Александр Филимонов

Протоколы Интернета

- ЛВС и удаленный доступ
- Внутренняя и внешняя маршрутизация
- Управление компонентами сети
- Формирование и интеграция потокового трафика

МАСТЕР_СИСТЕМ

Александр Филимонов

Протоколы Интернета

Санкт-Петербург

«БХВ-Петербург»

2003

УДК 681.3.06
ББК 32.973.202
Ф53

Филимонов А. Ю.

Ф53 Протоколы Интернета. — СПб.: БХВ-Петербург, 2003. — 528 с.: ил.

ISBN 5-94157-247-6

В книге подробно рассматриваются основные принципы построения и особенности применения наиболее популярных протоколов, используемых в современных сетях Интернета, а также направления их дальнейшего развития. Большое внимание уделено протоколам маршрутизации, управления компонентами сети и транспортным протоколам глобальной сети. Освещаются перспективные протоколы и технологии, предназначенные для обеспечения информационной безопасности современных вычислительных сетей, формирования и интеграции в них голосового и видеотрафика. Значительная часть книги посвящена рассмотрению протоколов таких популярных интернет-служб, как e-mail, WWW и др. Полнота представленного материала, подкрепленного практическими примерами, обеспечивает глубокое понимание принципов организации информационно-вычислительных сетей и успешную реализацию их на практике.

Для администраторов сетей,
сотрудников IT-отделов предприятий и студентов

УДК 681.3.06
ББК 32.973.202

Группа подготовки издания:

Главный редактор	*Екатерина Кондукова*
Зам. главного редактора	*Евгений Рыбаков*
Зав. редакцией	*Анна Кузьмина*
Редактор	*Юрий Рожко*
Компьютерная верстка	*Татьяны Олоновой*
Корректор	*Виктория Голуб*
Оформление серии	*Via Design*
Дизайн обложки	*Игоря Цырульникова*
Зав. производством	*Николай Тверских*

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 22.11.02.
Формат 70×100 1/16. Печать офсетная. Усл. печ. л. 42,57.
Тираж 4000 экз. Заказ №
"БХВ-Петербург", 198005, Санкт-Петербург, Измайловский пр., 29.

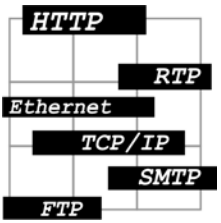
Гигиеническое заключение на продукцию, товар № 77.99.02.953.Д.001537.03.02
от 13.03.2002 г. выдано Департаментом ГСЭН Минздрава России.

Отпечатано с готовых диапозитивов
в Академической типографии "Наука" РАН
199034, Санкт-Петербург, 9 линия, 12.

ISBN 5-94157-247-6

Содержание

Глава 1



Обзор протоколов Интернета

Рождение TCP/IP. История создания Интернета

В научном и техническом мире существует мнение, что чаще всего перспективные и передовые технологии рождаются при изобретении нового оружия или средств защиты от него. Это объясняется тем, что система, предназначенная для работы в экстремальных условиях, надежно и устойчиво функционирует в мирное время. Именно так и случилось с Интернетом и протоколами семейства TCP/IP (Transmission Control Protocol/Internet Protocol).

Идея создания Интернета была предложена RAND Corporation (США) в связи с необходимостью построения коммуникационной отказоустойчивой сети, которая могла бы функционировать даже в том случае, если большая часть ее узлов вышла из строя, например, в случае ядерной войны. Решение состояло в том, чтобы создать сеть, где информационные пакеты могли бы передаваться от одного узла к другому без какого-либо централизованного контроля. В случае, если бы основная часть сети не работала, пакеты самостоятельно передвигались бы по доступным узлам до тех пор, пока не попали бы в точку своего назначения. Кроме того, сеть должна была быть достаточно устойчива к возможным ошибкам при передаче пакетов, т. е. обладать механизмом контроля пакетов и обеспечивать контроль доставляемой информации. Хотя проект был по своей сути военным, к его разработке привлекались научные институты и университеты.

В начале 70-х годов созданием такой сети занялось Агентство Передовых Проектов Национальной Безопасности США (United States Defense Advanced Research Project Agency — DARPA). Точнее, история Интернета началась в 1969 году, когда в Калифорнийском университете был установлен первый узел сети, получивший название ARPAnet (по имени компании, финансировавшей проект), и были созданы основы для построения сети с коммутируемыми пакетами. К началу 1971 года ARPAnet насчитывала уже около 20 узлов, и более 30 университетов получили возможность подключения к сети в рамках проекта.

К середине 70-х были предприняты попытки объединения различных пакетных сетей. В начале 80-х к ARPAnet были подключены первые локальные

сети, и для использования в построенной сети (в дальнейшем именуемой Интернетом) был выбран, адаптирован и затем повсеместно принят для работы набор протоколов Transmission Control Protocol/Internet Protocol (TCP/IP). Он сменил более ранний протокол NCP (Network Communication Protocol). TCP/IP вполне удовлетворял всем тем требованиям, которые на него возлагались. По сути дела, это и послужило началом широкого распространения TCP/IP. Что из этого вышло — мы можем судить по той скорости, с которой внедряется в нашу жизнь Интернет.

Существует много причин, почему протоколы семейства TCP/IP были выбраны за основу в сети Интернет. Это, прежде всего, возможность работать с этими протоколами как в локальных LAN (Local Area Network), так и в глобальных WAN (Wide Area Network) сетях, способность протоколов управлять большим количеством стационарных и мобильных пользователей, удобство для использования частными лицами или организациями, желающими подсоединиться прямо к Интернету или через фирмы, предоставляющие этот сервис. Протоколы Интернета обеспечивают высокий уровень взаимодействия между совершенно различными операционными системами, предоставляют средства для разработки на их основе приложений, использующих современный программный интерфейс. Но главную роль, безусловно, сыграло провидение разработчиков ARPAnet, когда они выбирали этот почти никому не известный протокол для построения своей тогда еще крохотной сети.

В 1986 году на базе существующей опорной сети ARPAnet, которая обладала производительностью 56 Кбит/с и объединяла шесть суперкомпьютерных центров США, было начато строительство новой сети. Решение о модернизации было принято вследствие загруженности существующей инфраструктуры. Заказ на проведение проекта получил консорциум Merit, MCI и IBM. При реализации этого проекта самой сложной проблемой был поиск специалистов, знающих технологию сетей и, в частности, TCP/IP. Однако уже через 8 месяцев система на базе каналов T1 (1,54 Мбит/с) была сдана в эксплуатацию. Она состояла из 13 коммутационных узлов, каждый из которых составляли 9 параллельно работающих компьютеров IBM, работающих под управлением Berkeley UNIX и соединенных локальной сетью. Узлы занимались маршрутизацией пакетов и сбором сетевой информации. Причем каждый был построен таким образом, что при выходе его из строя выполняемые им функции принимали на себя оставшиеся узлы.

Дальнейшее развитие Интернета шло подобно снежному кому. Ядро обрастало все большим и большим количеством узлов, соединений и подключенных сетей. И уже в 1989 году загрузка опорных линий достигла предельных значений. Поэтому в 1991 году была выполнена модернизация и сеть была переведена на линии T3 (45 Мбит/с), причем коммутационные компьютеры были заменены на ЭВМ IBM RS/6000. К тому времени сеть уже насчитывала 16 узлов и более 3500 подключенных сетей.

В настоящее время скорости информационного обмена сети Интернет и темпы ее роста продолжают увеличиваться.

Иерархическая система информационного взаимодействия

Информационное взаимодействие в сети Интернет строится в соответствии с правилами и требованиями общего международного стандарта ISO 7498 (ISO — International Organization for Standardization).

Этот стандарт имеет тройной заголовок "Информационно-вычислительные системы — Взаимодействие открытых систем — Эталонная модель". Обычно его называют короче — "Эталонная модель взаимодействия открытых систем". Публикация этого стандарта в 1983 году подвела итог многолетней работы многих известных телекоммуникационных компаний и стандартизующих организаций.

Основной идеей, которая положена в основу этого документа, является разбиение процесса информационного взаимодействия между системами на уровни с четко разграниченными функциями.

Идея такого разбиения не была революционной. Можно вспомнить, что слоистую архитектуру имели информационные взаимодействия в сетях SNA (Systems Network Architecture) и DNA (Digital Network Architecture).

В качестве прообраза модели взаимодействия OSI (Open System Interconnection) была использована структура, предложенная ANSI (American National Standards Institute). Основные работы по созданию текста документа были выполнены CCITT (Consultative Committee for International Telegraphy), а итоговый документ появился в виде стандарта ISO. Статус стандарта ISO важен для данного документа, поскольку ISO 7498 является стандартом стандартов в области телекоммуникаций.

Преимущества слоистой организации взаимодействия заключаются в том, что она обеспечивает независимую разработку уровневых стандартов, модульность аппаратуры и программного обеспечения информационно-вычислительных систем и способствует тем самым техническому прогрессу в данной области.

При использовании многоуровневой модели проблема перемещения информации между узлами сети разбивается на более мелкие и, следовательно, более легко разрешимые проблемы.

Многоуровневая модель четко описывает, каким образом информация проделывает путь через среду сети от одной прикладной программы, к примеру, обработки таблиц, до иной прикладной программы обработки тех же таблиц, находящейся на другом компьютере сети.

Предположим, например, что система А, изображенная на рис. 1.1, имеет информацию для отправки в систему В. Прикладная программа системы А начинает взаимодействовать с уровнем 4 системы А (верхний уровень), который, в свою очередь, начинает взаимодействовать с уровнем 3 системы

А, и т. д. — до уровня 1 системы А. Задача уровня 1 — отдавать, а потом забирать информацию из физической среды сети.

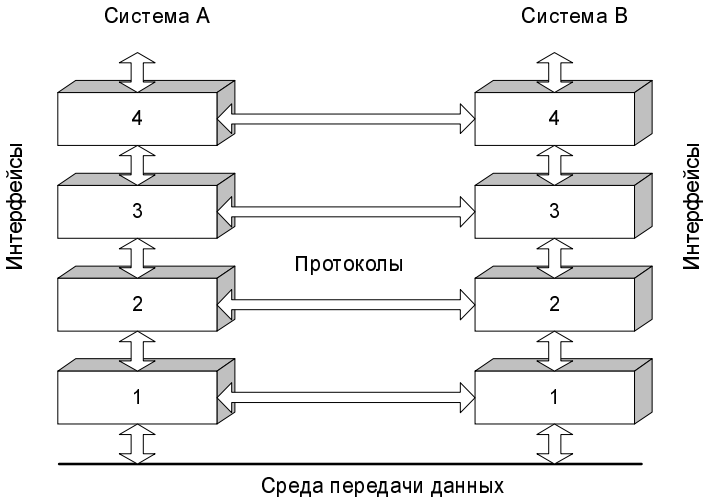


Рис. 1.1. Организация информационного взаимодействия двух систем

Примечание

Поскольку информация, которая должна быть отослана, проходит вниз через уровни системы, по мере этого продвижения она становится все меньше похожей на человеческий язык и все больше похожей на ту информацию, которую понимают компьютеры, а именно "единицы" и "нули".

После того как информация проходит через физическую среду сети и поступает в систему В, она последовательно обрабатывается на каждом уровне системы В в обратном порядке — сначала на уровне 1, затем на уровне 2 и т. д., пока, наконец, не достигнет прикладной программы системы В.

Многоуровневая модель не предполагает наличия непосредственной связи между одноименными уровнями взаимодействующих систем. Следовательно, каждый уровень системы А должен полагаться на услуги, предоставляемые ему смежными уровнями системы А, чтобы помочь осуществить связь с соответствующим уровнем системы В. Предположим, что уровень 4 системы А должен связаться с уровнем 4 системы В. Для того чтобы выполнить эту задачу, уровень 4 системы А должен воспользоваться услугами уровня 3 системы А, тогда уровень 4 будет называться "пользователем услуг", а уровень 3 — "источником услуг".

Информация по оказываемым услугам передается между уровнями в специальном информационном блоке, который называется *заголовком*. Заголовок обычно предшествует передаваемой прикладной информации.

Предположим, что система А хочет отправить в систему В какой-либо текст, называемый "данные" или "информация". Этот текст передается из прикладной программы системы А в верхний уровень этой системы. Прикладной уровень системы А должен передать определенную информацию в прикладной уровень системы В, поэтому он помещает управляющую информацию своего уровня в виде заголовка перед фактическим текстом, который должен быть передан. Построенный таким образом информационный блок передается в уровень 3 системы А, который может предварить его своей собственной управляющей информацией, и т. д.

Размеры сообщения увеличиваются по мере того, как оно проходит вниз через уровни до тех пор, пока не достигнет сети, где оригинальный текст и вся связанная с ним управляющая информация перемещаются в систему В и поглощаются уровнем 1 системы В. Уровень 1 системы В отделяет от поступившей информации и обрабатывает заголовок уровня 1, после чего он определяет, как обрабатывать поступивший информационный блок. Слегка уменьшенный в размерах информационный блок передается в уровень 2, который отделяет заголовок этого же уровня, анализирует его, чтобы узнать о действиях, которые он должен выполнить и т. д. Когда информационный блок наконец доходит до прикладной программы системы В, он должен содержать только оригинальный текст.

Структура заголовка и собственно данных относительна и зависит от уровня, который в данный момент анализирует информационный блок. Например, на уровне 2 информационный блок состоит из заголовка этого же уровня и следующих за ним данных. Однако данные уровня 2 могут содержать заголовки уровней 3 и 4. Кроме того, заголовок уровня 2 является просто данными для уровня 1. Помимо заголовка на каждом уровне системы информационный блок завершается соответствующей контрольной суммой КонтСум. Эта концепция иллюстрируется на рис. 1.2.

Данная модель напоминает собой вложенные друг в друга матрешки. Самая маленькая из них — это и есть пользовательские данные, а все остальные служат для доставки данных в точку назначения.

Иными словами, в результате работы этого механизма каждый пакет более высокого уровня вкладывается в "конверт" протокола нижнего уровня. Здесь уместно провести аналогию с обычными почтовыми отправлениями. Так, например, если вы пишете обычное письмо и вкладываете его в конверт с адресом, то текст письма будет информационным сообщением, которое вы хотите отправить, а конверт — заголовком "почтового" протокола. На почте ваше письмо перекладывают в мешок (протокол более низкого уровня) с письмами того же или близкого назначения и т. п. Электронные протоколы работают по той же схеме, только доставку и целостность обычных писем обеспечивает добросовестность служащих отделений связи, а электронным протоколам приходится следить за этим самостоятельно.

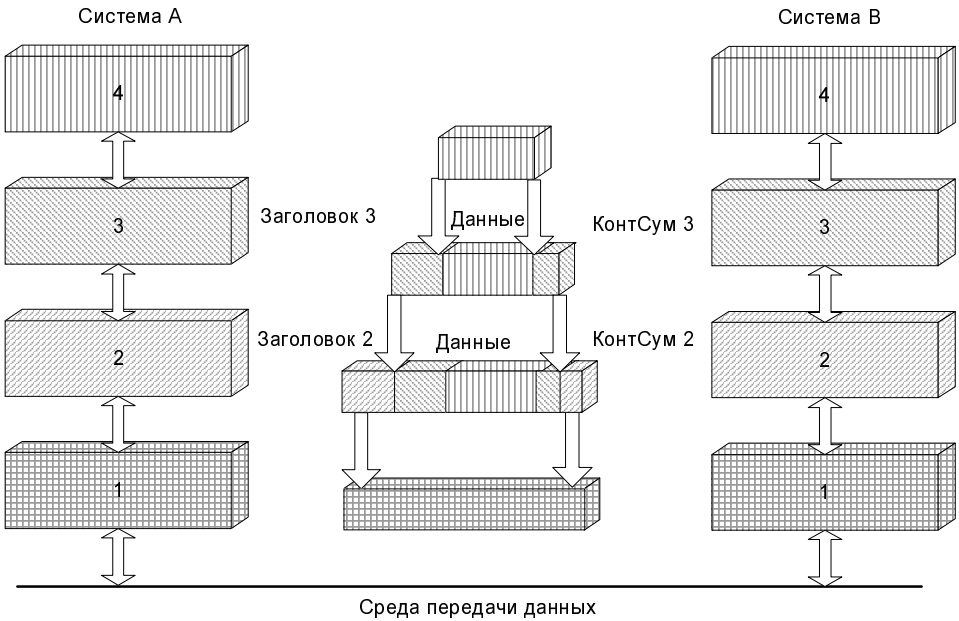


Рис. 1.2. Инкапсуляция блоков данных различных уровней

В соответствии с ISO 7498 выделяются семь уровней (слоев) информационного взаимодействия:

7. Уровень приложения (Application Layer)
6. Уровень представления (Presentation Layer)
5. Уровень сессии (Session Layer)
4. Транспортный уровень (Transport Layer)
3. Сетевой уровень (Network Layer)
2. Канальный уровень (DataLink Layer)
1. Физический уровень (Physical Layer)

Информационное взаимодействие двух или более систем, таким образом, представляет собой совокупность информационных взаимодействий уровневых подсистем, причем каждый слой локальной информационной системы взаимодействует только с соответствующим слоем удаленной системы.

Определение

Протоколом мы будем называть набор алгоритмов (правил) взаимодействия объектов одноименных уровней.

Слои (уровни) одной информационной системы также взаимодействуют друг с другом, причем в непосредственном взаимодействии участвуют только соседние уровни. Как правило, средний уровень пользуется услугами, которые ему предоставляет нижний уровень, а сам, в свою очередь, предоставляет услуги для верхнего уровня.

Определение

Интерфейсом мы будем называть совокупность правил, в соответствии с которыми осуществляется взаимодействие с объектом данного уровня.

Иерархическая организация сетевого взаимодействия позволяет обеспечивать преемственность разработанных структур и их быструю адаптацию к изменениям, происходящим в технологиях передачи данных. Например, при переходе на новый способ передачи данных по физическому носителю, изменения коснутся только нижних уровней и совсем не затронут верхние в том случае, если система протоколов организована в соответствии с требованиями ISO 7498. На практике требования данного стандарта реализуются в виде стека протоколов.

Определение

Стеком мы будем называть иерархически организованную группу взаимодействующих протоколов

Протоколы, которые входят в стек, имеют специализированный интерфейс и предназначены для взаимодействия только с протоколами соответствующих уровней данного стека. В качестве примеров таких стеков можно привести стек TCP/IP и протоколы X.25.

Уровни 7—5 считаются верхними и, как правило, не отражают специфики конкретной сети. Блок данных пользователя (сообщение) этими уровнями рассматривается как единое целое. Изменения могут испытывать только сами данные.

Уровни 1—3 и иногда 4 считаются нижними уровнями OSI. На каждом из этих уровней определяется свой формат представления данных. При прохождении по стеку с 4-го уровня до первого сообщение пользователя последовательно фрагментируется и преобразуется в последовательность блоков данных соответствующего уровня.

Определение

Процесс помещения фрагментированных блоков данных одного уровня в блоки данных другого уровня называют *инкапсуляцией*.

Обычно инкапсулируются данные протоколов верхних уровней в блоки данных протоколов нижних уровней (сетевой — канальный), но также может выполняться инкапсуляция для протоколов одноименных уровней (IP-X.25).

Уровень приложения

Протоколы, оперирующие на седьмом уровне OSI, предназначены для обеспечения доступа к ресурсам сети программ-приложений пользователя. На данном уровне определяется интерфейс с коммуникационной частью приложения. В качестве примера протоколов прикладного уровня можно привести протокол TELNET, который обеспечивает доступ пользователя к узлу сети в режиме удаленного терминала.

Уровень представления

На этом уровне обычно выполняется преобразование форматов данных, алгоритмы шифрования и компрессии данных.

Уровень сессии

На данном уровне устанавливаются, обслуживаются и разрываются сессии между представительными объектами приложений. В качестве примера протокола сеансового уровня можно рассмотреть протокол RPC (Remote Procedure Call). Как следует из названия, данный протокол предназначен для отображения результатов выполнения процедуры на удаленном узле. В процессе выполнения этой процедуры между приложениями устанавливается сеансовое соединение. Назначением этого соединения является обслуживание запросов, возникающих при взаимодействии приложения-клиент с приложением-сервер. Протоколы сеансового уровня непосредственно взаимодействуют с протоколами транспортного уровня. Организация сеансового соединения может вызвать установление транспортного соединения. Возможен, однако, вариант, когда несколько сеансовых соединений используют одно транспортное.

Транспортный уровень

Можно выделить два типа протоколов транспортного уровня — сегментирующие протоколы и дейтаграммные протоколы.

Сегментирующие протоколы транспортного уровня разбивают исходное сообщение на блоки данных транспортного уровня — сегменты. Основной функцией таких протоколов является обеспечение гарантированной доставки этих сегментов до объекта назначения и восстановление исходного сообщения. Для выполнения этой задачи протокол транспортного уровня устанавливает и разрывает соединение между узлами, управляет скоростью передачи сегментов и обеспечивает достоверность передачи и восстановления сообщения специальными контрольными процедурами. Функционирование протокола транспортного уровня состоит из трех фаз: фазы установления соединения, фазы передачи сегментов и фазы разрыва соединения.

Дейтаграммные протоколы не сегментируют сообщение и отправляют его одним куском, который называется *дейтаграмма*. При этом функции установления и разрыва соединения, управления потоком не нужны. Дейтаграммные протоколы просты для реализации, однако не обеспечивают достоверной доставки сообщений.

Поскольку транспортное соединение может быть использовано несколькими приложениями, для идентификации транспортных блоков этих приложений используются номера портов получателя и источника данных. Таким образом, сетевое взаимодействие на транспортном уровне осуществляется между портами — интерфейсами данного уровня.

Сетевой уровень

Задачей протоколов сетевого уровня является построение сетевого адреса и определение маршрута доставки информационных блоков протоколов верхних уровней (маршрутизация).

Для того чтобы блок данных был доставлен до какого — либо узла, этому узлу должен быть поставлен в соответствие известный передатчику сетевой адрес. Группы узлов, объединенные по территориальному принципу, образуют сеть. Для упрощения решения задачи маршрутизации сетевой адрес узла составляется из двух частей -адреса сети и адреса узла. Таким образом, задача маршрутизации распадается на две подзадачи — поиск сети и поиск узла в этой сети.

Например, сетевой адрес у протокола IP состоит из четырех байтов. Адрес сети может занимать в нем от 1 до 3 байтов. Маршрут может быть определен двумя способами — статическим способом и динамическим способом. Использование статической маршрутизации целесообразно в тех случаях, когда сеть имеет простую и стабильную архитектуру. Примером такой сети может быть локальная частная сеть небольшой организации, которая пользуется услугами только одного провайдера Интернета. Динамический способ является более предпочтительным для использования в сетях крупных организаций, имеющих в своем составе множество региональных офисов и нескольких провайдеров. Использование динамической маршрутизации в данном случае позволяет учитывать изменения, происходящие в сети. Динамический способ реализуется при использовании алгоритмов маршрутизации. Современные алгоритмы маршрутизации обеспечивают не только быстрое переключение маршрута в обход неисправного участка, но динамический анализ качества используемого канала передачи данных и перераспределение загрузки между параллельными каналами.

Канальный уровень

Назначением протоколов канального уровня является обеспечение передачи данных по физическому носителю — среде передачи. На канальном уровне данные передаются в виде блоков, которые называются кадрами. Тип используемой среды передачи и ее топология во многом определяют вид кадра

протокола транспортного уровня, который должен быть использован. При использовании топологий "общая шина" и "point-to-multipoint" средствами протокола канального уровня должны быть определены физические адреса, с помощью которых будет производиться обмен данными по разделяемой среде передачи и процедура доступа к этой среде. Примерами таких протоколов являются протоколы Ethernet (в соответствующей части) и HDLC (High-level Data Link Control). Протоколы транспортного уровня, которые предназначены для работы в среде типа "точка-точка", не определяют физических адресов и имеют упрощенную процедуру доступа. Примером протокола такого типа является протокол PPP (Point-to-Point Protocol).

Физический уровень

Протоколы физического уровня обеспечивают непосредственный доступ к среде передачи данных для протоколов канального и последующих уровней. Данные передаются протоколами данного уровня в виде битов (для последовательных протоколов) или групп бит (для параллельных протоколов). На данном уровне определяются набор сигналов, которыми обмениваются системы, параметры этих сигналов — временные и электрические, и последовательность формирования этих сигналов при выполнении процедуры передачи данных. Кроме того, на данном уровне формулируются требования к электрическим, физическим и механическим характеристикам среды передачи. Наиболее распространенным протоколом транспортного уровня до недавних пор был V.24, который, в частности, обеспечивал интерфейс последовательного обмена IBM PC.

Взаимодействие на сетевом уровне Интернета

Если быть наиболее точным, то данному набору протоколов больше подходит название "Комплекс протоколов Интернета". Этот комплекс охватывает целое семейство протоколов, прикладные программы и даже саму сеть. В его состав входят протоколы UDP (User Datagram Protocol), ARP (Address Resolution Protocol), ICMP (Internet Control Message Protocol), TELNET, FTP (File Transfer Protocol) и многие другие. Но, поскольку TCP и IP — наиболее известные протоколы этого комплекса, часто, ссылаясь на данный набор протоколов, используют термин TCP/IP. По сути, TCP/IP — это технология межсетевого взаимодействия.

TCP/IP был разработан совместно членами объединения, использующего процесс экспертной оценки документации, называемый Request for Comments (RFC). Несколько человек предложили структурировать и опубликовать описание RFC. Оно было изучено другими членами команды, в него были внесены изменения, и затем эти изменения отразились на структуре RFC.

Часть изменений были сделаны позднее. Со временем RFC превратился в развивающийся набор стандартов и стал применяться в построении продуктов, подчиняющихся одному или нескольким стандартам RFC. Этот процесс стал достаточно эффективно развиваться и со временем (первые RFC опубликованы в 1969 году) имеющиеся в протоколах ошибки были устранены.

Существующие RFC доступны для массового использования. В основном они предназначены для тех разработчиков и организаций, кто проектирует продукты и сервисы для использования их в сети Интернет, а также для тех, кто хочет лучше понимать идеи и принципы, заложенные в основу современного киберпространства.

Как бы мы не называли TCP/IP — это семейство протоколов. Часть из них обеспечивает выполнение "низкоуровневых" сетевых функций для множества приложений, таких, как работа с аппаратными протоколами, поддержка механизма доставки пакета по адресу назначения через множество сетей и узлов, обеспечение достоверности и надежности соединения и др. Другая часть протоколов предназначена для выполнения прикладных задач, таких, как передача файлов между компьютерами, отправка электронной почты или чтение гипертекстовой страницы WWW-сервера.

Рассмотрим, например, процесс отправки почты. Прежде всего, здесь используется протокол работы с почтой. Он определяет систему команд, которые один компьютер посылает другому. Эти команды обозначают, кто будет отправителем сообщения, кто получателем, в какой форме будет отправлено само сообщение. Этот протокол, как и другие прикладные протоколы, подразумевает, что между компьютерами уже установлено надежное соединение. Именно за установление и обеспечение этого соединения и отвечают "низкоуровневые" протоколы TCP и IP.

Задачей TCP является доставка всей информации компьютеру получателя, контроль последовательности передаваемой информации, повторная отправка недоставленных пакетов в случае сбоев работы сети. Кроме того, если сообщение достаточно большое, чтобы отправить его в одном пакете, TCP делит и отправляет его несколькими блоками. TCP также осуществляет контроль за составлением первоначального сообщения из этих блоков на компьютере получателя. Поскольку эти функции требуется выполнять для многих приложений, их объединили и выделили в единый протокол, который мог бы использоваться всеми прикладными протоколами. TCP можно рассматривать как библиотеку процедур, которую используют прикладные протоколы, если необходимо установить надежное и достоверное соединение между компьютерами.

Подобно тому как почтовый протокол использует TCP, сам TCP использует протокол IP, который обеспечивает доставку пакета по адресу, т. е. адресацию и маршрутизацию. Функции, которые предоставляет TCP, необходимы

для работы множества приложений, однако существуют приложения, для работы которых эти функции не требуются. Эти приложения используют вместо TCP свой протокол, обеспечивающий взаимодействие приложений, например UDP, которому для работы также необходим механизм, который бы осуществлял доставку пакета по адресу (т. е. уровня IP).

Адресация в IP-сетях

IP-адресация компьютеров в сети Интернет построена на концепции сети, состоящей из узлов и других сетей. Узел представляет собой объект сети, который может передавать и принимать IP-пакеты, например, компьютер рабочей станции или маршрутизатор. Обычно ошибочно понимают под узлом какой-либо сервер, однако в рамках концепции IP-сети и рабочие станции, и серверы — все являются узлами.

Узлы соединены друг с другом через одну или несколько сетей. IP-адрес любого из узлов состоит из адреса сети и адреса узла в этой сети. IP-адресация, в отличие, например, от IPX-адресации (IPX — Internetwork Packet Exchange), использует один идентификатор, объединяющий и адрес сети, и адрес узла. В соответствии с соглашением, адрес представляется четырьмя десятичными числами, разделенными точками. Каждое из этих чисел не может превышать 255 и представляет один байт 4-байтного IP-адреса.

Длина адреса IP составляет 32 бита, разделенных на две или три части. Первая часть обозначает адрес сети, вторая (если она имеется) — адрес подсети, и третья — адрес главной вычислительной машины. Адреса подсети присутствуют только в том случае, если администратор сети принял решение о разделении сети на подсети. Длина полей адреса сети, подсети и главной вычислительной машины являются переменными величинами.

Менеджер сети (администратор или программа) присваивает IP-адреса машинам в соответствии с тем, к каким IP-сетям они подключены. Старшие биты 4-байтного IP-адреса определяют номер IP-сети. Оставшаяся часть IP-адреса — номер узла. Напомним, что IP-адрес узла идентифицирует точку доступа модуля IP к сетевому интерфейсу, а не всю машину.

Сетевой IP-адрес (классы, структура)

Различают пять классов сетевых адресов Интернета. Все классы имеют различные структуры сетевого адреса. Крайние левые биты сетевого адреса Интернета обозначают класс сети. Перечислим эти пять классов сетевых адресов, имеющихся в сетях IP.

- **Класс A.** Сети класса A предназначены главным образом для использования крупными организациями, так как они обеспечивают всего 7 бит для поля адреса сети.

- **Класс B.** Сети класса B выделяют 14 бит для поля адреса сети и 16 бит для поля адреса главной вычислительной машины. Этот класс адресов обеспечивает хороший компромисс между адресным пространством сети и главной вычислительной машиной.
- **Класс C.** Сети класса C выделяют 22 бита для поля адреса сети. Однако сети класса C обеспечивают только 8 бит для поля адреса главной вычислительной машины, поэтому число главных вычислительных машин, приходящихся на сеть, может стать ограничивающим фактором.
- **Класс D.** Адреса класса D резервируются для групповой адресации в соответствии с официальным документом RFC-1112. В адресах класса D четыре бита наивысшего порядка устанавливаются на значения 1, 1, 1 и 0.
- **Класс E.** Адреса класса E также определены IP, но зарезервированы для использования в будущем. В адресах класса E все четыре бита наивысшего порядка устанавливаются в 1.

На рис. 1.3 изображена структура адресов сетей классов A—E.

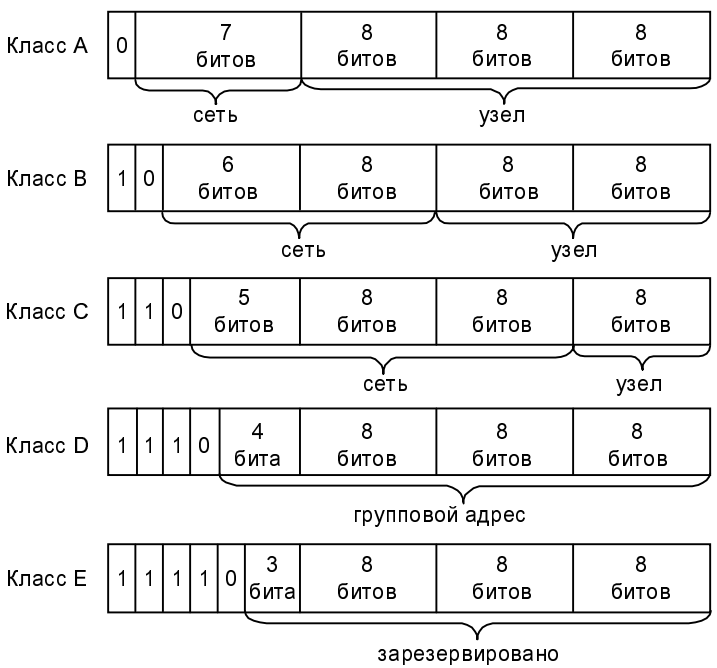


Рис. 1.3. Структура адресов сетей классов A–E

В табл. 1.1 приведены диапазоны адресов сетей и узлов сетей классов A, B, C.

Таблица 1.1. *Диапазоны адресов сетей и узлов сетей классов A, B, C*

Класс	Диапазон номера сети	Диапазон адресов узла
A	1—126	0.0.1—255.255.254
B	128.0—191.255	0.1—255.254
C	192.0.0—223.255.255	1—254

Адреса класса A предназначены для использования в больших сетях общего пользования. Они допускают большое количество номеров узлов. Адреса класса B используются в сетях среднего размера, например, сетях университетов и крупных компаний. Адреса класса C используются в сетях с небольшим числом компьютеров. Адреса класса D используются при обращениях к группам машин, а адреса класса E зарезервированы на будущее.

Специфические адреса Интернета

Для написания адреса IP обычно используется так называемая десятичная — точечная (dotted — decimal) нотация. При использовании этой формы представления адрес записывается в виде четырех десятичных чисел — по одному числу на байт, — разделенных точками:

14.0.0.1

Некоторые адреса и адресные диапазоны сети Интернет используются для реализации специфических режимов адресации. К ним относятся:

- ❒ адрес сети;
- ❒ локальный адрес узла;
- ❒ адреса типа Broadcast;
- ❒ адреса типа Multicast;
- ❒ адреса типа loop back;
- ❒ адреса частных сетей.

Адрес сети

Для обозначения адреса сети используется специальный адрес, который состоит из номера сети и использует в качестве идентификатора узла значение "все нули". Для приведенного выше примера адрес сети должен быть представлен следующим образом:

14.0.0.0

Локальный адрес

Для обозначения локального узла используется адрес, состоящий из нулевых байтов:

0.0.0.0

Broadcast-адреса протокола IP

Адреса данного типа используются в тех случаях, когда содержимое пакета адресовано группе абонентов сети. В сетях IP применяются два типа Broadcast-адресов:

- адреса типа Limited Broadcast;
- адреса типа Directed Broadcast.

Адреса первого из указанных типов используются в том случае, когда пакет адресован всей сети в целом. В этом случае все 32 разряда адреса формируются равными "1":

255.255.255.255

В тех случаях, когда пакет адресован всем узлам конкретной сети, должен быть использован адрес типа направленный (directed) Broadcast. У адресов данного типа единицами формируется только значение адреса узла. Поле адреса сети в данном случае соответствует номеру сети, в которую передается пакет. Ниже приведен пример адреса Directed Broadcast для сети 14.0.0.0.

14.255.255.255

Адрес loop back

Адрес типа loop back (петля) представляет собой виртуальный адрес, который присваивается любому интерфейсу, сконфигурированному для обработки пакетов протокола IP. Этот адрес имеет только локальное значение в пределах данного узла и может быть использован для проверки функционирования программных компонентов, используемых для реализации в данном устройстве стека протоколов TCP/IP. Для адресов данного типа используется одна из сеток класса A, в качестве номера узла обычно используется значение "1". Ниже приведен пример адреса типа loop back.

127.0.0.1

Адреса частных сетей

Как уже было отмечено выше, размер сети определяется максимальным числом активных узлов, которое, в свою очередь, ограничивается длиной поля узла сети в сетевом адресе.

Очень быстро выяснилось, что длина сетевого адреса IP (32 бита) является недостаточной для того, чтобы обеспечить одновременное информационное взаимодействие всех узлов в мире, поддерживающих протоколы TCP/IP.

Дело в том, что в тот момент, когда определялась структура этого сетевого адреса, невозможно было предположить, что сеть IP покроет весь мир и станет, по сути, безальтернативной.

Для того чтобы обеспечить уникальность используемых сетевых адресов IP, был создан специальный комитет, который называется IANA (Internet Assigned Number Authority) и предназначен для контроля над соблюдением правил назначения сетевых адресов. Непосредственное выделение уникальных сетевых адресов в регионах координируется центрами INTERNIC (Internet Network Information Center). Например, в Северной Америке эти функции наряду с некоторыми другими могут выполняться организацией ARIN (American Registry for Internet Numbers). В городах и регионах выделение адресного пространства осуществляется коммерческими организациями — провайдерами Интернета.

Однако потребности в адресном пространстве, которые существуют в настоящий момент, уже существенно превышают те возможности, которые могут быть реализованы при использовании существующей адресной схемы IP. Это объясняется появлением и широким распространением значительного числа приложений, которые функционируют в стеке TCP/IP HTTP (Hypertext Transfer Protocol), FTP и многие другие.

Помимо того что сам адрес IP не является достаточно длинным для того, чтобы использоваться для создания достаточного количества уникальных сетевых узлов, размер адресного пространства ограничивается также многочисленными "мертвыми" зонами. Действительно, если какой-либо организации (например, IBM) выделена сеть адресов IP класса A (9.0.0.0), размер адресуемого пространства сокращается приблизительно на 2^{24} независимо от того, может эта корпорация использовать это адресное пространство по назначению или нет. Для решения данной проблемы был разработан специальный механизм, который позволял использовать внеклассовое разбиение сетей. В этом случае сеть класса A может быть интерпретирована в качестве совокупности сетей класса B (например, 255 сетей 9.1.0.0—9.255.0.0). Однако этот метод не обеспечивает коренного решения проблемы и, кроме того, порождает дополнительные трудности (доработка протоколов маршрутизации и появление дополнительных "мертвых" зон). В рассмотренном выше примере адреса 9.1.0.0 и 9.1.255.255 уже не могут быть использованы для назначения узлам сети, поскольку являются адресом сети и адресом типа Directed Broadcast для данной сети, соответственно.

Вместе с тем было отмечено, что обеспечение глобальной уникальности используемого сетевого адреса не является актуальным для ряда классов корпоративных сетей, которые используют сервисы Интернета только для внутренних нужд. Для обеспечения возможности более широкого использования сервисов Интернета в корпоративных сетях комитет IANA зарезервировал специальное адресное пространство. Концепция частных сетей в Ин-

тернете и адресные диапазоны IP, которые могут быть использованы для построения данных сетей, приведены в RFC-1918.

Этот документ содержит определение нескольких категорий сетей, и относит к понятию "частная сеть" сети образованные из узлов, для обеспечения информационного взаимодействия между которыми не требуется обращения к глобальным ресурсам Интернета. К категории "частная сеть" этот документ относит также сети, узлы которых для обращения к информационным ресурсам глобальной сети используют специальные шлюзы.

В соответствии с RFC-1918, для построения сетей, которые попадают под определение "частная сеть", могут быть использованы диапазоны адресов Интернета, представленные в табл. 1.2.

Таблица 1.2. *Адреса "частных" сетей Интернет*

Класс	Начальный адрес	Конечный адрес	Число сетей
A	10.0.0.0	10.255.255.255	1
B	172.16.0.0	172.31.255.255	16
C	192.168.0.0	192.168.255.255	255

Диапазоны адресов, которые могут быть реально использованы для назначения узлам сети Интернет, представлены в табл. 1.3.

Таблица 1.3. *Адресное пространство Интернета*

Класс	Начальный адрес	Конечный адрес
A	0.1.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.1.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

Групповые адреса

Групповые IP-адреса предназначены для обращения к группе узлов по одному IP-адресу, а именно, адресу группы. Например, узлам P, Q и R назначен один групповой адрес — X. Тогда для того, чтобы отправить дейтаграмму всем трем узлам (P, Q и R), достаточно отправить ее по адресу X.

Групповые адреса принадлежат классу D и лежат в промежутке от 224.0.0.1 до 239.255.255.244. Из этого промежутка адресов группе узлов с определен-

ными IP-адресами присваивается дополнительный групповой адрес. Назначение этого адреса некоторому узлу означает, что этот узел становится приемником сетевого трафика, адресованного членам данной группы.

Внеклассовые сети Интернета

Для решения проблемы адресного дефицита в сети Интернет было предложено использовать внеклассовые сети. В сетевом адресе внеклассовых сетей граница между адресом сети и адресом узла определяется с точностью до бита. Таким образом, может быть создана сеть, у которой адрес сети занимает 30 разрядов, а адрес узла — 2 разряда сетевого адреса. Благодаря использованию таких внеклассовых сетей, адресное пространство сети Интернет может быть разделено на более мелкие непересекающиеся подпространства ("подсети") — subnets, с каждой из которых можно работать как с обычной сетью TCP/IP. Как правило, подсеть соответствует одной физической сети, например, одной ветви сети Ethernet.

Использование внеклассовых сетей позволяет экономить сетевые адреса и обеспечивает дополнительные удобства для администратора сети.

Допустим, ваша организация получила один сетевой адрес, например, адрес класса B. Предположим, что третий байт будет определять адрес подсети, а четвертый байт — адрес узла в ней, т. е. часть битов адреса сети заимствуется для использования их в качестве адреса подсети. На рис. 1.4 показан вариант организации подсети класса B.

Если администратор сети решил использовать восемь битов для организации подсети, то третий байт адреса IP-класса B обозначает адрес этой подсети. В приведенном примере адрес 128.10.1.0. относится к сети 128.10, подсети 1; адрес 128.10.2.0 относится к сети 128.10, подсети 2 и т. д.

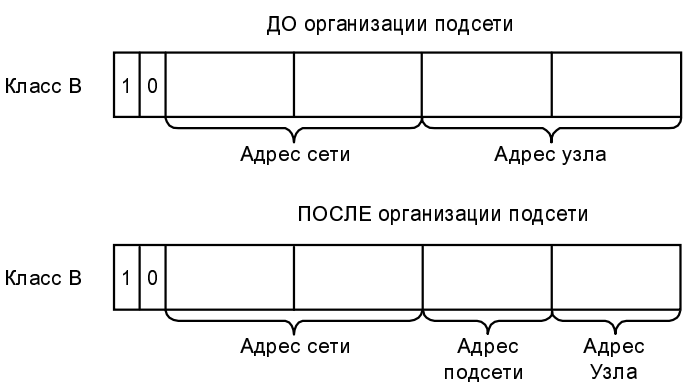


Рис. 1.4. Организация подсети сети класса B