экранов являются *фильтрующие маршрутизаторы*, *шлюзы сетевого уровня* и *шлюзы прикладного уровня*.

Персональный сетевой экран (Personal Firewall) выполняет функции, аналогичные сетевому (межсетевому) экрану: разграничение и контроль доступа к защищаемым ресурсам на уровне сетевых соединений.

Персональный сетевой экран, в отличие от межсетевых экранов, представляет собой программный агент, устанавливаемый непосредственно на защищаемый компьютер.

Персональный сетевой экран способен определить не только используемый протокол и сетевые адреса, но и программное обеспечение, устанавливающее или принимающее сетевое соединение, благодаря чему существенно повышаются возможности гранулированного и точного контроля и реализации дискретных политик. Специальный *режим обучения*, используемый для тонкой настройки персонального сетевого экрана под конкретную программную конфигурацию компьютера, позволяет свести к минимуму количество ошибок ложных срабатываний.

Основные функции персональных сетевых экранов:

– фильтрация сетевого трафика;

– анализ подозрительной активности при помощи встроенной системы обнаружения вторжений IDS;

– блокирование и удаление вредоносного программного обеспечения с помощью интегрированного антивируса.

Фильтрующий маршрутизатор представляет собой маршрутизатор или работающую на сервере программу, сконфигурированные таким образом, чтобы фильтровать входящие и исходящие пакеты.

Фильтрация пакетов осуществляется на основе информации, содержащейся в TCP- и IP-заголовках пакетов. Фильтрующий маршрутизатор обычно может фильтровать IP-пакеты на основе группы следующих полей заголовка пакета: IP-адрес отправителя, IP-адрес получателя, порт отправителя, порт получателя.

Шлюз сетевого уровня иногда называют *системой трансляции сетевых адресов* или *шлюзом сеансового уровня* модели OSI. Такой шлюз исключает прямое взаимодействие между *авторизованным клиентом* и *внешним хост-компьютером*.

Шлюз сетевого уровня принимает запрос доверенного клиента на конкретные услуги и после проверки допустимости запрошенного

сеанса устанавливает соединение с внешним хост-компьютером. После этого шлюз копирует пакеты в обоих направлениях, не осуществляя их фильтрации.

Для устранения ряда недостатков, которые присущи фильтрующим маршрутизаторам, межсетевые экраны должны использовать дополнительные программные средства для фильтрации сообщений сервисов типа TELNET и FTP. Такие программные средства называются *полномочными серверами* (серверами-посредниками), а *хост-компьютер*, на котором они выполняются, – *шлюзом прикладного уровня*.

Шлюз прикладного уровня исключает прямое взаимодействие между авторизованным клиентом и внешним хост-компьютером. Шлюз фильтрует все исходящие и входящие пакеты на прикладном уровне. Связанные с приложениями серверы-посредники перенаправляют через шлюз информацию, которая генерируется конкретными серверами.

Для достижения более высокого уровня безопасности и гибкости шлюзы прикладного уровня могут быть объединены с фильтрующими маршрутизаторами в одном межсетевом экране.

13.3.2. Программные методы защиты соединений

К *программным методам защиты* в сети Интернет могут быть отнесены *защищенные криптопротоколы*, которые позволяют надежно защищать соединения. К основным подходам и протоколам, обеспечивающим защиту соединений, относятся *SKIP-технология* и *протокол защиты соединения*, SSL.

13.3.2.1. SKIP-технология. Одной из технологий, предлагающей необходимые для применения в масштабах Internet универсальность и общность, является спецификация *SKIP* (Simple Key Management or Internet Protocol – простой протокол управления криптоключами в интерсети).

SKIP-технология – стандарт защиты трафика IP-пакетов, позволяющий на сетевом уровне обеспечить защиту соединения и передаваемых данных.

SKIP совместим с IP. Это достигается тем, что заголовок SKIP-пакета является стандартным IP-заголовком, и поэтому защищенный при помощи протокола SKIP пакет будет распространяться и маршрутизоваться стандартными устройствами любой TCP/IP-сети. Отсюда вытекает и аппаратная независимость SKIP. Протокол SKIP имплементируется в IP-стек выше аппаратно-зависимой его части и работает на тех же каналах, на которых работает IP.

Протокол *SKIP базируется на открытых криптографических ключах*. В качестве одного из возможных решений обеспечения защищенности ключа предлагаются рекомендации стандарта X.509. Однако кроме этого документа рассматриваются и другие представления сертификатов.

Существует два способа реализации SKIP-защиты трафика IP-пакетов:

– шифрование блока данных IP-пакета;

– инкапсуляция IP-пакета в SKIP-пакет.

SKIP-пакет похож на обычный IP-пакет. В поле данных SKIP-пакета полностью размещается в зашифрованном виде исходный IP-пакет. В этом случае в новом заголовке вместо истинных адресов могут быть помещены некоторые другие адреса. Такая структура SKIP-пакета позволяет беспрепятственно направлять его любому хост-компьютеру в сети Интернет, при этом межсетевая адресация осуществляется по обычному IP-заголовку в SKIP-пакете. Конечный получатель SKIP-пакета по заранее определенному разработчиками алгоритму расшифровывает криптограмму и формирует обычный TCP- или UDP-пакет, который и передает соответствующему модулю (TCP или UDP) ядра операционной системы.

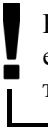
13.3.2.2. Протокол SSL/TLS. Соответствующее программное обеспечение, как и сам протокол, сегодня используется очень широко (в том числе программами Firefox, Safari, Internet Explorer).

SSL (Secure Sockets Layer – протокол защищенных сокетов) создает защищенное соединение между двумя сокетами, которое позволяет клиенту и серверу договориться об используемых параметрах, произвести взаимную аутентификацию, организовать тайное общение или обеспечить защиту целостности данных.

Идея SSL заключается в том, что, по сути дела, между прикладным и транспортным уровнями появляется новый уровень, принимающий запросы от браузера и отсылающий их по TCP для передачи серверу. Расположение SSL в структуре обычного стека протоколов показано на рис. 13.2.

Рис. 13.2. Расположение SSL
в структуре обычного стека протоколов

После установки защищенного соединения основная задача SSL заключается в поддержке сжатия и шифрования.



Если поверх SSL используется http, то этот вариант называется **HTTPS** (Secure http, защищенный HTTP), несмотря на то, что это обычный HTTP.

Код авторизации сообщения это зашифрованный отпечаток, вычисленный на основе содержимого сообщений. Для получения SSL-пакета каждая из сторон объединяет данные фрагмента, код авторизации сообщения, заголовки сообщения и шифруют это с использованием согласованного секретного ключа. При получении пакета каждая из сторон расшифровывает его и сверяет полученный код авторизации сообщения со своим. Если они не совпадают, то пакет был подделан, и наоборот.

Существует несколько версий протокола SSL. Протокол может обладать разными дополнительными функциями, среди которых наличие или отсутствие сжатия, тот или иной алгоритм шифрования.

SSL поддерживает разнообразные криптографические алгоритмы. Наиболее криптостойкий из них использует для шифрования *тройной DES* (Data Encription Standard) с тремя отдельными ключами и хеширование на основе алгоритмов SHA-1 (Secure Hash Algorithm) и MD5 для обеспечения целостности данных. Такое сочетание алгоритмов работает довольно медленно, поэтому применяется в основном при выполнении банковских операций и в других приложениях, в которых требуется высокий уровень защиты.

> ***SSL состоит из двух субпротоколов***, один из которых предназначен для установления защищенного соединения – *фаза рукопожатия* (Handshake), а второй – для использования этого соединения, т. е. для передачи данных (рис. 13.3).

Во время *рукопожатия* клиент и сервер используют шифрование открытым ключом для согласования секретного ключа, используемого сторонами при передаче данных, т. е. во второй фазе.

Клиент инициирует *рукопожатие*, посылая *hello* серверу со списком алгоритмов симметричного шифрования (Cipher Specs), поддерживаемых клиентом. Сервер отвечает похожим *hello*-сообщением с указанием наиболее подходящго алгоритма шифрования из полученного списка. Далее сервер отправляет сертификат, который содержит его публичный ключ.

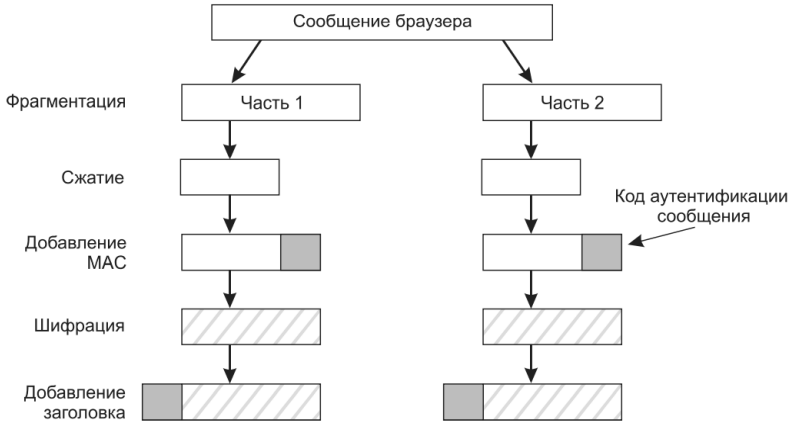


Рис. 13.3. Передача данных при использовании SSL

Сертификат – это набор данных, который подтверждает подлинность. Третья сторона, *центр сертификации* (CA), генерирует сертификат и проверяет его подлинность. Чтобы получить сертификат, сервер должен использовать безопасные каналы для отправки своего публичного ключа в центр сертификации. Он генерирует сертификат, который содержит его собственный ID, ID сервера, публичный ключ сервера и другую информацию. CA создает *отпечаток* (Digest) сертификата, который, по сути, является контрольной суммой. Далее CA создает подпись сертификата (Certificate Signature), которая формируется путем шифрования отпечатка сертификата тайным ключом CA.

Для проверки сертификата сервера клиент использует публичный ключ CA для расшифрования подписи. В дальнейшем клиент самостоятельно использует отпечаток сертификата сервера и сверяет его с расшифрованным (с помощью публичного ключа CA). Если они не совпадают, то принимается решение о том, что сертификат был подделан, и наоборот – при совпадении отпечатков сертификат является подлинным.

Клиент хранит у себя список публичных ключей, подтвержденных центром сертификации. Многие браузерные приложения имеют подобный список, находящийся непосредственно в их коде. Когда клиент установил подлинность сервера (который также может запросить сертификат у клиента), используется шифрование открытым ключом для согласования совместного *секретного ключа* при обмене информацией (как и в технологиях на основе симметричной криптографии).

Фаза рукопожатия завершается отправкой сообщений *finished*.

В *фазе передачи данных* (рис. 13.3) каждая сторона разбивает сообщения, поступающие от браузера, на блоки размером до 16 Кбайт. Если сжатие данных включено, каждый из этих блоков независимо сжимается. К ним прикрепляются коды авторизации сообщений, MAC (Message Authentication Code). Результат хешируется по согласованному алгоритму (обычно MD5). Хеш добавляется к каждому фрагменту в виде кода MAC.

Наконец, присоединяется заголовок фрагмента, и фрагмент передается по TCP-соединению.

Протокол SSL является универсальным средством, позволяющим динамически защищать соединение при использовании любого прикладного протокола (FTP, TELNET, SMTP, DNS и т. д.).

SSL сконфигурирован на уровне приложения, непосредственно над TCP, что позволяет более высокоуровневым протоколам (таким как HTTP или протоколу электронной почты) работать без изменений. Если SSL сконфигурирован корректно, то сторонний наблюдатель может узнать лишь параметры соединения (например, алгоритм шифрования), а также частоту пересылки и примерное количество данных, но не может читать и изменять их.

После того как протокол SSL был стандартизирован IETF (Internet Engineering Task Force), он был переименован в TLS (Transport Layer Security – протокол защиты транспортного уровня). Поэтому, хотя названия SSL и TLS взаимозаменяемы, они все-таки отличаются, так как каждое описывает разные версии протокола.

TLS был создан для работы над TCP, однако для работы с протоколами дейтаграмм, такими как UDP (User Datagram Protocol), была разработана специальная версия TLS, получившая название DTLS (Datagram Transport Layer Security).

Протокол TLS является двухуровневым и действует над транспортным протоколом. К первому уровню относятся *TLS Handshake Protocol*, *TLS Change Cipher Spec*, *TLS Alert Protocol*. Ко второму уровню относится *TLS Record Protocol*.

Стандартный порт для HTTP-трафика по протоколу SSL/TLS (HTTPS) – 443.

Протокол TLS предназначен для предоставления трех услуг всем приложениям, работающим над ним, а именно: *шифрования*, *аутентификации* и *целостности*. Технически не все три могут использоваться, однако на практике для обеспечения безопасности, как правило, используются все.

Для того чтобы установить криптографически безопасный канал данных, узлы соединения должны согласовать используемые методы шифрования и ключи.

Как можно было заметить, установление соединения SSL (фаза рукопожатия) является достаточно длительным и трудоемким процессом. Поэтому в TLS есть несколько оптимизаций. В частности, имеется процедура под названием *abbreviated handshake*, которая позволяет использовать ранее согласованные параметры для *восстановления соединения* (если стороны устанавливали TLS-соединение

в прошлом). Имеется также дополнительное расширение процедуры *handshake – TLS False Start*. Это расширение позволяет клиенту и серверу начать обмен зашифрованными данными сразу после согласования алгоритма шифрования, что сокращает процедуру на одну итерацию.

Чаще всего в TLS используется обмен ключами по алгоритму RSA. Этот алгоритм используется только в процедуре *TLS handshake* во время первоначальной настройки соединения. После завершения первой фазы, т. е. настройки *туннеля*, применяется симметричная криптография: собщение в пределах текущей сессии зашифровано/расшифровано именно установленными симметричными ключами (например, DES). Это необходимо для увеличения быстродействия, так как криптография с открытым ключом требует значительно больше вычислительной мощности.

SSL/TLS противостоит таким угрозам, как:
– *подмена идентификатора* клиента или сервера (с помощью надежной аутентификации);
– *раскрытие информации* (с помощью шифрования канала связи);
– *искажение данных* (с помощью кодов целостности сообщений).

13.3.2.3. Специализированные средства. К специализированным программным средствам защиты информации от несанкционированного доступа в компьютерных сетях также относятся так называемые ***proxy-серверы*** (Proxy-Servers, *proxy* – доверенное лицо, доверенность).

Идея использования proxy-сервера заключается в том, что весь трафик сетевого/транспортного уровней между локальной и глобальной сетями запрещается полностью – маршрутизация как таковая отсутствует, а обращения из локальной сети в глобальную происходят через специальные серверы-посредники.

Следует отметить, что, несмотря на все преимущества, этот метод не дает достаточной защиты против атак на более высоких уровнях, например на уровне приложения (вирусы, код Java и JavaScript).

Не следует забывать и о том, что встроенные средства защиты информации имеются также в сетевых операционных системах. Однако

они не всегда могут полностью решить возникающие на практике проблемы.

В качестве примера рассмотрим некоторые из названных систем.

Система **SFT** (System Fault Tolerance – система устойчивости к отказам) компании Novell включает три основных уровня.

1. Первый уровень (SFT Level I) предусматривает, в частности, создание дополнительных копий FAT и Directory Entries Tables, немедленную верификацию каждого вновь записанного на файловый сервер блока данных, а также резервирование на каждом жестком диске около 2% от объема диска. При обнаружении сбоя данные перенаправляются в зарезервированную область диска, а сбойный блок помечается как «плохой» и в дальнейшем не используется.

2. Второй уровень (SFT Level II) содержит дополнительные возможности создания «зеркальных» дисков, а также дублирования дисковых контроллеров, источников питания и интерфейсных кабелей.

3. Третий уровень (SFT Level III) позволяет применять в локальной сети дублированные серверы, один из которых является «главным», а второй, содержащий копию всей информации, вступает в работу в случае выхода «главного» сервера из строя.

Система контроля и ограничения прав доступа в сетях NetWare (защита от несанкционированного доступа) также содержит несколько уровней:

– *уровень начального доступа* (включает имя и пароль пользователя, систему учетных ограничений, таких как явное разрешение или запрещение работы, допустимое время работы в сети, место на жестком диске, занимаемое личными файлами данного пользователя, и т. д.);

– *уровень прав пользователей* (ограничения на выполнение отдельных операций и/или на работу данного пользователя как члена подразделения в определенных частях файловой системы сети);

– *уровень атрибутов каталогов и файлов* (ограничения на выполнение отдельных операций, в том числе удаления, редактирования или создания, идущих со стороны файловой системы и касающихся всех пользователей, пытающихся работать с данными каталогами или файлами);

– *уровень консоли файл-сервера* (блокирование клавиатуры файл-сервера на время отсутствия сетевого администратора до ввода им специального пароля).

Однако полагаться на данную часть системы защиты информации в ОС NetWare можно не всегда. Свидетельством тому являются

многочисленные инструкции в Интернете и готовые доступные программы, позволяющие взломать те или иные элементы защиты от несанкционированного доступа.

То же замечание справедливо по отношению к более поздним версиям ОС NetWare (вплоть до последней 6-й версии) и к другим «мощным» сетевым ОС со встроенными средствами защиты информации (Windows NT, UNIX). В связи с остротой проблемы защиты информации наблюдается тенденция интеграции (встраивания) отдельных, хорошо зарекомендовавших себя и ставших стандартными средств в сетевые ОС, а также разработка собственных «фирменных» аналогов известным программам защиты информации.

Так, в сетевой ОС NetWare 4.1 предусмотрена возможность кодирования данных по принципу «открытого ключа» (алгоритм RSA) с формированием электронной подписи для передаваемых по сети пакетов.

13.4. Безопасность беспроводных сетей и IoT

13.4.1. Безопасность Wi-Fi-сетей

С точки зрения практического использования беспроводных сетей очень актуальны вопросы безопасности и защиты передаваемых данных, так как для перехвата данных в общем случае достаточно просто оказаться в зоне действия сети.

Первоначально созданные в этой сфере технологии обладали невысокой степенью защиты, данная проблема остается актуальной и на сегодняшний день.

Для защиты передаваемых данных можно использовать разные методы (много полезной информации на этот счет содержится в упоминавшейся книге К. Митника «Искусство быть невидимым»). Далее проанализируем особенности практической реализации некоторых методов.

1. ***Использование MAC-адресов*** (Media Access Control ID). У каждого адаптера есть свой абсолютно уникальный код, установленный производителем. Эти адреса необходимо занести в списки адресов доступа у используемых для организации сети точек доступа. Все

остальные WLAN-адаптеры с неправильными адресами будут исключены из сети автоматически.

Важно отметить, что хакерский инструмент под названием *aircrack-ng* позволяет узнать авторизованный MAC-адрес подключенного к сети устройства. Затем хакер может сфальсифицировать этот MAC-адрес для подключения к маршрутизатору.

Кстати, узнать адрес своего устройства довольно просто. Для этого, например, в ОС Windows необходимо выполнить следующее:

1) нажать кнопку *Пуск*;
2) напечатать *cmd*;
3) выбрать пункт *Командная строка*;
4) в открывшемся окне оболочки ввести команду *getmac/v/fo list*.

В списке данных будет MAC-адрес.

Тот же поиск на устройствах фирмы Apple реализуется еще проще: в меню нужно выбрать пункт *Системные настройки* (*System Preferences*) – щелкнуть по значку *Сеть* (*Network*) – выбрать свой сетевой адаптер – выбрать MAC-адрес, нажав кнопку *Дополнительно* (*Advanced*).

Зная MAC-адреса, можно настроить маршрутизатор так, чтобы он предоставлял доступ к сети только этим устройствам и блокировал другие.

Чтобы можно было без труда подключить любое новое устройство к домашнему маршрутизатору, *альянс совметимости беспроводного оборудования* (Wi-Fi Alliance) разработал протокол WPS – Wireless Protected Setup – *защищенная настройка беспроводного соединения*. Этот протокол обычно реализован в виде кнопки на роутере (! формально любой человек, находящийся рядом с роутером, может нажать на эту кнопку), а также может быть представлен кодом, передаваемым посредством технологии NFC (Near Field Communication – *ближняя бесконтактная связь*).

Существуют офлайновые методы взлома WPS. Известный метод *PixieDust* позволяет хакеру получить доступ к беспроводному маршрутизатору.

WPS целесообразно отключать, а каждое новое мобильное устройство подключать вручную с использованием установленного пароля.

2. ***Использование ключей SSID*** (Service Set Identifier). Каждый легальный пользователь сети должен получить от администратора сети свой уникальный идентификатор SSID.

Все беспроводные маршрутизаторы по умолчанию транслируют данный идентификатор. Он включает в себя название модели маршрутизатора и выглядит, например, так: Linksys WTR12GL. Транслируя предустановленный идентификатор, можно скрыть, из какой именно точки исходит сигнал. Но при этом каждый желающий легко может узнать марку и модель используемого маршрутизатора. И если этот человек знает уязвимости (они свойственны практически любым устройствам и системам!) маршрутизатора, то может ими воспользоваться в своих интересах. Получить доступ к настройкам маршрутизатора можно, например, через веб-браузер или путем поиска сайтов, на которых указывается, что нужно напечатать в адресной строке браузера, чтобы подключиться напрямую к роутеру (см., например, по ссылке: routeripaddress.com/). Набрав локальный URL-адрес, нужно авторизоваться. Часто логином и паролем выступает фраза «admin». Можно попробовать другие распространенные варианты. Оказавшись в настройках маршрутизатора, нужно сменить установленный по умолчанию пароль.

Вопрос использования безопасных паролей – один из ключевых в решении проблемы защиты сетей, защиты данных от несанкционированного доступа.

3. ***Шифрование данных.*** Первые два способа не обеспечивают защиту от прослушивания и перехвата пакетов данных, поэтому защитить сеть в случае перехвата данных можно только с помощью шифрования.

Изначально стандарт 802.11 предусматривал аппаратный *протокол шифрования данных, WEP* (Wired Equivalent Privacy – защищенность, эквивалентная беспроводным сетям), основанный на *алгоритме шифрования RC4*. Однако в скором времени было обнаружено, что защищенную с его помощью сеть довольно легко взломать. Ранние версии предусматривали шифрование с использованием 40-битного *ключа*, более поздние – 64-, 128- или 256-битное шифрование. Но даже такая длина ключа в WEP не может обеспечить высокий уровень защиты сети, так как основная слабость данной технологии заключается в статичности ключа шифрования. Хотя при использовании данного ключа увеличивается время взлома и количество пакетов данных, которые нужно перехватить, чтобы

вычислить ключ, сама возможность взлома остается. Это абсолютно неприемлемо для определенного круга серьезных компаний и организаций.

> **!** Базовая защита передаваемых данных на основе протокола шифрования *WEP* считается бесполезной с точки зрения обеспечиваемого уровня криптостойкости, т. е. устойчивости к взлому.

На смену WEP была создана новая технология *WPA* (Wi-Fi Protected Access), разрабатывающаяся IEEE совместно с Wi-Fi Alliance.

> **!** Главной особенностью протокола WPA является *шифрование данных с динамически изменяемыми ключами* и *проверка аутентификации пользователей*. В отличие от WEP здесь используется *протокол целостности временных ключей*, *TKIP* (Temporal Key Integrity Protocol), который подразумевает обновление ключей перед началом каждой сессии шифрования и проверку пакетов на принадлежность к данной сессии. Протокол *WPA2* является наиболее надежным.

Для аутентификации пользователей применяются *сертификаты RADIUS* (Remote Authentication Dial-In User Service – сервер RADIUS должен подтвердить право доступа). Такой метод подразумевает, главным образом, корпоративное использование. Второй, упрощенный, вариант аутентификации требует предварительной установки разделяемых паролей на сетевые устройства (режим аутентификации PSK (Pre-Shared Keys)). Этот метод лучше всего применять в домашних условиях или там, где не происходит обмена важной информацией.

Изначально WPA была разработана как технология, которая со временем должна быть заменена новым стандартом 802.11i. Данный стандарт, с учетом всех уже существующих наработок, призван обеспечить надежное шифрование передаваемых данных, а также аутентификацию пользователей сети.

В большинстве уже выпущенных Wi-Fi-устройств (точки доступа, сетевые карты) можно установить протокол WPA посредством обновления программного обеспечения.

Когда в настройках маршрутизатора включено шифрование (по умолчанию – обычно отключено), то остальные устройства, которые подключаются к маршрутизатору, должны поддерживать этот режим. Большинство новых устройств автоматически поддерживают действующий режим шифрования, однако для выпущенных ранее это нужно проделывать вручную.

Таким образом, настроив WPA2 на маршрутизаторе, его нужно также настроить на ноутбуке или на мобильном устройстве. При этом следует иметь в виду, что отдельные ОС автоматически определяют используемый алгоритм шифрования.

Некоторые интернет-провайдеры по умолчанию встраивают в свои домашние маршрутизаторы функцию беспроводного доступа. Возможность беспроводного подключения создает очевидные удобства, но такие маршрутизаторы при отсутствии определенного уровня защиты могут стать источником проблем.

Дело в том, что мобильные устройства запоминают точки доступа Wi-Fi, к которым они (их пользователи) подключались. Может случиться так, что мобильное устройство поймало сигнал, совпадающий с профилем одного из недавних подключений. Это может быть не только точка доступа библиотеки или кафе, например, а фальшивая точка, которая была создана с помощью специальных приложений, например *WireShark* – известной программы для захвата и анализа сетевого трафика.

Некоторые мобильные средства выбирают сеть автоматически. По утверждениям, например, Apple, произведенные этой компанией мобильные средства осуществляют подключение в следующем порядке:

1) защищенная сеть, к которой устройство было подключено в последний раз;

2) другая защищенная сеть;

3) публичная незащищенная сеть.

Существуют возможности удалять из списка данные ненужных или нежелательных соединений.

В Windows надо для этого (с некоторыми вариациями в различных версиях) сбросить флажок *Подключаться автоматически* (*Connect Automatically*), который размещается рядом с именем точки доступа. Подобная цель может быть достигнута при инициализации цепочки операций: окно *Панель управления* (*Control Panel*) – выбрать пункт *Центр управления сетями с общим доступом* (*Network*

and Sharing Center) – щелкнуть мышью по названию-ссылке подключенной сети – нажать кнопку *Свойства беспроводной сети* (*Wireless Properties*) – сбросить флажок *Подключаться автоматически, если есть в радиусе действия* (*Connect automatically when this network in range*).

То же в macOS: приложение *Системные настройки* (*System Preferences*) – щелкнуть мышью по пункту *Сеть* (*Network*) – выбрать пункт *Wi-Fi* – нажать кнопку *Дополнительно* (*Advanced*) – сбросить флажок *Запоминать сети, к которым подключается компьютер* (*Remember network this computer has joined*).

В устройствах под управлением iOS и Android также можно «забыть» соответствующие Wi-Fi-соединения.

13.4.2. Особенности обеспечения безопасности IoT-сетей

Элементы IoT-сетей способны обмениваться данными без непосредственного участия человека. Превращение устройств в самостоятельные интернет-узлы привело к значительному снижению безопасности системы.

Безопасность IoT-устройств обеспечивается, прежде всего, сохранением целостности кода, проверкой подлинности пользователей и устройств, присвоением пользователям прав владения (в том числе генерируемыми данными), а также возможностью отражения виртуальных и физических атак.

Все «умные» устройства, подключенные к сети, передают через нее соответствующие их функционалу данные, которые являются мишенью для киберпреступников.

В первой половине 2019 г. специалисты из Лаборатории Касперского с помощью *ханипотов* (ресурс, представляющий собой приманку для злоумышленников) зафиксировали 105 млн атак на IoT-устройства, исходящих из 276 тыс. уникальных IP-адресов. Данный показатель в семь раз больше, чем в первой половине 2018 г., когда было обнаружено около 12 млн атак с 69 тыс. IP-адресов. Пользуясь слабой защитой IoT-продуктов, киберпреступники прикладывают много усилий для создания и монетизации *IoT-ботнетов*.

Количество кибератак на IoT-устройства стремительно увеличивается, поскольку все чаще пользователи и организации приобретают «умные» устройства, такие как маршрутизаторы или камеры видеорегистрации, но при этом не все заботятся об их защите. Киберпреступники, в свою очередь, видят все больше финансовых возможностей в использовании таких устройств. Они используют сети зараженных «умных» устройств для проведения DDoS-атак или в качестве прокси-сервера для других типов вредоносных действий.

К основным «слабым местам» IoT относятся:

– переход на IPv6;

– питание датчиков, камер и их сетевое подключение; известен случай проникновения в систему через датчик температуры, размещенный в аквариуме и даже через «умный» унитаз;

– проблемы стандартизации архитектуры и протоколов, сертификация устройств;

– отсутствие поддержки со стороны производителей для устранения уязвимостей;

– трудность или невозможность обновления ПО и ОС;

– использование текстовых протоколов и ненужных открытых портов;

– легкость попадания в сеть при слабости одного из гаджетов;

– использование недостаточно незащищенных мобильных технологий и облачной инфраструктуры;

– возможность создания крупных *ботнетов.*

> ***Поставщики услуг и устройств рынка IoT нарушают принцип сквозной информационной безопасности*** (ИБ), который рекомендован для всех ИКТ-продуктов и услуг. Согласно этому принципу, ИБ должна закладываться на начальной стадии проектирования продукта или услуги и поддерживаться вплоть до завершения их жизненного цикла.

Сети, построенные на совместимом через IPv4 оборудовании, могут быть не в состоянии полностью интегрировать IPv6 без замены своих последовательных терминальных серверов и серверов устройств ввода-вывода Ethernet.

Для компаний важно ускорить выпуск нового устройства на рынок. Некоторые производители предпочитают пожертвовать защищенностью ради получения преимущества перед конкурентами.

Многие компании и сегодня выпускают умные гаджеты, не вкладывая большие ресурсы денег и времени в тестирование кодов, доработку систем безопасности. По этой причине рынок растет очень быстро, технологии развиваются, но страдают пользователи.

Часто не обращается внимание на проблемы, возникающие как на стороне владельцев устройств, так и на те, над которыми должны задуматься разработчики.

> **!** В самом начале эксплуатации ***пользователю обязательно нужно заменить фабричный пароль***, который установлен по умолчанию, на свой личный, поскольку фабричные пароли одинаковы на всех устройствах и не отличаются стойкостью.

Боты позволяют злоумышленникам скрытно управлять зараженными устройствами. С появлением интернета вещей возникло больше возможностей для создания целых ботнетов, что связано с потерей физическими устройствами автономии – многие вещи перестали работать самостоятельно, они интегрируются в единую систему и не способны функционировать в отрыве от нее. А как уже упоминалось выше, процессы внутри IoT зачастую не контролируются.

Обязательным пунктом в обеспечении безопасности является защита устройств (вещей), которая во многом заключается в защите целостности кода. Необходима гарантия безопасности запуска кода – *криптографическая подпись*. Также требуется и специальная защита в процессе выполнения кода, позволяющая не допустить его переписывания различными хакерскими программами.

К сожалению, делают это далеко не все. Поскольку не все приборы имеют встроенные средства ИБ-защиты, владельцам также следует позаботиться об установке внешней защиты, предназначенной для домашнего использования, с тем, чтобы интернет-устройства не стали открытыми шлюзами в домашнюю сеть или прямыми инструментами причинения ущерба.

> **!** Одним из направлений решения рассматриваемого комплекса проблем с обеспечением безопасности IoT-сетей может стать использование ***технологии блокчейн***.

Эта технология позволяет сохранять протоколы обмена и результаты взаимодействия устройств IoT в децентрализованной системе. Распределенная архитектура блокчейна обеспечивает более высокую безопасность IoT-системы: даже если часть устройств будет подвержена взлому, это не скажется на работе системы в целом. Распределенная модель системы позволяет избавиться от взломанного устройства без ощутимого ущерба для взаимодействия между «здоровыми» объектами.

В контексте безопасности блокчейн может использоваться в сферах, где IoT развивается наиболее интенсивно. Например, это управление аутентификацией, проверка работоспособности разных сервисов, обеспечение неделимости информации и т. д. На сегодняшний день главная задача, которую поставили перед собой специалисты, – разработка на основе блокчейн-технологии распределенной базы данных и протокола обмена информацией между IoT-устройствами.

Выводы

1. Криптография представляет собой инструмент, который используется для обеспечения конфиденциальности информации и аутентичности.

2. Все алгоритмы шифрования можно разделить на два вида: с симметричными (закрытыми) ключами и с ассиметричными (открытыми) ключами. Алгоритмы с симметричными ключами искажают при шифровании значения битов последовательности итераций, параметризованных ключом. Наиболее популярными алгоритмами этого типа являются DES и AES. Алгоритмы с симметричным ключом могут работать в режиме электронного шифроблокнота, потокового шифра.

3. Алгоритмы с открытым ключом отличаются тем, что для шифрования и дешифрации используют разные ключи, причем ключ дешифрации невозможно вычислить по ключу шифрования. Эти свойства позволяют делать ключ открытым. Чаще всего применяется алгоритм RSA, основанный на сложности разложения больших чисел на простые сомножители.

4. Для противодействия различным атакам в сетях в основном используются межсетевые экраны и *proxy*-серверы, а также криптозащищенные протоколы (технология SKIP, протокол SSL).

5. Межсетевой экран – это система межсетевой зашиты, которая позволяет разделить сеть на две или более частей, а также реализовать набор правил, определяющих условия прохождения пакетов с данными через границу из одной общей части сети в другую. Межсетевые экраны пропускают через себя весь трафик, принимая для каждого проходящего пакета решение – пропустить его или отбросить.

6. Идея использования proxy-сервера заключается в том, что весь трафик сетевого/транспортного уровней между локальной и глобальной сетями запрещается полностью – маршрутизация как таковая отсутствует, а обращения из локальной сети в глобальную происходят через специальные серверы-посредники. Очевидно, что при этом обращения из глобальной сети в локальную становятся невозможными в принципе.

7. SSL/TLS – набор криптографических технологий, обеспечивающих аутентификацию, конфиденциальность и целостность данных. Это один из самых распространенных протоколов защиты в Интернете: он ясен и практически не требует дополнительных усилий со стороны пользователей. Единственное, что нужно для настройки SSL/TLS на сервере, – установить сертификат X.509, предварительно получив его от сертифицирующей организации (CA).

13.5. Использование стеганографии в сетевых технологиях

13.5.1. Общая характеристика стеганографических преобразований

Стеганография – это, в общем случае, наука о передаче скрытых сообщений или о скрытой передаче сообщений.

Компьютерная стеганография изучает способы *сокрытия* (Hiding) информации в компьютерных данных, представляющих собой различные файлы, программы, пакеты протоколов и т. п.

Компьютерная стеганография базируется на двух принципах. Первый состоит в том, что файлы, содержащие оцифрованное изображение или звук, могут быть до некоторой степени видоизменены без потери функциональности, в отличие от других типов данных, требующих абсолютной точности.

Второй принцип заключается в неспособности органов чувств человека различать незначительные изменения в цвете изображения или качестве звука, что особенно легко использовать применительно к объекту-файлу, который несет избыточную, т. е. спрятанную информацию, будь то 16-битный звук, 8- или 24-битное изображение.

Стеганографическая система (стегосистема[*]) – совокупность средств и методов, которые используются для формирования *скрытого канала* (Covert Channel) передачи информации.

Стеганосистема образует канал – **стеганоканал**, по которому передается (или в котором хранится) заполненный тайной информацией *контейнер* (модифицированный контейнер) – **стеганоконтейнер** (Steganographic Container). Контейнер в стеганографической системе является носителем сокрытой информации.

Процесс внедрения или осаждения (Embedding) информации в контейнер – **прямое стеганографическое преобразование**.

Процесс извлечения (Extraction) информации из стеганоконтейнера – **обратное стеганографическое преобразование**.

Ключом (подобно криптосистемам) в стеганографических системах выступает метод преобразования контейнера при осаждении информации.

Ключей может быть несколько (в том числе – и на основе криптографии). В этом случае стеганосистема будет относиться к классу **многоключевых**.

На рис. 13.4 представлена общая схема преобразований в стеганографической системе.

Стеганография – это наука о способах передачи (хранения) сокрытой информации, при которых скрытый канал организуется на базе и внутри открытого канала с использованием особенностей восприятия информации.

[*]В литературе встречаются два сокращенных термина, обозначающие саму стеганографическую систему или ее элементы: ***стего****система* и ***стегано****система.*

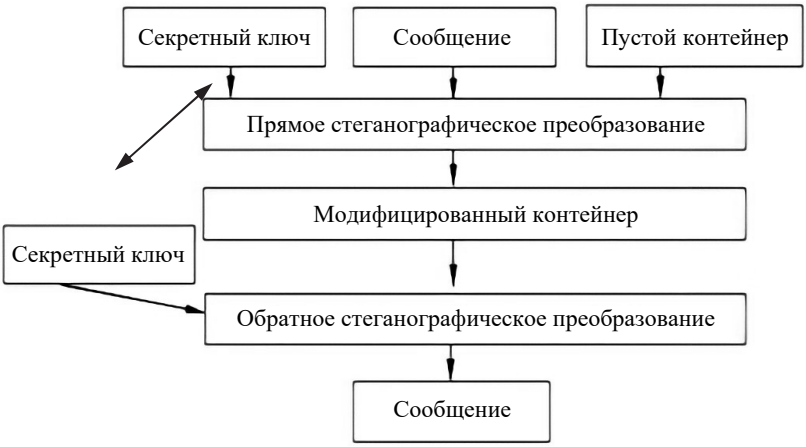


Рис. 13.4. Общая схема преобразований
в стеганографической системе

Для этой цели используются такие приемы:

– сокрытие факта существования скрытого канала связи;

– создание трудностей для обнаружения, извлечения или модификации передаваемых сокрытых сообщений внутри открытых сообщений-контейнеров;

– маскировка сокрытой информации в протоколе.

Выделяют следующие основные направления применения компьютерной стеганографии:

1) тайная передача или тайное хранение информации без совершения неправомерных, деструктивных действий по отношению к *чужим информационным ресурсам*;

2) организация деструктивных действий по отношению к чужим информационным ресурсам.

В первом случае реализуемые алгоритмы тайного хранения информации направлены *на защиту прав интеллектуальной собственности.* Здесь объектом авторского права выступает документ-контейнер, а тайная информация представляет собой авторские *цифровые водяные знаки* или *цифровые отпечатки пальцев*. При этом стеганоконтейнер может как храниться в соответствующем репозитории, так и передаваться по сети.

Во втором случае (деструктивные действия по отношению к чужим ресурсам) речь идет об организации *атак*.

Сейчас наблюдается новый и опасный тренд: ***все больше разработчиков вредоносного программного обеспечения и средств кибершпионажа прибегают к использованию стеганографии***.

Сетевое взаимодействие является ключевой функцией любой *вредоносной программы.*

Этому способствуют три главные причины:

– *стеганография позволяет скрыть сам факт загрузки/выгрузки данных*, а не только сами данные;

– *стеганография помогает обойти DPI-системы*, что актуально в корпоративных сетях; система DPI (Deep Packet Inspection – глубокая проверка пакетов), как видно из названия, выполняет глубокий анализ всех проходящих через нее пакетов. Термин «глубокий» подразумевает анализ пакета на верхних уровнях модели OSI, а не только по стандартным номерам портов. Помимо изучения пакетов по стандартным паттернам, по которым можно однозначно определить принадлежность пакета определенному приложению (по формату заголовков, номерам портов и т. п.), система DPI осуществляет и так называемый *поведенческий анализ трафика*, который позволяет распознать приложения, не использующие для обмена данными заранее известные заголовки и структуры данных. Популярными становятся интегрированные в маршрутизаторы решения DPI;

– *использование стеганографии может позволить обойти проверку в AntiAPT-продуктах*, поскольку последние не могут обрабатывать все графические файлы (их слишком много в корпоративных сетях, а алгоритмы анализа довольно дорогие).

Advanced Persistent Threat (APT) дословно переводится как «постоянная угроза повышенной сложности»; APT являются киберугрозами, к которым организации менее всего готовы. Это угрозы, использующие передовые методы для предотвращения обнаружения, а также гарантирующие устойчивость на скомпрометированном хосте при перезагрузках.

Механизмы AntiAPT-продуктов действуют на трех уровнях:

1) анализ сетевого трафика;
2) анализ контента;
3) анализ поведения рабочих станций.

Современные методы компьютерной стеганографии условно можно поделить на 2 класса:
1) методы ***сокрытия информации в контенте***, а также в ***альтернативных потоках данных*** (Alternate Data Streams, ADS) файловой системы NTFS;
2) методы ***сокрытия информации в структуре сетевых протоколов***.

13.5.2. Стеганографические методы на основе модификации контента

Методы рассматриваемого класса, как правило, основаны на внедрении тайной информации в документ-контейнер (файл аудио-видео- или текстовой информации) путем модификации параметров этого документа (цвет, пространственно-геометрические параметры структурных элементов, параметры форматов растровой и векторной графики и др.), а также на использовании специальных свойств компьютерных форматов данных*.

Здесь одним из наиболее распространенных, адаптированных к различным реализациям, в том числе – и на сетевом уровне, является *метод наименее значащих битов*, *НЗБ* (Least Significant Bit, LSB). Известная универсальность метода обусловлена конечными размерами различных структурных блоков двоичных данных, сетевых пакетов и т. п.: *младшие биты каждой структурной единицы бинарной последовательности являются менее значимыми, чем старшие*.

13.5.2.1. Сущность и особенности метода LSB. *Метод LSB основывается на ограниченных способностях зрения или слуха человека*, вследствие чего людям тяжело различать незначительные вариации цвета или звука.

Если документ-контейнер рассматривать на основе определенной цветовой модели (bitmap или RGB), то младшие разряды цифровых отсчетов, формирующих изображение, содержат очень мало полезной информации. Их заполнение (модификация) дополнительной информацией практически не влияет на качество восприятия, что и дает возможность скрытия конфиденциальной информации.

*Дополнительную информацию можно найти в изданиях [6, 27, 43–47].

Рассмотрим это на примере 24-битного растрового RGB-изображения. Каждая точка кодируется тремя байтами, каждый из которых определяет интенсивность красного (Red), зеленого (Green) и синего (Blue) цветов. Совокупность интенсивностей цвета в каждом из трех каналов определяет оттенок пикселя.

Представим пиксель тремя байтами в битовом виде, как это показано на рис. 13.5.

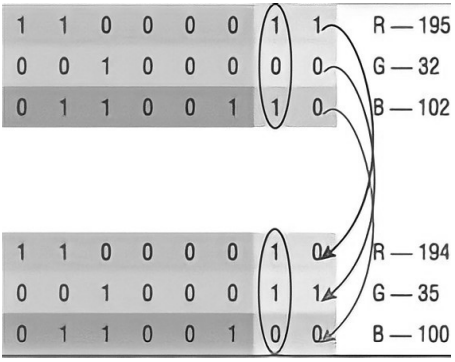


Рис. 13.5. Пример, показывающий принцип реализации метода LSB

Младшие биты (справа) дают незначительный вклад в изображение по сравнению со старшими. Замена одного или двух младших битов для человеческого глаза будет почти незаметна.

Эксперименты показывают, что не менее, чем в 98% случаев, эксперт не в состоянии визуально установить наличие модификаций до 3-х младших битов текстовых и графических документов.

Пример 13.3. Пусть необходимо в этом пикселе скрыть 6 битов – 101100. Разделим их на 3 пары и заместим этими парами младшие биты в каждом канале. Получим новый цвет, очень похожий на первоначальный. Оценим эффективность такого метода: используя 2 бита на канал, мы сможем прятать 3 байта информации на 4 пикселя изображения. А это уже где-то 25% картинки. Например, в мегабайтовом файле можно спрятать 250 Кбайт информации, причем для невооруженного глаза этот факт останется незаметным.

Недостатки метода LSB:

1) скрытое сообщение легко разрушить, например при сжатии или отображении;

2) не обеспечена секретность встраивания информации. Точно известно местоположение зашифрованной информации. Для преодоления этого недостатка можно встраивать информацию не во все пиксели изображения, а лишь в некоторые из них, определяемые по псевдослучайному закону в соответствии с ключом, известным только законному пользователю.

Альтернативным подходом является моделирование характеристик поведения LSB. Встраиваемое сообщение будет в этом случае частично или полностью зависеть от контейнера. Процесс моделирования является вычислительно трудоемким, кроме того, его надо повторять для каждого отдельно взятого контейнера. Главным недостатком этого метода является то, что процесс моделирования может быть повторен нарушителем, возможно обладающим большим вычислительным ресурсом, создающим лучшие модели, что приведет к обнаружению скрытого сообщения.

При использовании любого стеганографического ***метода на основе избыточности среды*** существует возможность повысить степень надежности скрытия, жертвуя при этом объемом скрываемых данных.

13.5.2.2. Сущность стеганографических методов на основе альтернативных потоков файловой системы NTFS. *NTFS* – New Technology File System – файловая система новой технологии – стандартная файловая система для семейства операционных систем Windows NT фирмы Microsoft. NTFS поддерживает хранение *метаданных*.

Как известно, в файловой системе NTFS каждый файл (или каталог) представлен как набор отдельных элементов, называемых *атрибутами*. Такие элементы, как имя файла, параметры безопасности и даже данные, являются атрибутами файла.

Каждый атрибут идентифицирован кодом *типа атрибута* и, необязательно, – *именем атрибута*. Так, например, имя файла содержится в атрибуте *Filename*, содержимое – в атрибуте *Data*, сведения о владельце и правах доступа – в атрибуте *Security Descriptor* и т. д.

Диск NTFS условно делится на две части. Первые 12% диска отводятся под так называемую MFT-зону – пространство, в которое растет метафайл MFT (Master File Table – общая таблица файлов). Запись каких-либо данных в эту область невозможна. MFT-зона всегда

держится пустой. Это делается для того, чтобы самый главный, служебный файл (MFT) не фрагментировался при своем росте. Остальные 88% диска представляют собой обычное пространство для хранения файлов.

Каждый файл в NTFS имеет несколько абстрактное строение – у него нет как таковых данных, а есть *потоки* (Streams). Главный из потоков хранит в себе данные файла. Но большинство атрибутов файла хранятся в потоках. Таким образом, получается, что сущностью файла является только его номер в MFT, а все остальное – опционально. Это дает возможность, например, прикрепить к файлу еще один поток, записав в него любые данные.

Содержимое каждого файла (атрибут *$DATA*) представляет собой набор потоков, в которых хранятся данные. Для каждого файла или каталога в NTFS существует как минимум один основной поток, в котором, собственно, и хранятся данные. Кроме основного потока, с файлом или каталогом могут быть связаны и ***альтернативные потоки*** (Alternate Data Stream, ADS), которые также могут содержать некоторые данные, никак не связанные с данными основного потока.

Основной поток файла не имеет имени и обозначается *$DATA*: " ". Альтернативные же потоки обязательно имеют имя, например *$DATA: "StreamData"* – альтернативный поток с именем *StreamData*.

При выполнении функций записи данных в файл они помещаются в основной поток данных. Когда мы открываем, например, блокнотом текстовый файл, то получаем доступ именно к данным основного потока. Данные же альтернативных потоков при использовании стандартного доступа не отображаются, как впрочем, нет даже никаких признаков их наличия. Тем не менее, данные альтернативных потоков, связанные с конкретным файлом или каталогом, могут быть доступны с помощью специальных программ или при использовании особого синтаксиса в командной строке Windows. Например, запись в файл stegano.txt текста *echo Main stream Data > stegano.txt* означает запись в основной неименованный поток. Но можно изменить команду: *echo Alternate stream Data > stegano.txt:stream1*, это значит записать текст *Alternate stream Data* в альтернативный поток со stegano.txt:stream1.

Альтернативные потоки данных дают возможность добавлять к файлу скрытую информацию. Все данные, записываемые в файл, по умолчанию попадают в основной поток данных. Когда мы открываем файл, то видим именно основной поток, ***альтернативные же потоки скрыты от пользователя и не отображаются с помощью обычных средств***. Такой подход ассоциируется со стеганографией.

Альтернативные потоки нельзя увидеть стандартными способами, хотя некоторые программы «умеют» читать скрытые в них данные. Для работы с потоками можно использовать командную строку. Перечислим некоторые средства для работы с альтернативными потоками.

Начиная с WindowsVista, команда *DIR* с ключом */r* позволяет просматривать не только файлы, но и альтернативные потоки. Пример показан на рис. 13.6.

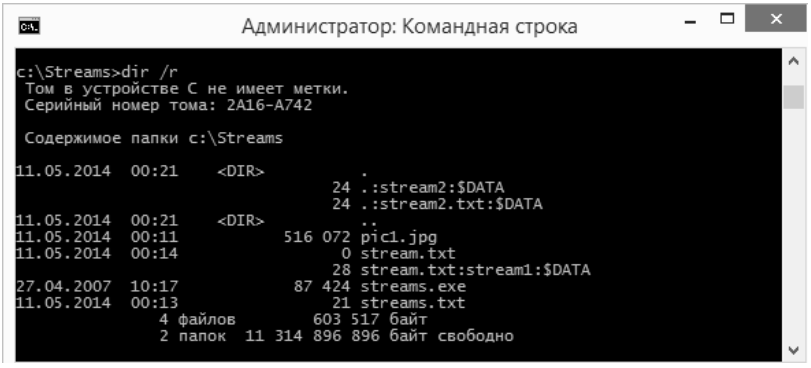


Рис. 13.6. Результат просмотра альтернативных потоков с помощью команды *DIR /r*

Для работы с альтернативными потоками существует несколько сторонних утилит, например консольная утилита *Streams* от Sysinternals.

NTFS Stream Explorer – программа для работы с альтернативными файловыми потоками NTFS и другими системными потоками, в том числе с расширенными атрибутами.

Альтернативные потоки используются как самой Windows, так и некоторыми программами.

К примеру, Internet Explorer делит сеть на 4 зоны безопасности и при загрузке файлов добавляет к ним метки, содержащие информацию о зоне, из которой они были загружены.

13.5.3. Стеганографические методы на основе использования структуры сетевых протоколов

13.5.3.1. DNS-туннелирование. Как мы уже знаем из главы 5, система доменных имен (DNS) – по сути, огромная телефонная книга интернета. DNS представляет собой ключевой компонент, обеспечивающий функционирование корпоративных и публичных компьютерных сетей, поскольку DNS является базовым протоколом, позволяющим администраторам делать запросы в базу данных DNS-сервера.

Пользуясь приведенными особенностями DNS, злоумышленники разработали технологию, позволяющую *скрытно общаться с компьютером-жертвой* путем внедрения управляющих команд и данных в протокол DNS. Эта идея и лежит в основе *DNS-туннелирования*. Гибкость протокола DNS позволяет встраивать произвольную информацию в качестве доменного имени, из-за чего возникает трудность в *различении вредоносных* и *безопасных запросов*. Одним из способов *сокрытия кибератаки* служит встраивание или инкапсуляция протоколов различных уровней модели OSI в доменное имя DNS-запроса.

> **!** Методика встраивания произвольной информации в запросы DNS называется **DNS-туннелированием**, так как подразумевает создание логического соединения между двумя конечными точками сети – туннеля.

Первое обсуждение такой атаки проводилось Оскаром Пирсаном (Oskar Pearson) в почтовой рассылке Bugtraq в апреле 1998 г. К 2004 г. DNS-туннелирование было представлено на Black Hat как хакерский метод в презентации Дэна Каминского (Dan Kaminsky). Таким образом, идея очень быстро переросла в настоящий инструмент атаки.

Известный способ передачи информации через DNS-туннель – *кодирование данных в субдомены с помощью алфавитного кодировщика отправителя*. При этом DNS-сервер с механизмом *рекурсивной*

обработки запроса (см. п. 5.3.3) инициирует запрос *авторитативному DNS-серверу* (здесь находится официальная база данных ресурсных записей зоны DNS; в базе указаны IP-адреса хост-серверов (например, веб-серверов, почтовых серверов) и серверов имен в зоне авторитативного DNS-сервера), который в случае DNS-туннелирования управляется злоумышленником.

Напомним: в случае *рекурсивного запроса* DNS-сервер опрашивает серверы (в порядке убывания уровня зон в имени), пока не найдет ответ или не обнаружит, что домен не существует (на практике поиск начинается с наиболее близких к искомому DNS-серверов, если информация о них есть в кэше и не устарела, сервер может не запрашивать другие DNS-серверы).

Благодаря тому, что данный протокол не блокируется межсетевыми экранами, DNS-туннелирование широко используется как для передачи команд управления зараженному узлу, так и для эксфильтрации данных.

Под термином **эксфильтрация данных** в широком смысле понимается неавторизированная передача произвольной информации, которая рассматривается как форма кражи данных.

Скрытый канал на основе DNS-туннелирования может использоваться в двух основных целях:
1) *для передачи украденной информации*, корпоративных секретов, интеллектуальной собственности путем кодирования и инкапсуляции данных в запрашиваемое доменное имя – ***туннелирование системы доменных имен***;
2) *для получения команд от удаленного злоумышленника*, в частности для организации распределенной ботнет-сети – ***генерирование доменных имен***; основной задачей является сокрытие реального месторасположения серверов злоумышленника; эту технологию обычно называют *Управление и контроль* (Command & Control, C2).

Рассмотрим порядок организации DNS-туннеля.

Для этого целесообразно вернуться (см. п. 5.3.3) к рассмотрению функцианальностей DNS. Система работает в режиме «запрос – ответ». Допустим, некто (**X**) ввел в строке своего браузера *qwerty.com*.

Рассмотрим работу DNS более подробно, пошагово.

1. Браузер, которым пользуется **X**, об IP-адресе *varonis.com* ничего не знает и с запросом IP-адреса через специальную программу – *резолвер* (Resolver) обращается к локальному серверу имен.

Локальный DNS-сервер – это сервер имен локальной сети или DNS-сервер интернет-провайдера, услугами которого пользуется **X**. Браузеру известно о существовании этого локального DNS, поскольку при настройках сетевого подключения **X** прописал IP-адреса DNS-серверов (*предпочитаемого* и/или *альтернативного*), один из которых будет отвечать на запросы, посылаемые браузером через резолвер – это и есть локальный или местный сервер сети клиента **X**. Для того чтобы посмотреть IP-адрес локального DNS-сервера, достаточно посмотреть свойства сетевого подключения, используемого на компьютере.

Резолверы играют важную роль в DNS: DNS-резолвер кэширует информацию. К примеру, сайт *varonis.com* расположен на машине с IP-адресом 35.239.147.27. Поэтому кэши резолверов со всего мира будут содержать следующее соответствие: *varonis.com → 35.239.147.27.*

2. Запрос на IP-адрес *varonis.com* доходит до местного сервера имен. Резолвер по возможности просматривает свой кэш на наличие необходимого IP-адреса. Если у резолвера есть необходимый IP-адрес, то он возвращает его клиенту **X**.

Предположим, что этот сервер о данном IP-адресе ничего не знает и посылает запрос одному из корневых серверов «.» (*root*).

3. Корневой сервер передает локальному серверу (резолверу) IP-адрес сервера, который поддерживает зону *.com*, т. е. пересылает Top-Level Domain, TLD.

4. Далее по полученному адресу локальный сервер имен обращается к DNS-серверу, который поддерживает *.com*.

5. Этот DNS-сервер, в свою очередь, по полученному запросу передает IP-адрес сервера, который поддерживает зону *qwerty.com*.

6. Местный DNS-сервер с запросом IP-адреса *varonis.com* обращается к DNS-серверу зоны *varonis.com*; этот сервер называется авторитативным; он хранит фактические записи сопоставления имени хоста с IP-адресом, который возвращается резолверу.

7. Локальный сервер имен получает IP-адрес *varonis.com* от авторитативного DNS-сервера зоны *varonis.com*.

8. Получив адрес *varonis.com*, локальный DNS-сервер сообщает его браузеру клиента **X**.

Результат операции представлен на рис. 13.7.

```
1010-4:~ andyg$
1010-4:~ andyg$ nslookup
> varonis.com
Server:         208.67.222.222
Address:        208.67.222.222#53

Non-authoritative answer:
Name:   varonis.com
Address: 35.239.147.27
>
>
```

Рис. 13.7. Пример работы DNS в режиме «запрос – ответ»

Вместо рассмотренного запроса (ввода вполне легитимного URL) можно ввести тайное сообщение (рис. 13.8) – простейший способ использования туннеля.

```
1010-4:~ andyg$ nslookup
> I_am_sneaking_out_important_data.com
Server:         208.67.222.222
Address:        208.67.222.222#53

** server can't find I_am_sneaking_out_important_data.com: NXDOMAIN
>
```

Источник: https://habr.com/ru/company/varonis/blog/513160/.

Рис. 13.8. Пример передачи тайного сообщения через DNS-туннель

Вместо разрешения запроса на сервере злоумышленник может извлечь из полученного домена нужную ему информацию. Например, таким способом можно передать информацию о системе пользователя. В ответ DNS-сервер злоумышленника также посылает некую информацию в зашифрованном виде, передавая ее в доменном имени более высокого уровня. Таким образом, злоумышленник имеет в запасе для каждого DNS-резолвера 255 символов. Налажен скрытый канал для передачи данных. А главное – невооруженным взглядом этого не видно.

В традиционной бот-сети, созданной на основе рассматриваемой технологии DNS-туннелирования, боты обычно заражаются троянским конем и используют *интернет-ретранслятор* (Internet Relay Chat, IRC – это протокол, определяющий набор правил для связи между клиентом и сервером через некоторый механизм связи, такой как чаты) для связи с центральным *сервером C&C* (Command & Control, C2). В зависимости от цели и структуры ботнета, сервер C&C

может выдавать команды, например, для начала атаки DDoS (распределенного отказа в обслуживании).

Существуют различные ***способы распознавания DNS-туннелирования***. Например, очевидно, что доменные имена для резолва при туннелировании значительно длиннее, чем обычно. Кроме того, такие имена, поскольку они, как правило, сгенерированы компьютером, значительно разнятся от тех, которые созданы на основе естественных языков, что дает возможность использования методов обработки естественных языков с целью *детектирования угроз сетевой безопасности*, *связанных с DNS-тунеллированием.*

На практике применяются ***два основных метода обнаружения DNS-туннелирования***:
1) ***анализ нагрузки***;
2) ***анализ трафика***.

При *анализе нагрузки* защищающаяся сторона ищет аномалии в данных, передаваемых в обе стороны, которые могут быть обнаружены *статистическими методами*: необычно выглядящие имена хостов, тип DNS-записи, которая не используется настолько часто, или нестандартная кодировка.

При *анализе трафика* оценивается число DNS-запросов к каждому домену по сравнению со среднестатистическим уровнем. Злоумышленники, использующие DNS-туннелирование, генерируют большой объем трафика (например, при организации DDoS-атак – десятки тысяч запросов в течение сравнительно небольшого промежутка времени) на сервер.

Для проверки того, насколько эффективно сеть компании или организации способна обнаруживать DNS-туннели и/или противодействовать им, можно воспользоваться готовыми *утилитами DNS-туннелирования.* Например, программа *Iodine* – позволяет туннелировать данные IPv4 через DNS-сервер; доступна на платформах Linux, MacOS, FreeBSD и Windows; можно использовать в различных ситуациях, когда доступ в Интернет защищен брандмауэром, но разрешены DNS-запросы.

Для обнаружения туннелирующих атак также можно воспользоваться готовыми утилитами, например *dnsHunter* – Python-модуль, написанный для Mercenary Hunt Framework и Mercenary-Linux.

Считывает *pcap*-файлы, извлекает DNS-запросы и производит сопоставление геолокации, что помогает при анализе; *reassemble_dns* – утилита на Python, читающая *pcap*-файлы и анализирующая DNS-сообщения.

13.5.3.2. Стеганография в скрытых каналах на основе стека TCP/IP.

Основные модели и типы скрытого сетевого взаимодействия. Как мы уже знаем, *скрытый канал может существовать в любом открытом канале, в котором существует некоторая избыточность*. А избыточность является непременным атрибутом практически любого сетевого протокола. И именно в отношении сетевых протоколов Б. Лэмпсон (B. Lampson) в 1973 г. ввел понятие *скрытого канала* [49].

Скрытые каналы в протоколах компьютерных сетей аналогичны методам сокрытия информации в звуковом, визуальном или текстовом контенте. В то время как классическая стеганография требует какой-либо формы контента в качестве прикрытия, *скрытые каналы требуют некоторого сетевого протокола в качестве носителя скрытой информации, передаваемой от одного абонента к другому*.

Для количественной характеристики методов сетевой стеганографии обычно используют два основных параметра:

1) *пропускную способность скрытого канала* – объем секретных данных, который может быть передан по каналу в единицу времени;

2) *вероятность обнаружения скрытого канала* или *стеганографическая стойкость* используемого метода сокрытия передаваемых данных.

Существуют различные модели скрытого сетевого взаимодействия двух абонентов (*A* и *B*) в зависимости от того, являются ли *A* и *B* отправителем и получателем открытого канала (*Overt Sender* и *Overt Receiver*) или же они действуют как посредники (*C*) и манипулируют открытым каналом между ничего не подозревающими пользователями (как это мы анализировали, например, на основе DNS-туннелирования).

Если отправитель (*A*) скрытого канала также является отправителем открытого канала, он может манипулировать открытым каналом

по желанию (например, чтобы максимизировать пропускную способность скрытого канала или его скрытность). Однако иногда скрытый отправитель может быть не в состоянии создать открытый канал или может не делать этого для большей скрытности. В этом случае отправитель может выступать в качестве *посредника*, встраивая скрытый канал в существующий открытый канал. Очевидно, что тогда скрытый отправитель не имеет контроля над открытым каналом, и максимальная пропускная способность скрытого канала зависит от существующего открытого канала.

Скрытый получатель (*B*) может быть получателем открытого канала, но для повышения скрытности получатель также может быть посредником (*C*), извлекающим скрытую информацию из открытого сообщения, предназначенного для невиновного получателя. Затем скрытый приемник должен (если возможно) удалить скрытый канал, предотвращая возможное обнаружение приемником или любыми другими промежуточными узлами.

На рис. 13.9 показаны возможные комбинации скрытых местоположений отправителя и получателя. Фактический сценарий связи зависит от применения скрытого канала.

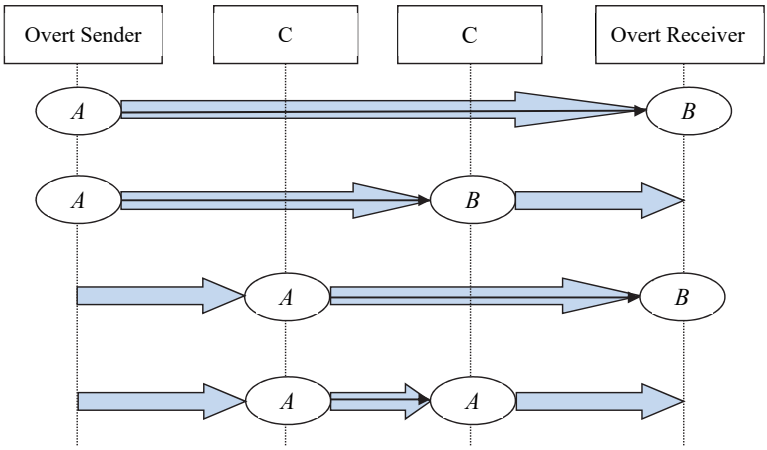


Рис. 13.9. Модели взаимодействия абонентов на основе скрытых каналов

Если скрытый канал используется для обхода цензуры, скрытые и явные отправитель/получатель, вероятно, будут идентичными. Если же канал используется хакером для внешней фильтрации данных,

то скрытые отправитель и получатель, скорее всего, будут посредниками (например, отправитель может находиться внутри стека сетевых протоколов скомпрометированной машины, а получатель – на маршрутизаторе, близком к краю скомпрометированной сети).

Традиционно скрытые каналы (Covert Channels) подразделяются (хотя принципиального различия между ними нет):

1) на ***каналы памяти*** (*скрытые каналы памяти*);

2) ***каналы временны́е*** (*скрытые каналы времени*).

Каналы памяти (Storage Channel) предполагают *прямую/косвенную запись значения объекта* (осаждение тайной информации) отправителем и *прямое/косвенное считывание значений объекта* (извлечение тайной информации) получателем. При этом обычно подразумевается, что у процессов с разными уровнями безопасности имеется доступ к некоторому ресурсу (например, к некоторым секторам диска).

> ***Тайная информация в скрытом канале памяти*** может быть размещена, например, в длинах пакетов или полях заголовков пакетов.

Временны́е каналы (Timing Channels) включают сигнальную информацию отправителя, *модулируя* (осаждая тайную информацию) *использование ресурсов* (например, использование ЦП) *с течением времени* таким образом, чтобы получатель мог наблюдать за этим и декодировать (извлекать) информацию.

> ***Тайная информация в скрытом временно́м канале*** может быть внедрена путем изменения скоростей передачи пакетов (или времени между пакетами) на основе их сортировки.

Данные подходы предполагают использование только одного сетевого протокола из стека TCP/IP для создания тайного канала. Модификация протокола может быть применена либо к протокольным блокам данных *PDU* (Protocol Data Unit), либо к *временным отношениям между обмениваемыми PDU*, либо одновременно к обоим из указанных параметров. Этот вид сетевой стеганографии можно назвать ***внутрипротокольной стеганографией***.

Кратко проанализируем методы, на основе которых создаются скрытые каналы.

Методы создания скрытых каналов памяти. Скрытые каналы могут быть закодированы в *неиспользуемых или зарезервированных битах заголовков кадров или пакетов*. Это особенно проблематично, если стандарты протоколов не требуют конкретных значений или получатели не проверяют стандартные значения. На этой основе могут создаваться следующие типы скрытых каналов:

– неиспользуемые биты поля *типа обслуживания* (Type of Service, ToS);

– *Заголовок IP*;

– поля *Флагов заголовка TCP*;

– бит *Don't Fragment* (DF) заголовка IP; биту DF можно присвоить произвольное значение, если отправителю известен размер *Максимальной единицы передачи* (Maximum Transfer Unit, MTU) пути к получателю и он отправляет только те пакеты, которые по размерам меньше MTU;

– поле *Указателя важности* (TCP Urgent Pointer, применяется для указания данных с высоким приоритетом) – 16-битное поле, которое принимается во внимание только для пакетов с установленным флагом URG;

– сегмент *Флага сброса соединения* (TCP Reset – сегменты TCP с установленным флагом RST обрывают соединение и обычно не содержат данных);

– поля заголовка IPv6 *Класс трафика* (Traffic Class) и *Метка потока* (Flow Label);

– заголовок *Параметров назначения* IPv6; этот заголовок содержит необязательную информацию об узле назначения пакета: если тип опции установлен таким образом, что получатель игнорирует эту опцию, скрытая информация может быть напрямую закодирована как данные опции.

Скрытая информация может быть *закодирована в кадрах или пакетах*: например, кадры Ethernet должны быть дополнены до минимальной длины 60 байтов. Если стандарт протокола не требует определенных значений байтов заполнения, можно использовать любые данные.

Заполнение заголовка IP и TCP до 4-байтовых границ (в случае наличия опций заголовка) и заполнение в IPv6 также может использоваться для передачи скрытых данных.

Размещение тайного сообщения в поле *Контрольной суммы заголовка* (Header Checksum) *IP*: поле контрольной суммы модифицируется для размещения секретной информации и добавляется расширение IP-заголовка с выбранным содержимым таким образом, чтобы измененная контрольная сумма снова была правильной; тот же метод можно использовать для контрольной суммы заголовка TCP, однако поскольку этот метод требует добавления расширения заголовка, скрытая информация также может быть размещена непосредственно в расширении заголовка.

Поскольку контрольная сумма в пакетах UDP не является обязательной, можно использовать его наличие или отсутствие для внедрения одного бита скрытой информации на пакет UDP.

В беспроводных сетях используются соединения с переменной частотой ошибок по битам (BER), они дают возможность *вводить искусственные поврежденные кадры*. Все станции, которые являются частью скрытого канала, обмениваются данными, отправляя некоторый процент своих кадров с намеренно созданными неверными контрольными суммами. Другие станции отбрасывают поврежденные кадры.

К тому же, разработаны иные возможности для создания скрытых каналов памяти.

Методы создания скрытых каналов времени. К наиболее известным относятся методы, созданные на основе модификации следующих параметров сетевых протоколов:

– использование поля *Время жизни* (Time-to-Live, TTL); в IPv4 TTL представляет собой 8-разрядное поле IP-заголовка, которое определяет максимальное количество *хопов* (hop – прыжок, участок между маршрутизаторами), которые пакет может пройти. Наличие этого параметра не позволяет пакету бесконечно ходить по сети. Каждый маршрутизатор при маршрутизации должен уменьшать значение TTL на единицу, но некоторые шлюзы можно настроить, чтобы игнорировать это. Пакеты, не достигшие адресата, но *время жизни* которых стало равно нулю, уничтожаются, а отправителю посылается сообщение *ICMP Time Exceeded*. Если требуется, чтобы пакет не был маршрутизирован, т. е. был принят только в своем сегменте, то выставляется TTL = 1. На отправке пакетов с разным временем жизни основана трассировка их пути прохождения (Traceroute). Максимальное значение TTL = 255. Обычное начальное значение TTL = 64 (Linux, Mac, Android, iOS), TTL = 128 (Windows).

Скрытый канал на основе поля TTL заголовка IP в качестве решения для отслеживания IP-потоков без использования поля адреса источника предусматривает участие маршрутизаторов, которые модулируют поле TTL-пакетов, чтобы нижестоящие получатели могли однозначно идентифицировать свой восходящий маршрутизатор. Примерно по этому же алгоритму создаются скрытые каналы на основе поля *Предела перехода* IPv6 (эквивалент IP TTL);

– использование полей *Временных меток* (Timestamp Fields); скрытая информация осаждается в младшие биты временных меток отправителя, поскольку они фактически случайны для медленных TCP-соединений. Вместо прямого изменения меток времени алгоритм замедляет поток TCP, чтобы метки времени на пакетах были действительными при их отправке. Алгоритм сравнивает *младший значащий бит* (LSB) каждого TCP-сегмента, сгенерированного системой, с текущим скрытым битом для отправки. Если младший бит соответствует скрытому биту, TCP-сегмент отправляется немедленно, в противном случае он задерживается на один тик временной метки;

– скрытый канал, реализованный посредством *сортировки пакетов*; набор из *n* пакетов, составляющих сообщение, можно отсортировать в виде *n!* последовательностей; требуется, чтобы порядковые номера пакетов определяли их исходный порядок в переданном сообщении; вместо того, чтобы фактически сортировать пакеты, этот метод изменяет только порядковые номера, тем самым сохраняя полезную нагрузку, отправляемую через несколько пакетов, нетронутой. Авторы предложили использовать порядковый номер заголовка аутентификации IPSec (AH) или инкапсуляции полезной нагрузки безопасности (ESP) [50], но, вероятно, можно использовать и другие порядковые номера (например, порядковый номер TCP).

Методы создания межпротокольных скрытых каналов. В отличие от внутрипротокольной стеганографии скрытые каналы можно создавать на основе использования *отношения между двумя или более различными сетевыми протоколами* для обеспечения секретной связи. Разработчики этого направления [42] определили его как *Padding Steganography*, PadSteg (Padding Steganography – стеганография заполнения*), или ***межпротокольную стеганографию***. PadSteg использует Ethernet (IEEE 802.3), ARP (Address Resolution Protocol), TCP и другие протоколы. Известно, что в IP-сетях протокол ARP

**Padding* (англ.) – заполнение, набивка.

используется в основном для определения аппаратного MAC-адреса (Media Access Control), когда известен только адрес сетевого протокола (IP-адрес). ARP важен для надлежащего функционирования любой коммутируемой локальной сети.

С точки зрения выбора и использования элементов анализируемой здесь стеганосистемы для их модификации (моделирования) при размещении/извлечении тайной информации методы на основе *PadSteg схожи с методами создания скрытых каналов памяти*.

Межпротокольную стеганографию можно реализовать на любом уровне модели OSI. Но, как правило, она используется для скрытой связи только на канальном, сетевом и транспортном уровнях.

PadSteg обеспечивает секретную связь в скрытой группе в среде LAN. В такой группе каждый хост, участвующий в обмене стеганограммами, должен иметь возможность находить и идентифицировать другие скрытые хосты. Для обеспечения этой функциональности должны быть указаны определенные механизмы. Например, протокол ARP вместе с «неправильным» заполнением кадра Ethernet используется для обеспечения локализации и идентификации членов скрытой группы.

Для обеспечения обмена секретными данными можно использовать Ethernet и ARP/TCP. Стеганограмма внедряется в заполнение кадра Ethernet, но всегда нужно «смотреть» на протокол другого уровня (ARP или TCP), чтобы определить, содержит ли он секретные данные или нет.

Пропускная способность скрытого канала на основе рассматриваемого метода приблизительно оценена в 32 бит/с.

Чтобы свести к минимуму потенциальную угрозу межпротокольной стеганографии для общественной безопасности, важна идентификация таких методов. Не менее значимой является разработка эффективных контрмер. Это требует глубокого понимания функциональности сетевых протоколов и способов их использования для стеганографии.

Однако учитывая сложность используемых в настоящее время сетевых протоколов, мало надежды на разработку универсального

и эффективного метода стегоанализа. Таким образом, системы безопасности после выявления каждого нового метода стеганографии должны быть адаптированы к новой потенциальной угрозе.

13.5.3.3. Нейтрализация скрытых каналов. Задача *обнаружения скрытых каналов* на основе сетевой стеганографии состоит в том, чтобы найти потенциальные скрытые каналы, которые могут быть реализованы в сети. Второй шаг – анализ выявленных каналов для оценки уровня угрозы каждого из них.

Если канал представляет угрозу для защищаемой сети, могут быть применены следующие меры: *устранение*, *ограничение*.

Противодействие скрытым каналам на основе устранения предусматривает блокировку уязвимых протоколов/портов. Например, протокол ICMP блокируется многими брандмауэрами, что делает неэффективными многие методы создания скрытых каналов. Очевидно, что в Интернете некоторые протоколы не могут быть заблокированы, потому что они жизненно необходимы (например, IP, TCP, DNS) или потому, что их службы слишком важны (например, электронная почта, Интернет). Однако в закрытой сети протоколы, которые могут стать контейнером для скрытых стеганоканалов, могут быть заблокированы или заменены версиями с меньшими рисками.

Если скрытый канал невозможно устранить, его пропускную способность целесообразно уменьшить с помощью методов ограничения. Для обнаружения действующих скрытых каналов могут быть использованы методы аудита и обнаружения. Эти методы основаны на обнаружении нестандартного или аномального поведения, о чем упоминалось выше.

Важно помнить: *защита хостов не может нейтрализовать скрытые сетевые каналы*, но может предотвратить их использование в некоторых сценариях приложений.

В плане нейтрализации скрытых каналов, описанных в подпункте 13.5.3.2, нужно иметь в виду следующее.

1. С неиспользуемыми или зарезервированными битами можно легко справиться, установив их равными нулю, а неизвестные расширения заголовка можно удалить.

2. Некоторые скрытые каналы основаны на том, что определенные поля заголовка не всегда используются (и их использование указывается другими полями заголовка). Эту особенность можно нейтрализовать, например, устанавливая значения IP ID, биты DF и Urgent Pointer в определенное, согласованное состояние. Кроме того, следует убедиться, что контрольные суммы всегда используются и являются правильными.

3. Ряд других полей заголовка можно переписать при определенных предположениях. Например, включить бит DF и установить IP ID и Fragment Offset равными нулю, если пакет меньше размера MTU (при условии согласования сторонами размера MTU).

4. Параметры *Время жизни* и *Временная метка* TCP также могут быть согласованы.

Скрытые каналы могут создаваться и использоваться не только злоумышленниками, но и для передачи данных аутентификации. Сетевые администраторы могут применять эти скрытые каналы для защиты связи, управления сетью, скрывая всю информацию от хакеров.

Выводы

1. Сейчас наблюдается новый и опасный тренд: все больше разработчиков вредоносного ПО и средств кибершпионажа прибегают к использованию *стеганографии*. Сетевое взаимодействие является ключевой функцией любой вредоносной программы.

2. Современные методы компьютерной стеганографии условно можно поделить на 2 класса:

– *методы сокрытия информации в контенте*, а также в альтернативных потоках данных (Alternate Data Streams, ADS) файловой системы NTFS;

– *методы сокрытия информации в структуре сетевых протоколов*.

3. Содержимое каждого файла (атрибут *$DATA*) представляет собой набор потоков, в которых хранятся данные. Для каждого файла

или каталога в NTFS существует как минимум один основной поток, в котором, собственно, и хранятся данные. Кроме основного потока с файлом или каталогом могут быть связаны и *альтернативные потоки* (Alternate Data Stream, ADS), которые также могут содержать некоторые данные, никак не связанные с данными основного потока.

4. *Альтернативные потоки* данных дают возможность добавлять к файлу *скрытую информацию*. Все данные, записываемые в файл, по умолчанию попадают в основной поток данных. Когда файл открывается, то виден именно основной поток, альтернативные же потоки скрыты от пользователя и не отображаются с помощью обычных средств. Такой подход ассоциируется со стеганографией.

5. *Скрытый канал на основе DNS-туннелирования* может использоваться в двух основных целях:

– *для передачи украденной информации*, корпоративных секретов, интеллектуальной собственности путем кодирования и инкапсуляции данных в запрашиваемое доменное имя – *туннелирование системы доменных имен*;

– *для получения команд от удаленного злоумышленника*, в частности для организации распределенной ботнет-сети, – *генерирование доменных имен*, основной задачей которого является сокрытие реального месторасположения серверов злоумышленника. Эту технологию обычно называют *Управление и контроль* (Command & Control, C2).

6. На практике применяются два основных метода обнаружения DNS-туннелирования:

– *анализ нагрузки*;

– *анализ трафика*.

7. Скрытые каналы в протоколах компьютерных сетей аналогичны методам сокрытия информации в звуковом, визуальном или текстовом контенте. В то время как классическая стеганография требует какой-либо формы контента в качестве прикрытия, *скрытые каналы требуют некоторого сетевого протокола в качестве носителя скрытой информации, передаваемой от одного абонента к другому*.

8. Скрытые каналы подразделяются (хотя принципиального различия между ними нет):

1) на *каналы памяти*;

2) *каналы времени*.

9. Защита хостов не может нейтрализовать скрытые сетевые каналы, но может предотвратить их использование в некоторых сценариях приложений.

Контрольные вопросы

1. Каковы принципы криптографической защиты информации?
2. Что такое симметричные криптосистемы?
3. Что такое ассиметричные криптосистемы?
4. Оценка эффективности использования пароля.
5. Опишите принцип дествия технологии SKIP.
6. В чем заключается суть и назначение прокола SSL/TLS?
7. Охарактеризуйте фазу рукопожатия протокола SSL.
8. Как формируется сообщение между клиентом и сервером после окончания фазы рукопожатия?
9. Что такое proxy-сервер?
10. Опишите принципы функционирования межсетевых экранов.
11. Поясните сущность стеганографических методов.
12. В чем состоят общность и различие между стеганографией и криптографией?
13. Что такое «скрытый канал связи»? Как он создается и как может использоваться?
14. Поясните сущность стеганографических методов на основе альтернативных потоков файловой системы NTFS.
15. Как использование альтернативных потоков влияет на безопасность компьютерных сетей?
16. Поясните сущность DNS-туннелирования.
17. Как создается DNS-туннель? Поясните на примере.
18. Как злоумышленники используют DNS-туннелирование?
19. Как можно выявлять и нейтрализовывать DNS-туннелирование?
20. Поясните особенности использования стеганографических методов для тайной передачи информации.
21. В чем состоит суть LSB-алгоритма?
22. Поясните сущность и особенности создания скрытых каналов на основе стека TCP/IP.
23. Охарактеризуйте основные параметры скрытых каналов.
24. Предложите и обоснуйте ваши методы создания скрытых каналов.
25. В чем состоит сущность методов противодействия угрозам безопасности информации со стороны скрытых каналов.

ЛИТЕРАТУРА

1. Урбанович, П. П. Компьютерные сети: учеб. пособие / П. П. Урбанович, Д. М. Романенко, Е. В. Кабак. – Минск: БГТУ, 2011. – 399 с.

2. Олифер, В. Г. Компьютерные сети. Принципы, технологии, протоколы: учебник / В. Г. Олифер, Н. А. Олифер. – 5-е изд. – СПб.: Питер, 2016. – 992 с.

3. Таненбаум, Э. Компьютерные сети / Э. Таненбаум, Д. Уэзеролл. – 5-е изд. – СПб.: Питер, 2012. – 960 с.

4. Ochrona informacji w sieciach komputerowych / pod red. prof. P. Urbanowicza. – Lublin: KUL, 2004. – 150 p.

5. Урбанович, П. П. Лабораторный практикум по дисциплинам «Защита информации и надежность информационных систем» и «Криптографические методы защиты информации». В 2 ч. Ч. 1. Кодирование информации: учеб.-метод. пособие / П. П. Урбанович, Д. В. Шиман, Н. П. Шутько. – Минск: БГТУ, 2019. – 116 с.

6. Урбанович, П. П. Лабораторный практикум по дисциплинам «Защита информации и надежность информационных систем» и «Криптографические методы защиты информации». В 2 ч. Ч. 2. Криптографические и стеганографические методы защиты информации: учеб.-метод. пособие / П. П. Урбанович, Н. П. Шутько. – Минск: БГТУ, 2020. – 226 с.

7. Олифер, В. Г. Сетевые операционные системы / В. Г. Олифер, Н. А. Олифер. – СПб.: Питер, 2002. – 544 с.

8. Кульгин, М. В. Компьютерные сети. Практика построения. Для профессионалов / М. В. Кульгин. – 2-е изд. – СПб.: Питер, 2003. – 464 с.

9. Бейли, Д. Волоконная оптика: теория и практика / Д. Бейли, Э. Райт; пер. с англ. – М.: КУДИЦ-ОБРАЗ, 2006. – 320 с.

10. Линн, С. Администрирование Microsoft Windows Server 2012 / С. Линн. – СПб.: Питер, 2014. – 304 с.

11. Windows Server 2012 R2. Полное Руководство. Том 1. Установка и конфигурирование сервера, сети, DNS, Active Directory и общего доступа к данным и принтерам / М. Минаси [и др.]. – Киев: Диалектика, 2015. – 960 с.

12. Челлис, Дж. Основы построения сетей: учеб. пособие / Дж. Челлис, Ч. Перкинс, М. Стриб. – М.: Лори, 2001. – 324 с.

13. Урбанович, П. П. Информационная безопасность и надежность систем: учеб.-метод. пособие / П. П. Урбанович, Д. М. Романенко, Е. В. Романцевич. – Минск: БГТУ, 2007. – 100 с.

14. Романенко, Д. М. Компьютерные сети: лабораторный практикум / Д. М. Романенко, Н. В. Пацей, М. Ф. Кудлацкая. – Минск: БГТУ, 2016. – 168 с.

15. Gorbunova, Yu. W-cyclic method of interleaving of the data for communication systems / Yu. Gorbunova, P. Urbanovich // Przegląd Elektrotechniczny. – 2012. – R. 88. – № 11b. – P. 344–345.

16. Дженнингс, Ф. Практическая передача данных: модемы, сети и протоколы / Ф. Дженнигс; пер. с англ. – М.: Мир, 1989. – 272 с.

17. Гейер, Д. Беспроводные сети. Первый шаг / Д. Гейер. – М.: Вильямс, 2005. – 192 с.

18. Пролетарский, А. В. Беспроводные сети Wi-Fi / А. В. Пролетарский, И. В. Баскаков, Д. Н. Чирков. – М.: Интуит, 2007. – 177 с.

19. Рошан, П. Основы построения беспроводных локальных сетей стандарта 802.11 / П. Рошан, Д. Лиэри; пер. с англ. – М.: Вильямс, 2004. – 304 с.

20. Титтел, Э. Networking Essentials / Э. Титтел, К. Хадсон, Дж. Майкл Стюарт. – СПб.: Питер, 1999. – 474 с.

21. Чеппел, Л. TCP/IP. Учебный курс / Лора А. Чеппел, Э. Титтел. – СПб.: БХВ-Петербург, 2003. – 960 с.

22. Скляр, Б. Цифровая связь. Теоретические основы и практическое применение / Б. Скляр. – М.: Вильямс, 2007. – 1104 с.

23. BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня: сб. материалов V Междунар. науч.-практ. конф., Минск, 13–14 марта 2019 г. в 2 ч. / Белорус. гос. ун-т информатики и радиоэлектроники; редкол.: В. А. Богуш [и др.]. – Минск, 2019. Ч. 1. – 403 с.

24. Elements of Modern Networking // The University of Mumbai [Electronic resource], 2021. – Mode of access: https://mu.ac.in/wp-content/uploads/2021/01/Modern-Networking.pdf. – Data of access: 12.10.2021.

25. 5 Trends Appear on the Gartner Hype Cycle for Emerging Technologies // Gartner [Electronic resource], 2019. – Mode of access: https://www.gartner.com/smarterwithgartner/5-trends-appear-on-the-gartner-hype-cycle-for-emerging-technologies-2019/. – Data of access: 12.10.2021.

26. Гладкий, М. В. Безопасность приложений на платформах облачных вычислений / М. В. Гладкий, П. П. Урбанович // Информационные технологии: тез. докл. 79-й науч.-техн. конф. профес.-преподават. состава, науч. сотрудников и аспирантов (с междунар. участием), Минск, 2–6 февр. 2015 г. / Белорус. гос. технол. ун-т. – Минск, 2015. – С. 18–19. – Режим доступа: https://elib.belstu.by/handle/123456789/25736. – Дата доступа: 25.10.2021.

27. Урбанович, П. П. Защита информации методами криптографии, стеганографии и обфускации: учеб.-метод. пособие / П. П. Урбанович. – Минск: БГТУ, 2016. – 219 с.

28. Zander, S. A Survey of Covert Channels and Countermeasures in Computer Network Protocols / S. Zander, G. Armitage, P. Branch // IEEE Communications Surveys & Tutorial. – 2007, 3rd Quarter. – P. 44–57.

29. Technical Specification: Speech and multimedia Transmission Quality (STQ); Quality of Experience; A Monitoring Architecture: ETSI TS 103 294. – European Telecommunications Standards Institute. – 2014. – 24 p.

30. Урбанович, П. П. Концепция создания взаимодополняемых алгоритмов обеспечения надежности и целостности информации в компьютерных сетях / П. П. Урбанович, Н. В. Пацей // Труды БГТУ. Сер. IV, Физ.-мат. науки и информатика. – 1997. – Вып. V. – С. 91–96.

31. Кунегин, С. В. Основы технологии АТМ: учеб.-метод. пособие / С. В. Кунегин. – М., 1999. – 80 с.

32. Пацей, Н. В. Комбинированное преобразование информации для повышения целостности и надежности данных (система 3К) / Н. В. Пацей, П. П. Урбанович // Труды БГТУ. Сер. IV, Физ.-мат. науки и информатика. – 1998. – Вып. VI. – С. 103–107.

33. Patsei, N. V. On the Design of Error Detection and Correction Cryptography Schemes / N. V. Patsei, P. P. Urbanovich // Eurocomm 2000 Information Systems for Enhanced Public Safety and Security: Conference Record. Munich, Germany, 2000. – P. 266–268.

34. Urbanovich, P. P. The algorithm for determining of the errors multiplicity by multithreshold decoding of iterative codes / Pavel P. Urbanovich, Marina F. Vitkova, Dmitri M. Romanenko // Przegląd Elektrotechniczny. – 2014. – Wyp. 90, № 3. – S. 235–238.

35. Velte, T. Cloud Computing: A Practical Approach by Anthony / T. Velte, A. Velte, R. Elsenpeter. – The McGraw-Hill Companies, 2010. – 352 p.

36. Filipek, Ł. Internet of things: concepts, risks, security / Ł. Filipek, P. P. Urbanovich // Информационные технологии: материалы 84-й науч.-техн. конф., посвящ. 90-летнему юбилею БГТУ и дню белорусской науки (с междунар. участием), Минск, 3–14 февр. 2020 г. / Белорус. гос. технол. ун-т. – Минск, 2020. – С. 10–14.

37. Stallings, W. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud / W. Stallings. – 1st edition. – Indianapolis: Published by Addison-Wesley Professional, 2016. – 560 p.

38. Урбанович, П. П. Элементы современных компьютерных сетей и сетевых технологий / П. П. Урбанович, М. Д. Плонковски // Передовые технологии и материалы будущего: сб. ст. IV Междунар. науч.-техн. конф. «Минские научные чтения – 2021», Минск, 9 дек. 2021 г.: в 3 т. / Белорус. гос. технол. ун-т. – Минск, 2021. – Т. 3. – С. 240–246.

39. Блащак, М. Атаки на многопользовательские компьютерные игры и некоторые методы защиты от них / М. Блащак, П. П. Урбанович // Труды БГТУ. Сер. 3, Физ.-мат. науки и информатика. – 2020. – № 1 (230). – С. 43–49.

40. Обзор интернета вещей // Рекомендация МСЭ-T/Y.2060. Сер. Y: Глобальная информационная инфраструктура, аспекты протокола Интернет и сети последующих поколений / Междунар. союз электросвязи. – Женева, 2012. – 22 с.

41. An agent-based QoE monitoring strategy for LTE networks / E. Grigoriou [at al.] // IEEE International Conference on Communications (ICC). – Thessaloniki, 2018. – P. 1–6.

42. Митник, К. Искусство быть невидимым. Как сохранить приватность в эпоху Big Data / К. Митник. – М.: Эксмо, 2019. – 464 с.

43. Jankowski, B. PadSteg: Introducing Inter-Protocol Steganography / B. Jankowski, W. Mazurczyk, K. Szczypiorski // Arxiv [Electronic resource]. – Mode of access: https://arxiv.org/ftp/arxiv/papers/1104/1104.0422.pdf. – Date of access: 15.11.2021.

44. Shutko, N. A method of syntactic text steganography based on modification of the document-container aprosh / N. Shutko, P. Urbanovich, P. Zukowski // Przegląd Elektrotechniczny. – 2018. – Wyp. 94, № 6. – S. 82–85.

45. Urbanovich, P. Theoretical Model of a Multi-Key Steganography System / P. Urbanovich, N. Shutko // Recent Developments in Mathematics and Informatics. Contemporary Mathematics and Computer Science. – 2016. – Vol. 2, Chapter 11. – P. 181–202.

46. Блинова, Е. А. Стеганографический метод на основе встраивания дополнительных значений координат в изображения формата SVG / Е. А. Блинова, П. П. Урбанович // Труды БГТУ. Сер. 3, Физ.-мат. науки и информатика. – 2018. – № 1 (206). – С. 104–109.

47. Вахаб, А. Методы цифровой стеганографии на основе модификации цветовых параметров изображения / А. Вахаб, Д. М. Романенко // Труды БГТУ. Сер. 3, Физ.-мат. науки и информатика. – 2018. – № 1 (206). – С. 94–98.

48. Блинова, Е. А. Стеганографический метод на основе встраивания скрытых сообщений в кривые Безье изображений формата SVG / Е. А. Блинова, П. П. Урбанович // Журнал Белорусского государственного университета. Математика. Информатика. – 2021. – № 3. – С. 68–83.

49. Lampson, B. W. A note on the confinement problem / B. W. Lampson // Communications of the ACM. – 1973. – P. 613–615.

50. Zander, S. Covert channels and countermeasures in computer network protocols / S. Zander, G. Armitage, P. Branch // IEEE Communications Surveys & Tutorials. – 2007. – No. 9 (3). – P. 44–57.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

РУССКОЯЗЫЧНЫЕ ТЕРМИНЫ И ПОНЯТИЯ

Адаптер (Adapter) – устройство либо программа для согласования параметров входных и выходных сигналов в целях сопряжения объектов.

Административная система (Management System) – система, обеспечивающая управление сетью либо ее частью.

Адрес (Address) – закодированное обозначение пункта отправления либо назначения данных.

Адрес IP – адрес, однозначно определяющий компьютер в сети (адрес состоит из 32 двоичных разрядов и не может повторяться во всей сети TCP/IP). Адрес IP обычно разбивается на четыре октета по восемь двоичных разрядов (один байт); каждый октет преобразуется в десятичное число и отделяется точкой, например 102.54.94.97.

Альтернативный поток данных (Alternative Data Stream) – метаданные, связанные с объектом файловой системы NTFS: файл в NTFS представляет собой набор потоков, в которых хранятся данные. По умолчанию все данные находятся в основном потоке, но при необходимости к файлу можно добавлять дополнительные, альтернативные потоки данных. Альтернативные потоки данных дают возможность добавлять к файлу скрытую информацию.

Аналоговый сигнал (Analog Signal) – сигнал, величина которого непрерывно изменяется во времени. Аналоговый сигнал обеспечивает передачу данных путем непрерывного изменения во времени.

Аналого-дискретное преобразование (Analog-to-Digital Conversion) – процесс преобразования аналогового сигнала в дискретный сигнал.

Анонимные подключения (Anonymous Connections) – эта функция, которая разрешает удаленный доступ к ресурсам компьютера по учетной записи компьютера без предъявления имени и пароля с правами, определяемыми этой учетной записью.

Архитектура (Architecture) – концепция, определяющая модель, структуру, выполняемые функции и взаимосвязь компонентов сети. Архитектура охватывает логическую, физическую и программную

структуры и функционирование сети, а также элементы, характер и топологию взаимодействия элементов.

Асинхронная передача (Asynchronous Transfer) – метод передачи основанный на пересылке данных по одному символу. При этом промежутки между передачами символов могут быть не равными.

База данных (БД) – совокупность взаимосвязанных данных, организованная по определенным правилам в виде одного или группы файлов.

Базовый порт ввода-вывода (Base I/O Port) – адрес памяти, по которому центральный процессор и адаптер проверяют наличие сообщений, которые они могут оставлять друг для друга.

Безопасность данных (Data Security) – концепция защиты программ и данных от случайного либо умышленного изменения, уничтожения, разглашения, а также несанкционированного использования.

Блок данных (Data Unit) – последовательность символов фиксированной длины, используемая для представления данных или самостоятельно передаваемая в сети.

Бод (Baud) – термин, используемый для измерения скорости модема, который описывает количество изменений состояния, происходящих за одну секунду в аналоговой телефонной линии.

Булева алгебра (Boolean Algebra) – алгебраическая структура с тремя операциями: И, ИЛИ, НЕ.

Буфер (Buffer) – временная область, которую устройство использует для хранения входящих данных перед тем, как они смогут быть обработаны на входе, или для хранения исходящих данных до тех пор, пока не появится возможность их передачи.

Виртуальная сеть (Virtual Network) – сеть, характеристики которой в основном определяются ее программным обеспечением.

Виртуальные локальные вычислительные сети (ВЛВС) – логические наложения на коммутируемое объединение сетей, определяющие группы пользователей. Это означает, что пользователь или система, подключенные к физическому порту, могут участвовать в нескольких ВЛВС-группах, поскольку логическая сеть не обязана подчиняться ограничениям физической. Границы ВЛВС задают область локального вещания. Обычно потоки данных в ВЛВС коммутируются на уровне 2, в то время как трафик между ВЛВС маршрутизируется с использованием внешнего маршрутизатора.

Витая пара (Twisted-Pair Cable) – два скрученных изолированных провода, которые используются для передачи электрических сигналов.

Волновое сопротивление, импеданс (Impedance) – полное электрическое сопротивление переменному току, включающее активную и реактивную составляющие. Измеряется в омах.

Выделенная линия (Dedicated Line) – (точка – точка) частная или адресуемая линия, наиболее популярная в глобальных вычислительных сетях. Обеспечивает полнодуплексную полосу пропускания в результате установки постоянного соединения каждой оконечной точки через мосты и маршрутизаторы с несколькими ЛВС.

Выделенный сервер (Dedicated Server) – сетевой сервер, который действует только как сервер и не предназначен для использования в качестве клиентской машины.

Гигабайт (Gigabyte) – обычно 1000 Мбайт. Точно 1024 Мбайт, где 1 Мбайт равен 1 048 576 байтам.

Гиперсреда (Hyperenvironment) – технология представления любых видов информации в виде блоков, ассоциативно связанных друг с другом, не требующая подтверждения о приеме от принимающей стороны.

Гипертекст (Hypertext) – текст, представленный в виде ассоциативно связанных друг с другом блоков.

Гипертекстовый протокол HTTP – протокол сети Internet, описывающий процедуры обмена блоками гипертекста.

Главный контроллер домена (Primary Domain Controller, PDC) – это компьютер, на котором устанавливается Windows NT Server в режиме PDC для хранения главной копии базы данных учетных записей.

Глобальная вычислительная сеть, ГВС (Wide Area Network, WAN) – компьютерная сеть, использующая средства связи дальнего действия.

Группа (Group) – совокупность пользователей, определяемая общим именем и правами доступа к ресурсам.

Данные (Data) – информация, представленная в формализованном виде, пригодном для автоматической обработки при возможном участии человека.

Дейтаграммы (Datagrams) – сообщения, которые не требуют подтверждения о приеме от принимающей стороны. Термин, используемый в некоторых протоколах для обозначения пакета.

Дефрагментация (Defragmentation) – процесс воссоздания больших PDU (пакетных блоков данных) на более высоком уровне из набора более мелких PDU с нижнего уровня.

Диагностическое программное обеспечение (Diagnostic Software) – специализированные программы или специфические системные компоненты, которые позволяют исследовать и наблюдать систему с целью определения корректности функционирования при необходимости определения причины проблемы.

Дискретный сигнал (Discrete Signal) – сигнал, имеющий конечное, обычно небольшое, число значений. Практически всегда дискретный сигнал имеет два либо три значения. Нередко его называют также цифровым сигналом.

Домен (Domain) – совокупность компьютеров, использующих операционную систему Windows NT Server, имеющих общую базу данных и систему защиты. Каждый домен имеет неповторяющееся имя.

Доменная система имен (Domain Name System, DNS) – система обозначений для сопоставления адресов IP и имен, понятных пользователю, используется в сети Internet. Система DNS иногда называется службой DNS.

Доступ (Access) – операция, обеспечивающая запись, модификацию, чтение или передачу данных.

Драйвер (Driver) – компонент операционной системы, взаимодействующий с внешним устройством или управляющий выполнением программ.

Драйвер устройства (Device Driver) – это программа, которая обеспечивает взаимодействие между операционной системой и конкретными устройствами с целью ввода-вывода данных для этого устройства.

Единица данных протокола (Protocol Data Unit, PDU) – это единичный блок информации, передаваемый между одноранговыми объектами компьютерной сети. PDU состоит из управляющей информации, зависящей от протокола, и пользовательских данных.

Единообразный локатор ресурсов (Uniform Resource Locator, URL) – идентификатор, или адрес ресурсов, в сети Internet. Обеспечивает гипертекстовые связи между документами WWW.

Жесткий диск (Hard Disk) – накопитель данных в вычислительных системах.

Заголовок кадра (Frame Preamble) – служебная информация канального уровня модели OSI, добавляемая в начало кадра.

Запрос прерывания (IRQ, Interrupt Request) – сигнал, посылаемый центральному процессору от периферийного устройства. Сообщает о событии, обработка которого требует участия процессора.

Запросчик, или редиректор (Requester, LAN Requester), – программа, находящаяся на компьютере клиенте. Переадресует на соответствующий сервер запросы на сетевые услуги со стороны работающих на этом же компьютере приложений.

Затухание (Attenuation) – ослабление сигнала при удалении его от точки испускания.

Звезда (Star Topology) – вид топологии, при котором каждый компьютер подключен к центральному компоненту, называемому концентратором.

Зеркальные диски (Disk Mirroring) – уровень 1 технологии RAID, при которой часть жесткого диска (или весь жесткий диск) дублируется на одном или нескольких жестких дисках. Позволяет создавать резервную копию данных.

Импульсно-кодовая модуляция, ИКМ (Pulse Code Modulation, PCM) – метод преобразования аналогового сигнала телефонии в дискретный сигнал.

Интернет – совокупность компьютеров, объединенных в глобальную сеть.

Информационная сеть (Information Network) – сеть, предназначенная для обработки, хранения и передачи данных.

Информационная система (Information System) – объект, способный осуществлять хранение, обработку или передачу данных. К этой системе относятся: компьютеры, программы, пользователи и другие составляющие, предназначенные для процесса обработки и передачи данных.

Информационно-поисковая система – (Information Retrieval System, IRS) – система, предназначенная для поиска информации в базе данных.

Информация (Information) – данные, обработанные адекватными им методами.

Инфракрасный канал (Infrared Channel) – канал, использующий для передачи данных инфракрасное излучение. Инфракрасный канал работает в диапазоне высоких частот, где сигналы мало подвержены электрическим помехам.

Кабель (Cable) – один либо группа изолированных проводников, заключенных в герметичную оболочку.

Кадр (Frame) – блок информации канального уровня.

Кадр данных (Data Frame) – базовая упаковка битов, которая представляет собой PDU (пакетный блок данных), посланный с одного компьютера на другой по сетевому носителю.

Канал (Channel, Link) – среда или путь передачи данных.

Канал передачи данных (Data Channel) – кабели и инфраструктура сети.

Канальный уровень (Date Link Layer) – второй уровень модели OSI. Здесь из последовательности битов, поступающих от физического уровня, формируются кадры.

Квитирование, или *рукопожатие* (Handshake), – подтверждение установления связи.

Клиент (Client) – компьютер в сети, который запрашивает ресурсы или услуги от некоторых других компьютеров.

Клиент-сервер (Client-Server) – модель вычислений, при которой некоторые компьютеры запрашивают услуги (это клиенты), а другие отвечают на такие запросы на услуги (это серверы).

Коаксиальный кабель (Coaxial Cable) – кабель, состоящий из изолированных друг от друга внутреннего и внешнего проводников. Коаксиальный кабель имеет один либо несколько центральных медных проводников, покрытых диэлектрической изоляцией, которая для защиты центральных проводников от внешних электромагнитных воздействий покрыта металлической оплеткой (сеткой) либо трубкой.

Коллизия (Collision) – ситуация, когда две рабочие станции пытаются одновременно занять канал (использовать рабочую среду – кабель).

Коммуникационная сеть (Communication Network) – сеть, предназначенная для передачи данных, также она выполняет задачи, связанные с преобразованием данных.

Коммутатор (Switch) – устройство или программа, осуществляющие выбор одного из возможных вариантов направления передачи данных.

Коммутатор кадров (Frame Switch) – многопортовый мост уровня доступа к среде передачи, работающий со скоростью этой среды и гарантирующий на порядок более высокую пропускную способность при связывании клиентских и серверных систем по сравнению с концентраторами для среды с разделяемым доступом. При сегментации ЛВС коммутаторы кадров обеспечивают лучшие показатели цена/производительность и меньшие задержки, чем традиционные связки мостов и маршрутизаторов.

Коммутатор ячеек (Cell Switch) – устройство, реализующие ATM-коммутацию данных, разделенных на короткие ячейки фиксированного размера. Ориентация на установление соединений позволяет

ATM обеспечивать классы (качество) обслуживания, пригодные для всех видов мультимедийного трафика, включая данные, голос и видео.

Концентратор, или *хаб* (Concentrator or Hub), – связующий компонент сети, к которому подключаются все компьютеры в сети топологии «звезда». Концентратор обеспечивает связь компьютеров друг с другом при использовании витой пары, также применяется в сетях FDDI для подключения компьютеров в центральном узле.

Концентратор (Multi Station Access Unit, MSAU) – устройство для доступа ко множеству станций, осуществляющее маршрутизацию пакета к следующему узлу в сетях на основе метода доступа с передачей маркера.

Корпоративная сеть (Corporate Network) – крупномасштабная сеть, обычно соединяющая многие локальные сети.

Логический диск (Logical Disk) – часть физического диска, отформатированная под конкретную файловую систему и имеющая свое буквенное наименование.

Логический канал (Logical Channel) – путь, по которому данные передаются от одного порта к другому. Логический канал прокладывается в одном либо нескольких последовательно расположенных физических каналах и через уровни области взаимодействия.

Локальная группа (Local Group) – в Windows NT Server учетная запись, определенная на конкретном компьютере. Включает учетные записи пользователей данного компьютера.

Локальная сеть (Local-Area Network) – сеть, системы которой расположены на небольшом расстоянии друг от друга.

Магистраль (Backbone) – основной кабель, от которого кабели трансиверов идут к компьютерам, повторителям и мостам.

Манчестерское кодирование (Manchester Encoding) – схема передачи двоичных данных, применяемая во многих сетях. При передаче бита, равного 1, в течение временного интервала, который отведен для его передачи, значение сигнала меняется с положительного на отрицательное. При передаче бита, равного 0, в течение временного интервала, который отведен для его передачи, значение сигнала меняется с отрицательного на положительное.

Маркер (Token) – уникальная комбинация битов. Получая маркер, рабочая станция в ЛВС имеет право начать передачу данных.

Маршрутизатор (Router) – протокол, ориентированное устройство, соединяющее две сети, иногда с абсолютно разными уровнями MAC (канальный уровень, контроль доступа к среде).

Маршрутизация (Routing) – процесс определения в коммуникационной сети пути, по которому блок данных может достигнуть адресата.

Маска сети (Network Mask) – 32-битное число, по которому можно определить диапазон IP-адресов, находящихся в одной IP-сети/подсети.

Масштабируемость (Scalability) – это возможность увеличить вычислительную мощность Web-сайта или компьютерной системы (в частности, выполнение большего числа операций или транзакций за определенный период времени) за счет установки большего числа процессоров или их замены на более мощные.

Мегабайт (Megabyte) – 1 048 576 байтов.

Метод доступа (Access Method) – способ определения рабочей станции, которая сможет следующей использовать ЛВС. Кроме того, точно так называется набор правил, используемых сетевым оборудованием, чтобы направлять поток сообщений через сеть, а также один из основных признаков, по которым различают компоненты сетевого оборудования.

Метод доступа к каналу (Channel Access Method) – правила, используемые для определения компьютера, который может посылать данные по сети, тем самым предотвращающие потерю данных из-за коллизий.

Метод множественного доступа с прослушиванием несущей и разрешением коллизий (Carrier Sense Multiple Access with Collision Detection, CSMA/CD) – метод доступа к каналу связи, который устанавливает следующий порядок: если рабочая станция хочет воспользоваться сетью для передачи данных, она сначала должна проверить состояние канала, начинать передачу станция может, если канал свободен. В процессе передачи станция продолжает прослушивание сети для обнаружения возможных конфликтов. Если конфликт возникает в случае, когда два узла попытаются занять канал, то обнаружившая конфликт интерфейсная плата выдает в сеть специальный сигнал и обе станции одновременно прекращают передачу.

Метод обработки запросов по приоритету (Request Processing Method by Priority) – метод доступа к каналу связи, где всем узлам сети предоставляется право равного доступа. Концентратор опрашивает каждый порт и проверяет наличие запроса на передачу, затем решает этот запрос в соответствии с приоритетом.

Метод с передачей маркера или полномочия (Token Passing Multiple Access, TPMA) – метод доступа к каналу связи, в котором от

компьютера к компьютеру передается маркер, дающий разрешение на передачу сообщения. При получении маркера рабочая станция может передавать сообщение, присоединяя его к маркеру, который переносит его по сети. Каждая станция, находящаяся между передающей и принимающей, «видит» это сообщение, но только станция-адресат принимает его. При этом она создает новый маркер.

Микроядро (Microkernel) – это центральная часть операционной системы, которая выполняет основные функции управления системой.

Модем (Modem) – сокращение от МОДулятор/ДЕМодулятор. Устройство связи, которое позволяет компьютеру передавать данные по обычной телефонной линии. При передаче преобразует цифровые сигналы в аналоговые, а при приеме наоборот – аналоговые в цифровые.

Монитор сети (Network Monitor) – это программно-аппаратное устройство, которое отслеживает сетевой трафик. Проверяет пакеты на уровне кадров, собирает информацию о типах пакетов и ошибках.

Мост (Bridge) – это прибор, позволяющий рабочим станциям одной сети обращаться к рабочим станциям другой. Мосты используются для разделения ЛВС на маленькие сегменты, выполняют соединение на канальном уровне модели OSI. Мост преобразует физический и канальный уровни различных типов, используется для увеличения длины или количества узлов.

Мост-маршрутизатор (Bridge-Router) – сетевое устройство, которое объединяет лучшие функции моста и маршрутизатора.

Мультиплексор (Multiplexor) – устройство, позволяющее разделить канал передачи на два или более подканала. Может быть реализован программно. Кроме того, используется для подключения нескольких линий связи к компьютеру.

Нейронная сеть (Neural Network) – сеть, образованная взаимодействующими друг с другом нервными клетками либо моделирующими их поведение компонентами.

Несущая (Carrier) – непрерывный сигнал, на который накладывается другой сигнал, несущий информацию.

Неэкранированная витая пара (UTP, Unshielded Twisted Pair) – кабель, в котором изолированная пара проводников скручена с небольшим числом витков на единицу длины. Скручивание проводов уменьшает электрические помехи извне при распространении сигналов по кабелю.

Оболочка (Shell) – программное обеспечение, которое реализует взаимодействие пользователя с операционной системой (пользовательский интерфейс).

Обработка запросов по приоритету (Demand Priority) – высокоскоростной метод доступа к каналу, используемый сетями 100VG-Any LAN в топологии «звезда».

Общий ресурс (Shared Resource) – любое устройство, данные или программа.

Одноранговая архитектура (Peer-to-Peer Architecture) – концепция информационной сети, в которой каждая абонентская система может предоставлять и потреблять ресурсы.

Октет – байт.

Оперативная память (Main Memory) – память, предназначенная для хранения данных и команд, необходимых процессору для выполнения им операций.

Оптический кабель (Optical Cable) – кабель, передающий сигналы света. Для создания оптического кабеля используются световоды, каждый из которых имеет несколько слоев защитных покрытий, улучшающих механические и оптические характеристики этих световодов.

Оптический канал (Optical Channel) – канал, предназначенный для передачи сигналов света.

Оптоволокно (Optical Fiber) – среда, по которой цифровые данные передаются в виде модулированных световых импульсов.

Пакет (Packet) – это единица информации, передаваемой между станциями сети. Используется на сетевом уровне модели OSI.

Пароль (Password) – признак, подтверждающий право пользователя или прикладной программы на использование какого-нибудь ресурса.

Передача данных (Data Communications) – процесс транспортирования данных из одной системы в другую.

Повторитель, или *репитер* (Repeater), – устройство, усиливающее сигналы с одного отрезка кабеля и передающее их в другой отрезок без изменения содержания. Повторители увеличивают максимальную длину трассы ЛВС.

Полномочие (Token) – специальный символ или группа символов, разрешающая системе передачу кадров.

Полоса пропускания (Bandwidth) – разность между максимальной и минимальной частотой в заданном диапазоне; диапазон частот, на которых может работать носитель.

Пользователь (User) – юридическое либо физическое лицо, использующее какие-либо ресурсы, возможности.

Порт (Port) – точка доступа к устройству либо программе. Различают физические и логические порты.

Провайдер (Provider) – организация, которая обеспечивает подключение к Internet и другие услуги за определенную плату.

Протокол (Protocol) – набор правил, регламентирующих порядок сборки пакетов, содержащих данные и управляющую информацию на рабочей станции-отправителе для передачи их по сети, а также порядок разборки пакетов по достижении ими рабочей станции-получателя.

Рабочая станция (*станция, компьютер, ПК, клиент*) – это рабочие станции, которые используют ресурсы сервера и предоставляют удобные интерфейсы пользователя.

Распределитель (Hub) – центр ЛВС или кабельной системы с топологией «звезда». В этой роли могут быть файл-серверы или концентраторы. Они содержат сетевое программное обеспечение и управляют коммуникациями внутри сети, а также могут работать как шлюзы к другим ЛВС.

Редиректор для ОС (Redirector) – сетевое программное обеспечение, которое принимает запросы ввода-вывода для удаленных файлов, именованных каналов или почтовых слотов и затем переназначает их сетевым сервисам другого компьютера. Для Windows NT редиректоры выполнены как драйверы файловой системы.

Редиректор для протоколов (Redirector) – компонент набора протоколов или сетевой операционной системы, ответственный за перехват запросов от приложений и распределение их между локальной или удаленной службами сети.

Реестр (Registry) – архив БД Windows NT для хранения информации о конфигурации компьютера, включая аппаратные средства, установленное программное обеспечение, установки окружения и др.

Сеанс (Session) – сообщение, в котором предполагается создание логической связи для обмена сообщениями. Сначала сеанс должен быть установлен, затем происходит обмен сообщениями. После окончания обмена сеанс необходимо закрыть.

Сегмент (Segment) – часть сети, ограниченная ретранслирующими устройствами (повторителями, мостами, маршрутизаторами и шлюзами).

Сервер (Server) – это компьютер сети, предоставляющий сервис другим объектам по их запросам.

Сервис (Service) – процесс обслуживания объектов.

Сетевая служба (Network Service) – вид сервиса, предоставляемого сетью.

Сеть (Network) – взаимодействующая совокупность сетевых узлов, связанных друг с другом каналами связи, предназначенная для передачи информации.

Слот адаптера (Adapter Slot) – гнездо, встроенное в материнскую плату.

Стандарт RS-232 – промышленный стандарт для последовательных соединений.

Структурная избыточность (Structure Redundancy или Hardware Redundancy) – структурный параметр, отражающий превышение общего числа связей и/или элементов системы над необходимым минимальным числом связей и/или элементов.

Телекоммуникация (Telecommunication) – область деятельности, предметом которой являются методы и средства передачи информации.

Терминал (Terminal) – устройство ввода-вывода данных и команд в систему или сеть.

Тестирование (Testing) – процесс проверки правильности функционирования устройства либо программного обеспечения.

Тип кадра (Frame Type) – один из четырех стандартов, которые определяют структуру пакета Ethernet: Ethernet 802.3, Ethernet 802.2, Ethernet SNAP или Ethernet II.

Транзакция (Transaction) – короткий во времени цикл взаимодействия объектов, включающий запрос – выполнение задания – ответ.

Трансивер (Transciever) – устройство, предназначенное для передачи данных с сетевых интерфейсных плат в физическую среду.

Трафик (Traffic) – поток данных.

Удаленная регистрация (Remote Logon) – подключение по сети к другому компьютеру пользователя, зарегистрированного на своем ПК по своей учетной записи.

Удаленный доступ (Dial-up) – доступ к системе или по сети к другому компьютеру пользователя, зарегистрированного на своем ПК по своей учетной записи.

Узел (Node) – точка присоединения к сети; устройство, подключенное к сети.

Утилита (Utility) – программа, выполняющая какую-либо функцию сервиса.

Учетная запись (Account) – это информация, которая хранится в базе данных Windows NT (учетная запись пользователя, компьютера, группы).

Факсимильная связь (Faximile) – процесс передачи через коммуникационную сеть неподвижных изображений и текста.

Фрагментация (Fragmentation) – процесс разделения длинного пакета данных с более высокого уровня на последовательность более коротких пакетов на нижнем уровне.

Центральный процессор (Central Processing Unit, CPU) – управляющий и вычислительный модуль компьютера. Устройство, которое интерпретирует и выполняет команды.

Циклический избыточный код (Cyclical Redundancy Code, CRC) – число, получаемое в результате математических преобразований над пакетом данных и исходными данными. При доставке пакета вычисления повторяются. Если результат совпадает, то пакет принят без ошибок.

Цифровая линия (Digital Line) – линия связи, передающая информацию только в двоичной (цифровой) форме.

Цифровая сеть комплексных услуг (Integrated Services Digital Network, ISDN) – цифровая сеть связи, обеспечивающая коммутацию каналов и коммутацию пакетов.

Четность (Parity) – способ контроля за безошибочной передачей блоков данных с помощью добавления контрольных битов.

Шина (Bus) – канал передачи данных, отдельные части которого называются сегментами.

Широковещательная передача (Broadcast) – технология передачи сигналов, таких как сетевые данные, посредством использования передатчика какого-либо типа для посылки этих сигналов по коммуникационному носителю.

Шифрование (Encryption) – преобразование информации для ее защиты от несанкционированного доступа.

Шлюз (Gateway) – устройство, посредством которого соединяются сети разных архитектур.

Экран (Shielding) – металлическая оплетка или цилиндр, навитый из фольги. Защищает передаваемые данные, уменьшая внешние электрические помехи, которые называются шумом.

Экранированная витая пара (Shielded Twisted-Pair, STP) – витая пара, окруженная заземленной металлической оплеткой, которая служит экраном.

Электронная почта (e-mail) – компьютерная система обмена сообщениями, где текст и файлы могут быть посланы от одного пользователя к одному или многим другим пользователям в той же сети.

Эталонная модель взаимодействия открытых систем (OSI, Open System Interconnection) – семиуровневая модель, которая стандартизирует уровни услуг и виды взаимодействия между системами в информационной сети при передаче данных.

Эфир (Ether) – пространство, через которое распространяются волны электромагнитного спектра и прокладываются каналы радиосетей и инфракрасных сетей. Электромагнитное поле не нуждается в специальном носителе.

Язык HTML – инструментальное программное обеспечение, использующее технологию гипертекста.

Язык описания страниц (Page Description Language) – язык программирования, который описывает вид страницы для печати. Используется для компоновки изображения страницы.

Язык структурированных запросов (SQL, Structured Query Language) – язык управления базами данных, используемый для запроса, обновления и управления реляционными базами данных.

Ячеистая топология сети (Mesh Network Topology) – топология, используемая в глобальных вычислительных сетях. К любому узлу существует несколько маршрутов.

АНГЛОЯЗЫЧНЫЕ ТЕРМИНЫ И ПОНЯТИЯ

10Base2 (известный как «тонкий» Ethernet) – обозначение технологии Ethernet по стандарту IEEE 802.3 со скоростью передачи данных 10 Мбит/с для тонкого коаксиального кабеля.

10Base5 (также известен как «толстый» Ethernet) – обозначение технологии Ethernet по стандарту IEEE 802.3 со скоростью передачи данных 10 Мбит/с для толстого коаксиального кабеля.

10Base-FL – стандарт на сегменты сети Ethernet на оптоволоконном кабеле.

10Base-T – обозначение технологии Ethernet в соответствии со стандартом IEEE 802.3 со скоростью передачи данных 10 Мбит/с для кабеля «витая пара».

100Base-FX – обозначение технологии Fast Ethernet по стандарту IEEE 802.3 сети Fast Ethernet для передачи больших сообщений по многомодовому оптоволокну в полудуплексном и полнодуплексном режимах.

100Base-T4 – обозначение технологии Fast Ethernet по стандарту IEEE 802.3 со скоростью 100 Мбит/с для четырехпарной витой пары. Вместо кодирования 4B/5B в этом методе используется кодирование 8B/6T.

100Base-TX – обозначение технологии сети Fast Ethernet по стандарту IEEE 802.3 для передачи больших сообщений с использованием метода MLT-3 для передачи сигналов 5-битовых порций кода 4B/5B по витой паре, а также наличие функции автопереговоров (Auto-negotiation) для выбора режима работы порта.

1000Base-CX – стандарт на сегменты сети Gigabit Ethernet на экранированной витой паре.

1000Base-LX – стандарт на сегменты сети Gigabit Ethernet на оптоволоконном кабеле с длиной волны света 1,3 мкм.

1000Base-SX – стандарт на сегменты сети Gigabit Ethernet на оптоволоконном кабеле с длиной волны света 0,85 мкм.

Access – см. термин «доступ».

Access Auditing (*Access Control*) – контроль доступа.

Accounting Management – управление учетом использования ресурсов (сети).

Addressing – адресация, способ указания объектов в сети либо в системе.

Administration – администрирование, управление сетью.

Analog Network – аналоговая сеть, передающая и обрабатывающая аналоговые сигналы.

Analog-to-Digital Conversion – аналого-дискретное преобразование, процесс преобразования аналогового сигнала в дискретный.

Animation – анимация, виртуальная реальность, мнимый мир, создаваемый аудиовидиосистемой в воображении пользователя.

Application layer – прикладной уровень модели OSI, обеспечивающий прикладным процессам средства доступа к области взаимодействия.

Archivator – архиватор, программа, обеспечивающая сжатие данных.

Arithmetic and Logical Unit (ALU) – арифметико-логическое устройство, часть процессора, выполняющая арифметические и логические операции над данными.

Asynchronous Transfer Mode (ATM) – асинхронный способ передачи данных, пакетно-ориентированный метод скоростной передачи.

Banyan Network – баньяновая сеть, скоростная распределительная сеть с каскадной адресацией.

Baud – бод, единица скорости передачи данных. Число бод равно количеству изменений сигнала (потенциала, фазы, частоты), происходящих в секунду. Для двоичных сигналов нередко считают, что бод равен биту в секунду, например 1200 бод = 1200 бит/с.

Beaconing (*iBeacon)* – технология на основе API-сервиса iOS, которая начиная с версии 7 позволяет передавать данные между беспроводными устройствами – маяками (*beacon*) – и устройствами, поддерживающими Bluetooth.

Binary Code – двоичный код, алфавит кода ограничен двумя символами (0, 1).

Bipolar Code – биполярный код. Алфавит кода ограничен тремя символами (–1, 0, +1), где единицы представляются чередующимися импульсами. Отсутствие импульсов определяет состояние нуля.

Bit (BInary digiT) – бит, наименьшая единица информации в двоичной системе счисления.

Bridge – мост, сетевое оборудование для преобразования физического и канального уровней различных типов.

Broadband Channel – широкополосный канал.

Broadcasting – широковещание.

Bus – шина.

Byte – байт, единица количества информации, равная восьми битам.

Cable – кабель, длинномерное изделие для передачи сигналов.

Cache Memory – кэш-память, буферное запоминающее устройство, работающее со скоростью, обеспечивающей функционирование процессора без режимов ожидания.

Carrier – несущая, непрерывный сигнал, на который накладывается другой сигнал, дающий информацию.

Cellular Packet Radio Network – сотовая пакетная радиосеть.

Channel – канал, среда или путь, по которому передаются данные.

Circuit Switching – коммутация каналов, предоставление последовательности каналов сети для монопольного использования при передаче данных во время сеанса.

Client – клиент, объект использующий сервис, предоставляемый другими объектами.

Client-Server Architecture – архитектура «клиент-сервер».

Clock Rate – тактовая частота.

Closed Channel – закрытый канал.

Communication Network – коммуникационная сеть, предназначенная для передачи данных, также она выполняет задачи, связанные с преобразованием данных.

Communication Protocol Stack – стек коммуникационных протоколов.

Compiler – компилятор, программа-транслятор, преобразующая код в язык машинных команд (исполняемый файл).

Confidention – конфиденциальность, доверительность, секретность.

Configuration Management – управление конфигурацией.

Conformance – конформность, соответствие объекта его нормативно-технической документации. Конформность объекта определяется в результате процесса его тестирования.

Connection – соединение.

Console – консоль, одна либо несколько абонентских систем для работы с платформой управления сетью.

Covert Channel – скрытый канал.

Databank – банк данных.

Database – база данных.

Database Management System (DBMS) – система управления базой данных (СУБД).

Database Server – сервер базы данных.

Datagram – дейтаграмма, сообщение, которое не требует подтверждения о приеме от принимающей стороны.

Data Link Layer – канальный уровень, уровень модели OSI, отвечающий за формирование и передачу блоков данных и обеспечивающий доступ к каналу связи области взаимодействия.

Data Management – управление данными.

Data Processing – обработка данных.

Data Protection – защита данных.

Data Security – безопасность данных.

Data Security Architecture – архитектура безопасности данных, архитектура, определяющая методы и средства защиты данных.

Data Transfer – пересылка данных.

Data Unit – блок данных.

Decoding – декодирование.

Dedicated Channel – выделенный канал.

Designator – распределитель.

Determinate Access – детерминированный доступ, множественный доступ.

Device – устройство.

Diagnostic – диагностика.

Dialog – диалог.

Digital Network – дискретная сеть.

Digital Signal – цифровой сигнал, дискретный сигнал.

Digit-to-Analog Conversion – дискретно-аналоговое преобразование, процесс преобразования дискретного сигнала в аналоговый.

Direct Memory Access (DMA) – прямой доступ к памяти.

DirectDraw – часть набора драйверов DirectX, поддерживающих непосредственную работу с видеокартой и позволяющих, например, прямую запись в видеопамять.

Directory – каталог.

Directory Network Service – сетевая служба каталогов.

DirectX – набор драйверов, образующий интерфейс между программами в среде Windows и аппаратными средствами.

Disk Drive – дисковод.

Disk Operating System (DOS) – дисковая операционная система (ДОС).

Domain – домен, группа компьютеров, находящаяся в одном месте (здание, этаж, организация) и управляемая СОС.

Driver – компонент операционной системы, взаимодействующий с устройством либо управляющий выполнением программ.

Duplex Channel – дуплексный канал, осуществляет передачу данных в обоих направлениях.

Electronic Mail – электронная почта, средства передачи сообщений между пользователями в сети.

Emulation – эмуляция, организация структуры одного объекта, при которой его функционирование неотличимо от другого объекта.

Encryption – шифрование (зашифрование), способ изменения данных с целью засекречивания.

Enterprise Network – корпоративная сеть, локальная сеть большого предприятия.

Ether – эфир, пространство, через которое распространяются волны электромагнитного спектра и прокладываются каналы, радиосетей и инфракрасных сетей.

Ethernet Network – сеть Ethernet, тип локальной сети, предложенный корпорацией Xerox.

Explorer – программа-браузер для просмотра Web-страниц.

External Device – внешнее устройство.

External Memory – внешняя память, непосредственно не доступная процессору.

Fast Ethernet – сеть Fast Ethernet, тип скоростной сети Ethernet со скоростью передачи данных 100 Мбит/с.

Fault Management – управление неисправностями (отказами в сети).

Faximile – факсимильная связь, процесс передачи через коммуникационную сеть неподвижных изображений и текста.

Fiber Channel Network – сеть Fiber Channel, тип скоростной локальной сети, основанной на использовании оптических каналов.

Fiber Distributed Data Interface (FDDI) – оптоволоконный распределенный интерфейс данных.

Fiber-Optic Link – волоконно-оптическая линия связи.

Flash Memory – флэш-память, память на основе полупроводниковой технологии.

Frame – кадр.

Frame Relay – ретрансляция кадров.

Frequency Band – полоса частот.

Frequency Division Multiple Access (FDMA) – множественный доступ с разделением частоты.

Frequency Modulation – частотная модуляция.

Functional Profile – функциональный профиль.

Gateway – шлюз.

Global Network – глобальная сеть.

Gopher – интерактивная оболочка для поиска, присоединения и использования ресурсов и возможностей Internet. Интерфейс с пользователем осуществлен через систему меню.

Graphic Interface – графический интерфейс.

Hardware – техническое обеспечение.

Hardware Description Language (HDL) – язык описания технических средств.

Hardware Platform – аппаратная платформа.

Heterogeneous Network – гетерогенная сеть, в которой работают системы различных фирм производителей.

Hierarchical Addressing – иерархическая адресация, при которой адреса объединяют в группы, отражая их взаимосвязь.

High-level Language – язык высокого уровня.

Host Computer – главный компьютер в архитектуре «терминал – главный компьютер».

Hypermedia – гиперсреда.

Hypertext – гипертекст.

Hypertext Markup Language (HTML) – гипертекстовый язык разметки.

Hypertext Transfer Protocol (HTTP) – гипертекстовый протокол передачи.

Identification – идентификация.

Information – информация.

Information Network – информационная сеть.

Infrared Channel – инфракрасный канал.

Infrared Network – инфракрасная сеть.

Infrared Radiation – инфракрасное излучение.

Infrastructure – инфраструктура.

Input/output Device – устройство ввода-вывода.

Input/output Interface – интерфейс ввода-вывода.

Integrated Services Digital Network (ISDN) – цифровая сеть с интегральным обслуживанием.

Intelligent Hub – интеллектуальный концентратор. Интеллект концентраторов состоит в том, что они могут выполнять операции мониторинга и управления сетью.

Interconnection Area – область взаимодействия.

Internet Network – сеть Internet.

Interpreter – интерпретатор, программа, анализирующая построчно команды или операторы программы и непосредственно выполняющая их.

Java Language – язык Java, имеющий объектно-ориентированную архитектуру, предложенный корпорацией SUN Microsystems.

Java Script Language – язык JavaScript.

Key – ключ.

Knowledge Base – база знаний (БЗ).

Light Guide – световод.

Link Access Procedure (LAP) – процедура доступа к каналу.

Loader – загрузчик, программа, выполняющая функции загрузки объектного модуля в операционную память и динамического формирования загрузочного модуля.

Local-Area Network (LAN) – локальная сеть.

Locking – блокировка.

Logical Address – логический адрес, символический условный адрес объекта.

Logical Channel – логический канал.

Low-Level Language – язык низкого уровня.

Machine Language – машинный язык.

Macro Instruction – макрокоманда.

Manageable Hub – управляемый концентратор. Еще одно название для интеллектуальных хабов. Каждый порт управляемого концентратора можно независимо конфигурировать, включать или выключать, а также организовать его мониторинг.

Manchester Coding – манчестерское кодирование.

Message – сообщение, единица данных на прикладном уровне.

Mirroring – зеркализация.

Modular Hub – модульный концентратор, или хаб. В основе модульного хаба лежит шасси, в которое помещаются специальные платы или модули. Каждый из модулей функционирует подобно автономному концентратору, а модули взаимодействуют друг с другом через шину шасси.

Narrowband Channel – узкополосный канал.

NetWare Network – сеть NetWare.

Network – сеть.

Network Analyzer – анализатор сети.

Network Basic Input/Output System (NetBIOS) – сетевая базовая система ввода-вывода.

Network Compatibility – совместимость сети.

Network Layer – сетевой уровень.

Network Manageability – управляемость сети.

Network Management – управление сетью.

Network Operating System (NOS) – сетевая операционная система (СОС).

Network Performance – производительность сети.

Network Printer – сетевой принтер.

Network Service – сетевая служба.

Network Transparency – прозрачность сети.

Neural Network – нейронная сеть.

Notebook Personal Computer – блокнотный персональный компьютер.

Object Linking and Embedding Technology (OLE) – технология связи и внедрения объектов

Object-Oriented Architecture – объектно-ориентированная архитектура.

Object-Oriented Database (OODB) – объектно-ориентированная база данных.

Online Mode – интерактивный режим.

Optical Fiber – оптическое волокно.

Packet – пакет, единица данных на сетевом уровне.

Packet (Batch) Mode – пакетный режим.

Packets Encapsulation – инкапсуляция пакетов.

Packet Switching – коммутация пакетов.

Paging Device – пейджер, устройство радиовызова.

Parity – четность, паритет.

Pascal Language – язык Pascal.

Password – пароль.

PCI Bus – шина PCI.

Peer-to-Peer Architecture – одноранговая архитектура.

Performance Management – управление эффективностью (в сети).

Permission – разрешение.

Physical Address – физический адрес.

Physical Interconnection Facility – физические средства соединения.

Physical Layer – физический уровень.

Physical Link – физический канал.

Physical Medium – физическая среда.

Ping – утилита проверки связи с удаленной ЭВМ.

Presentation Layer – представительский уровень.

Quantization – квантование, разбиение диапазона значений аналогового сигнала на конечное число интервалов (кванты).

Quantum – квант.

Radio Channel – радиоканал.

Radio Local-Area Network – локальная радиосеть.

Radio Network – радиосеть.

Real-Time System – система реального времени, функционирование которой зависит не только от логической корректности вычислений, но и от времени, за которое эти вычисления производятся.

Redirector – редиректор.

Relational Database (RDB) – реляционная база данных.

Relay System – ретрансляционная система.

Remote Access – удаленный доступ.

Repeater – повторитель, репитер.

Resource – ресурс.

Resource Sharing – совместное использование ресурса.

Ribbon Cable – плоский кабель.

Rout – маршрут, путь.

Router – маршрутизатор.

Routing – роутинг.

Routing Table – таблица маршрутизации.

Security Management – управление защитой данных или безопасностью (сети).

Serial Interface – последовательный интерфейс.

Server – сервер.

Service – сервис.

Session – сеанс.

Session Layer – сеансовый уровень.

Sharing (разделение) – совместное использование.

Simulation – моделирование.

Software – программное обеспечение.

Stackable Hub – стековый хаб. Стековые хабы действуют как автономные устройства с единственным отличием, они позволяют организовать стек – группу концентраторов, работающих как одно логическое устройство. С точки зрения сети стек концентраторов является одним хабом.

Stand-alone – автономный.

Stand-alone Hub – автономный хаб. Устройство с несколькими (обычно от 4 до 32) портами, способное функционировать независимо. Обычно автономные концентраторы поддерживают способ наращивания числа портов.

Subnet Mask – маска подсети.

Switch – коммутатор.

Synchronizing – синхронизация.

Syntax – синтаксис.

Telecommunications – телекоммуникации.

Telefax – факс-аппарат.

Telephone Mail – электронная почта.

Telephone Network – телефонная сеть.

Telnet – удаленный доступ. Дает возможность абоненту работать на любой ЭВМ сети Internet как на своей собственной.

Time Sharing – разделение времени.

Token – полномочие.

Topology – топология.

Traffic – трафик.

Transaction – транзакция, короткий во времени цикл взаимодействия объектов, включающий запрос – выполнение задания – ответ.

Translator – транслятор, программа, преобразующая программу, написанную на одном языке, в программу, представленную на другом языке.

Transparency – прозрачность, объект считается прозрачным для пользователя либо программы в том случае, когда они, работая через (сквозь) объект, не видят его.

Transport Layer (транспортный уровень) – уровень, на котором пакеты передаются через коммуникационную сеть.

Unauthorized Access – несанкционированный доступ.

Uninterruptible Power Supply (UPS) – источник бесперебойного питания.

Unipolar Code – униполярный код.

Unique Address – уникальный адрес.

Universal Code (UNICODE) – это универсальный код, стандарт 16-разрядного кодирования символов. Код идет на смену использовавшимся до сих пор 7–8-битовым обозначениям.

UNIX Operating System (операционная система) – сетевая операционная система (СОС), созданная фирмой Bell Laboratory.

User – пользователь, юридическое либо физическое лицо, использующее какие-либо ресурсы, возможности.

User Interface – интерфейс пользователя.

Utility – утилита, программа, выполняющая какую-либо функцию сервиса.

Verification – верификация, процедура проведения анализа с целью установления подлинности, проверки истинности.

Video Board – видеоплата, одноплатный контроллер, вставляемый в компьютер и осуществляющий в режиме реального времени аналого-дискретное преобразование в потоки дискретных сигналов.

Video Bus – видеошина, предназначенная, в первую очередь, для передачи изображений.

Video Conferencing – видеоконференция, методология проведения совещаний и дискуссий между группами удаленных пользователей с использованием движущихся изображений.

Viewer – визуализатор, программа просмотра документов на экране.

Wavequide – волновод.

Whois – адресная книга сети Internet.

Wiring Concentrator – связной концентратор.

Workstation – рабочая станция, использующая ресурсы сервера и предоставляющая удобные интерфейсы пользователя.

АНГЛОЯЗЫЧНЫЕ СОКРАЩЕНИЯ

ACF (Advanced Communications Function) – дополнительная коммуникационная функция.

ACP (ANSI Code Page) – кодовая страница ANSI.

ACPI (Advanced Configuration and Power Interface) – современный интерфейс конфигурирования и управления энергопотреблением.

ACS (Advanced Connectivity System) – дополнительные системы связи.

ADC (Analog Digital Converter) – аналого-цифровой преобразователь (АЦП). Предназначен для преобразования аналогового сигнала в цифровой.

ADS (Alternate Data Streams) – альтернативный поток данных.

ADSL (Asymmetric Digital Subscriber Line) – асимметричная цифровая абонентская линия.

AFP (AppleTalk File Protocol) – файловый протокол AppleTalk). Протокол удаленного управления файлами Macintosh.

AMPS (Advanced Mobile Phone Service) – мобильная телефонная система; аналоговый стандарт сотовой связи, относящийся к сетям первого поколения (1G).

ANR (Automatic Network Routing) – автоматическая сетевая маршрутизация.

ANSI (American National Standards Institute) – Американский национальный институт стандартов.

API (Application Programming Interface) – интерфейс прикладных программ. Набор процедур, которые вызываются прикладной программой для осуществления низкоуровневых операций, исполняемых операционной системой.

APPC (Advanced Program-to-Program Communication) – высокоуровневый протокол для взаимодействия программ.

ARCU (Application-Resource-Context-User) – модель «приложение – ресурс – контекст – пользователь».

ARP (Address Resolution Protocol) – протокол разрешения адреса.

ARQ (Automatic Repeat Request) – автоматический запрос повторной передачи.

ASCII (American Standard Code for Information Interchange) – американский стандартный код для информационного обмена.

ASMP (ASymmetric Multi Processing) – асимметричная мультипроцессорная обработка.

ASP (Active Server Page) – технология, позволяющая создавать динамические Web-приложения.

ASP (Application Service Provider) – провайдер прикладных услуг.

AT (Advanced Technology) – усовершенствованная технология.

AT&T (American Telephone and Telegraph) – американский телефон и телеграф, американский транснациональный телекоммуникационный конгломерат.

ATM (Asynchronous Transfer Mode) – асинхронный режим передачи. Тип коммутационной технологии, при котором по сети передаются небольшие ячейки фиксированного размера.

ATP (AppleTalk Protocol) – транзакционный сеансовый протокол AppleTalk.

AUI (Attachment Unit Interface) – интерфейс подключаемого модуля. Интерфейс для подключения внешнего трансивера, установленного на магистральном коаксиальном кабеле.

BASE – сокращение BASEband, основная полоса канала.

BASIC (Beginning All-purpose Symbolic Instruction Code) – система символического кодирования для начинающих.

BBS (Broadcast Bulletin System) – широковещательная система объявлений. Электронная доска объявлений, компьютерный аналог доски объявлений.

BDC (Backup Domain Controller) – вторичный контроллер домена.

BI (Business Intelligence) – система бизнес-аналитики.

BIOS (Basic Input/Output System) – базовая система ввода-вывода.

BISDN (Broadband Integrated Services Digital Network) – широкополосная цифровая сеть с интегральным обслуживанием.

BNA (Broad Network Access) – широкий сетевой доступ (в облаке).

BNS (Broadband Network Service) – широкополосный сетевой сервис.

BSC (Base Station Controller) – контроллер базовой станции.

BSS (Base Station Subsystem) – подсистема базовой станции.

BTS (Base Transceiver Station) – базовая трансиверная станция.

B-WIN (Broadband-Wissenschafts Nets) – широкополосная исследовательская сеть.

CAS (Column Address Strobe) – строб адреса столбца, сигнал, используемый при работе с динамической памятью.

CASE (Computer-Aided Software Engineering) – компьютерная разработка программного обеспечения.

CBN (Cloud-Based Network) – облачная сеть.

CDDI (Copper Distributed Data Interface) – распределенный проводной интерфейс передачи данных.

CDMA (Code Division Multiple Access) – множественный доступ с кодовым разделением каналов.

CDPD (Cellular Digital Packet Date) – сотовые дискретные пакетные данные, сотовая пакетная радиосеть.

CD-ROM (Compact Disk Read Only Memory) – компакт-диск с памятью только для чтения.

CGI (Common Gateway Interface) – общий интерфейс шлюза.

CGM (Computer Graphics Metafile) – метафайл компьютерной графики.

CHAP (Challenge Handshake Authentication Protocol) – протокол аутентификации с предварительным (косвенным) согласованием вызова.

CLNP (Connection Less Network Protocol) – сетевой протокол без организации соединений.

CMIP (Common Management Information Protocol) – общий протокол управления информацией.

CPI (Common Programming Interface) – общий программный интерфейс.

CPU (Central Processing Unit) – центральное процессорное устройство.

CRC (Cycle Redundancy Check) – контроль цикличной избыточности.

CSMA/CD (Carrier Sense Multiple Access with Collision Detection) – множественный доступ с прослушиванием несущей и разрешением коллизий.

CWIS (Campus Wide Information System) – глобальная информационная система.

DAS (Double Attached Station) – станция сети FDDI с двойным подключением к магистральному кольцу или концентратор.

DBMS (Database Management System) – система управления базами данных (СУБД).

DC (Data Center) – центр обработки данных.

DDC (Display Data Channel) – интерфейс обмена данными между компьютером и монитором.

DDE (Dynamic Date Exchange) – динамический обмен данными.

DDP (Datagram Delivery Protocol – протокол доставки дейтаграмм). Протокол передачи данных Apple, используемый в AppleTalk.

DECT (Digital Enhanced Cordless Telecommunications) – стандарт беспроводной телефонии домашнего или офисного назначения.

DHCP (Dynamic Host Configuration Protocol) – протокол динамической конфигурации хоста.

DIPC (Data Integration Platform Cloud) – облачная платформа интеграции данных; это унифицированная платформа для репликации данных в реальном времени, преобразования данных, слежения за качеством данных и управления данными; разработана корпорацией Oracle.

DLC (Dada Link Control) – протокол управления каналом передачи данных.

DLL (Dynamic Linked Library) – динамическая библиотека.

DMA (Direct Memory Access) – прямой доступ к памяти.

DMS (Data Management Server) – сервер управления данными.

DNS (Domain Name System) – доменная система имен.

DPS (Workstations/Data Processing Systems) – рабочие станции/системы обработки данных.

DRAM (Dynamic Random Access Memory) – динамическая память прямого доступа, память, схемотехнически выполненная в виде двумерной матрицы (строки и столбцы) конденсаторов.

DSL (Digital Subscriber Line) – цифровая абонентская линия.

DVI (Digital Video Interactive) – система аппаратного сжатия движущихся видеоизображений.

DVD (Digital Versatile Disk) – цифровой универсальный диск, самый современный стандарт хранения информации на оптическом (лазерном) диске.

DW (Data Warehouse) – хранилище данных.

EAP (Extensible Authentication Protocol) – расширяемый протокол аутентификации.

EBCDIC (Extended Binary Coded Decimal Interchange Code) – схема кодировки IBM. Используется мэйнфреймами и ПК.

ECC (Error Correction Code) – код коррекции ошибок.

ED (End Delimeter) – конечный ограничитель.

EDGE (Enhanced Data Rates for GSM Evolution) – цифровая технология для мобильной связи, которая функционирует как надстройка над 2G- и 2.5G-сетями.

EHCS (Event Hub Cloud Service) – это управляемый облачный сервис, который обеспечивает высокодоступную и масштабируемую

платформу обмена сообщениями для работы с потоковыми данными; разработан корпорацией Oracle.

EIA (Electronic Industries Alliance) – альянс отраслей электронной промышленности в США.

EIR (Equipment Identity Register) – регистр идентификации оборудования.

EISA (Enhanced Industry Standard Architecture) – 32-разрядная архитектура системной шины для ПК на базе процессора Intel.

EMS (Element Management System) – система управления элементами сети.

ESP (Encapsulating Security Payload) – протокол, который обеспечивает конфиденциальность, целостность и аутентификацию пакета при передаче данных.

ETR (Early Token Release) – алгоритм раннего освобождения маркера.

FAQ (Frequently Asked Questions) – часто задаваемые вопросы.

FCS (Frame Check Sequence) – поле контрольной суммы.

FDDI (Fiber Distributed Date Interface) – распределенный интерфейс передачи данных по волоконно-оптическому кабелю. Технология ЛВС, использующая скорость передачи 100 Мбит/с.

FDMA (Frequency Division Multiple Access) – множественный доступ с разделением частоты.

FDSE (Full Duplex Switched Ethernet) – полнодуплексная коммутируемая сеть Ethernet.

FEC (Forward Error Correction) – упреждающая (или прямая) коррекция ошибок.

FQDN (Fully Qualified Domain Name) – полностью определенное имя домена.

FTAM (File Transfer, Access, and Management) – протокол передачи, доступа и управления файлами.

FTP (File Transfer Protocol) – протокол передачи файлов. Позволяет обмениваться файлами по сети.

GDI (Graphics Device Interface) – интерфейс графического устройства.

GIF (Graphics Interchange Format) – файлы растровых изображений, в которых используется не более 256 индексированных цветов.

GRE (Generic Routing Encapsulation) – общая инкапсуляция маршрутов, протокол туннелирования сетевых пакетов, разработанный компанией Cisco Systems.

GSM – от названия группы Groupe Special Mobile, позже переименован в Global System for Mobile Communications.

GUI (Graphics User Interface) – графический интерфейс пользователя.

HAL (Hardware Abstraction Layer) – уровень аппаратных абстракций.

HDL (Hardware Description Language) – язык описания технических средств.

HDLC (High-level Data Link Control) – протокол управления каналом передачи данных высокого уровня.

HLR (Home Location Register) – регистр исходного положения.

HP (Hewlett-Packard) – Хьюлитт – Паккард (корпорация HP).

HSTR (High-Speed Token Ring) – высокоскоростной Token Ring.

HTML (Hyper Text Markup Language) – язык гипертекстовой разметки.

HTTP (Hyper Text Transfer Protocol) – протокол передачи гипертекста.

IaaS (Infrastructure as a Service) – инфраструктура как услуга.

IANA (Internet Assigned Numbers Authority) – служба распределения номеров.

IBM (International Business Machines) – международные бизнес-машины, название известной корпорации.

IBSS (Independent Basic Service Set) – независимая базовая зона обслуживания (или P2P).

ICANN (the Internet Corporation for Assigned Names and Numbers) – интернет-корпорация по распределению имен и адресов.

ICMP (Internet Control Message Protocol) – протокол управления сообщениями Интернета.

IDE (Integrated Device Electronic) – интерфейс жестких дисков.

IDS (Intrusion Detection System) – система распознавания вторжений или атак.

IDS/IPS (Intrusion Detection System/Intrusion Prevention Systems) – системы обнаружения/предотвращения вторжений.

IEEE (Institute of Electrical and Electronics Engineers) – Институт инженеров электротехники и электроники.

IIS (Internet Information Server) – компонент Microsoft Back ofFice, который действует как web-сервер в среде Windows NT.

IMAP (Internet Message Access Protocol) – протокол доступа к электронной почте. Разработан на смену SMTP.

IMP (Interface Message Processors) – устройства для управления трафиком в сети.

IoT (Internet of Things) – интернет вещей.

IP (Internet Protocol) – протокол Интернет, сетевой протокол стека TCP/IP, который предоставляет адресную и маршрутную информацию.

IPsec (Internet Protocol Security) – набор протоколов для обеспечения защиты данных, передаваемых по IP-сети.

IPTV (Internet Protocol Television) – телевидение по протоколу Интернета.

IPX (Internetwork Packet Exchange) – протокол межсетевого обмена пакетами, предназначенный для адресации и маршрутизации пакетов в сетях Novell.

IrMC (Infrared Mobile Communications) – мобильная связь в инфракрасном диапазоне.

IRQ (Interrupt Request) – запрос на прерывание.

ISA (Industry Standard Architecture) – системная шина IBM PC/IT. Позволяет подключить к системе различные адаптеры, установив дополнительную плату в гнездо расширения.

ISAPI (Inernet Server API) – интерфейсы прикладного программирования фирмы Microsoft.

ISDN (Integrated Services Digital Network) – цифровая сеть с интеграцией служб (услуг), ЦСИС.

ISO (International Organization for Standardization) – организация стандартизации различных стран.

JTM (Job Transfer and Manipulation) – сетевая служба передачи и управления заданиями.

L2TP (Layer Two Tunneling Protocol) – протокол туннелирования, основанный на протоколе L2F (Layer Two Forwarding), разработанном компанией Cisco, и протоколе PPTP.

LAN (Local-Area Network) – локальная сеть.

LAP (Link Access Procedure) – процедура доступа к каналу.

LAT (Local-Area Transport) – немаршрутизируемый протокол фирмы Digital Equipment Corporation, DEC).

LLC (Logical Link Control) – логический контроль связи.

LSB (Least Significant Bit) – метод наименее значащих битов, НЗБ.

LSL (Link Support Layer) – уровень поддержки связи.

LTE (Long-Term Evolution) – долговременное развитие; относится обычно к обозначению стандарта беспроводной связи 4G LTE.

MAC (Media Assess Control) – контроль доступа к среде.

MAPI (Messaging Application Program Interface) – интерфейс прикладных программ обработки сообщений.

MAU (Multiple Access Unit) – устройство с множественным доступом).

MCA (Micro Channel Architecture) – 32-битная системная шина в ПК IBM PS/2.

MIB (Management Information Base) – базы управляющей информации.

MMF (Multi Mode Fiber) – многомодовый кабель.

MNP (Microcom Network Protocol) – серия стандартов, предназначенная для сжатия информации и исправления ошибок при асинхронной передачи данных по телефонным линиям.

MPLS (Multiprotocol Label Switching) – многопротокольная коммутация по меткам (с использованием IP).

MSAU (или *MAU*, MultiStation Access Unit) – многостанционные устройства доступа; концентратор.

MSC (Mobile Switching Center) – мобильный центр коммутации.

MTBF (Mean Time Between Failure) – среднее время между отказами; параметр надежности системы или устройства.

MTSO (Mobile Telecommunications Switching Office) – коммутатор мобильных телекоммуникаций.

MU-MIMO (Multi-User Multiple-Input/Multiple-Output) – многопользовательский многоканальный вход-выход.

NAF (Network Access Facilities) – средства доступа к сети.

NAS (Network-Attached Storage) – сетевой накопитель.

NBP (Name Binding Protocol) – транспортный протокол связывания имен AppleTalk.

NCP (NetWare Core Protocol) – базовый протокол сетей NetWare.

NDIS (Network Device Interface Specification) – спецификация интерфейса сетевого устройства, программный интерфейс, обеспечивающий взаимодействие между драйверами транспортных протоколов и соответствующими драйверами сетевых интерфейсов. Позволяет использовать несколько протоколов, даже если установлена только одна сетевая карта.

NetBEUI (NetBIOS Extended User Interface) – протокол ЛВС, поддерживаемый всеми СОС фирмы Microsoft, обеспечивает транспортные услуги для NetBIOS.

NetBIOS (Network Basis Input/Output System) – интерфейс прикладных программ для ЛВС. Устанавливает соединение между компьютерами.

NFS (Network File System) – сетевая файловая система.

NFV (Network Functions Virtualization) – виртуализация сетевых функций.

NGN, или *NGNe* (Next Generation Network), – сети последующих поколений.

NIS (Network Information System) – сетевая информационная система. Обеспечивает способ доступа к данным, благодаря которому все узлы сети могут использовать единую БД, содержащую все учетные записи пользователей сети и имена всех сетевых узлов.

NLM (NetWare Loadable Module) – загружаемый модуль NetWare.

NLSP (NetWare Link Service Protocol) – протокол канального сервиса NetWare.

NMS (Network Management Server) – сервер управления сетью.

NOS (Network Operating System) – сетевая операционная система.

NRZ (Non-Return to Zero) – без возврата к нулю. Метод двоичного кодирования информации, при котором единичные биты представляются положительным значением, а нулевые отрицательным.

NSAPI (Netscape API) – интерфейсы прикладного программирования фирмы Netscape.

ODBC (Open Database Connectivity) – открытый доступ к базам данных.

ODI (Open Data-Link Interface) – интерфейс открытого драйвера.

ODSS (On-Demand Self-Service) – самообслуживание по запросу.

OFDM (Orthogonal Frequency Division Multiplexing) – мультиплексирование с разделением по ортогональным частотам.

OFDMA (Orthogonal Frequency Division Multiple Access) – множественный доступ с ортогональным частотным разделением каналов, одна из важнейших фунуций повышения производительности сети.

OLAP (Online Analytical Processing) – онлайн-аналитическая обработка.

OLE (Object Linking and Embedding) – связь и внедрение объектов.

OME (Open Messaging Environment) – среда открытых сообщений.

OSA (Open Scripting Architecture) – архитектура открытых сценариев.

OSI (Open System Interconnection) – взаимодействие открытых систем.

OSINT (Open Source Intelligence) – разведка на основе открытых источников.

OSPM (Operating System Power Management) – непосредственное управление энергопотреблением операционной системы.

OSPF (Open Shortest Path First) – протокол динамической маршрутизации.

PaaS (Platform as a Service) – платформа как услуга.

PAP (Password Authentication Protocol) – протокол простой проверки подлинности, предусматривающий отправку имени пользователя и пароля на сервер.

PCI (Peripheral Component Interconnect) – соединение внешних устройств, шина PCI.

PCN (Personal Communications Network) – персональные сети связи.

PCS (Personal Communication Services) – персональные службы связи.

PDC (Primary Domain Controller) – первичный контролер доменов, ПК под управлением Windows NT Server, на котором хранятся БД учетных записей домена.

PDU (Protocol Data Unit) – единица данных протокола.

PKI (Public Key Infrastructure) – инфраструктура системы с открытыми ключами.

PLC (Power Line Carrier) – связь через линии электропередач, ЛЭП.

PnP (Plug-and-Play) – технология самонастраиваемого оборудования.

PoE (Power over Ethernet) – передача электроэнергии через Ethernet.

PPP (Point-to-Point Protocol) – протокол «точка – точка». Протокол, предназначенный для работы на двухточечной линии (линии, соединяющей два устройства). Протокол канального уровня.

PSTN (Public Switched Telephone Network) – телефонные сети связи общего пользования, ТфОП.

PTM (Packet Transfer Mode) – пакетный способ передачи.

PVC (Permanent Virtual Connection) – постоянные виртуальные соединения (в сетях ATM).

QoE (Quality of Experience) – качество восприятия.

QoS (Quality of Service) – качество обслуживания.

RAC (Remote Access Client) – клиент удаленного доступа.

RAID (Redundant Array of Independent Disks) – избыточный массив независимых дисков.

RAM (Random Access Memory) – память с произвольным доступом.

RARP (Reverse Address Resolution Protocol) – реверсивный протокол разрешения адреса.

RAS (Remote Access Server) – сервер удаленного доступа.

RDBMS (Relational Database Management System) – система управления реляционными базами данных.

RFS (Remote File System) – удаленная файловая система.

RIF (Routing Information Field) – поле маршрутной информации.

RIP (Routing Information Protocol) – протокол обмена маршрутной информацией, протокол маршрутной информации.

RIP (Routing Internet Protocol) – протокол взаимодействия маршрутизаторов в сети; протокол маршрутной информации.

RPC (Remote Procedure Call) – вызов удаленных процедур.

RRAS (Routing and Remote Access Service) – служба маршрутизации и удаленного доступа.

RTOS (Real-Time Operating System) – операционная система реального времени.

RTP (Real-Time Transport Protocol) – транспортный протокол передачи в реальном времени.

SaaS (Software as a Service) – программное обеспечение как услуга.

SAN (Storage-Area Networks) – сеть хранения данных.

SAP (Service Access Point) – точка доступа к службе. Точка, в которой услуга какого-либо уровня OSI становится доступной ближайшему вышележащему уровню. Точки доступа именуются в соответствии с уровнями, обеспечивающими сервис.

SAS (Single Attached Station) – станция сети FDDI с одинарным подключением.

SCADA (Supervisory Control and Data Acquisition) – диспетчерское управление и сбор данных.

SCCP (Signaling Connection Control Part) – подсистема управления соединениями сигнализации.

SDH (Synchronous Digital Hierarchy) – синхронная дискретная иерархия. Европейский стандарт на использование оптических кабелей в качестве физической среды для скоростных сетей передачи на большие расстояния.

SDLC (Synchronous Data Link Control) – протокол синхронной передачи данных.

SDN (Software-Defined Network) – сеть, определяемая программным обеспечением – виртуальная сеть; программно-конфигурируемая сеть.

SFD (Start of Frame Delimiter) – признак начала кадра.

SID (Security Identification) – идентификатор безопасности.

SKIP (Simple Key Management or Internet Protocol) – простой протокол управления криптоключами в Интернете.

SLA (Service Level Agreement) – соглашение об уровне обслуживания.

SLIP (Serial Line Internet Protocol) – Интернет-протокол для последовательных линий. Протокол последовательной посимвольной передачи данных. Позволяет компьютеру использовать IP (и, таким образом, становится полноправным членом сети), осуществляет связь с миром через стандартные телефонные линии и модемы, а также непосредственно через интерфейс RS-232.

SMF (Single Mode Fiber) – одномодовый кабель.

SMTP (Simple Mail Transfer Protocol) – простой протокол электронной почты.

SNA (System Network Architecture) – архитектура систем связи, которая предназначена для обмена данными между ПК различных типов.

SNAP (SubNetwork Access Protocol) – протокол доступа к подсети.

SNMP (Simple Network Management Protocol) – простой протокол сетевого управления. Протокол сетевого администрирования SNMP очень широко используется в настоящее время. Управление сетью входит в стек протоколов TCP/IP.

SONET (Synchronous Optical Network) – синхронная оптическая сеть.

SPX (Sequenced Packet Exchange) – протокол, осуществляющий передачу сообщений с установлением соединений в сетях Novell.

SQL (Structured Query Language) – язык структурированных запросов.

SSL (Secure Socket Layer) – протокол защищенных сокетов; обеспечивает секретный обмен сообщениями для протоколов прикладного уровня стека TCP/IP.

STP (Spanning Tree Protocol) – протокол связующего дерева; канальный протокол, основной задачей которого является устранение петель в топологии произвольной сети Ethernet, в которой есть один или более сетевых мостов, связанных избыточными соединениями.

STP (Shielded Twisted Pair) – экранированная витая пара. Имеются защита в виде экрана для каждой пары и общий внешний экран в виде сетки.

SVC (Switched Virtual Connection) – коммутируемые виртуальные соединения (в сетях ATM).

TCP (Transmission Control Protocol) – протокол управления передачей.

TDI (Transport Driver Interface) – интерфейс транспортного драйвера.

TDMA (Time Division Multiple Access) – множественный доступ с разделением во времени.

TFTP (Trivial File Transfer Protocol) – простейший протокол передачи файлов.

TIFF (Tagged Image Format File) – спецификация формата файла изображения.

TLD (Top-Level Domain) – домены верхнего уровня.

TLI (Transport Level Interface) – интерфейс транспортного уровня.

TP4 (Transmission Protocol 4) – протокол передачи класса 4.

TPDDI (Twisted Pair Distributed Data Interface) – электрическая реализация архитектуры FDDI на витой паре.

TPMA (Token Passing Multiple Access) – множественный доступ с передачей полномочия или метод с передачей маркера.

UDP (User Datagram Protocol) – пользовательский протокол дейтаграмм.

UMTS (Universal Mobile Telecommunication System) – универсальная система мобильной связи.

UNI (User-to-Network Interface) – сетевой интерфейс пользователя. Набор правил, который определяет взаимодействие оконечного оборудования и сети ATM с физической и информационной точек зрения.

UNC (Universal Name Convention) – стандартный метод именования в сети, имеющий вид \\сервер\общий_ресурс.

UPS (Uninterruptible Power Supply) – источник бесперебойного питания.

URL (Uniform Resource Locator) – адрес универсального указателя ресурсов.

USSD (Unstructured Supplementary Service Data) – неструктурированные дополнительные сервисные данные.

UTP (Unsealing Twist Pare) – неэкранированная витая пара.

UUCP (Unix-to-Unix Copy Protocol) – протокол копирования от Unix к Unix.

UWC (Universal Wireless Communications) – технология EDGE.

VESA (Video Electronics Standard Association) – ассоциация стандартов электронной графики.

VGA (Video Graphics Array) – видеографическая матрица.

VHDL (Very High-speed integrated circuit Hardware Description Language) – язык описания технических средств сверхскоростных интегральных схем.

VLR (Visitor Location Register) – регистр местонахождения посетителей.

VNF (Virtual Network Functions) – виртуальные сетевые функции.

VoIP (Voice over Internet Protocol) – IP-телефония.

VPI (Virtual Path Identifier) – идентификатор виртуального пути (в технологии ATM).

VPN (Virtual Private Network) – виртуальная частная сеть.

WAIS (Wide Area Information Server) – протокол глобального информационного сервера.

WDM (Wavelength Division Multiplexing) – мультиплексирование с разделением по длине волны.

WDMA (Wavelength Division Multiple Access) – множественный доступ с разделением длины волны.

WINS (Windows Internet Name Service) – сетевая служба Windows, используемая для определения IP-адреса по имени NetBIOS.

WWW (Word Wide Web) – всемирная паутина.

Учебное издание

Урбанович Павел Павлович
Романенко Дмитрий Михайлович

КОМПЬЮТЕРНЫЕ СЕТИ И СЕТЕВЫЕ ТЕХНОЛОГИИ

Учебное пособие

Редактор *Т. Е. Самсанович*
Компьютерная верстка *А. Н. Петрова*
Дизайн обложки *П. П. Падалец*
Корректор *Т. Е. Самсанович*

Подписано в печать 01.08.2022. Формат 60×84 1/16.
Бумага офсетная. Гарнитура Таймс. Печать ризографическая.
Усл. печ. л. 39,5. Уч.-изд. л. 33,0.
Тираж 150 экз. Заказ .

Издатель и полиграфическое исполнение:
УО «Белорусский государственный технологический университет».
Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/227 от 20.03.2014.
Ул. Свердлова, 13а, 220006, г. Минск.

Переплетно-брошюровочные процессы
произведены в УП «Донарит».
Ул. Октябрьская, 25, офис 2, 220030, г. Минск