МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ВОЛОГОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

А.А. СУКОНЩИКОВ

МЕТОДЫ, СРЕДСТВА И ПРОТОКОЛЫ ДОСТУПА К СРЕДЕ И УДАЛЕННЫМ ИНФОРМАЦИОННЫМ РЕСУРСАМ

Утверждено редакционно-издательским советом ВоГТУ
в качестве учебного пособия

Вологда

2013

УДК 004.772:004.057.4(075.8)
ББК 32.81
С 89

Рецензенты:

А.В. Тупицин, канд. техн. наук, доцент, начальник управления информатики и автоматики Вологодского отделения Сбербанка России;

В.Е. Марлей, д-р техн. наук, профессор, заведующий кафедрой вычислительных систем и информатики Государственного университета морского и речного флота им. адмирала С.О. Макарова

Суконщиков А.А.

С 89 **Методы, средства и протоколы доступа к среде и удаленным информационным ресурсам:** учебное пособие / А.А. Суконщиков.– Вологда: ВоГТУ, 2013. – 147 с.

ISBN 978-5-87951-500-9

Соответствует программе курса «Методы, средства и протоколы доступа к среде и удаленным информационным ресурсам». В учебном пособии рассмотрены основные методы, средства и протоколы доступа, что является частью любой информационной системы. Рассмотрены методы построения первичных и беспроводных сетей. Достаточно подробно раскрывается сетевой уровень, отвечающий за маршрутизацию и доставку данных в информационно-управляющих системах. Приведены примеры использования некоторых приложений сети Интернет, таких протоколов, как Telnet, NFS, SMTP, SNMP, и т.д.

Учебное пособие предназначено для студентов высших учебных заведений для направлений магистерской подготовки 220400.68 – Управление в технических системах и 230100.68 – Информатика и вычислительная техника. Может быть полезно инженерам, аспирантам при изучении и проектировании сетей АСУ.

Учебное пособие разработано в рамках проведения научно-исследовательской работы по заданию Министерства образования и науки РФ «Ситуационная интеллектуальная система поддержки принятия решения на основе извлечения знаний и параллельных вычислений в инфокоммуникационных сетях» (№ 01201253136).

УДК 004.772:004.057.4(075.8)
ББК 32.81

ISBN 978-5-87951-500-9

ПРЕДИСЛОВИЕ

Учебное пособие предназначено для студентов высших учебных заведений для направлений магистерской подготовки 220400.68 – УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ и 230100.68 – ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА. Назначение данного учебного пособия: для изучения теоретической и практической части дисциплины, а также для самостоятельной работы магистров. Пособие соответствует рабочей программе курса и основным задачам его освоения.

Материалы курса охватывают вопросы первичных и беспроводных сетей, особенно полно рассматриваются вопросы взаимодействия протокольных уровней. Вопросы, рассмотренные в учебном пособии, могут быть использованы в курсовом и дипломном проектировании при разработке методов и средств доступа к информационным ресурсам, а также в лабораторных занятиях при моделировании различных типов сетей.

В учебном пособии приведено множество справочных материалов, что позволяет использовать его для проектирования реальных сетей, поэтому может быть полезно инженерам, аспирантам при изучении и проектировании корпоративных вычислительных сетей.

Уважаемые читатели, если Вы заметите ошибки или неточности в издании учебного пособия, просим направлять свои отзывы и пожелания по E-mail: avt@vstu.edu.ru.

ВВЕДЕНИЕ

Целью настоящего учебного пособия является изложение сведений о методах, протоколах и средствах доступа к среде и информационным ресурсам, главный акцент сделан на современные сетевые стандарты и соответствующие им средства и протоколы. Для того чтобы понять проблемы функционирования современных методов, в пособии рассматриваются следующие вопросы:

- средства построения первичных сетей (волоконно-оптических, сети T1/E1, PDH, SDH, DWDM);
- беспроводные типы сетей, особенности подуровня MAC, взаимосвязи сетей;
- маршрутизация на сетевом уровне, протоколы TCP/IP и стандарты OSI/ISO, технология работы протоколов: IP, ARP, ICMP;
- протоколы маршрутизации в IP-сетях, особенности протокола RIP и OSPF;
- статическая и динамическая маршрутизация в информационных сетях;
- стандартные ARPA сервисы: файловый доступ, стандартные Berkeley-сервисы удаленного доступа, сетевая файловая система, технологии сетевого управления, электронная почта;
- драйверы сетевых адаптеров, установка стека TCP/IP.

Изучение рассмотренных методов, протоколов и средств поможет студентам в проектировании и работе на современных корпоративных сетях.

1. ПЕРВИЧНЫЕ СЕТИ

Эти сети являются основой вторичных сетей — компьютерных и телефонных. Как было показано выше, обеспечение QoS зависит от имеющегося резерва пропускных способностей, т. е. от производительности первичной сети. Без развития первичных сетей невозможен прогресс сетевых технологий. Такие сети называют также ***опорными*** и ***базовыми***.

Современные первичные сети основаны на коммутации каналов. Для создания абонентского канала коммутаторы первичных сетей поддерживают один из методов мультиплексирования и коммутации.

Методы мультиплексирования

В настоящее время для мультиплексирования абонентских каналов используются:

- Техника частотного мультиплексирования — Frequency Division Multiplexing, **FDM**
- Мультиплексирование с разделением времени — Time Division Multiplexing, **TDM**
- Мультиплексирование по длине волны — Wave Division Multiplexing, **WDM**

Частотное мультиплексирование

Речевой телефонный канал имеет спектр 300 – 3400 Гц, т.е. на его передачу необходима пропускная способность 3100 Гц. Кабельные же системы между телефонными коммутаторами имеют пропускную способность в сотни мегагерц. Для передачи производится модуляция высокочастотного сигнала низкочастотным. Таким образом, спектр модулированного сигнала переносится в другой частотный диапазон. Высокочастотный сигнал делится на полосы по 4000 Гц (3100 + 900 – страховой промежуток). В сетях на основе FDM-коммутации принято несколько уровней уплотненных каналов.

Первый уровень — 12 абонентских каналов. Они образуют ***базовую группу*** с полосой частот 48 КГц (от 60 до 108 КГц). Такой канал называют ***уплотненным***.

Второй уровень составляют 5 базовых групп, которые образуют ***супергруппу*** с полосой частот 240 КГц (от 312 до 552 КГц). Такая супергруппа передает данные 60 абонентских каналов.

Третий уровень — это десять супергрупп, которые образуют ***главную группу***. Она передает данные 600 абонентов и имеет полосу пропускания 2520 КГц (564 – 3084 КГц).

Мультиплексирование по длине волны

Первичные сети с мультиплексированием по длине волны (WDM и DWDM) используют тот же принцип частотного разделения, но информационным сигналом в них является не электрический ток, а свет. Используется инфракрасный диапазон с длинами волн от 850 до 1565 нм, что соответствует частотам 196 – 350 ТГц. В магистральном канале обычно мультиплексируются достаточно много спектральных каналов: 16, 32, 40, 80 или 160. Если используется 16 и более каналов, такая техника часто называется плотной, т.е. Dense WDM или DWDM. Внутри спектрального канала данные могут кодироваться как дискретным, так и аналоговым способом.

Коммутация каналов на основе разделения времени

Как уже упоминалось, эта техника носит название «Мультиплексирование с разделением времени» (Time Division Multiplexing, TDM). Наибольшее распространение в этом классе получили каналы типа T1/E1.

1.1. Каналы T1/E1

Эти каналы появились уже давно. Они были предложены для передачи вызовов между телефонными станциями (АТС) (см. рис. 1.1). Это дуплексные цифровые каналы.

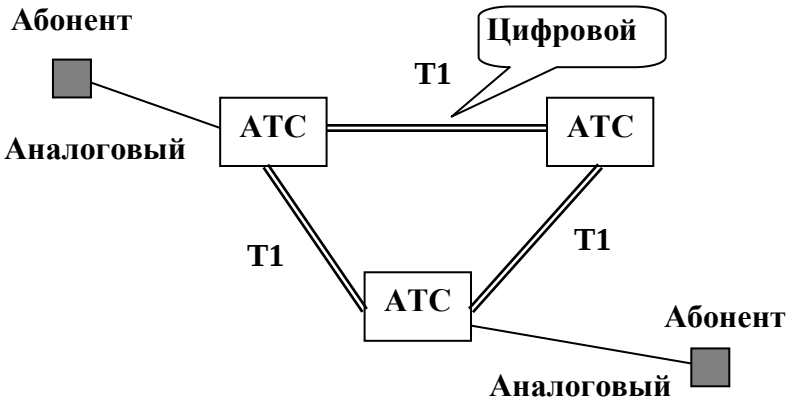


Рис. 1.1

Для передачи используются две пары витых проводников (по паре в каждую сторону). В 90-е годы эти каналы стали очень популярны в качестве

средства подключения абонентов (небольших фирм, корпоративных сетей) к сети Интернет.

Метод AMI

В канале применяют биполярное кодирование (*bipolar encoding*) (рис. 1.2). У этого метода есть и другое название — альтернативное кодирование логических единиц (*Alternative Mark Inversion, AMI*).

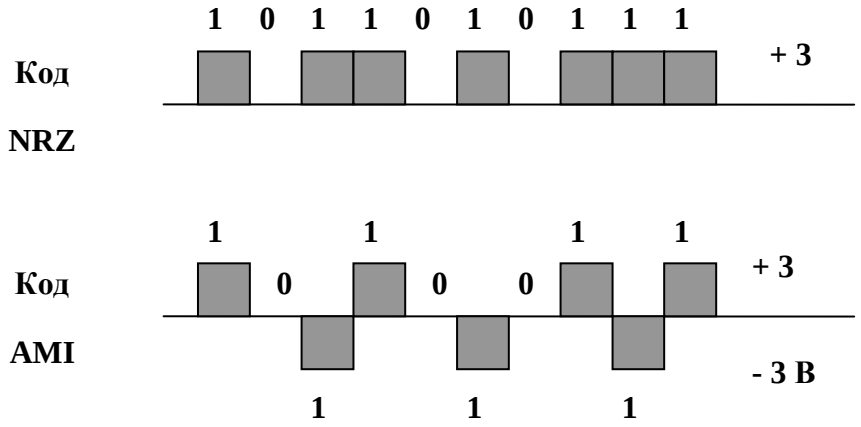


Рис. 1.2

Отсутствие напряжения в линии соответствует нулю, а для представления единиц используются по очереди положительные и отрицательные импульсы.

Проблема синхронизации

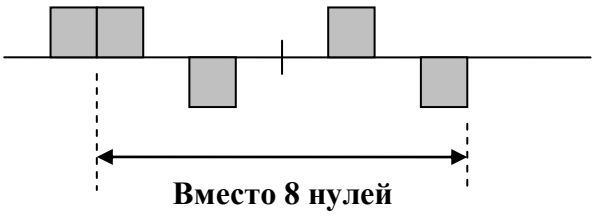


Рис. 1.3

Длинная последовательность логических нулей может привести к потере синхронизации у приемника. Для борьбы с этим применяют метод биполярной замены 8 нулей – *Bipolar 8 Zero Substitution, B8ZS*. Каждая обнаруженная передатчиком группа из 8 нулей заменяется специальной последовательностью (рис. 1.3). При приеме же на ее место опять вставляется 8 нулей. Для идентификации этой последовательности используется передача без инвертирования двух единиц (что недопустимо при нормальном режиме работы кода AMI).

Импульсно-кодовая модуляция

Первоначально каналы T1/E1 предназначались для передачи телефонных разговоров, но не по аналоговой, а по цифровой линии. Как уже говорилось, по обычному телефонному каналу передается аналоговый сигнал в диа-

пазоне частот от 300 до 3400 Гц. Для перевода этого аналогового сигнала в цифровую форму применяется импульсно-кодовая модуляция (ИКМ) — *Pulse Code Modulation, PCM*. Для этой цели вводится блок АЦП, который переводит амплитуду аналогового сигнала в цифровой отсчет из 8 бит. Частота снятия таких отсчетов выбрана с учетом теоремы Найквиста, в соответствии с которой для адекватного преобразования сигнала из аналоговой формы в цифровую частота дискретизации должна в 2 раза превышать частоту дискретизируемого сигнала. Таким образом, цифровая линия должна обладать пропускной способностью

$$8 \times 8000 = 64 \text{ Кбит/с}.$$

Мультиплексирование

В линии T1 собираются вместе 24 цифровых канала по 64 Кбит/с, т.е. в итоге пропускная способность составляет 1,544 Мбит/с. Для объединения применяется временное мультиплексирование каналов. Вся доступная полоса частот канала делится на временные интервалы по 125 мкс. Устройство монополизирует всю полосу частот на период такого элементарного интервала.

Техника мультиплексирования TDM иногда называется также техникой синхронного режима передачи — *Synchronous Transfer Mode, STM*. Аппаратура TDM-сетей — мультиплексоры, коммутаторы и демультиплексоры — работает в режиме разделения времени, поочередно обслуживая в течение цикла своей работы все абонентские каналы (рис. 1.4).

Цикл работы оборудования TDM равен 125 мкс, что соответствует периоду следования замеров голоса в цифровом абонентском канале. Каждому соединению выделяется один квант времени цикла работы аппаратуры, называемый также **тайм-слотом. Мультиплексор** принимает информацию по N входным каналам от конечных абонентов, каждый из которых передает данные со скоростью 64 Кбит/с — один байт каждые 125 мкс. В каждом цикле мультиплексор выполняет следующие действия:

- прием из каждого канала очередного байта данных
- составление из принятых байтов уплотненного кадра, называемого также обоймой;
- передача обоймы в выходной канал со скоростью N x 64 Кбит/с.

Порядок следования байтов в обойме соответствует номеру входного канала. Количество обслуживаемых мультиплексором каналов зависит от скорости. Для мультиплексора T1 – это 24 входных канала, и выходная скорость составляет 1,544 Мбит/с.

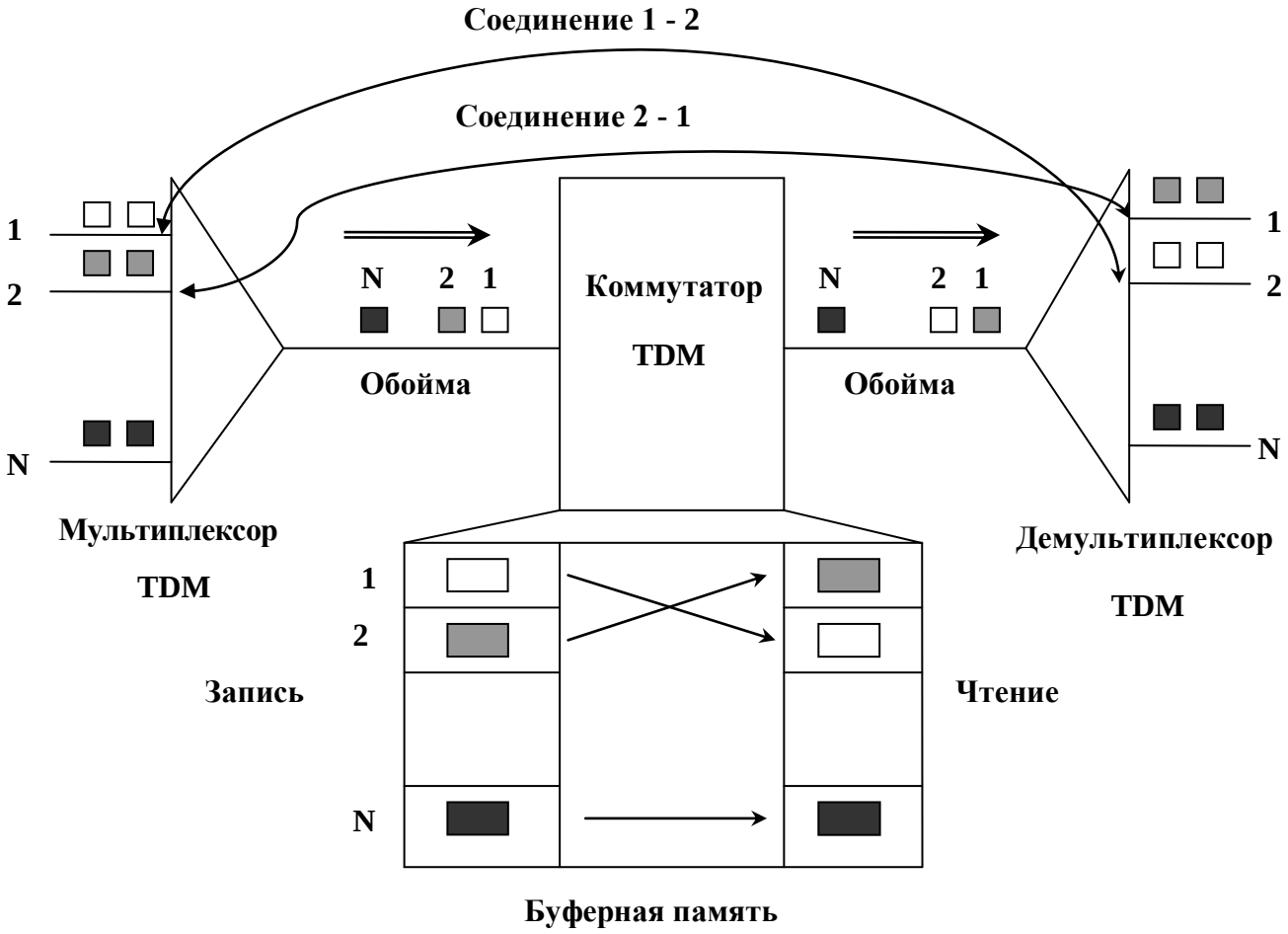


Рис. 1.4

У **демультиплексора** задача обратная – разборка обоймы и распределение байтов по выходным каналам. При этом считается, что порядок байта в обойме соответствует номеру выходного канала.

Коммутатор записывает байты в память и заново формирует обойму, ставя байт в нужное место (соответствующее номеру выходного канала). Выделенный номер тайм-слота все время остается в распоряжении входного канала (даже если по нему ничего не передается!).

1.2. Сети PDH

На базе внедренных в телефонии каналов T1 к настоящему времени сформировалось два поколения таких цифровых первичных сетей:

- Технология **плезиохронной цифровой передачи** (*Plesiochronous Digital Hierarchy, PDH*). «Плезо» означает «почти», т.е. это почти синхронная передача.

- **Синхронная цифровая иерархия** (*Sinchronous Digital Hierarchy, SDH*). В США технология SDH называется SONET.

Первой была разработана рассмотренная выше технология T1. Четыре канала T1 объединяются в канал следующего уровня — T2 (6,312 Мбит/с). Семь каналов T2 образуют канал T3 (44,736 Мбит/с) и т. д.

С середины 70-х годов каналы T1 стали сдаваться телефонными компаниями в аренду. Эта технология PDH была позже стандартизована CCITT с внесением некоторых изменений. В результате версия CCITT стала применяться в Европе (это каналы E1, E2, E3 и т. д. — 2,048; 8,488; 34,368 и т. д. Мбит/с), а в США, Канаде, Японии продолжает применяться стандарт T1 (T2, T3, …). На практике используются в основном каналы: T1/E1 и T3/E3.

Реализация мультиплексирования

Мультиплексор T1 передает в кадре 24 байта данных абонентов (со скоростью 1,544 Мбит/с) и вставляет один бит синхронизации. Таким образом, скорость пользовательских каналов составляет 64000 x24 = 1,536 Мбит/с, а еще 8 Кбит/с добавляют биты синхронизации.

При передаче компьютерных данных канал T1 предоставляет для пользовательских данных 23 канала, а 24-й канал остается для служебных целей (в основном, для восстановления искаженных кадров).

В канале E1 для служебных целей отводится 2 байта из 32. Это нулевой и 16-й байты (для целей синхронизации). Для передачи пользовательских данных остается 30 каналов. Пользователь может арендовать несколько каналов по 64 Кбит/с в канале T1/E1. Такой канал называется дробным (*fractional*) каналом T1/E1. Для усиления сигнала T1/E1 через каждые 1800 м ставятся регенераторы и аппаратура контроля линии. Наиболее высокоскоростная технология PDH — это DS-4.

Для технологии T1 — это 4032 голосовых канала (274 Мбит/с).
Для технологии E1 — это 1920 голосовых канала (139 Мбит/с).

Недостатки технологии PDH

- Сложность мультиплексирования и демультиплексирования данных. Например, для извлечения одного канала из потока T3 надо демультиплексировать канал до T2, затем – до T1, а уже затем демультиплексировать канал T1.
- Отсутствие встроенных средств контроля и управления сетью. Нет и процедур поддержки отказоустойчивости.
- Слишком низкие (по понятиям современных сетей) скорости передачи. Иерархия скоростей E1 заканчивается на 139 Мбит/с, а современные опто-

волоконные каналы позволяют передавать со скоростью в десятки гигабит в секунду.

Эти недостатки были устранены в сетях SDH.

1.3. Сети SDH

Технология синхронной цифровой иерархии (*Sinchronous Digital Hierarchy, SDH*) разработана для создания надежных транспортных каналов, позволяющих гибко формировать цифровые каналы в широком диапазоне скоростей – от единиц мегабит в секунду до десятков гигабит в секунду. Основная область применения — первичные сети операторов связи. Эти сети относятся к классу полупостоянных сетей – формирование канала происходит по инициативе оператора связи. Поэтому здесь чаще всего вместо термина «коммутация» используют термин «кросс-коннект» (*cross-connect*).

Это сети с коммутацией каналов. Используется мультиплексирование с разделением времени TDM. Информация адресуется путем относительного временного положения внутри составного кадра. Обычно эти сети используются для объединения большого числа более низкоскоростных каналов PDH. На рис. 1.5 приведен пример структуры такой сети.

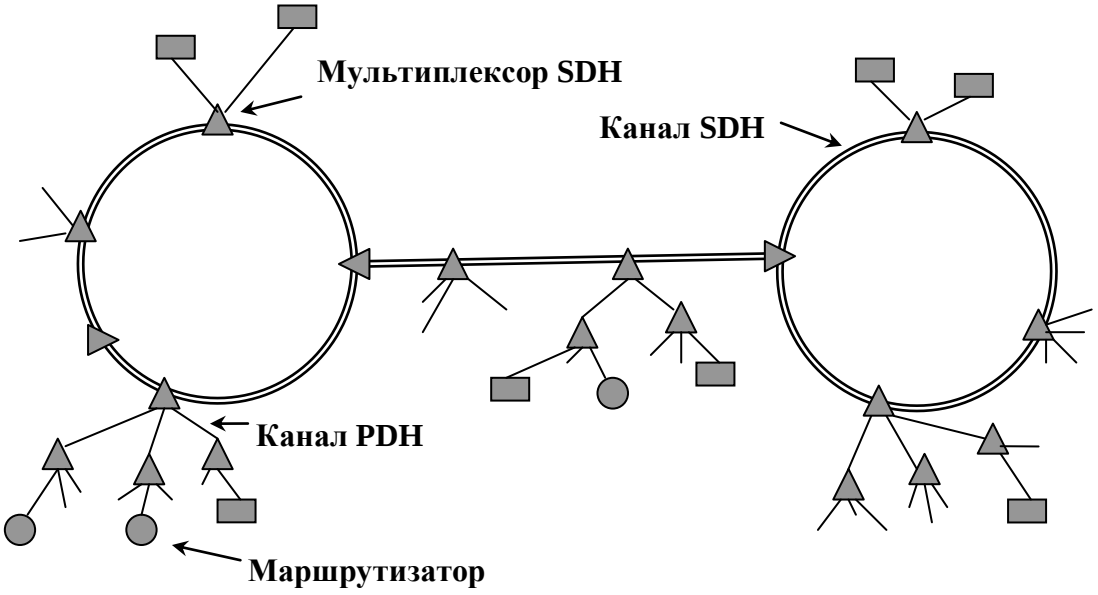


Рис. 1.5

Достоинства сетей SDH

Эти сети обладают следующими преимуществами.

1. Гибкая иерархическая система мультиплексирования цифровых потоков с различными скоростями. Возможность ввода и вывода пользовательской информации без демультиплексирования потока в целом.

2. Отказоустойчивость сети. Использование резервных маршрутов и резервного оборудования. Переход на резервный путь обычно требует не более 50 мс.

3. Мониторинг и управление сетью на основе той информации, которая встроена в заголовки кадров.
4. Высокое качество транспортного обслуживания для трафика любого вида — голосового, видео и компьютерного. Техника мультиплексирования TDM, лежащая в основе SDH, обеспечивает трафику каждого абонента гарантированную пропускную способность, а также низкий и фиксированный уровень задержек.

Сети SDH составляют сегодня фундамент практически всех крупных телекоммуникационных сетей — региональных, национальных и международных. Эти сети легко интегрируются с сетями DWDM, обеспечивая передачу информации по оптическим магистралям со скоростями сотни гигабит в секунду за счет мультиплексирования с разделением по длине волны.

В DWDM сетях сети SDH играют роль сетей доступа, т.е. ту же роль, что играют по отношению к ним сети PDH (рис. 1.6).

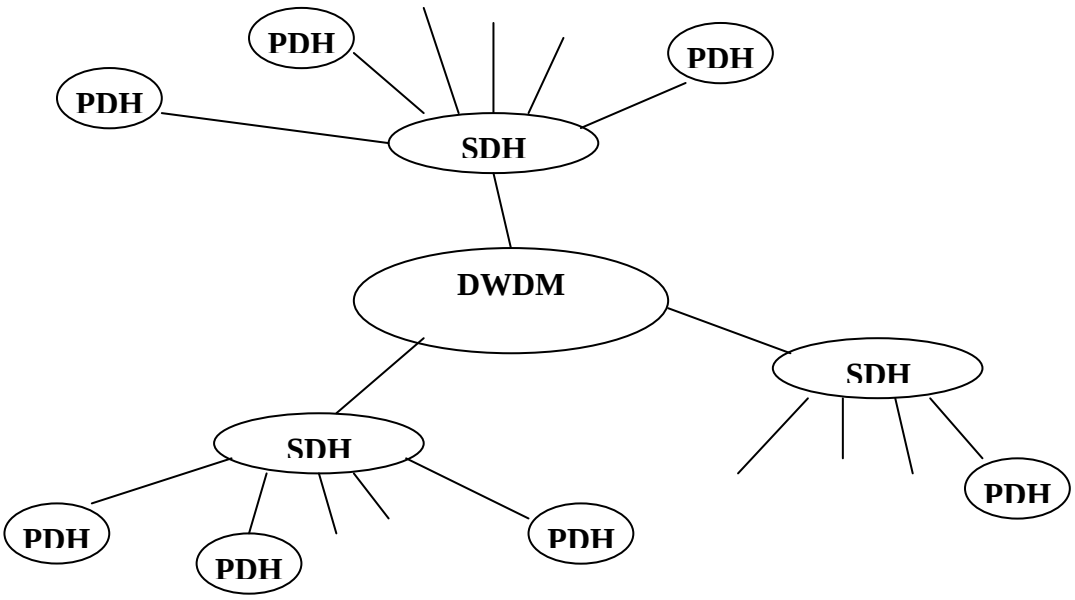


Рис. 1.6

Возникновение сетей SDH

Эта сеть разработана компанией Bellcore под названием *Synchronous Optical Nets* (*SONET*) – синхронные оптические сети. Она явилась развитием технологии PDH (T1/T3, E1/E3), которая к тому времени уже перестала справ-

ляться с потоками информации. Была поставлена также задача избавиться от недостатков сетей PDH (таких, как невозможность выделения низкоскоростного потока без полной разборки высокоскоростного и отсутствие встроенных средств обеспечения отказоустойчивости и управления сетью).

Первый вариант сети SONET появился в 1984 г. Затем эта технология была стандартизирована институтом стандартов США ANSI. Далее в результате длительной работы был создан стандарт ITU-T на сети SDH.

Иерархия скоростей

Эта технология обеспечивает следующий диапазон скоростей:

SONET: OC-1 = 51,84 Мбит/с … OC-768 = 39,81 Гбит/с

SDH: STM-1 = 15,52 Мбит/с … STM-256 = 39,81 Гбит/с

Обозначение STM = *Synchronous Transport Module* (*level* 1 — 256)

Структура кадров

Кадры STM-N имеют сложную структуру, позволяющую агрегировать в общий магистральный поток потоки SDH и PDH разных скоростей, а также выполнять операции ввода-вывода без полного мультиплексирования магистрального потока. Операции мультиплексирования и ввода-вывода выполняются с использованием **виртуальных контейнеров** (*Virtual Container, VC*), которые позволяют переносить через сеть блоки данных PDH. Каждый блок VC содержит, кроме блоков PDH, служебную информацию. К служебной информации относится, например:

- заголовок пути контейнера (*Path OverHead, POH*) — в нем переносится информация о процессе прохождения контейнера от начальной до конечной точки (сообщения об ошибках);
- индикатор установления соединения между конечными точками и т. д.

Это, разумеется, увеличивает размер самого блока. Например, VC-12 переносит 32 байта потока данных E1, но при этом состоит из 35 байт.

В сети SDH определены следующие типы VC для транспортировки основных типов блоков данных PDH:

- VC-11 — 1,5 Мбит/с
- VC-12 — 2 Мбит/с
- VC-2 — 6 Мбит/с
- VC-3 — 34 Мбит/с
- VC-4 — 140 Мбит/с

Виртуальные контейнеры являются единицей коммутации мультиплексоров SDH. В каждом мультиплексоре имеется таблица соединений, ее называют таблицей кросс-соединений, в которой, например, указано:

…
VC-12 порта P1 соединен с VC-12 порта P5
VC-3 порта P8 соединен с VC-3 порта P9
…

Эту таблицу формирует вручную администратор сети. Она обеспечивает сквозные пути для соединения конечных точек.

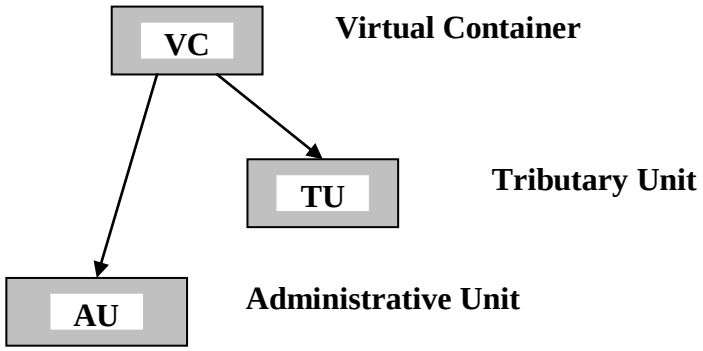


Рис. 1.7

Используется понятие указателя (pointer) – одно из ключевых понятий технологии SDH. Указатель определяет текущее положение VC в структуре более высокого уровня – трибутарном блоке (*Tributary Unit, TU*) или в административном блоке (*Administrative Unit, AU*) (рис. 1.7).

Система указателей позволяет мультиплексору находить положение пользовательских данных в синхронном потоке байтов кадров STM-N и «на лету» извлекать их оттуда. Трибутарные блоки могут объединяться в группы (TUG), которые, в свою очередь, входят в административные блоки. Группа из N административных блоков (AUG) и образует, в конечном счете, полезную нагрузку кадра STM-N. Помимо этого в кадр входит заголовок с общей для всех административных блоков служебной информацией (рис. 1.8).

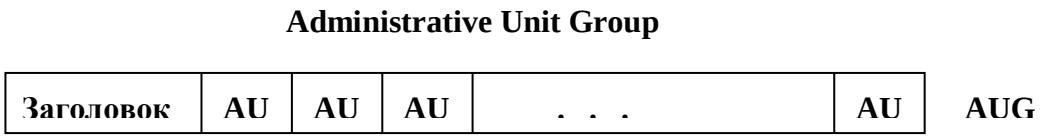


Рис. 1.8

Схема мультиплексирования SDH предоставляет разнообразные возможности для объединения пользовательских потоков PDH. Например, возможна следующая структура (рис. 1.9):

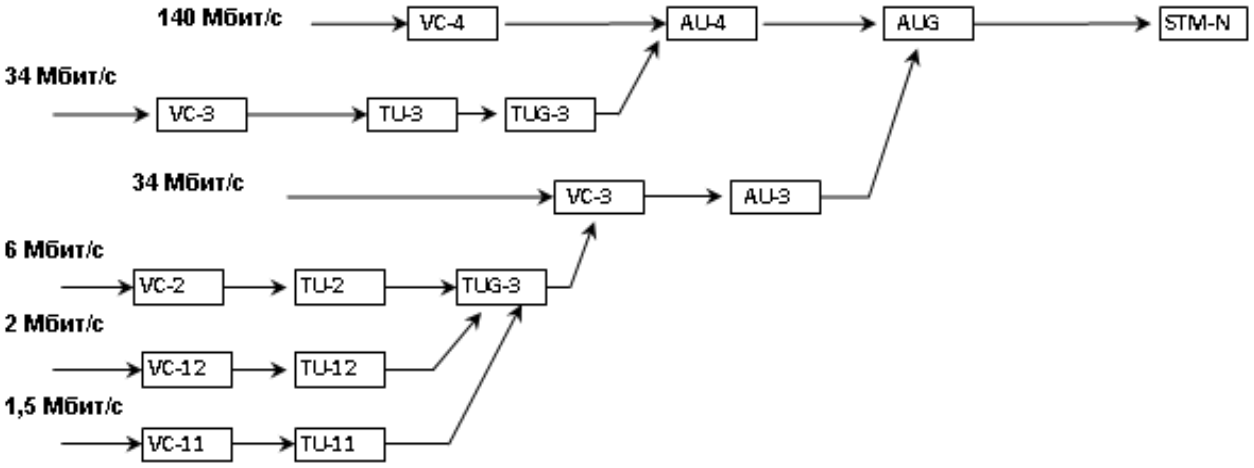


Рис. 1.9

Например, для кадра STM-1 можно реализовать такие варианты:

- один поток E4
- 63 потока E1
- 1 поток E3 и 42 потока E1.

Типы оборудования

Основным элементом сети SDH является мультиплексор. Он оснащен определенным количеством портов PDH и SDH. Порты мультиплексора делятся на:

- Агрегатные;
- трибутарные (сокращенно называются «трибами»).

Трибутарные порты часто называют **портами ввода-вывода**. Агрегатные порты определяют также в качестве **линейных портов**. Мультиплексоры SDH делятся на (рис. 1.10):

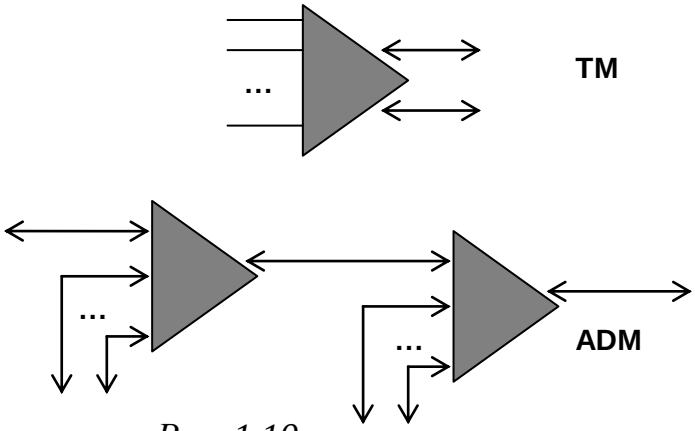


Рис. 1.10

- Терминальные мультиплексоры (*Terminal Multiplexor, TM*);
- Мультиплексоры ввода-вывода (*Add-Drop Multiplexo, ADM*).

Разница определяется положением мультиплексора в сети. Терминальный мультиплексор (TM) завершает агрегатные каналы, мультиплексируя в них большое количество каналов ввода-вывода. Мультиплексор ввода-вывода (ADM)

транзитом передает агрегатные каналы, занимая промежуточное положение на магистрали (в кольце, в цепи или смешанной топологии). При этом еще производится и ввод-вывод трибутарных каналов в агрегатный канал.

Агрегатные порты мультиплексора поддерживают максимальный для данной модели уровень скорости STM-N (например, STM-4 или STM-64). Иногда различают так называемые кросс-коннекторы (*Digital Cross-Connect, DXC*), которые выполняют операции над произвольными виртуальными контейнерами. Большинство же производителей выпускают **универсальные мультиплексоры**, которые могут использоваться как терминальные, ввода-вывода и кросс-коннекторы. Кроме мультиплексоров в состав сети SDH могут входить **регенераторы**. Они позволяют бороться с затуханием сигнала. Выполняется преобразование:

Свет → Электрический ток → Усиление → Свет

Стек протоколов

Стек протоколов SDH включает 4 уровня протоколов (рис. 1.11).

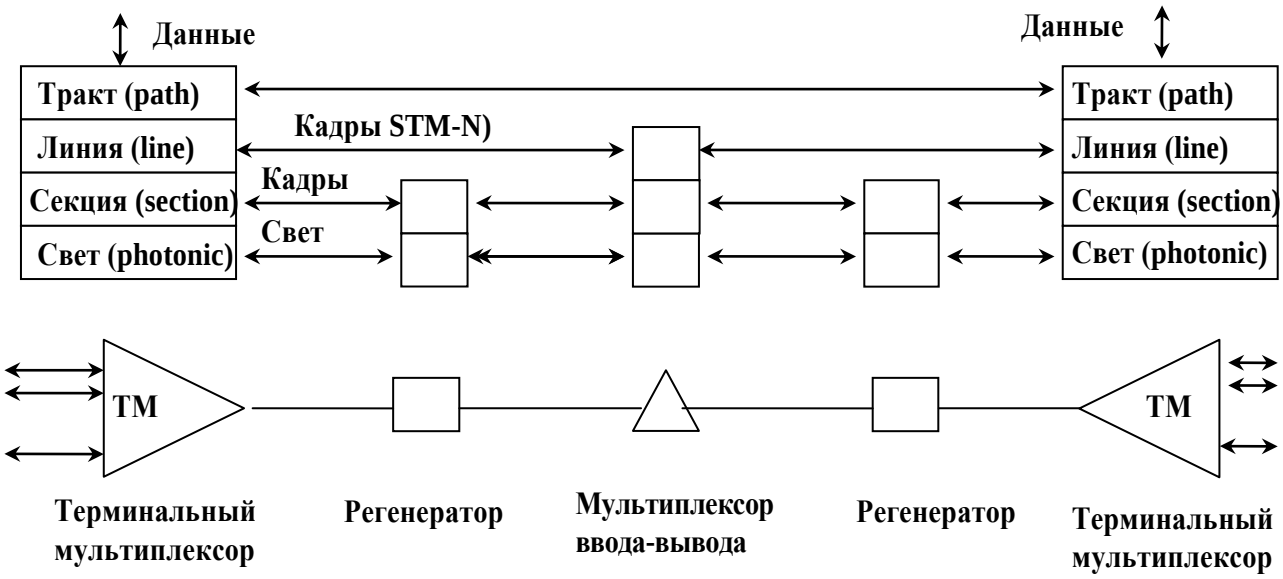


Рис. 1.11

▪ **Физический уровень**, называемый также в стандартах фотонным (*photonic*), имеет дело с кодированием битов информации. Используется потенциальный код NRZ.

▪ **Уровень секции** (*section*) поддерживает физическую целостность системы. Секцией в технологии SDH называют каждый непрерывный отрезок оптоволоконного кабеля, который соединяет пару устройств SONET/SDH (например,

мультиплексор, регенератор). Секцию часто называют также **регенераторной секцией**. Протокол регенераторной секции имеет дело с определенной частью заголовка кадра, которая называется заголовком регенераторной секции (RSOH), и на основе служебной информации может проводить тестирование секции и поддерживать операции административного контроля.

▪ **Уровень линии** (*line*). Отвечает за передачу данных между мультиплексорами. Протокол работает с кадрами STM-N, выполняя мультиплексирование и демультиплексирование, а также вставку и удаление пользовательских данных. Этот протокол отвечает также за проведение операций по реконфигурированию линии в случае отказа какого-то ее сегмента – оптоволокна, порта или соседнего мультиплексора. Линию часто называют также **мультиплексорной секцией.**

▪ **Уровень тракта** (path). Отвечает за доставку данных между двумя конечными пользователями сети. Тракт (путь) — это составное виртуальное соединение между пользователями. Протокол тракта должен принимать данные, поступающие в пользовательском формате (например, в формате T1) и последовательно преобразовывать их в синхронные кадры STM-N, т.е.:

$$\mathbf{T1} \rightarrow \text{VC} \rightarrow \text{TU} \rightarrow \text{TUG} \rightarrow \text{AU} \rightarrow \text{AUG} \rightarrow \mathbf{STM\text{-}N}$$

Кадры STM-N

Рассмотрим в качестве примера структуру самого простого кадра STM-1 (рис. 1.12).

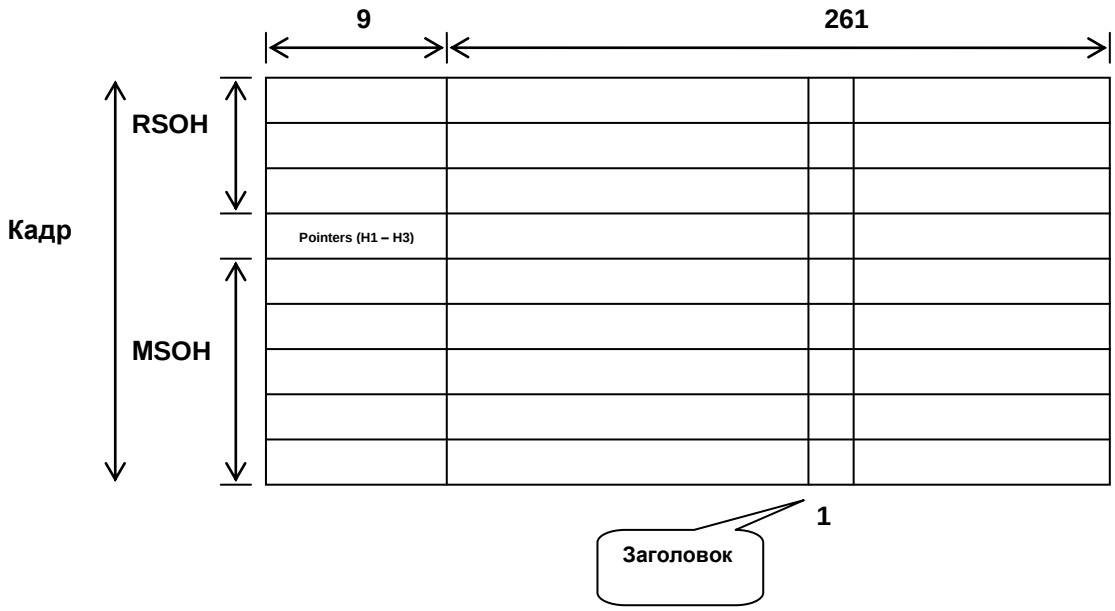


Рис. 1.12

Кадр представлен в виде матрицы, содержащей 270 столбцов и 9 строк. Первые 9 байтов каждой строки отводятся под служебные данные заголовков, из остальных 261 байта – 260 отводится под полезную информацию (данные VC, TU, TUG, AU, AUG), а один байт используется для заголовка тракта, что позволяет контролировать соединение из конца в конец. Заголовок регенераторной секции RSOH включает:

- Синхробайты
- Байты контроля ошибок для регенераторной секции
- Три байта канала передачи данных (*Data Communication Channel*, *DCC*), работающего со скоростью 192 Кбит/с
- Один байт служебного аудиоканала (64 Кбит/с)
- Резервные байты (для использования местными операторами связи)

Указатели H1, H2 H3 задают положение начала виртуального контейнера VC-4 или трех виртуальных контейнеров VC-3. Заголовок протокола мультиплексной секции (MSOH) содержит:

- Байты контроля ошибок для мультиплексной секции
- Шесть байтов канала передачи данных (*Data Communication Channel*, *DCC*), работающего со скоростью 576 Кбайт/с
- Два байта протокола автоматической защиты трафика, обеспечивающего живучесть сети.
- Байт передачи сообщений статуса системы синхронизации
- Резервные байты.

Типовые топологии

Чаще всего используются: кольца, линейные цепи и ячеистая топология, близкая к полносвязной. **Кольцо SDH** (рис. 1.13) строится из мультиплексоров ввода-вывода (ADM), имеющих, по крайней мере, два агрегатных порта. Пользовательские потоки вводятся и выводятся через трибутарные порты.

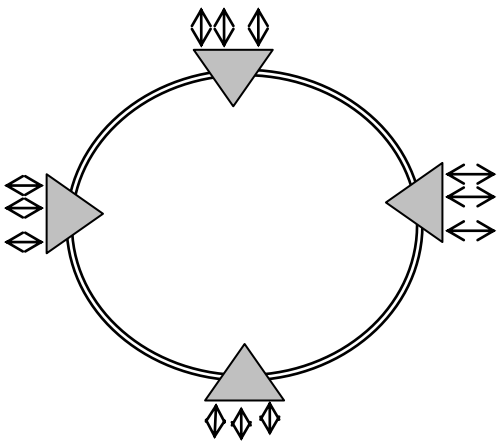


Рис. 1.13

Топология обладает отказоустойчивостью – при однократном обрыве кабеля данные можно направить по кольцу в обратном направлении. Кольцо обычно строится на основе кабеля с двумя оптическими волокнами (но иногда для повышения надежности используют и 4 волокна).

Цепь – это линейная последовательность мультиплексоров, из которых 2 крайних – терминальные, а остальные – мультиплексоры ввода-вывода (рис. 1.14). Такую сеть удобно располагать, например, вдоль магистрали железной дороги или трубопровода.

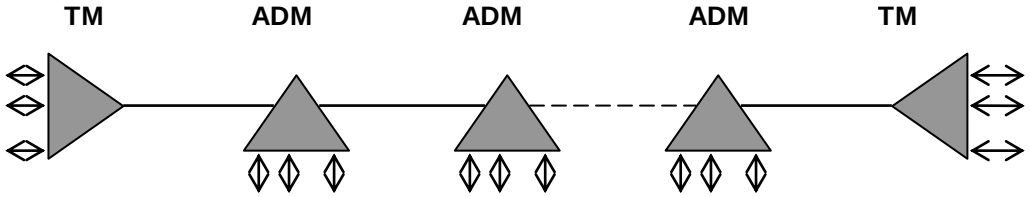


Рис. 1.14

Может применяться также и плоское кольцо, обеспечивающее более высокий уровень отказоустойчивости за счет использования двух дополнительных волокон в магистральном канале и по одному дополнительному агрегатному порту у терминальных мультиплексоров (рис. 1.15).

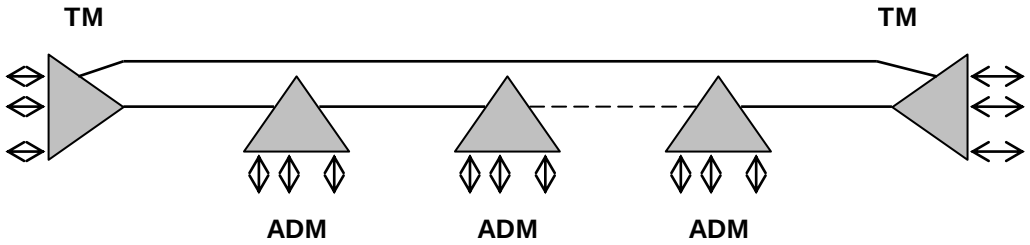


Рис. 1.15

Ячеистая топология (рис. 1.16) является наиболее общим случаем. При этом обеспечивается очень высокий уровень производительности и надежности.

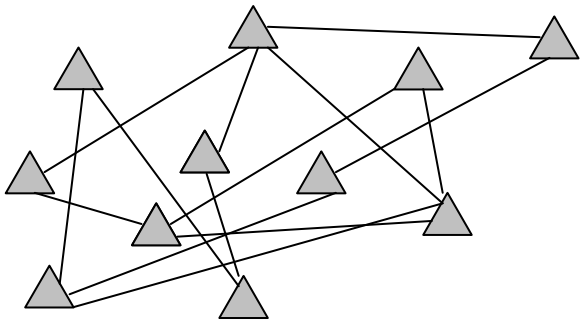


Рис. 1.16

Методы обеспечения живучести сети

В SDH-сетях применяются разнообразные средства обеспечения отказоустойчивости. Это позволяет очень быстро (за десятки мс) восстанавливать работоспособность, в случае отказа канала, карты мультиплексора или мультиплексора в целом. Механизмы обеспечения отказоустойчивости определяются в SDH как автоматическое защитное переключение (*Automatic Protection Switching, APS*), что отражает тот

факт, что в этом случае происходит переход на резервный путь или резервный элемент. Сети, поддерживающие такой механизм, определяются в стандартах SDH как selfhealing – самоизлечивающиеся. Применяются следующие виды автоматической защиты:

- ***Equipment Protection Swithing, EPS*** — защита блоков и оборудования SDH
- ***Card Protection, CP*** — защита агрегатных и трибутарных карт мультиплексора
- ***Multiplex Section Protection, MSP*** — защита мультиплексной секции (между двумя смежными мультиплексорами)
- ***Sub-Network Connection Protection, SNC-P*** — защита пути (соединения) через сеть для определенного виртуального контейнера
- ***Multiplex Section Shared Protection Ring, MS-SPRing*** — разделенная между пользовательскими соединениями защита путей в кольцевой топологии.

В сетях SDH примеряют следующие три схемы защиты: «1+1», «1:1» и «1:N».

- **Защита 1+1** означает, что резервный элемент выполняет ту же работу, что и основной. Например, при защите трибутарной карты трафик проходит как через рабочую (резервируемую), так и через защитную (резервную).
- **Защита 1:1**. Защитный элемент в нормальных условиях не выполняет функции защищаемого элемента, а переключается на них только в случае отказа.
- **Защита 1:N**. Предусматривается выделение одного защитного элемента на N защищаемых.

- **Защита блоков (EPS)** применяется для таких блоков мультиплексора, как процессорный блок, блок коммутации, блок питания, блок ввода сигналов синхронизации. Схема защиты: 1+1 или 1:1.
- **Защита карт (CP)** – агрегатных и трибутарных. Схемы – 1:1, 1+1 и 1:N. Защита типа 1+1 обеспечивает непрерывность сервиса по транспортировке данных.

При использовании двух трибутарных карт по схеме 1+1 используется еще и специальная карта-переключатель, которая коммутирует входные и выходные потоки карт.

- Для **защиты мультиплексной секции MSP** обычно применяется схема 1+1. При отказе время переключения на резервный путь не должно превышать 50 мс.
- **Защита соединения SNC-P** обеспечивает переключение определенного пользовательского соединения на альтернативный путь при отказе основного пути. Схема защиты 1+1.

- **Защита с разделением кольца MS-SPRing**. Здесь не резервируется заранее пропускная способность для каждого соединения. Вместо этого резервируется половина пропускной способности кольца и эта резервная полоса выделяется для соединений динамически, по мере необходимости, т.е. при обнаружении факта отказа линии или мультиплексора.

Синхронизация в сетях SDH

Устойчивая работа SDH-сети зависит от качества синхронизации между узлами. Применяется иерархический метод принудительной синхронизации с парами таймеров «ведущий-ведомый». Для обеспечения отказоустойчивости мультиплексор SDH может использовать несколько дублирующих источников синхронизации. Это такие источники, как:

- Сигнал внешнего сетевого таймера с частотой 2048 КГц. Он называется также первичным эталонным генератором (*Primary Reference Clock, PRC*). Точность не ниже $1x10^{-11}$. Используются цезиевые или рубидиевые (атомные) часы. Калибровка выполняется по сигналам мирового скоординированного времени (*Universal Time Coordinated, UTC*).
- Сигнал вторичного задающего генератора. Он является ведомым по отношению к первичному.
- Сигнал от спутника глобальной системы позиционирования (*Global Positioning System, GPS*).
- Сигнал внутреннего таймера узла SDH. Его точность обычно невелика – $(1 – 5)x10^{-6}$.
- Сигнал 2048 КГц, выделяемый из линейного или трибутарного сигнала STM-N. Его точность обычно составляет $5x10^{-8}$.
- Сигнал пользовательского (трибутарного) интерфейса PDH.

Основными источниками надежной и точной синхронизации являются сигналы первичного эталонного генератора и сигналы, выделяемые из кадров STM-N.

Создается иерархия синхронизирующих источников — сеть, состоящая из нескольких слоев генераторов, называемых также стратум-таймерами. Эта сеть содержит один генератор слоя Stratum 1 и несколько генераторов уровней Stratum 2 – Stratum 4 (рис. 1.17).

Генераторы слоя Stratum 2 устанавливаются обычно в транзитных узлах сети. Генераторы Stratum 3 устанавливаются чаще всего в терминальных узлах сети. Для каждого генератора установлены следующие требования к точности:

- Stratum 1 — 10^{-11}
- Stratum 2 — $1{,}6x10^{-8}$
- Stratum 3 — $4{,}6x10^{-6}$
- Stratum4 — $20x10^{-6}$

При потере синхронизации от более высокого уровня генераторы Stratum 2 и Stratum 3 должны перейти в режим удержания частоты (*holdover*) и от них требуется возможность автономной работы в течение не менее 24 часов.

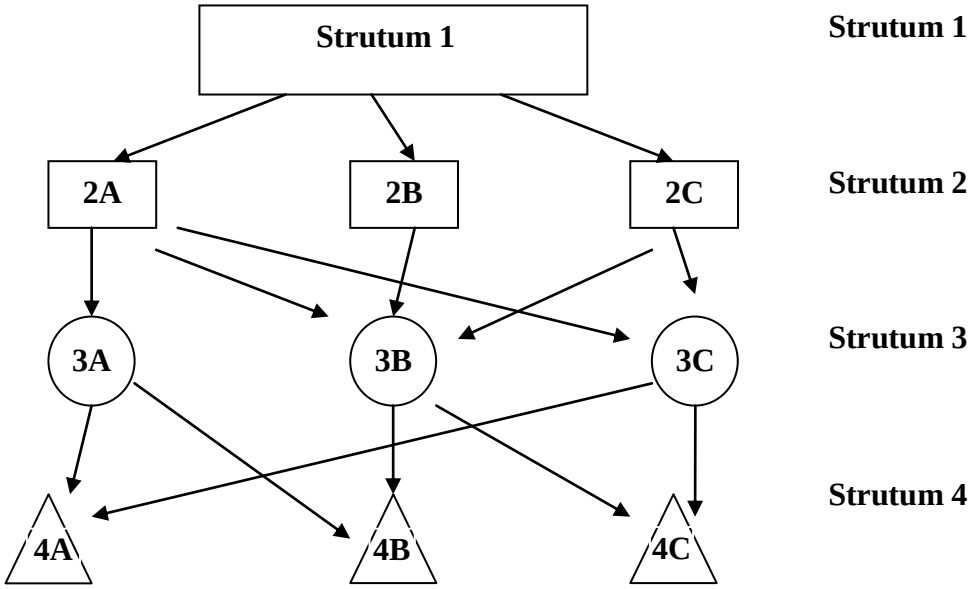


Рис. 1.17

Для надежной работы сети у каждого мультиплексора SDH должно быть несколько альтернативных источников синхронизации. Но в каждый момент времени используется один — самый точный источник.

1.4. Сети DWDM

Технология плотного волнового (спектрального) мультиплексирования (*Dense Wave Division Multiplexing, DWDM*) предназначена для создания оптических магистралей нового поколения, работающих на мультимегабитных и терабитных скоростях.

Информация в оптическом волокне передается одновременно большим количеством световых волн. Сети DWDM работают по принципу коммутации каналов — при этом каждая световая волна представляет собой отдельный **спектральный канал**.

Каждая волна несет свою собственную информацию, при этом оборудование DWDM не занимается непосредственно проблемами передачи данных

на каждой волне, т.е. способом кодирования информации и протоколом ее передачи. Устройства DWDM занимаются только объединением различных волн в одном световом пучке, а также выделением из общего сигнала информации каждого спектрального канала.

Принцип работы

Оборудование DWDM позволяет передавать по одному оптическому каналу 32 и более волн различной длины в окне прозрачности 150 нм. При этом каждая волна может переносить информацию со скоростью до 10 Гбит/с (при применении протоколов STM-64 или 10GE). Ведутся работы по повышению этой скорости до 40-80 Гбит/с. Мультиплексирование DWDM называется «плотным», т.к. в нем используется существенно меньшее (чем у предшествующей технологии WDM) расстояние между длинами волн.

В рекомендации G.692 определен частотный план с разнесением частот на 100 ГГц ($\Delta\lambda$ = 0,8 нм). Для передачи используется 41 волна (от 128 нм (191 ТГц) — до 160 нм (192 ТГц)). Определен также частотный план с разнесением на 50 ГГц ($\Delta\lambda$ = 0,4 нм), что позволяет передавать в этом диапазоне 81 волну. Имеются экспериментальные образцы с разнесением на 25 ГГц. Такая технология называется *High Dence WDM (HDWDM).*

Как видно на рисунке 1.18, необходимо обеспечить высокую точность частоты, чтобы не допустить перекрытия спектра каналов.

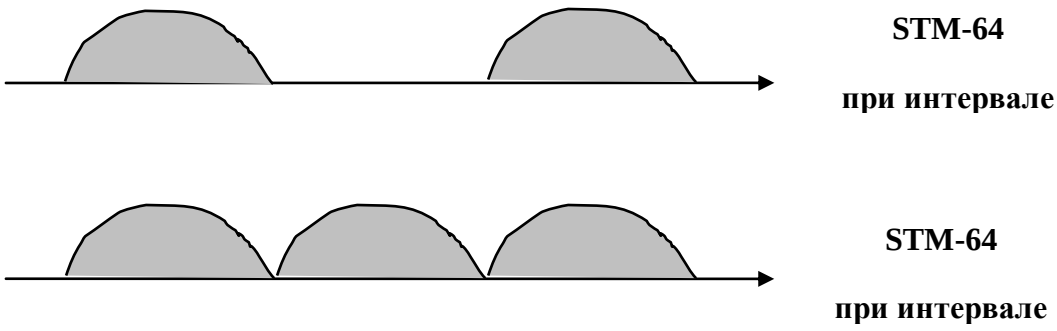


Рис. 1.18

Успех технологии DWDM связан с появлением волоконно-оптических усилителей на основе кварца, легированного эрбием. Эти устройства непосредственно усиливают световые сигналы в диапазоне 150 нм без промежуточного преобразования в электрический ток (как это делается в регенераторах сетей SDH). Протяженность участка между оптическими усилителями может достигать 10 км и более. Длина же мультиплексной секции достигает

3000 км (что требует установки промежуточных усилителей). Оптические усилители используются и на самих мультиплексорах.

Последние исследования позволили создать еще один класс оптических усилителей, работающих в L-диапазоне (4-е окно прозрачности) — от 170 до 1605 нм. Использование этого диапазона с расстоянием между волнами в 50 или 25 ГГц позволяет передавать одновременно 80 – 160 и более длин волн и тем самым повысить скорость трафика в одном направлении до 1,6 Тбит/с.
Развитием этой технологии являются также полностью оптические сети (All-Optical Networks). В них все операции по мультиплексированию / демультиплексированию, вводу-выводу и кросс-коммутации выполняются без преобразования сигнала в электрическую форму. Для этих сетей используется специальное сокращение «O-O-O» (в отличие от традиционных сетей «O-E-O». Пока создаются только фрагменты сетей на базе «O-O-O».

Преимущества технологии DWDM

Эта технология обладает следующими преимуществами.

- Повышение коэффициента использования частотного диапазона оптоволокна.
- Отличная масштабируемость – достаточно просто добавить новые спектральные каналы.
- Экономическая эффективность – не требуется электрическая регенерация на длинных маршрутах.
- Независимость от протокола передачи – через магистраль DWDM можно передавать трафик сетей любого типа.
- Независимость спектральных каналов друг от друга.
- Совместимость с технологией SDH. Мультиплексоры DWDM оснащаются интерфейсами STM-N, способными принимать и передавать данные мультиплексоров SDH.
- Совместимость с технологиями Ethernet – Gigabit Ethernet и 10GE.
- Стандартизация на уровне Международного союза по телекоммуникациям ITU-T.

Типовые топологии

1) ***Цепь «точка – точка».***

В такой структуре (рис. 1.19) транспондер (*transponder*) — это устройство, преобразующее электрический сигнал в сигнал с нужной длиной волны. Кроме дуплексного обмена с использованием двух оптических волокон можно

применять и одно оптоволокно — половина волн частотного диапазона передает информацию в одну сторону, а вторая половина – в обратную сторону.

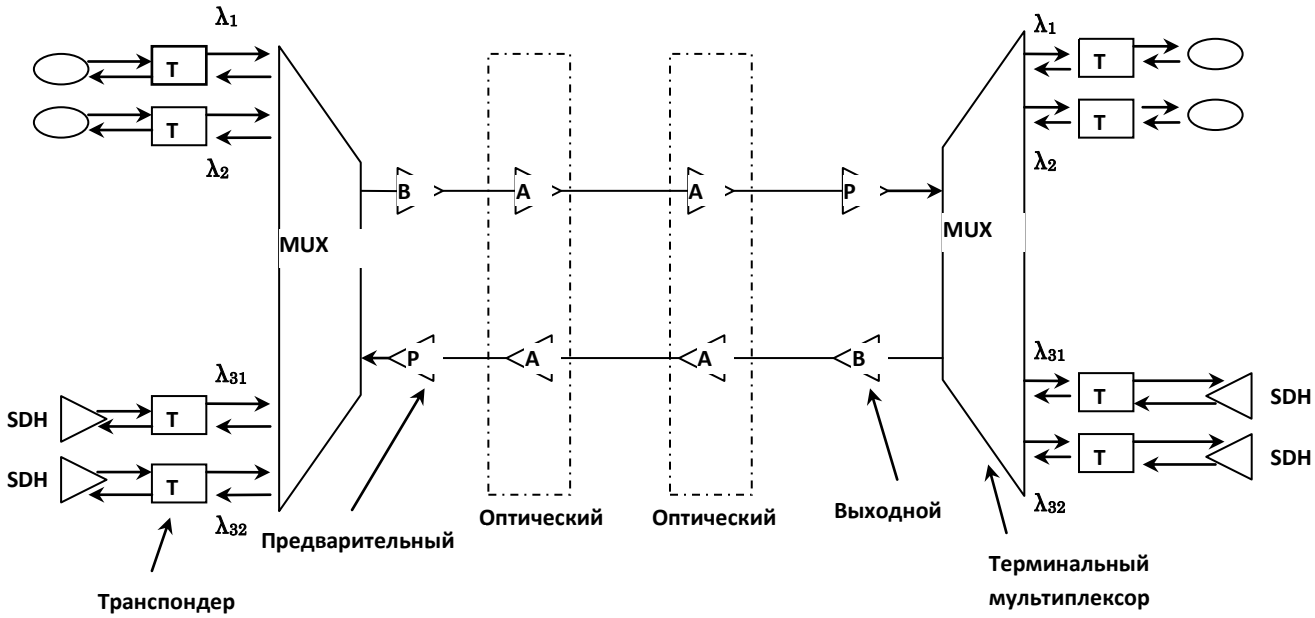


Рис. 1.19

2) Цепь с промежуточным подключением

Здесь промежуточные узлы выполняют функции мультиплексоров ввода-вывода (рис. 1.20). Используются оптические мультиплексоры ввода-вывода (*OADM*), которые могут вывести из оптического сигнала волну определенной длины и ввести туда сигнал с той же длиной волны. Мультиплексор OADM может выполнять операцию как оптическими средствами, так и с промежуточным электрическим преобразованием.

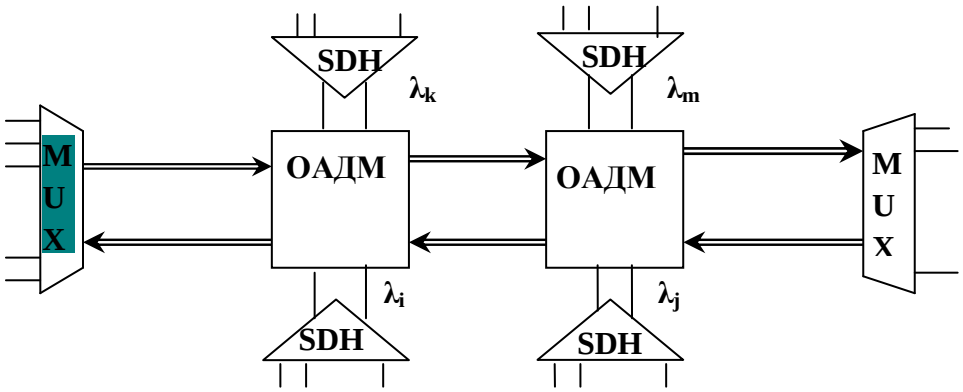


Рис. 1.20

3) Кольцо

При такой топологии (рис. 1.21) обеспечивается повышенная живучесть за счет резервных путей. Могут применяться те же методы защиты трафика, что и в сетях SDH (например, установление двух путей – основного и резервного).

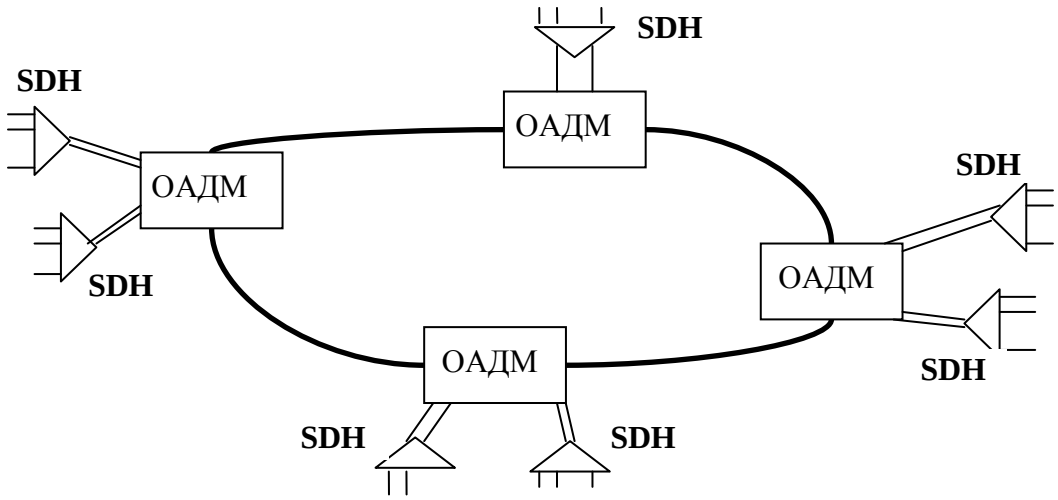


Рис. 1.21

4) Ячеистая топология

Такая топология (рис. 1.22) отличается большей гибкостью, производительностью и отказоустойчивостью. Для реализации необходимо наличие оптических кросс-коннекторов (ОХС), поддерживающих произвольную коммутацию. Оборудование этого типа пока еще только начинает выпускаться.

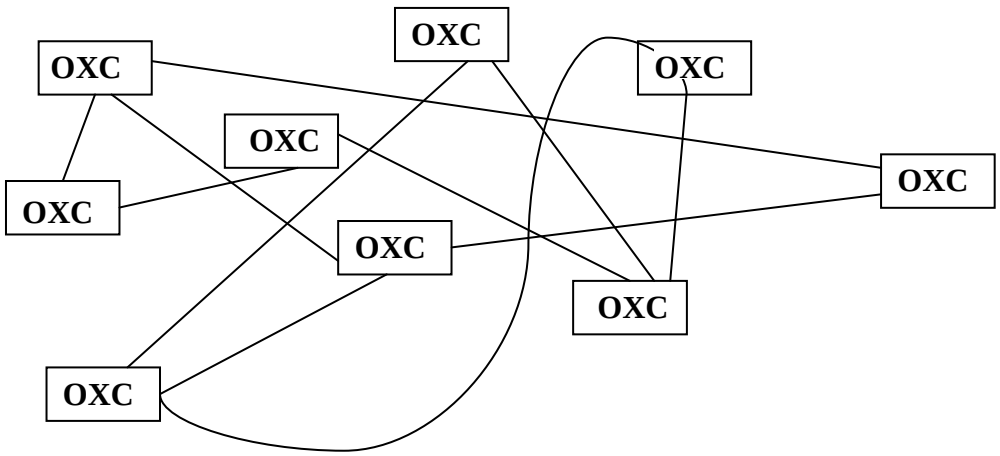


Рис. 1.22

Аппаратура

В сетях DWDM используются лазерные диоды с фиксированной длиной волны. Для оптических усилителей применяется волокно с добавлением ионов эмбия (для L-диапазона – ионы тулия). Имеется также лазер накачки, переводящий атомы эмбия в возбужденное состояние. Наибольшее применение нашло одномодовое оптоволокно со смещенной ненулевой дисперсией.

2. БЕСПРОВОДНЫЕ СЕТИ

Беспроводные локальные сети по стандарту IEEE 802.11 обеспечивают доступ к ИС, когда обычные кабельные решения оказываются непригодными. С повышением производительности и снижением стоимости беспроводные сети становятся все более популярными.

Беспроводная связь позволяет поддерживать два основных режима:

- одноранговый режим, дающий возможность конфигурировать беспроводную сетевую среду для равноправных клиентов;
- стационарный режим, обеспечивающий включение беспроводных локальных сетей в большую сетевую структуру в качестве составной части.

Различают полную мобильность пользователей, которая допускает перемещение от одной точки доступа к другой без потери связи с сервером, и частичную, когда обеспечивается только возможность подключения к различным устройствам доступа, расположенным по пути следования. Для объединения в сети отдельных зданий используется стационарное подключение типа “точка-точка” или точечное подключение к многоточечному радиомосту.

С помощью стандарта Radio Ethernet решается проблема "последней мили" (правда, в отдельных случаях эта "миля" может составлять от 100 м до 25 км). Сеть Radio Ethernet сейчас обеспечивает пропускную способность до 54 Мбит/с и позволяет создавать защищенные беспроводные каналы для передачи мультимедийной информации.

С точки зрения технологии передачи данных, системы беспроводной связи подразделяются на два типа ─ радиочастотные (РЧ) и инфракрасные (ИК).

2.1. Физический уровень (PHY)

Рассмотрим более подробно технические решения, которые заложены в конструкцию РЧ- и ИК-адаптеров. Стандарт IEEE 802.11 состоит из набора спецификаций, которые описывают физический уровень (PHY) и метод дос-

тупа к среде (MAC) на канальном уровне. В спецификации физического уровня определены два РЧ-стандарта и один ИК-стандарт. Оба РЧ-стандарта работают в одной и той же полосе частот 2400-2483,5 МГц, выделенной для промышленного, научного и медицинского использования Министерством связи РФ. РЧ-стандарты различаются методом передачи радиосигнала: с непосредственной модуляцией несущей частоты (Direct-Sequence Spread Spectrum, DSSS) и со скачкообразной перестройкой частоты (Frequency-Hopping Spread Spectrum, FHSS).

Непосредственная модуляция несущей частоты (DSSS)

Связь по методу DSSS более чувствительна к условиям эксплуатации беспроводной системы и внешним помехам. Однако аппаратура, использующая передачу с непосредственной модуляцией сигнала, в принципе позволяет получить более высокую пропускную способность. Типичная скорость передачи для устройств типа DSSS составляет до 8 Мбит/с, в то время как для аппаратуры на базе FHSS она не достигает и 2 Мбит/с.

Особенностью метода DSSS является то, что передача информации осуществляется на единственной заранее выделенной частоте, а защита от помех осуществляется за счет псевдошумового кодирования (PN): включения в поток данных избыточных битов, называемых маркерами, количество которых достигает 11 на каждый информационный бит (11-элементный PN-код Баркера). Основное назначение маркеров ─ коррекция сигнала при ошибках передачи, включая помехи от внешних источников. При таком методе необходим корреляционный приемник для расшифровки сигналов, настроенный на соответствующий расширенный код и удаляющий маркеры при преобразовании сигнала в исходную форму. На спектр передачи вокруг выбранного канала накладывается маска шириной 30 МГц. В качестве метода модуляции применяется фазовая манипуляция. При этом на частоте 902 МГц пропускная способность передатчика составляет до 2 Мбит/с, а для полосы 2,4 ГГц ─ не более 8 Мбит/с.

Главное преимущество DSSS заключается в том, что этот стандарт очень эффективен на заданном частотном спектре. Но у него есть и недостатки, например, схема модуляции требует применения линейных цепей передатчика и приемника, которые потребляют больше энергии.

Скачкообразная перестройка частоты (FHSS)

В устройствах ступенчатого изменения частоты расширение частотного спектра достигается перестройкой частоты передачи через определенный ин-

тервал времени. В США максимальное время работы на одной частоте составляет 400 мс, но в других странах оно другое. По интерфейсу FHSS этот интервал равен 20 мс. При этом новый частотный канал выбирается по случайному закону, который известен соответствующему приемному устройству. Это гарантирует устойчивый прием в условиях помех и конфиденциальность передачи. Частотные каналы расположены в пределах 500 кГц для диапазона радиочастот 902-928 МГц (используется 50 частотных каналов) и в пределах 1 МГц ─ для диапазона 2,4 - 2,484 ГГц (75 каналов). Помехи при такой модуляции сведены к минимуму.

FHSS использует модуляцию с двухуровневой гауссовой частотной манипуляцией (2GFSK) для скорости передачи данных 1 Мбит/с и четырехуровневой GFSK для 2 Мбит/с. Минимальное отклонение составляет 110 кГц. Мощность определяется местными правилами, но большинство систем работает на уровне 100 МВт.

Главный недостаток FHSS ─ необходимость расходовать время на переключение частот, вместо того чтобы передавать данные. При заданной скорости передачи пропускная способность FHSS будет ниже, чем у DSSS. Между этими методами существуют также различия в характере распространения сигналов, которые влияют на качество работы в разных средах.

Инфракрасный канал (ИК)

Метод передачи сигнала с помощью инфракрасного канала (ИК) или ИК-излучения известен давно и нашел широкое применение в различных областях науки и техники. Однако до недавнего времени никто не использовал эту технологию для создания компьютерных сетей. ИК-физический уровень отличается от других тем, что в его основе лежат не радиоволны, а инфракрасное излучение с длинами волн 850-950 нм. Известно, что препятствием для распространения сигнала ИК-излучения может стать любой предмет плотнее воздуха, находящийся на его пути. Однако этот недостаток, наряду с ограниченной дальностью помехоустойчивой связи, превращается в неоспоримое преимущество, когда речь идет о создании систем со специальными средствами защиты информации. Перехватить поток данных в такой сети практически невозможно. Кроме того, ИК-приемники не чувствительны к помехам на радиочастотах.

Недостаточная проникающая способность ИК-излучений накладывает дополнительные ограничения на конфигурацию сети: связь возможна только в пределах прямой видимости. В качестве адаптеров для приемопередачи ин-

формации в ИК-диапазоне применяются два типа устройств: линейные (на линии прямой видимости) и рассеивающие. Рассеянное излучение использует как прямые, так и отраженные лучи. В нем обеспечивается фазово-импульсная модуляция (PPM) методом блочного кодирования и две скорости передачи данных: 1 Мбит/с со схемой кодирования 16-PPM и 2 Мбит/с со схемой кодирования 4-PPM. В обоих вариантах для обнаружения ошибок используется код Грея.

За счет фокусировки линейные устройства передачи могут обеспечить дальность помехоустойчивой связи до 300 м. Однако для этого необходима точная пространственная юстировка положения приемопередающих пар. Устройства связи, использующие рассеянное ИК-излучение, не требуют точной юстировки по положению и обеспечивают мобильный режим работы сети, т.е. допускают свободное перемещение приемопередатчика в пределах помещения. Обычно они служат для связи клиентов беспроводной сети с обычной локальной сетью. Дальность связи при этом находится в пределах 10 м.

Главный недостаток ИК-технологии заключается в том, что для максимальной дальности действия он требует прямой видимости. Этот сдерживающий фактор может стать преимуществом при работе на ограниченном пространстве.

2.2. Особенности подуровня MAC

Чтобы преодолеть различия между кабельными и беспроводными локальными сетями, на подуровне MAC (Medium Access Control) обеспечиваются функции управления, специфические для радиосетей. Например, с этой целью определяется функция точки подсоединения, т.е. базовой станции, через которую осуществляется беспроводное подключение сети.

Отдельные станции распознают точку подсоединения по протоколу MAC и с системами, не нуждающимися в таких точках, когда устанавливается временное соединение между двумя узлами.

MAC использует метод доступа CSMA/CA ─ множественный доступ с контролем несущей, но вместо алгоритма обнаружения коллизий (используемого в стандарте IEEE 802.3) в нем используется алгоритм предотвращения коллизий. Применяя функцию DCF, схема MAC через интерфейс физического уровня контролирует состояние беспроводной среды и начинает передачу только при отсутствии активности других передатчиков. Существует два вопроса в реализации этого метода.

Первый вопрос решается при одноадресной передаче (схема подобной сети приведена на рис. 2.1) пакетов. При этом требуется подтверждение приема для каждого передаваемого фрейма. Поскольку из-за проблемы “невидимого” узла (на этой же частоте может работать какое-либо другое устройство) стандартный алгоритм обнаружения несущей утрачивает надежность, то для резервирования полосы пропускания с целью устранения возможного конфликта используются управляющие фреймы RTS/CTS (запрос на передачу и разрешение на передачу), хотя это и влечет дополнительные накладные расходы, которые несколько снижают производительность (до 20%).

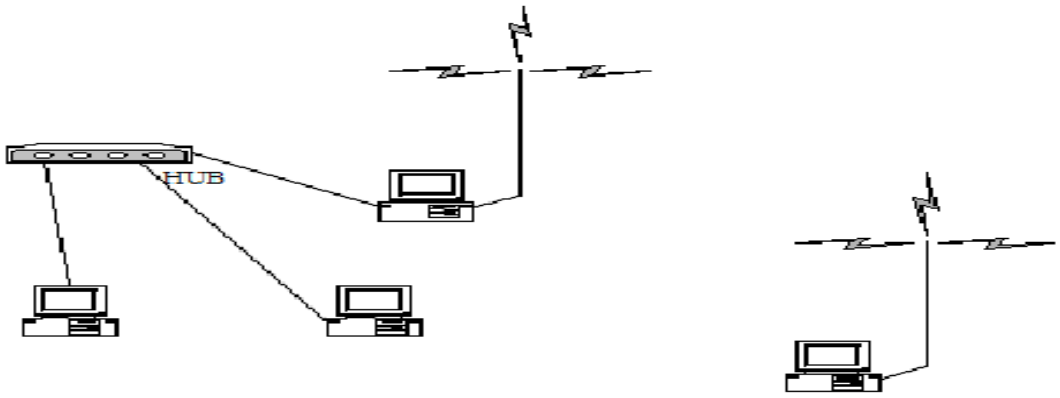


Рис. 2.1

Второй вопрос связан с разработкой спецификации MAC канального уровня, которая бы координировала работу множества логических сетей в одном и том же радиочастотном канале (схема подобной сети приведена на рис.2.2). С ним тесно связан второй вопрос, касающийся разработки механизмов, позволяющих устройствам автоматически менять узел доступа при перемещении между ними. При этом возникает проблема “невидимых” узлов, когда два беспроводных устройства способны передавать и принимать сигналы из общего узла доступа, но не имеют прямой связи друг с другом. Функция Point Coordination Function (PCF) используется в качестве факультативного элемента данного стандарта и реализует метод, исключающий конкуренцию за доступ к среде. По определенному этой функцией сценарию одна из станций работает как координатор, управляющий доступом к среде передачи данных на основе некоторой схемы опроса и исходя из приоритетов рабочих станций. Станция-координатор управляет доступом к среде всех обслуживаемых станций. Соединение выбранной для связи станции происходит по функции DCF.

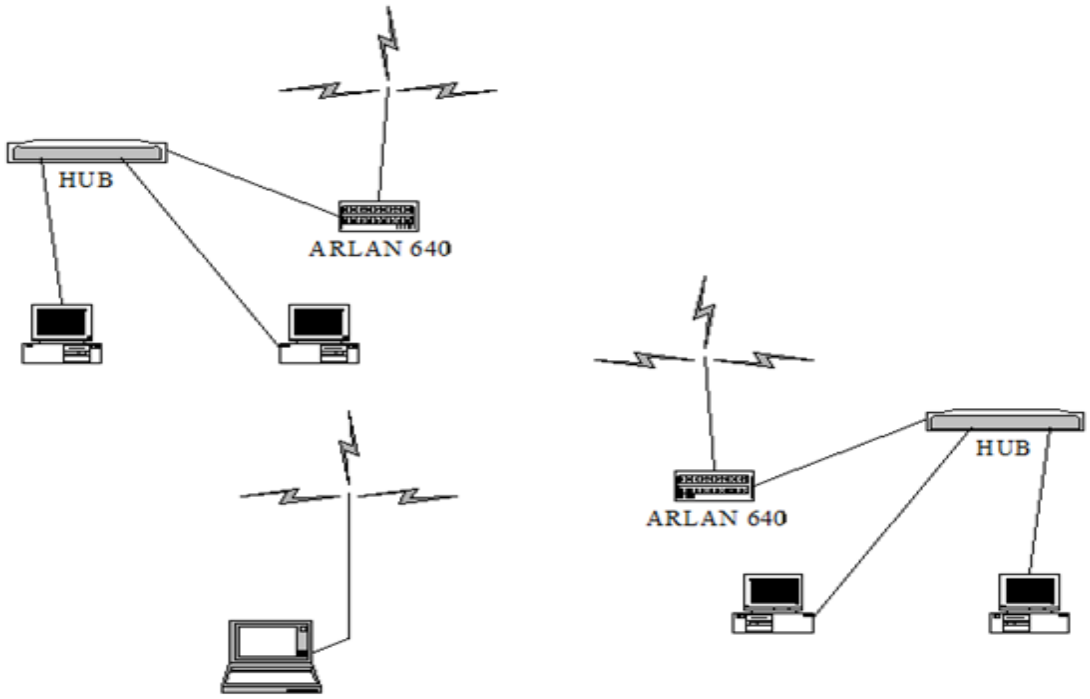


Рис. 2.2

Таким образом, функции DCF и PCF отличаются лишь схемой организации доступа к среде передачи.

Помимо основного доступа к среде подуровень MAC канального уровня стандарта IEEE 802.11 обеспечивает также защиту данных, систему поддержки связи при перемещениях рабочих станций в пространстве и управление питанием в блокнотных компьютерах. Для защиты данных используется метод шифрования, основанный на алгоритме RSA RC4 PRNG. Могут быть обеспечены пять уровней защиты:

- Уровень защиты 1. Технология шумоподобного сигнала.
- Уровень защиты 2. Идентификатор шумового кода для каждого устройства.
- Уровень защиты 3. DES алгоритм или скремблер.
- Уровень защиты 4. Сетевой пароль.
- Уровень защиты 5. Лист ограничения доступа к узлу сети.

Стандарт 802.11 предусматривает защиту информации в беспроводной сети с помощью мер, которые обеспечивают безопасность "на уровне, характерном для проводных сетей" (Wired Equivalent Privacy, WEP). Эти меры включают механизмы и процедуры аутентификации и шифрования.

Особенности стандарта IEEE 802.11

Наиболее "широкополосный" стандарт из семейства Radio Ethernet ─ это 802.11a, последняя редакция которого была утверждена в 1999 году. Предельная для него скорость передачи данных ─ 54 Мбит/с (в спецификациях определены три обязательные скорости ─ 6, 12 и 24 Мбит/с, а также пять необязательных ─ 9, 18, 36, 48 и 54 Мбит/с). Данный стандарт предусматривает работу в диапазоне 5 ГГц, в качестве метода модуляции используется ортогональное частотное мультиплексирование (OFDM), позволяющее снизить до минимума межсимвольные искажения в радиоканале. Применение OFDM дает возможность передавать полезные сигналы параллельно на нескольких частотах диапазона, что существенно повышает пропускную способность канала. Для оборудования, использующего диапазон частот 5 ГГц, характерна более высокая (чем для области 2,4 ГГц) потребляемая мощность передатчиков и меньший радиус действия ─ около 100 м.

Стандарт 802.11b (его окончательная редакция, принятая в 1999 году, известна как Wi-Fi - Wireless Fidelity) ориентирован на радиочастотный диапазон 2,4 ГГц и по пропускной способности ─ до 11 Мбит/с ─ практически соответствует проводному Ethernet-каналу. Базовая радиотехнология Wi-Fi ─ прямое расширение сигнала с помощью восьмиразрядных последовательностей Уолша. Стандарт предусматривает автоматическое понижение скорости передачи информации при ухудшении качества сигнала. Четкие механизмы роуминга в нем также не определены.

Стандарт 802.11d в настоящее время находится в стадии разработки. Тем не менее, уже сейчас можно сказать, что его спецификации будут включать универсальные процедуры физического уровня (формирования каналов, генерации псевдослучайных последовательностей частот, использования дополнительных параметров для информационной базы управления).

Предварительный вариант стандарта 802.11e, утвержденного в конце 2001 года, обеспечивает полную совместимость устройств, базирующихся на данном стандарте и стандартах 802.11 a и b. Спецификации 802.11e создавались для поддержки мультисервисных сетей, а, следовательно, в них заложена более высокая (чем в версиях 802.11a и b) функциональность, обеспечивающая потоковую передачу мультимедиа-данных с гарантированным качеством обслуживания (QoS).

Помимо перечисленных стандартов серии 802.11 рабочая группа IEEE по Radio Ethernet разрабатывает также отдельные спецификации, дополняю-

щие некоторые из стандартов серии 802.11 или используемые в данной технологии общие механизмы передачи сигнала.

Так, спецификации 802.11g расширяют возможности стандарта 802.11b путем применения более эффективного по сравнению с DSSS метода модуляции OFDM. Предполагается, что они будут гарантировать обратную совместимость с оборудованием, построенным на базе 802.11b.

В задачи рабочей группы, занимающейся спецификациями 802.11h, входит дополнение уровней MAC и PHY алгоритмами оптимального выбора частот, разработка средств управления использованием спектра и контроля излучаемой мощности. Предполагается, что их решение будет базироваться на протоколах DFS (Dynamic Frequency Selection) и TCP (Transmit Power Control), созданных ETSI. Процедура динамической регулировки мощности для 802.11h предполагает ее изменение в зависимости от уровня помех с последующим переходом на другой радиоканал в том случае, если повышением мощности не удается обеспечить требуемое отношение сигнал/шум.

Отдельную группу составляют спецификации, относящиеся к информационной безопасности. Сегодня проблемами защиты данных в беспроводных сетях стандарта 802.11 занимается самостоятельное подразделение 802.11i совместно с комитетом IEEE 802.1. Новый стандарт получил номер 802.1X и будет применим не только в беспроводных, но и в проводных сетях.

Повторители, концентраторы (хабы)

Повторитель ─ неинтеллектуальный концентратор и служит только для усиления сигнала и передачи всех поступающих к нему фреймов всем подключенным к нему узлам сети, т.е. все пользователи получают одинаковый трафик и, таким образом, разделяют пропускную способность сети. Проводя аналогию с сетью, построенной на коаксиальном кабеле, можно сказать, что в данном случае все пользователи находятся в одном сегменте сети.

Концентратор (HUB) ─ интеллектуальный повторитель, который выполняет кроме этой функции, следующие:

- объединяет сегменты одного протокола MAC-уровня, но использующие различные типы физической среды, например 10Base5, 10Base2, 10BaseT и 10BaseF (на оптоволокне) сетей Ethernet;
- осуществляет автосегментацию, то есть автоматически отключает порт при повреждении кабеля данного сегмента;

- является управляемым по протоколам SNMP (Simple Network Manager Protocol) или IEEE 802.1, то есть обеспечивает удаленный контроль состояния портов, сбор статистики, включение/отключение портов;
- осуществляет паролирование портов или шифрацию проходящей информации на аппаратном уровне;
- поддерживает резервные связи между повторителями.

Мосты

Мост (MAC-dridge, bridge) ─ это устройство, которое обеспечивает взаимосвязь двух (реже нескольких) локальных сетей посредством трансляции кадров MAC-уровня (т.е. кадров Ethernet или кадров Token Ring и т.д.) из одной ИС (которую иногда называют в этом случае сегментом) в другую сеть того же канального протокола.

Каждому из рассмотренных ранее типов локальных сетей присущи ограничения на длину и конфигурацию физической среды, пропускную способность сети, число подключаемых станций. Повторители снимают некоторые ограничения (по длине, конфигурации и, возможно, по совмещению различных сред передачи), но остаются ограничения на количество повторителей в сети (не больше 4-х) и на пропускную способность ИС, так как при использовании повторителей среда передачи остается единой и пакет от любой станции захватывает среду монопольно. Тем самым при увеличении числа станций общая среда передачи данных, в которой складываются трафики всех станций, становится узким местом. Мост, в отличие от повторителя, не повторяет кадр, появившийся в одном сегменте сети, на среде другого сегмента, если адрес получателя фрейма принадлежит станции, подключенной к тому же сегменту, что и станция-отправитель. Если же адрес получателя принадлежит станции другого сегмента, то мост транслирует кадр на среду другого сегмента в соответствии с правилами доступа. Счетчик повторителей при этом сбрасывается в ноль, поэтому каждый из сегментов может включать до 4-х повторителей.

По своему принципу действия мосты подразделяются на два типа. Мосты первого типа выполняют так называемую маршрутизацию от источника (Source Routing) ─ метод, разработанный фирмой IBM для своих сетей. Этот метод требует, чтобы узел-отправитель фрейма размещал в нем информацию о пути его маршрутизации сети Token Ring). Другими словами, каждая станция должна выполнять функции по маршрутизации фреймов. Второй тип мостов осуществляет прозрачную передачу фреймов (Transparent Bridges). Это основной тип мостов, и далее мы будем говорить только о мостах этого типа.

Так как для уровня MAC и выше расположенных уровней мост прозрачен, то он изолирует трафик одного сегмента от трафика другого сегмента, фильтруя фреймы. Мост не только снижает нагрузку в объединенной сети, но и уменьшает возможности несанкционированного доступа; так фреймы, предназначенные для циркуляции внутри одного сегмента, физически не появляются на других, что исключает их "прослушивание" станциями других сегментов.

Для мостов доступна информация только о MAC-адресах сетевых устройств, которая и является основой для принятия решения по управлению сетевыми фреймами. К верхним уровням сетевой адресной информации мосты не имеют доступа (работают только на канальном уровне MAC) и поэтому могут принимать весьма ограниченные решения относительно выбора пути следования фрейма через сеть.

По принципу передачи фреймов мосты подразделяются на инкапсулирующие (encapsulating) и транслирующие (translational). Инкапсулирующие мосты упаковывают при передаче пакеты канального уровня одной сети в пакеты канального уровня другой сети. После прохождения пакета по второй сети аналогичный мост удаляет оболочку промежуточного протокола, и пакет продолжает свое движение в исходном виде. Очевидно, что при таком методе взаимодействие со станциями второй сети невозможно, эта сеть используется только как промежуточное транспортное средство.

Транслирующие мосты выполняют преобразование из одного протокола канального уровня в другой. Такой мост имеет преимущества ─ меньше накладные расходы, так как не нужно передавать два заголовка канального уровня. Станции другой сети становятся доступными.

Современные мосты также могут поддерживать ряд дополнительных возможностей, которые относятся к так называемым интеллектуальным свойствам.

В первую очередь, это относится к возможностям управления мостами по протоколу SNMP (более подробно он будет рассматриваться позже). Это позволяет:

- Конфигурировать порты мостов.
- Производить сбор статистики и анализ трафика. Например, для каждой подключенной к сети станции можно получить информацию о том, когда она последний раз посылала фреймы в сеть, о числе фреймов и байт, переданных в сеть, о числе фреймов и байт, переданных за пределы сети, число переданных широковещательных фреймов и т.п.

- Устанавливать дополнительные фильтры на порты концентратора по физическим адресам сетевых устройств с целью усиления защиты от несанкционированного доступа или повышения эффективности работы отдельных сегментов.
- Оперативно получать сообщения обо всех возникающих проблемах в сети и локализовать их.
- Проводить диагностику модулей концентраторов.
- Просматривать в графическом формате изображения панелей модулей, установленных в удаленных мостах, включая текущее состояние индикаторов.

Таким образом, мост является относительно несложным (обычно двухпортовым) устройством и может обеспечивать простой в эксплуатации и относительно недорогой способ межсетевого взаимодействия. Функции моста могут выполнять некоторые управляющие модули интеллектуальных концентраторов. Некоторые мосты по протоколам сетевого уровня могут выполнять и функции маршрутизатора.

Маршрутизация на уровне MAC для мостов

Еще одна важная характеристика моста ─ поддержка алгоритма "покрывающего дерева" ─ Spanning Tree Aigoritm (STA), определенного стандартом IEEE 802.1D. Этот алгоритм позволяет мостам поддерживать в сети резервные пути. Так как мост работает на MAC-уровне и сам не имеет MAC-адреса, то в объединенной сети в каждый момент времени между двумя любыми сетями должен существовать только один маршрут. Иначе могут появиться фреймы-дубли, которые не могут правильно отрабатываться логикой. Мосты, поддерживающие алгоритм STA, позволяют иметь в сети петли, но наряду с физической конфигурацией объединенной сети существует активная конфигурация, динамически изменяющаяся и обеспечивающая единственный маршрут между двумя любыми узлами локальной сети. Перед началом работы и после реконфигурации объединенной сети происходит формирование новой активной конфигурации ─ таблицы фильтрации. Входы таблицы фильтрации могут быть статическими (создаются диспетчером моста) и динамическими, которые создаются в процессе самообучения. В процессе самообучения мост анализирует адрес отправителя MAC-фреймов, принимаемых портом, и при определенном состоянии порта обновляет динамические входы таблицы. Такой процесс называется "обучением" моста. Состояние порта отражает состояние его технических средств (работоспособны или нет) и его логическое состоя-

ние (включен/выключен, доступ по паролю). Таблица фильтрации должна содержать всю необходимую информацию для правильной работы транслятора фреймов. Эта таблица содержит совокупность входов и отражает маршрутную информацию для уникальных MAC-адресов станций объединенной ЛВС.

Главными параметрами моста являются:

- размер внутренней адресной таблицы;
- скорость фильтрации (filtering);
- скорость маршрутизации (forwarding).

Максимальная емкость адресной таблицы определяет максимальное количество MAC-адресов, с которыми может оперировать мост. Для пользователя это выражается максимальным количеством MAC-адресов, которые можно подключить к одному порту моста. Типичные значения лежат в пределах от 500 до 8000. При появлении новых адресов при полностью заполненной таблице мост вытесняет старые адреса и помещает в нее новые. Поэтому это обстоятельство только замедлит работу моста, но не вызовет его отказа.

Скорость фильтрации и маршрутизации пакетов характеризует производительность моста. Если она ниже максимально возможной скорости передачи фреймов для конкретного протокола, то мост может являться причиной задержек и снижения производительности. Если выше ─ значит стоимость данного моста скорее всего выше минимально возможной. Например, для протокола Ethernet максимальная производительность составляет 14880 фреймов в секунду (для фреймов 64 байта + 8 байтов преамбулы).

Коммутаторы

Принцип работы коммутатора (switch) во многом схож с принципом работы моста. Однако коммутатор является многопортовым устройством, и за счет использования внутренней высокоскоростной шины и внутренних буферов каждый подключенный через коммутатор сетевой узел (или группа узлов) может использовать полную полосу пропускания сети, а не разделять ее с устройствами, подключенными к другим портам. Существуют коммутаторы, осуществляющие коммутацию между различными сетевыми технологиями, например 10 Мбит/с Ethernet и 100 Мбит/с FDDI или Fast Ethernet. Это позволяет подключить серверы, являющиеся обычно центром сетевого трафика, по высокоскоростной технологии и тем самым ликвидировать узкое место в сети. Многие современные коммутаторы обеспечивают дополнительные возможности, включая фильтрацию фреймов, выбор между Bridging или Switching ре-

жимами работы, а также и некоторые функции маршрутизации (работает на канальном уровне MAC).

Switch - концентраторы

В настоящее время различные фирмы называют свои концентраторы коммутирующими концентраторами (Switching hubs), вкладывая в это понятие различный смысл. Фирма LANNET для сетей Ethernet предлагает различать 4 случая использования этого термина:

- динамические переключатели портов (Dynamic port switches);
- динамические переключатели сегментов (Dynamic segment switches);
- статические переключатели портов (Static port switches);
- статические переключатели модулей (Static module switches).

Статические переключатели были разработаны для решения проблемы реконфигурации сегментов сети. Они позволяют быстро переключить какую-либо станцию с одного сегмента сети Ethernet на другой. Статические переключатели используются в модульных концентраторах, имеющих несколько внутренних шин Ethernet. Между входными портами концентратора и внутренними шинами такого концентратора работает статический переключатель. Каждый порт может быть статически связан с одной из внутренних шин, которые разделяются между всеми приписанными к ним портами на основании стандартного для сетей Ethernet алгоритма случайного доступа. Переключение станции с сегмента на сегмент осуществляется администратором сети с помощью соответствующего программного обеспечения. Статический switch не увеличивает пропускную способность сети, а только позволяет ее удобно реконфигурировать.

При статическом переключении портов каждый порт может быть назначен любой из внутренних шин. Статический переключатель модулей позволяет приписывать какой-либо внутренней шине целиком весь многопортовый модуль.

Динамические переключатели были разработаны для повышения пропускной способности сети Ethernet. Динамический переключатель работает подобно телефонному коммутатору, который позволяет одновременно соединять нескольких абонентов и обеспечивать несколько параллельных соединений типа “точка-точка”. Способность к установлению параллельных связей является важнейшим преимуществом динамических переключателей. Например, динамический переключатель на 8 портов позволяет одновременно иметь 4

связи между портами, каждая из которых производится на стандартной для Ethernet скорости 10 Мбит/с. Таким образом, внутренняя производительность такого коммутатора будет составлять 40 Мбит/с. Динамические переключатели могут быть двух типов ─ с коммутацией "на лету" и с буферизацией фреймов.

Коммутатор первого типа осуществляет коммутацию фреймов с использованием конвейерного метода обработки фрейма. При поступлении начальных байтов фрейма на какой-либо порт коммутатор просматривает первые шесть байтов (после 8 байтов преамбулы), в которых содержится MAC-адрес назначения фрейма. По этому адресу определяется выходной порт фрейма, и если он не занят, то образуется динамический канал связи между входным и выходным портами коммутатора, имеющий пропускную способность 10 Мбит/с. Таким образом, первые байты фрейма начинают поступать на выходной порт коммутатора почти сразу же после поступления начальных байтов фрейма на его входной порт.

В наиболее быстродействующих коммутаторах задержка передачи фрейма составляет около 40 мкс. В том случае, когда порт назначения занят, коммутатор буферизует фрейм.

Коммутатор второго типа запоминает фрейм полностью, а потом обрабатывает его как мост. Но за счет специальной внутренней организации, обычно состоящей в наличии для каждого порта процессора обработки фреймов, передача фреймов между портами производится одновременно. Буферизующие коммутаторы работают несколько медленнее, чем конвейерные, но зато они исключают распространение поврежденных фреймов за счет их полного контроля, в то время как конвейерные не могут обеспечить эту нужную функцию.

Для определения выходного порта коммутатор строит таблицу фильтрации аналогично мосту. Для пользователя функции switch-концентратора идентичны функциям быстродействующего многопортового моста.

Динамическое переключение портов и сегментов определяется количественными возможностями switch-концентратора по операциям с MAC-адресами по каждому входу. Если к каждому входному порту допускается подключение только одного MAC-адреса, то это динамическая коммутация портов. Если же порт допускает подключение к нему некоторого количества MAC-адресов, то говорят о динамической коммутации сегментов. Концентратор с динамической коммутацией портов можно использовать только на нижнем иерархическом уровне концентраторов сети, например, для соединения ряда станций с несколькими серверами.

Концентратор с динамической коммутацией сегментов можно использовать как быстродействующий "позвоночник" (backbone) сети предприятия, объединяющий отдельные сегменты сети за счет своей внутренней высокой пропускной способности.

Кроме своего основного назначения ─ повышения пропускной способности связей в сети ─ динамический коммутатор позволяет локализовывать потоки информации в сети, а также контролировать и управлять этими потоками, образуя виртуальные сегменты. Так как динамический коммутатор передает фрейм только между источником и адресатом, то он, как и обычный мост, локализует трафик, что уменьшает нагрузку в сети и повышает конфиденциальность информации.

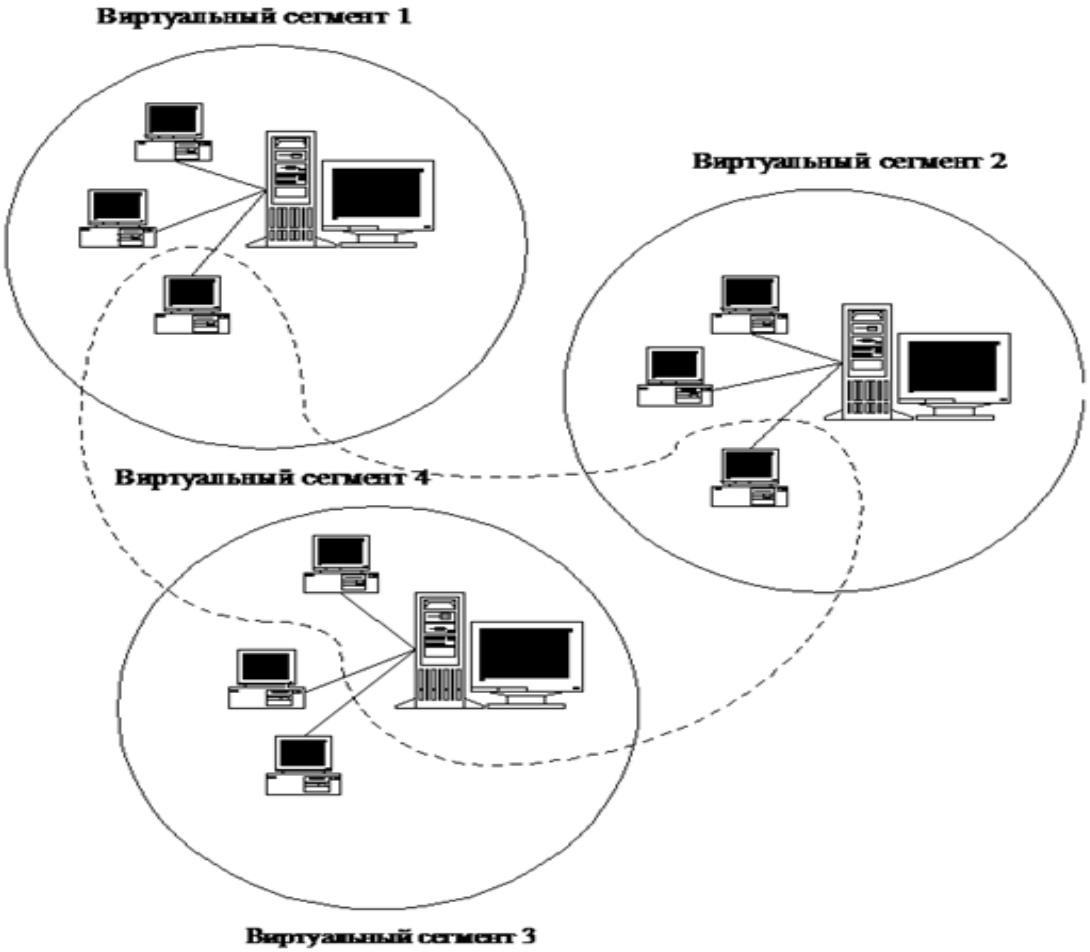


Рис. 2.3

Кроме того, динамический коммутатор позволяет контролировать и разрешать передачу фреймов только для определенных сочетаний адресов источника и приемника. Разрешенные пары адресов образуют виртуальный сегмент сети, доступ внутрь которого остальным адресам запрещен. Таким образом,

легко реализуются различные права доступа пользователей сети на уровне MAC-адресов их станций. Каждая станция может входить в несколько виртуальных сегментов, соответствующих связям работников предприятия между собой (рис. 2.3).

Как видно из описания, динамический switch выполняет все функции статического, но при этом обеспечивает повышение пропускной способности сети, что не может обеспечить статическая коммутация. Динамический коммутатор может быть реализован различными способами. Фирма LANNET делит динамические коммутаторы, с точки зрения их реализации, на четыре класса.

Автономный аппаратный концентратор, основанный на "матричном" или "перекрестном" коммутаторе, рассчитан на коммутацию строго определенного количества портов (обычно 4, 6 или 8) с помощью быстродействующей аппаратной матрицы, которая часто выполняется как "заказная" БИС (ASIC - App11ca11оп Specific Integrated Circuit). Такой коммутатор имеет обычно очень высокую скорость коммутации и невысокую стоимость, но его возможности по количеству коммутируемых портов ограничены; он реализует только один MAC-протокол (то есть Ethernet), и его трудно кооперировать с другими протоколами (то есть организовывать связи Ethernet - Token Ring, Ethernet - FDDI и т. п.), такой коммутатор имеет либо ограниченные возможности по управлению (организация виртуальных сегментов), либо эти возможности отсутствуют. Типичными представителями этого класса являются коммутаторы фирмы Kalpana Inc (типа dunamic switch). Технология фирмы Kalpana используется и в ряде коммутаторов других фирм, например, в коммутаторах HP LAN Switch фирмы Hewlett-Packard, ESE фирмы SynOptics.

Распределенный коммутатор с программной реализацией функций. Эти модульные коммутаторы построены на базе распределенной архитектуры "мост- маршрутизатор", используя процессор общего назначения на каждом модуле. Они обеспечивают различную скорость передачи данных между модулями на шасси. Благодаря своим маршрутным возможностям они могут обслуживать различные протоколы и проверять ошибки фреймов во время передачи.

Преимуществами таких коммутаторов являются: модульность, расширяемость, распределенная архитектура, возможность работы со многими протоколами и обнаружения ошибочных пакетов, отсутствие уязвимых мест благодаря распределенной архитектуре. Недостатком являются ограниченная скорость межмодульного взаимодействия и низкая скорость коммутации из-за

программной реализации функций. Представитель этого класса устройств ─ модуль Dragon-Switch фирмы Ungermann-Bass.

Централизованный коммутатор с программной реализацией функций использует центральный процессор для осуществления переключений, обрабатывая фреймы как маршрутизатор. Обычно он использует для связи между модулями общую технологию, такую как FDDI. Благодаря своим возможностям маршрутизации концентраторы могут работать с различными протоколами и обнаруживать ошибочные фреймы. Недостатки те же, что и у устройств предыдущего класса, плюс снижение надежности и пропускной способности из-за наличия центрального процессора.

Распределенный аппаратный коммутатор с высокоскоростной внутренней шиной. Эти концентраторы базируются на модулях, использующих аппаратные специализированные процессоры (реализуемые на "заказных" БИС), которые динамически коммутируют порты и используют общую высокоскоростную протокольно-независимую внутреннюю шину для передачи фреймов между модулями. Высокоскоростная шина является "общим знаменателем" различных протоколов. Преимуществом такого подхода является модульность, распределенная конструкция, обеспечивающая отказоустойчивость и расширяемость, высокая скорость коммутации из-за аппаратной ее реализации, легкость интеграции различных протоколов (Ethernet - Token Ring, Ethernet -FDDI, Ethernet - ATM и т. п.). Представителями этого класса являются концентраторы LET3б и LET10 фирмы LANNET с модулями LSE208 (switching Ethernet), LSE808 (switching Ethernet) и LSF100 (switching FDDI).

Большинство крупных фирм-производителей сетевого оборудования предлагает многомодульные коммутаторы в качестве "коммутационного центра" корпоративной ИС. Такие коммутаторы отражают тенденцию перехода от полностью распределенных локальных сетей на коаксиальном кабеле к централизованным коммуникационным решениям. Модульные корпоративные коммутаторы могут включать несколько десятков модулей различного назначения (повторителей, мостов и маршрутизаторов), объединенных в одном устройстве с модулями агентами протокола SNMP, и позволяют реализовать централизованное управление и обслуживание, что очень удобно в ИС большого размера.

Модульный коммутатор масштаба предприятия обычно обладает внутренней шиной или набором шин очень высокой производительности ─ до нескольких десятков гигабит в секунду, что позволяет реализовать одновременные соединения между модулями с высокой скоростью, гораздо большей, чем

скорость передачи данных по кабелю локальной сети в случае распределения модулей по отдельным устройствам ─ повторителям, мостам и маршрутизаторам. Такой коммутатор обычно поддерживает технологию коммутации (switching), причем коммутируются не только сегменты Ethernet, но и кольца Token Ring и FDDI.

За счет большого выбора модулей разного типа можно подобрать необходимую для конкретного предприятия конфигурацию такого коммутатора, хотя обычно модули концентратора уступают в функциональном отношении отдельным устройствам аналогичного назначения, то есть отдельные маршрутизаторы обладают, как правило, более полным набором поддерживаемых протоколов, по сравнению с модулями маршрутизаторами.

Конкурентная борьба между фирмами-производителями "чистых" маршрутизаторов или "чистых" мостов и фирмами, специализирующимися на модульных коммутаторах, приводит к положительным для потребителей результатам ─ их характеристики постоянно улучшаются, постепенно сближаясь.

Ввиду того, что отказ корпоративного модульного коммутатора приводит к очень тяжелым последствиям, в их конструкцию вносится большое количество средств обеспечения отказоустойчивости ─ дублированных источников питания, источников бесперебойного питания, обеспечиваются режимы "горячей" замены модулей.

2.3. Технология VLAN (IEEE 802.1Q)

В последнее время к основным технологиям Fast Ethernet добавляются все новые и новые. Одной из таких технологий является VLAN (Virtual Local-Area Network). VLAN представляет собой набор пользователей или портов коммутатора, сгруппированных вместе в безопасный автономный широковещательный домен. Основной целью создания VLAN в сети является возможность значительно оптимизировать работу локальной сети за счет разгрузки отдельных ее сегментов от "лишнего" трафика и решить некоторые вопросы безопасности в сети, разграничив доступ пользователей. Кроме того, VLAN позволяет контролировать и эффективно подавлять широковещательные штормы, которые в больших сетях иногда останавливают работу целых сегментов. Как правило, в такой ситуации обычный коммутатор не может исправить положение.

Еще одним достоинством VLAN является возможность объединения нескольких портов в один единый канал ─ транк, скорость работы которого уве-

личивается пропорционально количеству объединенных портов. Для примера, максимальная скорость работы транка из 4 портов составит 800 Мбит/с. Транковое соединение позволяет преодолеть эффект "бутылочного горла" при подключении к серверу или при организации магистрали между коммутаторами.

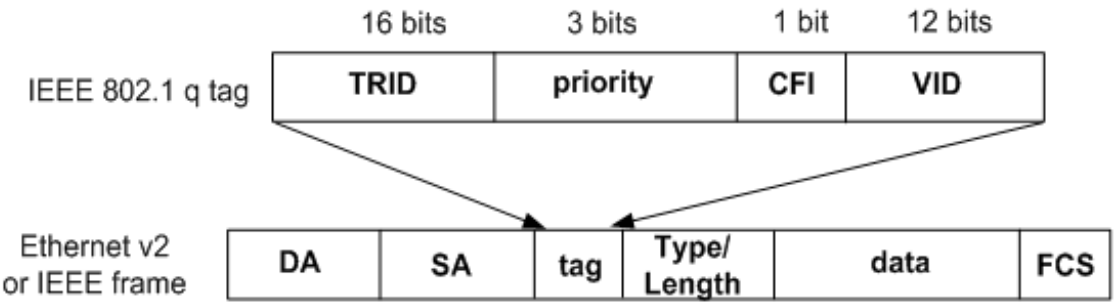
Конфигурирование коммутаторов с функцией VLAN предельно просто и осуществляется при помощи кнопок на передней панели устройства. При этом конфигурация записывается в энергонезависимую память и не "стирается" при выключении питания. Сети VLAN имеют следующие преимущества:

- функциональные рабочие группы;
- контроль за широковещательным трафиком ИС;
- повышенная безопасность ИС.

С помощью технологии VLAN можно создавать рабочие группы, основываясь на функциональности, а не на физическом расположении сегментов. Она позволяет администратору логически создавать, группировать и перегруппировывать сетевые сегменты без изменения физической инфраструктуры и отсоединения пользователей и серверов. VLAN обеспечивает дополнительные преимущества для безопасности. Пользователи одной рабочей группы не могут получить доступ к данным другой группы, потому что каждая VLAN ─ это закрытая и логически определенная группа.

Дополнительное преимущество VLAN 802.1Q на базе портов в том, что можно изменять топологию сети без физического перемещения станций или изменения кабельных соединений. Станции можно "перемещать" в другую VLAN и, соответственно, общаться и совместно использовать ресурсы с членами в новой VLAN, просто изменив настройки порта с одной VLAN (например, VLAN отдела продаж) на другую (VLAN отдела маркетинга). Таким образом, VLAN обеспечивают гибкость при перемещениях, изменениях и наращивании ИС.

Способность VLAN 802.1Q по извлечению меток из заголовков пакетов позволяет VLAN работать с существующими коммутаторами и сетевыми адаптерами, которые не распознают метки. Способность добавления меток позволяет VLAN распространяться через множество 802.1Q-совместимых коммутаторов по одному физическому соединению и позволяет активизировать алгоритм покрывающего дерева (Spanning Tree) на всех портах и работать в обычном режиме. Структура фрейма Enternet/IEEE802.3 с использованием VLAN (IEEE802.1 q) приведена на рис. 2.4.



IEEE 802.1 q поля	Описание
TRID	16- bit Tag Protocol Identifiel, which indicates that an 802.1 q tag follows.
Priority	3-bit IEEE 802.1 q priority, which provides 8 levels of prioritization
CFI	Canonical Format Indicator, which indicates whether the MAC addresses are in canonical (0) or noncanonical (1) format.
VID	12-bit VLAN identifier that allows 4096 unique VLAN values: VLAN numbers 0.1, and 4095 are reserved.

Рис. 2.4

Транкинг портов используется для объединения нескольких портов вместе для образования высокоскоростного канала передачи данных. Включенные в транк порты называются членами группы. Один из портов в группе выступает в качестве "связывающего". Поскольку все члены группы в транке должны быть настроены для работы в одинаковом режиме, все изменения настроек, произведенные по отношению к "связывающему" порту, относятся ко всем членам транковой группы. Таким образом, для настройки портов в группе необходимо только настроить "связывающий" порт.

Коммутаторы поддерживают различное количество транковых групп в зависимости от модели. Коммутатор воспринимает все порты в транковой группе как один порт. Как следствие, транковые порты не блокируются при использовании алгоритма покрывающего дерева (Spanning Tree).

Алгоритм покрывающего дерева (Spanning Tree Algorithm, STP) является средством, с помощью которого мосты могут предотвращать образование петель в топологии сети. Петля возникает, когда пакеты из контрольной точки A могут перемещаться в контрольную точку B в той же сети по более чем одному пути. Для примера образования такой петли возьмем простейшую сеть, состоящую из двух станций A и B и двух мостов 1 и 2. Для простоты разделим сеть на две половины, верхнюю и нижнюю. Станция A подключена к верхней

половине сети, а станция В ─ к нижней. Оба моста 1 и 2 соединяют верхнюю и нижнюю части сети (рис. 2.5). Здесь показана петля, образованная двумя мостами, поскольку пакеты, идущие от станции А к станции В, могут проходить либо через мост 1, либо через мост 2.

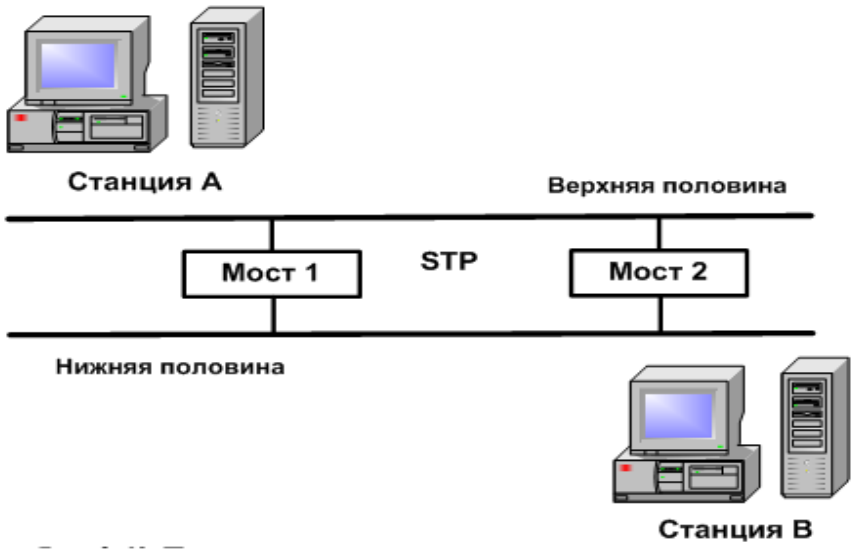


Рис. 2.5

Можно положиться на сетевого администратора, который должен исключить возможность образования петель в сети, но такое решение крайне нежелательно. Даже если администратор имеет время и желание для предотвращения таких вещей, он не застрахован от ошибок. Используя алгоритм покрывающего дерева, администратор сети может не заботиться о возникновении петель, мосты сами об этом позаботятся.

2.4. Использование технологии Ethernet для построения мультисервисных ИС

До недавнего времени технология Ethernet рассматривалась только как коммуникационная технология для передачи данных в локальных вычислительных сетях. Однако стремительный рост скорости от 10 Мбит/с до 1 Гбит/с (и в самом ближайшем времени до 10 Гбит/с) при невысокой стоимости единицы переданной информации, относительная простота дизайна и обслуживания обеспечивают неизбежный устойчивый интерес к технологии Ethernet. В том числе и со стороны операторов связи и провайдеров услуг Интернет.

Интерес этот до поры до времени был чисто теоретическим, поскольку были неочевидны гарантии надежности в сетях Ethernet, а также отсутствовали какие бы то ни было механизмы обеспечения качества обслуживания. Тем не менее, автору известен не один оператор, решившийся и в таких условиях

на предоставление услуг своим клиентам с помощью простой сети Fast Ethernet на магистрали, где «гарантиями качества обслуживания» (Quality of Service, QoS) являлась серьезная по тем временам полоса пропускания, заведомо и намного превышающая весь входящий трафик. Появление концепции мультисервисных услуг на базе протокола IP, подразумевающей интеграцию данных, телефонии и, возможно, трафика видео, предъявило более серьезные требования к сети по надежности и обеспечению QoS, чем требования, которые существовали ранее. Кроме передачи обычного трафика данных, не критичного к задержкам и количеству потерянных пакетов, технологии Ethernet предстояло решить задачу передачи интегрированного трафика.

Первым шагом «на верном пути» стало появление стандарта IEEE 802.1Q/p, описывающего распределенные виртуальные сети (tag switching VLAN) и определяющего механизм приоритезации трафика на канальном уровне. Несмотря на то, что стандарт был направлен на изменение ситуации в секторе ЛВС и корпоративных сетях, оборудование, поддерживающее 802.1Q/p, уже тогда позволило «операторам-первопроходцам» создавать группы VLAN и обеспечивать приемлемое качество обслуживания путем присвоения приоритетов виртуальным сетям (в соответствии с восемью классами).

Однако, появление сетей с маршрутизацией и приоритезацией между VLAN не могло полностью удовлетворить требования, предъявляемые к сети оператора. Чтобы справиться с потенциальными проблемами передачи телефонного разговора по пакетной сети, необходимо было обеспечить качество услуг «из конца в конец» и еще большую гибкость в управлении полосой пропускания.

Под качеством обслуживания QoS в общем случае принято понимать предоставление пользователям и приложениям в сети предсказуемого сервиса по доставке данных. Конкретное же определение и параметры качества обслуживания главным образом определяются типом приложения. Так, например, для передачи голосового трафика, важнейшими параметрами QoS являются задержка и вариация задержки на определенном интервале времени, в то время как потеря некоторой части пакетов допустима. Параметры качества обслуживания можно разбить на три группы:

- параметры пропускной способности (минимальная, средняя и максимальная скорость передачи);
- параметры задержек передачи пакетов (средние и максимальные величины задержек и вариаций задержек);
- параметры надежности передачи (уровень потерь и искажений пакетов).

Измерение указанных параметров производится на определенном интервале времени. Чем меньше этот временной интервал, тем более жесткие требования предъявляются к ИС, а следовательно, ко всем ее элементам, поскольку обеспечение QoS «из конца в конец» требует взаимодействия всех узлов на пути трафика и определяется надежностью, функциональностью и производительностью самого «слабого звена». Например, очевидно, что невозможно гарантировать обеспечение приоритетной обработки VLAN в распределенной коммутируемой сети Ethernet, если по маршруту распространения данных установлен хотя бы один концентратор (hub Ethernet).

Маршрутизаторы

Маршрутизатор (router) используется для соединения различных физических сетей. Однако, в отличие от мостового соединения, в этом случае каждая сеть остается логически независимой. В отличие от моста маршрутизатор является более сложным устройством. Он работает на более высоком уровне сетевых протоколов (в основном, на сетевом), за счет чего обеспечивает возможность соединения сетей различных типов и более надежную защиту от несанкционированного доступа. Маршрутизаторы используются в сетях со сложной топологией, допускающей наличие в сети петель, что требует выбора подходящего маршрута из нескольких альтернатив (мосты в таких конфигурациях использоваться не могут, или же они будут рассматривать альтернативные пути только как резервные). Маршрутизатор обычно является модульным устройством, т.е. имеет несколько слотов для установки различных модулей с необходимыми интерфейсами, что обеспечивает высокую гибкость при построении ИС.

Маршрутизаторы также отличаются более широкими функциональными возможностями по обработке фреймов, поэтому они также используются в тех случаях, где требуется сложное управление трафиком. Маршрутизаторы более сложны в работе, чем мосты, так как требуют определенной квалификации при их конфигурировании, работе и управлении. Поэтому в филиалах предприятий, где нет соответствующих специалистов, стараются использовать мосты, оставляя маршрутизаторы для штаб-квартир и центральных отделений. Маршрутизаторы могут также использоваться для связи сетей с различными протоколами канального уровня, например, Ethernet и Token Ring, хотя некоторые мосты также справляются с этой работой.

В базовые функции маршрутизатора входит создание и поддержание таблицы маршрутизации (построение этой таблицы и ее использование будут

рассмотрены ниже), и выбор на основании этих данных (и данных, находящихся в сетевом пакете, например, IPX или IP сетевой адрес, включающий, как правило, номер сети и номер узла) направления передачи каждого фрейма. При этом, в отличие от моста, разрешается одновременное существование нескольких активных путей между узлами сети. Маршрутизатор “знает” все возможные пути и их характеристики между двумя любыми узлами сети и при этом может направлять фреймы по кратчайшему из этих путей, используя различные критерии и различные алгоритмы маршрутизации, часто достаточно сложные в математическом отношении.

Поэтому маршрутизатор является обычно мощным вычислительным устройством, с одним или даже несколькими процессорами, чаще всего RISC-архитектуры, со сложным программным обеспечением на базе ОС Unix. То есть, сегодняшний маршрутизатор ─ это специализированный компьютер, имеющий скоростную внутреннюю шину или шины (с пропускной способностью 600 - 2000 Мбит/с), часто использующий симметричное или ассиметричное мультипроцессирование. Большая вычислительная мощность позволяет маршрутизаторам наряду с основной работой по выбору оптимального маршрута выполнять и ряд вспомогательных высокоуровневых функций.

К дополнительным функциям, выполняемым маршрутизаторами, относятся:

- возможность назначения приоритетов при передаче фреймов;
- определение дополнительных источников информации для таблиц маршрутизации;
- динамическое перераспределение нагрузок между различными направлениями сети;
- создание фильтров;
- сжатие данных для передачи по телефонным линиям, а также другие возможности, зависящие от фирмы-изготовителя и модели маршрутизатора.

Наряду с функцией маршрутизации интеллектуальные маршрутизаторы обладают следующими важными расширенными возможностями:

- приоритет одного протокола над другими, что бывает полезно в случае недостаточной полосы пропускания кабельной системы;
- защита от "штормов“ широковещательных пакетов (Broadcast storm).

Одна из характерных неисправностей сетевого оборудования ─ самопроизвольная генерация с высокой интенсивностью широковещательных пакетов. Обычный мост слепо передает такие пакеты во все стороны, засоряя,

таким образом, сеть. Борьба с широковещательным штормом в сети, соединенной мостами, требует от администратора отключения портов, генерирующих широковещательные пакеты. Маршрутизатор не распространяет такие поврежденные пакеты, потому что в его обязанности не входит копирование широковещательных пакетов.

Издержками реализации интеллектуальных функций является более медленная обработка пакетов маршрутизатором по сравнению с мостом. Хороший мост может передавать пакеты между портами со скоростью, близкой к максимальной скорости протокола, то есть до 14880 пакетов в секунду для сетей Ethernet, в то же время средний маршрутизатор обрабатывает пакеты со скоростью от 2000 до 5000 пакетов в секунду, хотя в последнее время появились и более быстрые маршрутизаторы, которые по скорости приближаются к мостам.

Немаршрутизируемые протоколы, такие как NetBIOS, NetBEUI или LAT, маршрутизаторы обрабатывают как мосты.

Шлюзы

Шлюзы (gateways) оперируют на верхних уровнях модели OSI (сеансовом, представления и прикладном). Они представляют наиболее развитый метод подсоединения сетевых сегментов. Необходимость в сетевых шлюзах возникает при объединении двух систем, имеющих совершенно различную архитектуру. Например, шлюз приходится использовать для соединения локальных сетей с протоколом TCP/IP и стандартом SNA (супер-ЭВМ). Эти две архитектуры не имеют ничего общего, и поэтому требуется полностью переводить весь поток данных, проходящих между двумя системами.

2.5. Типовые структуры локальных сетей в корпоративных ИС

Несмотря на разнообразие топологий связей в конкретных локальных сетях, существует несколько базовых решений, которые могут составлять основу для разработки индивидуального решения. Все базовые решения основаны на разбиении сети на горизонтальные подсети, вертикальные подсети и сети кампусов. Сети кампусов могут простираться на несколько километров, но при этом глобальные соединения не требуются. Сети кампусов имеют позвоночник (backbone) или главную сеть и подсети, подобные ребрам. Подсети подсоединяются к позвоночнику при помощи различного активного сетевого оборудования. Такое разбиение хорошо отражает и функциональную структуру предприятия, и пространственную структуру зданий.

По мере продвижения снизу вверх по иерархии подсетей изменяются и используемые устройства образования локальных связей. Для образования связей в отделах и рабочих группах используются обычно простые концентраторы-повторители, которые с помощью мостов или маршрутизаторов подключаются к вертикальному позвоночнику сети здания. Сети зданий объединяются в позвоночник кампуса с помощью маршрутизаторов, которые используются и как устройства соединения локальных сетей с глобальными.

Одной из типичных структур для сети небольшого здания является структура, представленная на рис. 2.6. Здесь на каждом этаже для соединения рабочих станций и серверов используются концентраторы-повторители для сегментов Ethernet и колец Token Ring.

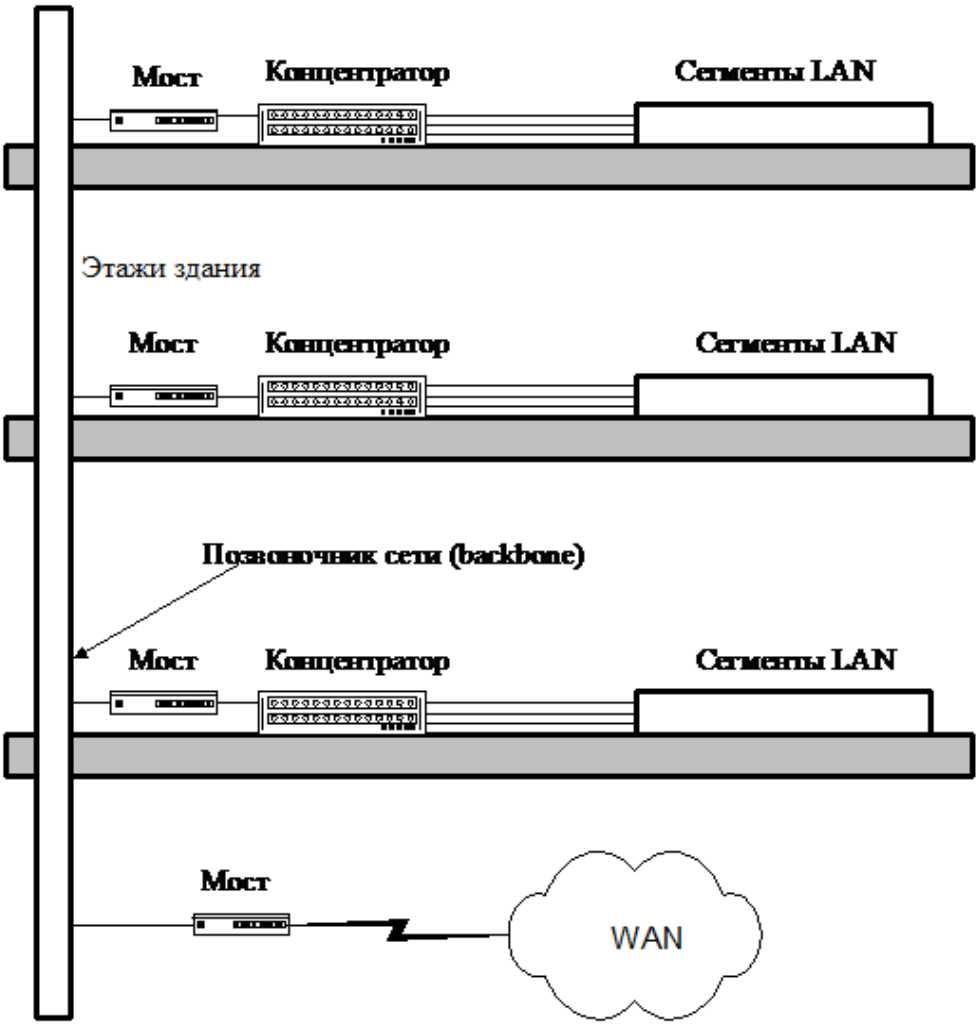


Рис. 2.6

Для локализации трафика в подсетях этажей они присоединяются к кабелю вертикального позвоночника через мосты. Если на этаже имеется несколько подсетей, то может использоваться несколько мостов. Такая схема не

обладает большой применимостью, так как ориентирована в основном на сети Ethernet, использующие в качестве вертикального позвоночника коаксиальный кабель. Использование подсетей Token Ring на этажах или FDDI в качестве вертикальной сети приведет к большим сложностям, так как немодульные мосты редко поддерживают трансляцию протоколов.

Поэтому для более сложных сетей, объединяющих различные канальные протоколы, используется другое базовое решение, в котором вместо мостов применяются маршрутизаторы, которые решают задачи согласования различных канальных протоколов, защищают корпоративную сеть от широковещательных штормов и обеспечивают другие высокоуровневые функции управления трафиком.

Кроме того, они могут использоваться для образования избыточных связей, как между этажами, так и между зданиями, то есть такая структура сети может развиваться проще, чем структура, построенная полностью на мостах.

Однако распределение маршрутизаторов по этажам усложняет управление сетью, так как каждый маршрутизатор требует индивидуальной настройки и управления. Кроме того, высокая стоимость маршрутизаторов удорожает стоимость всей сети при большом количестве этажей. Поэтому чаще используется схема построения сети с базовым маршрутизатором, объединяющим на своей внутренней магистрали подсети этажей. Такую структуру называют структурой со стянутым в точку позвоночником (collapsed backbone).

Сеть с вырожденным позвоночником (рис. 2.7) обладает рядом преимуществ, по сравнению с сетью на распределенных маршрутизаторах:

- маршрутизатор в сети один, поэтому такое решение дешевле;
- маршрутизотор в сети один, поэтому им легче централизованно управлять;
- внутренняя магистраль маршрутизатора обычно намного быстрее, чем позвоночник, реализующий канальный протокол локальной сети, поэтому связи между подсетями происходят намного быстрее, общая производительность сети увеличивается.

Еще более гибким решением будет использование в качестве устройства с вырожденной магистралью корпоративного модульного концентратора. Для многих предприятий такой концентратор позволит подобрать нужный набор модулей ─ мостов, маршрутизаторов или коммутируемых мостов – для нужной связи подсетей друг с другом на скоростной магистрали концентратора. Для не слишком больших сетей можно отказаться от концентраторов на эта-

жах, а использовать модули- повторители для прямой связи подсетей с центральным концентратором.

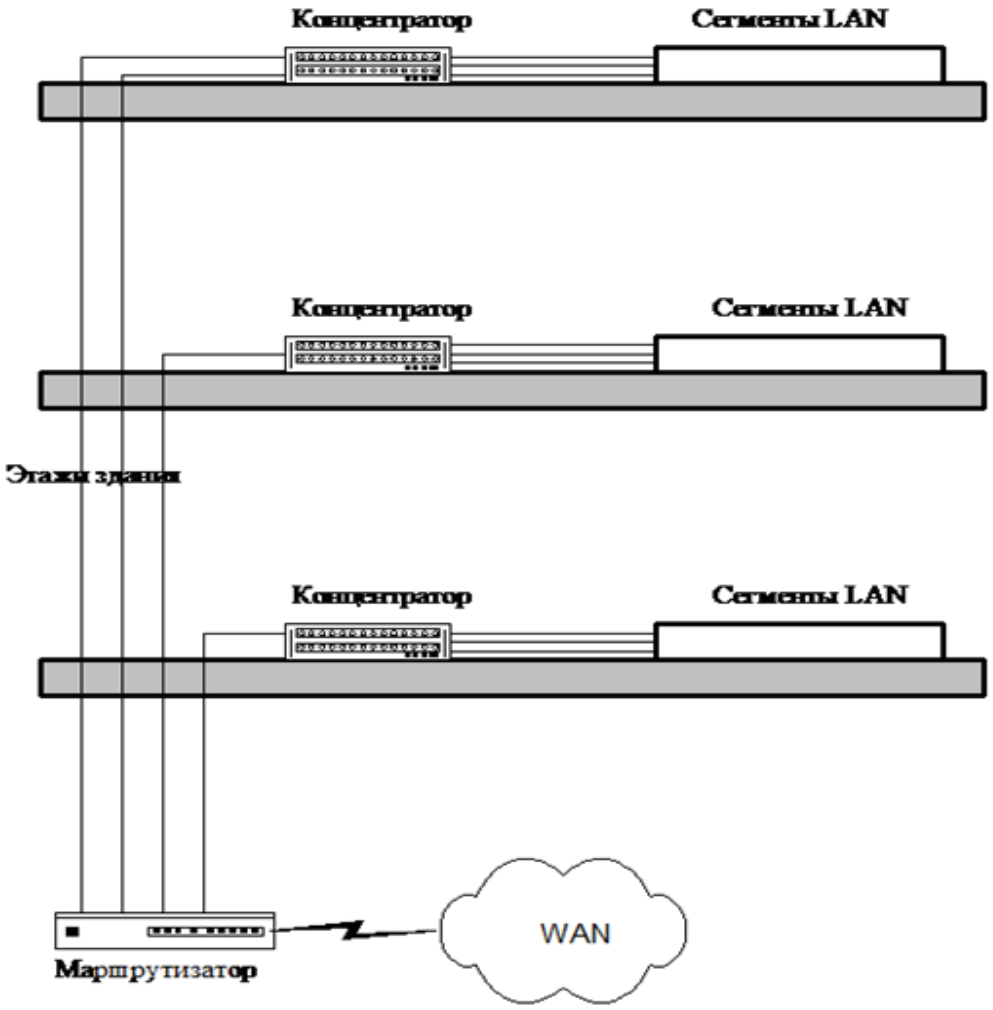


Рис. 2.7

Switch-концентраторы, работающие по технологии коммутации, обычно используются на нижнем уровне иерархии структурированной сети в качестве быстродействующей замены повторителям или обычным мостам. Обособленный коммутирующий концентратор типа ES-100 фирмы Kalpana или EtherSwitch фирмы Hewlett-Packard можно использовать на этажах для соединения отдельных станций или серверов, требующих большой полосы пропускания. Такой подход называется микросегментацией, так как каждый сегмент состоит из одного устройства. Альтернативой такому подходу является использование коммутирующих модулей в модульном центральном коммутаторе, собирающих на свои входы отдельные быстрые серверы или подсети.

3. СЕТЕВОЙ УРОВЕНЬ

3.1. Маршрутизация

Сетевой уровень отвечает за соединение любых двух станций в ИС (это довольно просто в небольшой локальной сети и, как правило, в основном реализуется на канальном уровне). ИС – это некоторый набор сегментов, подсетей или локальных сетей в различных местах, связанных вместе. Сетевой уровень должен знать все адреса и все возможные соединения в сети. Его задача - найти маршрут от одного компьютера к другому. Этот путь может быть очень сложным, проходящим через несколько других компьютеров - маршрутизаторов. Возможно, существует более одного пути для того, чтобы попасть из одного места ИС в другое.

Для того чтобы иметь информацию о текущей конфигурации, маршрутизаторы обмениваются маршрутной информацией между собой по специальным протоколам маршрутизации (это несетевые протоколы типа IP или IPX). Протоколы маршрутизации называются еще протоколами обмена маршрутной информацией. Наиболее популярными протоколами этого класса являются протоколы RIP, RIPv2 и 0SPF, хотя используются и другие. Более совершенным является протокол маршрутизации OSPF (Open Shortest Path First), который основан на построении каждым маршрутизатором графа связей в сети с учетом их состояний и выборе кратчайшего пути на графе связей. OSPF. Кроме протоколов RIP, RIPv2 и OSPF используются и другие протоколы, например IS-IS.

Протокол маршрутизации RIP

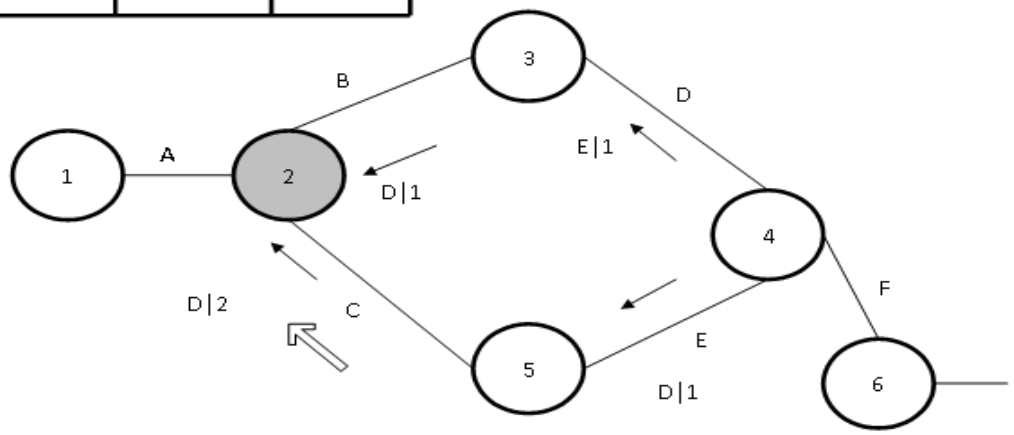
Протокол RIP называется протоколом дистанционно-векторного типа. В нем маршрутизаторы не составляют себе полную картину графа связей сети, а пользуются его упрощенным представлением. Каждый маршрутизатор представляет сеть как вектор расстояний от себя до других сетей (а сети в маршрутизируемых сетевых протоколах нумеруются). Каждый элемент этого вектора соответствует определенному порту маршрутизатора, поэтому при выборе маршрута к сети с определенным номером маршрутизатор просто выбирает тот порт, которому приписано минимальное расстояние до этой сети. Маршрутизаторы, работающие по протоколу RIP, узнают вектора дистанций путем взаимного обучения, то есть они периодически рассылают по сети широковещательные пакеты, содержащие информацию об известных им на данный момент сетях и расстояниях до них. Эта информация итерационно распро-

страняется по сети, и через несколько шагов каждый маршрутизатор имеет данные о достижимых для него сетях и о расстояниях до них (рис. 3.1).

Можно убедиться, что при использовании протокола RIP решение находится не оптимальное, а приближенное. Например, путь в один шаг по низкоскоростной линии связи окажется более предпочтительным, чем путь в два шага по скоростной локальной сети, кроме того, дополнительный маршрут может быть использован только в случае недостижимости основного маршрута.

Преимуществом протокола RIP является его вычислительная простота, а недостатками – увеличение трафика при периодической рассылке широковещательных пакетов, неоптимальность найденного маршрута и простой имеющихся сетевых путей.

Сеть	Расстояние	Сосед
A	1	-
B	1	-
C	1	-



После 2-х шагов:

Сеть	Расстояние	Сосед
A	1	-
B	1	-
C	1	-
D	2	3
E	2	5
D	3	5
	3	5

Рис. 3.1

3.2. Протоколы TCP/IP и стандарты OSI/ISO

Международной организацией по стандартизации (International Standard Organization, ISO) была разработана спецификация, названная моделью Взаимодействия открытых систем (Open Systems Interconnection, OSI). Однако более общие и функциональные продукты OSI не вытеснили технологию TCP/IP. Главная причина этого заключается в том, что протоколы TCP/IP сумели набрать предусмотренную в новых продуктах функциональность еще до того, как последние появились на рынке. Под семейством (стеком) протоколов TCP/IP понимают весь набор реализаций стандартов RFC. Он охватывает все семейство протоколов, прикладные программы и саму сеть. В состав семейства входят протоколы TCP, UDP, IP, ARP, ICMP, TELNET, FTP и многие другие. TCP/IP - это технология межсетевого взаимодействия, технология Internet. Отображение этого семейства протоколов на модель OSI приведено в табл. 3.1.

Основополагающим элементом для всех этих протоколов является Internet Protocol (IP). Этот протокол реализует распространение информации по IP-сети. Протокол IP осуществляет передачу информации от узла к узлу ИС в виде дискретных блоков - пакетов. При этом IP не несет ответственности за надежность доставки информации, целостность или сохранение порядка потока пакетов и не решает задачу передачи информации. Эту задачу решают два других протокола TCP (Transmission Control Protocol, протокол управления передачей данных) и UDP (User Datagram Protocol, дейтаграммный протокол передачи данных), которые “лежат” над IP (используют процедуры протокола IP для передачи информации, добавляя к ним свою дополнительную функциональность). Протокол IP решает задачу маршрутизации этих пакетов в ИС.

Протоколы TCP и UDP реализуют различные режимы доставки данных: TCP - протокол с установлением соединения, а UDP является дейтаграммным протоколом (без установления соединения), т. е. таким, в котором каждый блок передаваемой информации (пакет) обрабатывается и распространяется от узла к узлу ИС не как часть некоторого потока, а как независимая единица информации - дейтаграмма (datagram). Необходимость в UDP обусловлена тем, что IP работает как сетевой протокол, решающий задачу доставки информации от узла к узлу ИС, в то время как UDP “умеет” различать приложения и передает информацию от приложения к приложению, которых на одном сетевом узле может быть несколько.

Таблица 3.1

Отображение протоколов TCP/IP на модель OSI

<table>
<tr><th>OSI модель</th><th>ARPA</th><th>Berkeley</th><th>NFS</th><th>Структура продуктов</th></tr>
<tr><td>7. Прикладной уровень</td><td>ftp
telnet</td><td>rcp
rlogin
rexec
rwho
ruptime
Sendmail</td><td>NFS
NIS
VHE</td><td rowspan="3">Сервисы</td></tr>
<tr><td>6.Представления</td><td>SMTP</td><td></td><td>XDR</td></tr>
<tr><td>5.Сеансный</td><td colspan="2">BCD IPC (Berkeley socket's)</td><td>RCP</td></tr>
<tr><td>4. Транспортный</td><td colspan="2">TCP/UDP</td><td>UDP</td><td rowspan="4">Сетевые уровни</td></tr>
<tr><td>3. Сетевой</td><td colspan="3">IP</td></tr>
<tr><td>2.Канальный</td><td colspan="3"></td></tr>
<tr><td>1.Физический</td><td colspan="3"></td></tr>
</table>

Над транспортными протоколами TCP и UDP лежат протоколы, реализующие те или иные прикладные службы, такие как обмен файлами (File Transfer Protocol, FTP) и сообщениями электронной почты (Simple Mail

Transfer Protocol, SMTP), терминальный доступ к удаленным сераерам (Telnet), сетевое управление (Simple Network Manager Protocol, SNMP) и т.д.

Иерархию управления в TCP/IP-сетях представляют в виде пятиуровневой диаграммы, приведенной на рис. 3.2. Название блока данных, передаваемого по сети, зависит от того, на каком уровне системы протоколов он находится:

- для прикладных программ, работающих по протоколу TCP, данные, передаваемые по сети, называются *потоками*;
- для прикладных программ, работающих по протоколу UDP, данные, передаваемые по сети, называются *сообщениями*;
- блоки данных, которыми обмениваются транспортный и сетевой уровни TCP, называются *сегментами*;
- аналогичные блоки данных для протокола UDP называются *пакетами* или *UDP-дейтаграммами*;
- блоки данных, которыми обмениваются сетевой и канальный уровни, называются *IP-пакетами* или *дейтаграммами*;
- блок данных, который передается по физической среде передачи данных, называется *фреймом.*

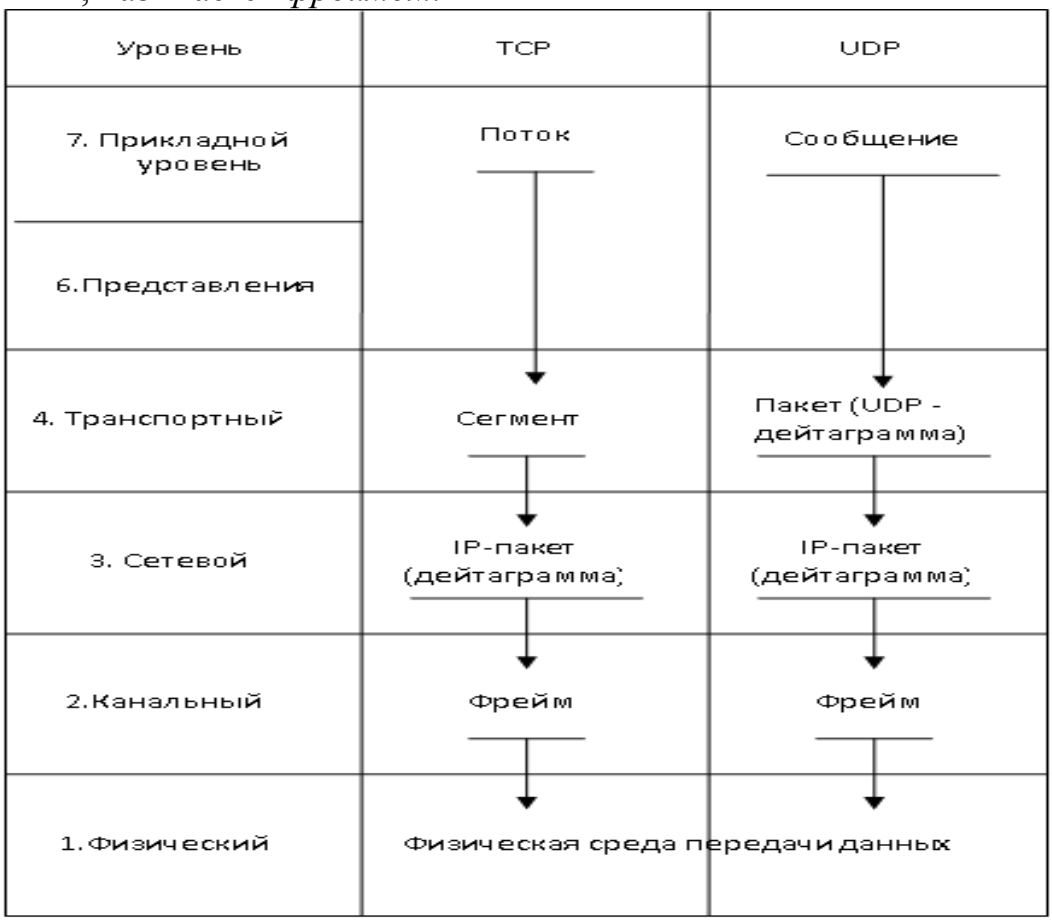


Рис. 3.2

Физический уровень описывает ту или иную среду передачи данных. На канальном уровне лежит аппаратно зависимое программное обеспечение (например, драйвер сетевого адаптера), реализующее распространение информации на том или ином отрезке среды передачи данных. TCP/IP является программной реализацией и никаких ограничений на программное обеспечение этих уровней не накладывает.

Сетевой уровень и есть протокол IP, главными задачами которого являются:

- формирование дейтаграмм;
- поддержание схемы адресации в ИС;
- маршрутизация дейтаграмм в ИС (выбор пути через множество промежуточных узлов) при доставке информации от узла-отправителя до узла-адресата;
- фрагментация и сборка дейтаграмм (пересылка IP-пакетов через физическую среду с возможной разбивкой на фреймы);
- предоставление вышележащим уровням единого унифицированного и аппаратно независимого интерфейса для доставки информации. Достигается при этом канальная (аппаратная) независимость и обеспечивается многоплатформное применение приложений, работающих над TCP/IP.

Сохранение порядка и целостности потока пакетов - задачи других протоколов - TCP и UDP, относящихся к следующему транспортному уровню. Выше - на уровне приложений - лежат прикладные задачи, запрашивающие сервис у транспортного уровня.

Концепция построения сети Интернет

Вся ИС представляется как множество значимых для нас компьютеров (хостов), подключенных к единой сети Internet, внутренняя структура которой неважна, что и подчеркивается изображением этой интерсети в виде некоего расплывчатого облака. Это облако, в свою очередь, состоит из маленьких облачков, называемых физическими сетями, и соединяющих их маршрутизаторов (router). Физические сети могут не являться носителями протокола IP. Это могут быть локальные сети, работающие под управлением некоторых аппаратно зависимых протоколов, или коммуникационные системы произвольной физической природы. Все функции IP исполняют хосты и маршрутизаторы.

3.3. Технология работы протокола IP

Протокол IP обеспечивает негарантированную (unreliable) доставку ннформации – пакета. Пакет может быть утерян, дублирован, задержан, доставлен с нарушением порядка. IP не имеет инструментов для определения таких ситуаций.

Протокол IP обеспечивает только дейтаграммную (datagram) доставку (или доставку без установления соединения). Последовательно исходящие от отправителя пакеты могут распространяться по различным петлям в сети, теряться и менять порядок. Протокол IP производит максимально обеспеченную доставку информации в том смысле, что потеря пакета происходит лишь в той ситуации, когда протокол не находит никаких физических средств для его доставки.

Физические объекты (хосты, маршрутизаторы, подсети) в IP-сети идентифицируются при помощи имен, называемых IP-адресами. Объект, наделяемый IP-адресом, скорее представляет собой сетевое соединение, а не физическое устройство. Например, хост, соединенный с двумя сетями, имеет два IP-адреса, относящихся к каждой из сетей.

IP-адреса представляют собой 32-битовые идентификаторы, структура которых оптимизирована для решения основной задачи протокола IP- маршрутизации. Обычно для удобства представления IP-адресов используется цифровое написание IP-адресов - адрес записывается как десятичное представление 4 байт, разделенных точками, например 192.168.13.60.

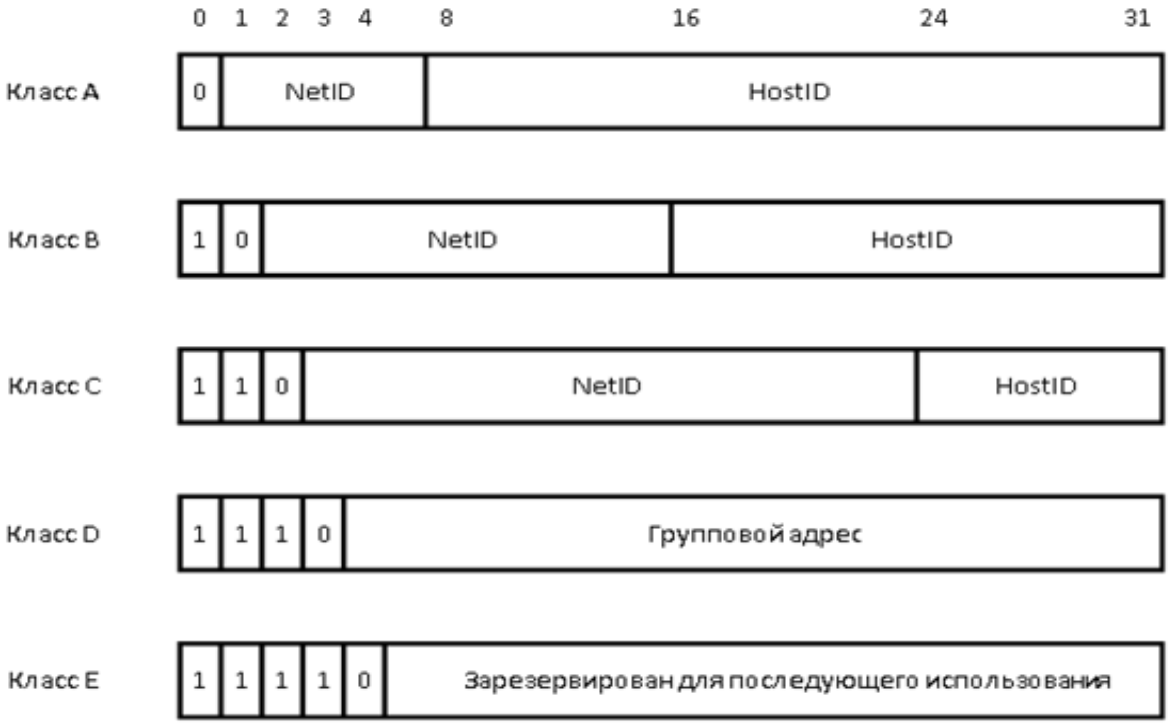
В общем случае каждый адрес можно представить как пару идентификаторов (NetID, HostID), где NetID - идентификатор сети, а HostID - идентификатор хоста. Практически каждый IP-адрес должен быть представлен в виде одной из первых трех показанных на рис. 3.3 битовых структур. Все IP-адреса разделены на 5 классов, но практическое применение находят, в основном, три первых.

Класс А определен для сетей с огромным (от 65 535 до 16777 21) числом хостов. В адресе этого класса 7 бит отведены под поле идентификатора ИС (NetID), а 24 бит - под поле идентификатора хоста (HostID).

Адреса класса В используются для среднемасштабных ИС, в которых содержится от 256 до 65 536 хостов; под поля NetID и HostID отводится 14 и 16 бит соответственно.

Адреса класса С предназначены для ИС с числом хостов менее 256, под HostID отведено 8 бит и под NetID - 21 бит. IP-адрес построен таким обра

зом, чтобы поля NetID и HostID можно было выделить быстро. Эффективность маршрутизации, осуществляемой маршрутизаторами и основанной только на полях NetID, зависит от эффективности выделения этого поля.



Класс	Диапазон значений первого байта	Возможное количество сетей	Возможное количество узлов
A	1 - 126	126	16777214
B	128 - 191	16382	65534
C	192 - 223	2097150	254
D	224 - 239	-	2^28
E	240 - 247	-	2^27

Рис. 3.3

IP-адрес идентифицирует сетевое соединение, а не хост: Адрес a1 на рис. 3.3 будет содержать идентификатор сети Netl, а адрес a2 – идентификатор сети Net2. Отсюда вытекает и следующее принципиальное ограничение, связанное с IP-адресацией: если хост переносится из одной подсети в другую, он должен обязательно изменить IP-адрес.

IP-адресация поддерживает адреса, обращенные к множеству хостов и/или сетей. Среди таких адресов различают два класса: широковещательные (broadcast), обращенные “ко всем”, и групповые (multicast), обращенные к заданному множеству объектов. Заполнение адресных полей нулями означает обращение к данному объекту (this), а заполнение единицами - обращение ко всем объектам (all).

Специальные IP-адреса, формируемые по этим правилам, показаны на рис. 3.4.

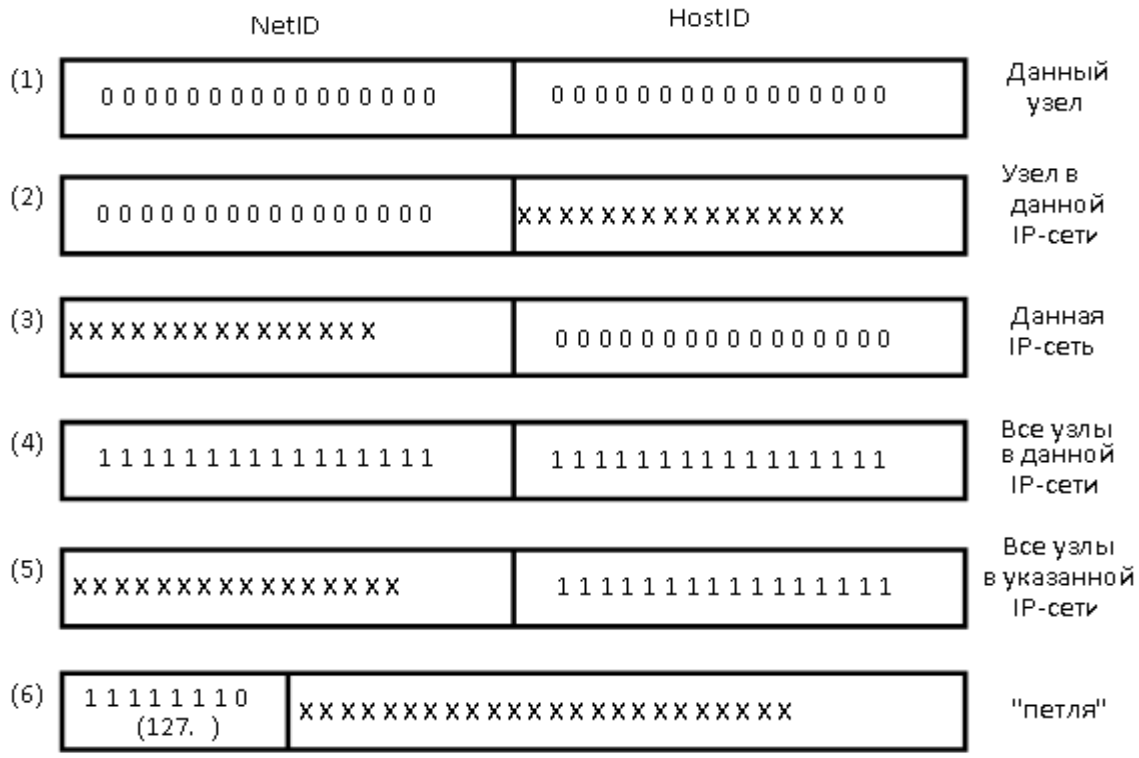


Рис. 3.4

Адрес 1 является “пустышкой”. Он может использоваться в инициализованной процедуре, когда рабочая станция не знает (или хочет согласовать) своего IP-адреса, и только как адрес отправителя, но никогда как адрес получателя пакета.

Адрес 2, в котором идентификатор сети (NetID) заполнен нулями, а идентификатор хоста (HostlD) имеет некоторое значение, есть адрес конкретного хоста в сети, из которой он получил пакет. Такой адрес применяется в том случае, когда хост-отправитель не знает идентификатора сети, в которой работает. (Такая ситуация возможна, например, при инициализации бездиско-

вой рабочей станции, которая при включении в сеть вообще “ничего не знает” ни о сети, ни о себе.) Этот адрес используется только как адрес получателя и никогда – как адрес отправителя.

Адрес 3, в котором поле NetID имеет некоторое значение, а поле HostID заполнено нулями, трактуетстся протоколом IP как адрес некоторой сети (но ни одного из хостов данной сети, поскольку состоящий из нулей HostID запрещен).

Адрес 5, в котором идентификатор сети имеет осмысленное значение, а поле идентификатора хоста заполнено единицами, называется широковещательным адресом (direct broadcast address), обращенным ко всем хостам в данной подсети. Наряду с ним в качестве применяется также локальный, или ограниченный, адрес (limited, или local broadcast address), целиком заполненный единицами (адрес 4), полезный, когда идентификатор сети неизвестен. Использование этого адреса не рекомендуется.

Адрес 6 - тестовый адрес (loopback address), в котором первый байт имеет значение 127, а оставшееся поле не специфицировано (обычно заполняется единицами). Он используется для задач отладки и тестирования, не является адресом никакой сети, и маршрутизаторы никогда не обрабатывают его. Он используется для взаимодействия процессов в пределах одной машины. Когда программа посылает данные по IP-адресу 127.0.0.1, то образуется как бы "петля". Данные не передаются по сети, а возвращаются модулям верхнего уровня, как только что принятые. Поэтому в IP-сети запрещается присваивать машинам IP-адреса, начинающиеся со 127.

Согласно принятому правилу узлу в IP-сети нельзя присвоить номер 0. IP-адрес, номер узла в котором 0, позволяет ссылаться на сеть, как на отдельный узел. Это повышает эффективность маршрутизации.

Выбор IP адреса. Одно из важнейших решений, которое необходимо принять при установке сети, заключается в выборе способа присвоения IP-адресов машинам. Этот выбор должен учитывать перспективу роста сети. Иначе, в дальнейшем вам придется менять адреса. Когда к сети подключено несколько сотен машин, изменение адресов становится почти невозможным.

Групповые адреса. Групповой адрес (multicast address), в отличие от широковещательного, применяется при обращении к выделенной группе хостов (но не ко всем хостам) некоторой физической сети или группы сетей. В этом случае используются адреса класса D (рис. 3.3). Групповая адресация в TCP/IP регламентируется входящим составной частью в IP протоколом Internet Group Management Protocol (IGMP).

Групповой адрес может объединять хосты из разных физических сетей. Это достигается путем использования специальных протоколов групповой маршрутизации на маршрутизаторах, о которых ни отправитель, ни получатель групповой дейтаграммы ничего знать не должны. Теоретически каждый хост может в любой момент подключиться к определенной адресной группе или выйти из нее.

Групповые адреса назначаются Network Information Center и разделяются на два класса:

- постоянные - для непрерывно существующих групп (так называемые well-known addresses, всем известные адреса);
- временные - для организуемых на некоторый срок групп, которые существуют до тех пор, пока в группе сохраняется хотя бы один член (так называемые transient multicast groups, временные адресные группы).

Распространение групповых сообщений по интерсети ограничивается временем жизни (time-to-live) IP-пакета. IP-адреса физических сетей, подключенных к Internet, распределяет Network Information Center (NetID). Автономные корпоративные IP-сети могут самостоятельно решать задачи IP адресации.

Сети и подсети IP. Адресное пространстоо любой сети Internet может быть разделено на непересекающиеся подпространства - "подсети", с каждой из которых можно работать как с обычной сетью TCP/IP. Таким образом, единая IP-сеть организации может строиться как объединение подсетей. Подсетъ соответствует одной физической сети, например, одному сегменту сети Ethernet.

Конечно, можно просто назначить для каждой физической сети свой сетевой номер, например, номер класса С. Однако, такое решение имеет существенные недостатки: зачастую впустую тратятся дефицитные сетевые адреса, которых в настоящее время катастрофически не хватает; кроме того, - это перегрузка таблиц маршрутизации в сети Internet, т.к. машины вне вашей сети должны поддерживать записи о маршрутах доступах каждой из этих IP-сетей.

Подсети позволяют избежать этих недостатков. Ваша организация должна получить один сетевой номер, например, номер класса В. Стандарты TCP/IP определяют структуру IP-адресов. Для IP-адресов класса В первые два байта являются номером сети. Оставшаяся часть IP- адреса может использоваться как угодно. Например, третий байт будет определять номер подсети, а четверый - номер узла в ней. Такая конфигурация подсетей описывается в файлах, определяющих маршрутизацию пакетов. Это описание является локальным для вашей организации и не видно вне ее. Все машины вне вашей ор-

ганизации видят одну большую IP-сеть. Следовательно, они должны поддерживать только маршруты доступа к шлюзам, соединяющим вашу IP-сеть с остальным миром. Изменения, происходящие в IP-сети организации, не видны вне ее. Вы легко можете добавить новую подсеть, новый шлюз и т.д. Такое представление подсетей, как суммы одной большой IP-сети, называется суммаризацией и значительно повышает производительность работы маршрутизаторов.

Распределение IP-адресов сетей и подсетей. Каждой физической сети, например, Ethernet или Token Ring, назначается отдельный номер подсети или номер сети. В некоторых случаях имеет смысл назначать одной физической сети несколько подсетевых номеров. Например, предположим, что есть сеть Ethernet, охватывающая три здания. Ясно, что при увеличении числа машин, подключенных к этой сети, придется ее разделить на несколько отдельных сетей Ethernet. Для того чтобы избежать необходимость менять IP-адреса, когда это произойдет, можно заранее выделить для этой сети три подсетевых номера - по одному на здание. Такая адресация позволяет сразу определить, где находится та или иная машина.

Необходимо выбрать ***маску подсети***. Она используется сетевым программным обеспечением для выделения номера подсети из IP-адресов. Биты IP-адреса, определяющие номер IP-сети, в маске подсети должны быть равны 1, а биты, определяющие номер узла, в маске подсети должны быть равны 0. Стандарты протоколов TCP/IP определяют количество байт, задающих номер сети. Часто в IP-адресах класса B третий байт используется для задания номера подсети. Это позволяет иметь 256 подсетей, в каждой из которых может быть до 254 узлов. Маска подсети в такой системе равна 255.255.255.0. Но, если должно быть больше подсетей, а в каждой подсети не будет при этом более 60 узлов, то можно использовать маску 255.255.255.192. Это позволяет иметь уже 1024 подсети и до 62 узлов в каждой (номер узла 0 – это адрес сети и "все единицы" адрес BroadCast в этой сети).

Обычно маска подсети указывается в файле конфигурации сетевого программного обеспечения. Протоколы TCP/IP позволяют также запрашивать эту информацию по сети. Маска подсети действует только в рамках той сети, для которой она определена. Маску подсети целесообразно использовать для выделения в рамках единой сети подсетей с автономным администрированием. Приоритетом всегда пользуется стандартная схема классов сетей.

В качестве коммуникационного протокола IP специфицирует три основных элемента:

- блок данных, с которыми работает протокол (пакет IP);
- механизмы распространения (маршрутизации) пакетов;
- способы обработки конфликтных ситуаций.

Структура пакета IP. Пакет IP состоит из заголовка и блока данных (рис.3.5,а). Вся функциональность протокола IP основывается на обработке и интерпретации полей заголовка (рис. 3.5,б), структура поля SERVICE TYPE (рис. 3.5,в).

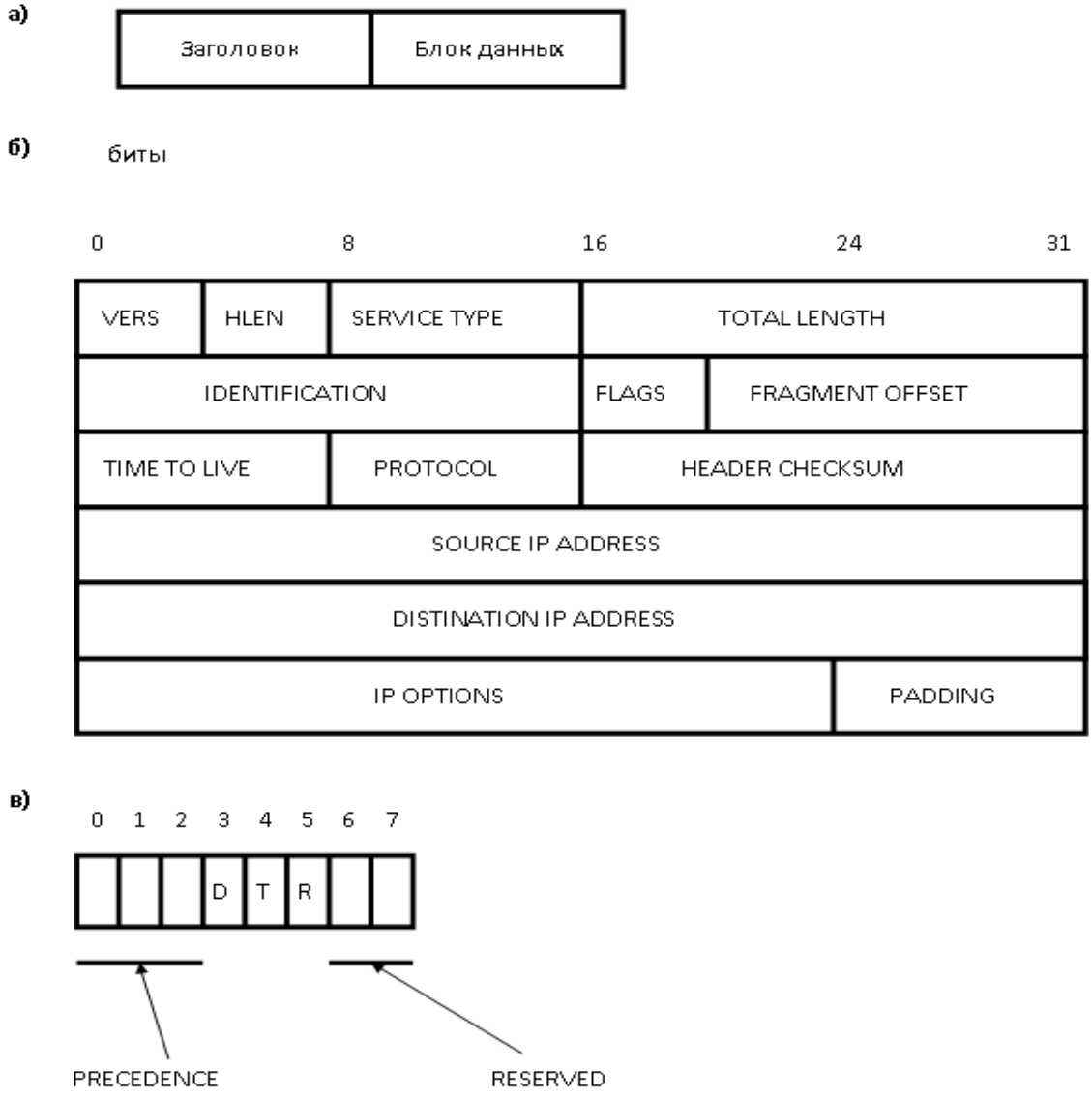


Рис. 3.5

4-битовое поле "версия" (VERS) используется для устранения конфликтов, которые могут возникать при изменении версии протокола IP, когда часть хостов работает по одной, а часть - по другой версии протокола. Если поле

“версия” содержит значение, отличное от текущей версии протокола, пакет уничтожается. Текущей является четвертая версия IP.

Поле “длина заголовка” (HLEN, H[eader] LEN[gth]) дает значение длины заголовка пакета, измеренное в 32 -битовых словах. Это поле предусматривает изменение длины заголовка и интерпретацию его расширения в соответствии с полями IP OPTIONS и PADDING.

Поле “тип сервиса” (SERVICE TYPE) структурировано.

Субполе “приоритет” (PRECEDENCE) поля SERVICE TYPE принимает восемь значений: от 0 (нормальный приоритет) до 7 (сетевое управление). Биты D,T,R этого поля ответственны за тип транспортировки, который “запрашивает” пакет. Установка этих битов в состояние “1” требует соответственно низкой задержки при передаче пакета (от Delay - задержка), высокой пропускной способности (Throughput) и высокой надежности (Reliability). Последние два бита поля SERVICE TYPE не используются.

Поле “полная длина” (TOTAL LENGTH) заголовка пакета задает полную (включая заголовок и данные) длину пакета, измеренную в октетах (байтах). Как видно из рис. 3.5,б, полная длина пакета IP максимально может достигать 65535 байт.

Протоколу IP, предназначенному для межсетевого взаимодействия, приходится ориентироваться на различия в реализациях физических сетей, одним из которых является ограничение на максимальную длину фрейма, разрешенную в той или иной физической сети (Maximum Transfer Unit, MTU). Поэтому протокол IP вынужден решать задачу, более свойственную транспортному протоколу, - разбивку больших пакетов на малые (и наоборот, их сборку). Это требуется делать в тех случаях, когда на вход некоторой физической сети поступает пакет, превосходящий по длине MTU для данной сети. Такая операция в IP называется фрагментированием (fragmentation).

Фрагментирование осуществляется следующим образом. Блок данных исходного (большого) пакета разделяется так, чтобы размер полученных фрагментов в сумме с длиной заголовка не превышал MTU для физической сети, в которую направляются фрагменты. При этом фрагменты упаковываются в пакеты, заголовки которых очень похожи на заголовок исходного пакета. Чтобы понять, что данные пакеты содержат фрагменты одного большого пакета и обеспечить его последующую сборку, производится установка специальных признаков в поле FLAGS; смещения, по которым разрезался исходный блок данных, помещаются в поле FRAGMENT OFFSET; а в поле IDENTIFICATION записывается один общий для всех фрагментов идентифи-

катор, указывающий на принадлежность фрагментов к одному большому блоку данных. Например, для сети Ethernet значение MTU равно 100.

Поле “время жизни” заголовка пакета (TIME TO LIVE) указывает время, в течение которого пакет должен существовать в сети. Хосты и маршрутизаторы, обрабатывающие данный пакет, уменьшают значения этого поля в период обработки и хранения пакета. Когда время жизни истекает, пакет уничтожается. При этом отправитель сообщения уведомляется о потере пакета. Наличие конечного времени жизни пакета обеспечивает защиту от таких нежелательных событий, как передача пакета по циклическому маршруту, перегрузка сетей.

Назначение *оставшихся полей* следующее:

- PROTOCOL (8 бит) - указывает протокол вышележащего уровня (TCP или UDP), которому предназначена информация, содержащаяся в поле данных пакета IP;
- HEADER CHECKSUM (16 бит) - используется для контроля целостности заголовка пакета IP;
- SOURCE IP ADDRESS (32 бита) - IP-адрес отправителя пакета;
- DESTINATION IP ADDRESS (32 бита) - IP- адрес получателя пакета;
- IP OPTIONS (изменяемая длина) - применяется для указания необязательных параметров IP, связанных с режимами безопасности или маршрутизации;
- PADDING (изменяемая длина) - дополняет заголовок пакета таким образом, чтобы он составлял целое число 32-битовых слов.

Отражение IP-адресов на физические MAC-адреса устройств в ИС

Концепция IP-сети требует установления соответствия между IP-адресами и аппаратно зависимыми MAC-адресами устройств внутри физических сетей.

IP-адрес должен однозначно определять сетевое соединение и в сетях X.25 или Token Ring, где используются конфигурируемые и достаточно короткие адресные поля, и в сетях Ethernet, где каждый сетевой адаптер имеет 48-битовый серийный номер, который устанавливают производители оборудования под управлением IEEE (Institute for Electric and Electronic Engineers).

Задачу определения физического адреса хоста по его IP-адресу решают два входящих в IP в виде составных частей протокола: ARP (Address Resolution Protocol) и RARP (Reverse Address Resolution Protocol). Схема отображения протокола ARP на модель OSI приведена на рис. 3.6.

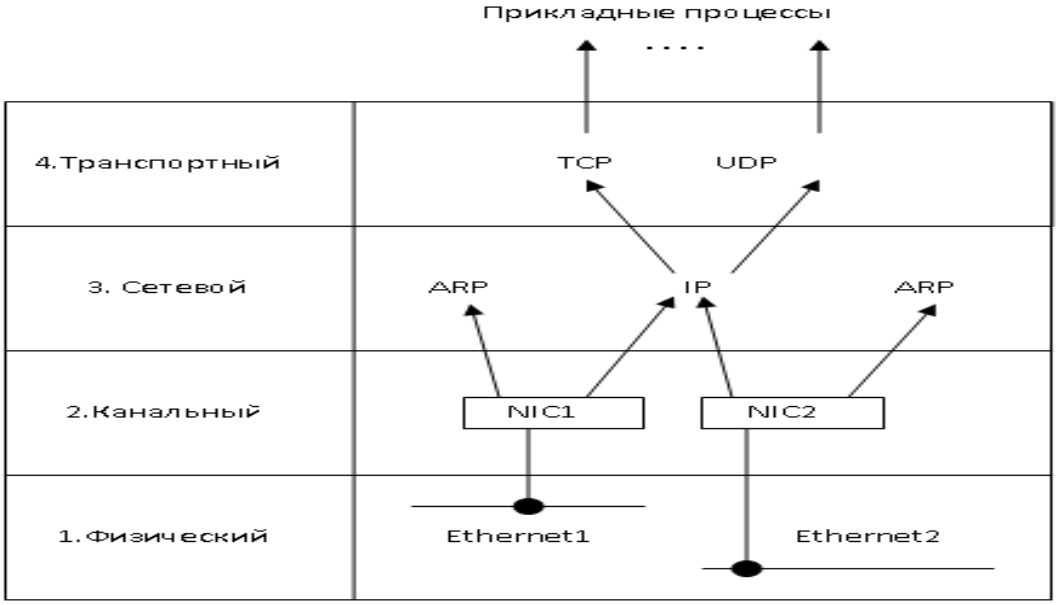


Рис. 3.6

3.4. Протокол ARP

Рассмотрим, как при посылке IP-пакета определяется Ethernet-адрес получателя. Для преобразования IP-адресов в Ethernet-адреса используется протокол ARP (Address Resolution Protocol - протокол преобразования адресов). Это преобразование выполняется только для отправляемых IP-пакетов, так как только в момент отправки создаются заголовки IP и Ethernet.

Преобразование адресов выполняется путем поиска в таблице, называемой ARP-таблицей, хранящейся в памяти и содержащей строки для каждой машины в ИС. В двух столбцах содержатся IP- и Ethernet-адреса. Если требуется преобразовать IP-адрес в Ethernet-адрес, то ищется запись с совпадающим IP-адресом. На рис. 3.7,а приведен пример упрощенной ARP-таблицы.

Принято все байты 4-байтного IP-адреса записывать десятичными числами, разделенными точками. При записи 6-байтного Ethernet-адреса каждый байт указывается в 16-ричной системе и отделяется двоеточием.

ARP-таблица необходима, потому что IP-адреса и Ethernet-адреса выбираются независимо, и нет какого-либо одного алгоритма для преобразования одного адреса в другой. IP-адрес выбирает администратор ИС с учетом положения машины в сети. Если машину перемещают в другую часть сети, то ее IP-адрес должен быть изменен. Ethernet-адрес выбирает изготовитель сетевого интерфейсного оборудования из выделенного для него по лицензии адресного

пространства. Когда у машины заменяется плата сетевого адаптера, то меняется и ее Ethernet-адрес.

а) Пример ARP-таблицы.

IP-адрес	Ethernet-адрес
223.1.2.1	08:00:39:00:2F:C3
223.1.2.3	08:00:5A:21:A7:22
223.1.2.4	08:00:10:99:AC:54

б) Пример ARP-запроса..

IP-адрес отправителя	223.1.2.1
Ethernet-адрес отправителя	08:00:39:00:2F:C3
Искомый IP-адрес	223.1.2.2
Искомый Ethernet-адрес	<пусто>

в) Пример ARP-ответа..

IP-адрес отправителя	223.1.2.2
Ethernet-адрес отправителя	08:00:28:00:38:A9
Искомый IP-адрес	223.1.2.1
Искомый Ethernet-адрес	08:00:39:00:2F:C3

г) ARP-таблица после обработки ответа.

IP-адрес	Ethernet-адрес
223.1.2.1	08:00:39:00:2F:C3
223.1.2.2	08:00:28:00:38:A9
223.1.2.3	08:00:5A:21:A7:22
223.1.2.4	08:00:10:99:AC:54

Рис. 3.7

Обычно прикладная программа, такая как TELNET, отправляет прикладное сообщение модулю TCP, а тот посылает соответствующее транспортное сообщение модулю IP. Прикладной программе, модулю TCP и модулю IP IP-адрес места назначения известен. К этому времени IP-пакет уже составлен и готов к передаче драйверу Ethernet, но сначала необходимо определить Ethernet-адрес назначения. Для этого используют ARP-таблицу.

ARP-таблица заполняется автоматически модулем ARP, по мере необходимости. Если с помощью существующей ARP-таблицы не удается преобразовать IP-адрес, то происходит следующее:

1. По сети передается широковещательный ARP-запрос. Пример пакета с ARP-запросом приведен на рис.3.7,б.
2. Исходящий IP-пакет ставится в очередь.
3. Возвращается ARP-ответ, содержащий информацию о соответствии IP- и Ethernet-адресов. Эта информация заносится в ARP-таблицу. Пример пакета с ARP-ответом приведен на рис. 3.7,в.
4. Для преобразования IP-адреса в Ethernet-адрес у IP-пакета, поставленного в очередь, используется ARP-таблица. Обновленная ARP-таблица приведена на рис.3.7,г.
5. Ethernet-фрейм передается по сети Ethernet.

Таким образом, если преобразование какого-то IP-адреса в ARP-таблице пропущено, то IP-пакет ставится в очередь, а необходимую для этого преобразования информацию получают с помощью запроса и ответа протокола ARP, после чего IP-пакет передают дальше. Если в ИС нет узла с искомым IP-адресом, то ARP-ответа не будет и не будет записи в ARP-таблице. Протокол IP уничтожит IP-пакеты, направляемые по этому адресу. Каждый узел имеет отдельную ARP-таблицу для каждого своего сетевого адаптера.

Эта схема хороша при условии, что узел ИС знает свой IP - адрес. А как быть, если, например, узел - бездисковая рабочая станция, у которой только что включили питание и она не только ничего не знает ни о себе, ни об окружающих, но и не может произвести дистанционную загрузку операционной системы, которая хранится где-то на сетевом диске. В этом случае используется протокол RARP. Узел широковещательно вызывает обслуживающий его сервер, закладывая в запрос свой физический адрес. (При этом узел может даже не знать адреса сервера). В сети находится по меньшей мере один обслуживающий такие запросы сервер (RARP-сервер), который распознает запрос от рабочей станции, выбирает из некоторого списка свободный IP-адрес и шлет узлу сообщение, включающее необходимую информацию - динамически выделенный узлу IP-адрес, свой физический и IP-адрес и т. д. Поскольку при таком механизме отказ RARP-сервера очень критичен, то обычно сеть конфигурируется так, чтобы протокол RARP поддерживало несколько серверов в сети.

Протокол ARP с представителем. Протокол ARP с представителем является альтернативным методом, позволяющим шлюзам, принимать все необ-

ходимые решения о маршрутизации. Он применяется в сетях с широковещательной передачей, где для отображения IP-адресов в Ethernet-адреса используется протокол ARP или ему подобный. Будем предполагать, что имеем дело со средой Ethernet. Протокол ARP с представителем, во многом аналогичен использованию маршрутов по умолчанию и сообщению перенаправления. Однако он использует иной механизм сообщения узлам в сети маршрутов. Протокол ARP с представителем не затрагивает таблиц маршрутов, все делается на уровне адресов Ethernet.

Чтобы использовать протокол, нужно настроить узел ИС так, как будто все машины в мире подключены непосредственно к вашей локальной среде Ethernet. В ОС UNIX это делается командой "route add default 128.6.4.2 0", где 128.6.4.2 - IP-адрес вашей машины, а метрика 0 говорит о том, что все IP-пакеты, которым подходит данный маршрут, должны посылаться напрямую по локальной среде Ethernet.

Когда нужно послать IP-пакет узлу в локальной среде Ethernet, ваша машина должна определить Ethernet-адрес места назначения. Для этого она использует ARP-таблицу. Если в ARP-таблице уже есть запись, соответствующая IP-адресу места назначения, то из нее просто берут Ethernet-адрес места назначения. Если такой записи нет, то посылается широковещательный ARP-запрос. Узел с искомым IP-адресом назначения принимает его и в ARP-ответе сообщает свой Ethernet-адрес. Эти действия соответствуют обычному протоколу ARP, описанному выше.

Протокол ARP с представителем основан на том, что шлюзы работают как представители удаленных узлов. Предположим, в подсети 128.6.5 (см. рис.3.8) имеется узел 128.6.5.2 (машина А). Она желает послать IP-пакет машине 128.6.4.194, которая подключена к другой сети Ethernet (машина В в подсети 128.6.4). Существует шлюз с IP-адресом 128.6.5.1, соединяющий эти две подсети (шлюз R).

Если в ARP-таблице машины А нет маршрута доступа к машине В, то машина А посылает ARP-запрос машине В. Фактически машина А спрашивает: "Если Вы - машина с IP-адресом 128.6.4.194, то сообщите мне Ваш Ethernet-адрес". Машина В не может ответить на запрос самостоятельно, т.к. она подключена к другой сети Ethernet и никогда даже не увидит этот ARP-запрос. Однако шлюз R может работать от его имени. Шлюз R отвечает: "Я здесь, IP-адресу 128.6.4.194 соответствует Ethernet-адрес 2:7:l:0:EB:CD", где 2:7:l:0:EB:CD в действительности является Ethernet-адресом шлюза. Машина А теперь считает, что машина 128.6.4.194 подключена непосредственно к той

же локальной среде Ethernet, что и она имеет Ethernet-адрес 2:7:l:0:EB:CD. Теперь всегда, когда есть IP-пакет, который должен быть послан по адресу 128.6.4.194, машина будет использовать указанный Ethernet-адрес. Поскольку это адрес шлюза, он забирает этот пакет и переправляет его по назначению.

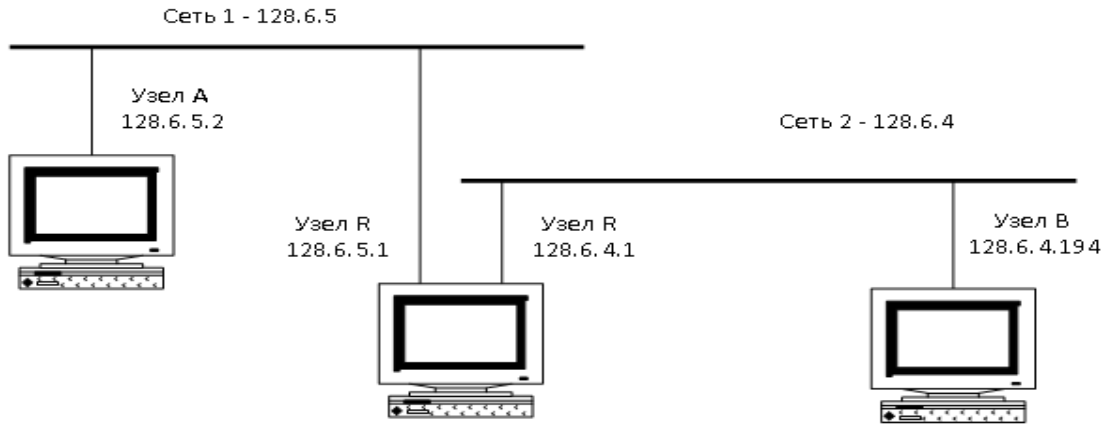


Таблица маршрутов.

Адрес назначения	Флаг вида маршрутизации	Шлюз	Адаптер
128.6.4.194	косвенная	128.6.5.1	lan0

Рис. 3.8

Заметим, что полученный эффект такой же, как если бы в таблице маршрутов была запись (рис. 3.8) за исключением того, что маршрутизация выполняется на уровне модуля ARP, а не модуля IP.

Вообще для целей маршрутаризации лучше использовать таблицу маршрутов. Это то, для чего она создана. Однако есть несколько случаев, когда протокол ARP с представителем очень полезен:

- в IP-сети есть машина, которая не умеет работать с подсетями;
- в IP-сети есть машина, которая не может соответствующим образом реагировать на сообщения перенаправления;
- нежелательно выбиратъ какой-либо шлюз как маршрут по умолчанию;
- программное обеспечение не способно восстанавливаться при сбоях на маршрутах.

Иногда протокол ARP с представителем выбирают из-за удобства, т.к. он упрощает работу по начальной установке таблицы маршрутов. Простейший способ начальной установки - определить маршрут по умолчанию, то есть использовать команду типа "route add defallt ...", как в ОС UNIX. При изменении IP-адреса шлюза эту команду приходится менять во всех машинах.

Если же использовать протокол ARP с представителем, т.е. в команде установки маршрута по умолчанию указать метрику 0, то при замене IP-адреса шлюза команду начальной установки менять не придется, так как протокол ARP с представителем не требует явного задания IP-адресов шлюзов. Любой шлюз может ответить на ARP-запрос.

Для того чтобы избавить пользователей от обязательной начальной установки маршрутов, некоторые реализации TCP/IP используют протокол ARP с представителем по умолчанию в тех случаях, когда не находят подходящих записей в таблице маршрутов.

3.5. Протокол ICMP

Если происходит "нештатное" событие, например потеря пакета, то для его обработки инициализируется специальный протокол, называемый Internet Control Message Protocol (ICMP). Он призван выяснить природу ошибки, сформировать сообщение о ней и передать его приложению, сформировавшему пакет. Протокол выполняет следующие основные функции:

- обмен тестовыми сообщениями для выяснения наличия и активности узлов сети (примером может служить команда ping);
- анализ достижимости узла получателя, сброс пакетов, направляемых к недостижимым узлам;
- оценка работоспособности удаленных узлов сети;
- изменение маршрутов и коррекция таблиц маршрутизации узлов, выполняющих функции шлюзов и маршрутизаторов (redirect);
- уничтожение пакетов с истекшим временем жизни;
- синхронизация времени в узлах сети;
- управление потоком (путем регулирования частоты посылки пакетов узлами-отправителями).

Одной из наиболее интересных среди перечисленных функций является изменение маршрутов: если некоторый маршрутизатор определяет, что хост использует неоптимальный путь для доставки пакета, он при помощи протокола ICMP корректирует маршрутную таблицу хоста. Это один из механизмов автоматической оптимизации доставки информации и адаптации сетей TCP/IP

к изменениям топологии. Функции протокола ICMP проиллюстрированы на рис. 3.9.

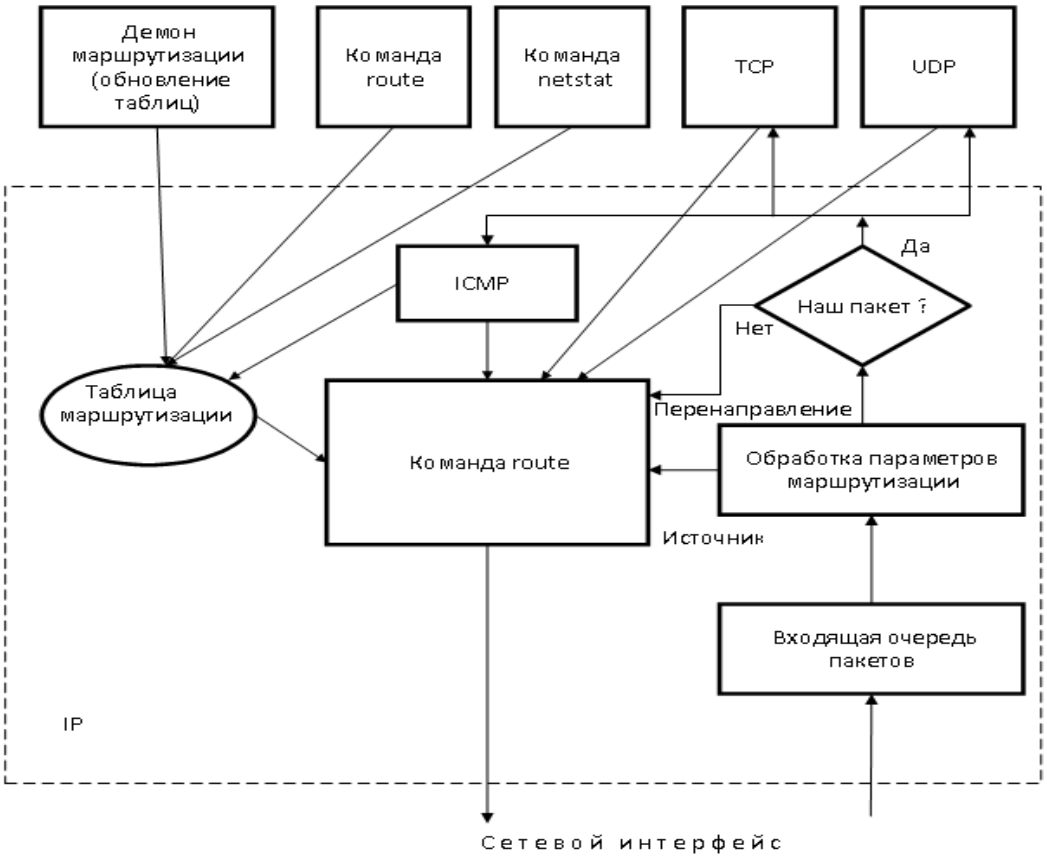


Рис. 3.9

3.6. Протоколы маршрутизации в IP-сетях

Важнейшая функция протокола IP - маршрутизация. Блок-схема машины, реализующей маршрутизацию, показана на рис. 3.9. Центральным элементом этой схемы является маршрутный вычислитель. Ему на вход поступают пакеты: от вышележащих протоколов (TCP, UDP), от протокола ICMP (например, контрольное сообщение о потере пакета), из входящей очереди IP, лежащей непосредственно “над” сетевым интерфейсом.

Маршрутизатор работает со специальной структурой данных - таблицей маршрутизации (routing table), которая указывает, куда нужно перенаправлять пакет с заданным адресом, т.е. отдать некоторому хосту (прямая маршрутизация) или некоторому маршрутизатору (косвенная маршрутизация) или передать в некоторую подсеть. При этом в таблице маршрутизации специфицируется и непосредственно физический сетевой интерфейс по ARP-таблице, через который должен быть передан пакет (у выполняющей маршрутизацию машины таких сетевых интерфейсов может быть несколько).

Различают статическую и динамическую маршрутизации в ИС. Статическая, основывающаяся на жестко заданных маршрутных таблицах, наиболее приемлема для хостов, работающих в небольших сетях, поскольку хост не должен тратить много “сил” на задачи маршрутизации. Часто таблицу маршрутизации на нем конфигурируют как небольшой список, состоящий из хостов-соседей и маршрутизатора по умолчанию (default gateway), которому хост и отдает все пакеты, маршрутизация которых у него вызывает какие-либо затруднения. Эта же логика работает и на уровне маршрутизаторов: маршрутизатор, обслуживающий локальную сеть, обязан знать все о своей локальной сети, однако информацией о внешних сетях он может и не владеть, обращаясь при необходимости к владеющим этой информацией внешним маршрутизаторам. В сетях TCP/IP маршрутизация осуществляется на основе неполной информации.

Замечателен набор средств динамической маршрутизации в сетях TCP/IP, позволяющих конкретному хосту (или маршрутизатору), взаимодействуя со смежными узлами, обновлять и корректировать информацию в таблицах маршрутизации. Исполняет протоколы динамической маршрутизации программа, называемая маршрутизационным демоном (демоном в ОС UNIX называют процесс, запускаемый при загрузке ядра операционной системы и работающий в фоновом режиме вплоть до выключения машины).

Прежде чем решать задачу маршрутизации, как поиска оптимального пути доставки информации, следует установить, с кем можно связаться вообще. Эта задача становится весьма нетривиальной в силу разделения маршрутизаторов на базовые (core gateways), подключаемые непосредственно к магистральным каналам, и прочие, обслуживающие произвольным образом соединенные подсети, называемые в мире IP автономными системами (autonomous systems). Она не всегда решается просто, поскольку маршрутизация в сетях IP осуществляется на основе неполной информации и достижимость (reachability) некоторого узла не очевидна. Задачи распространения информации до достижимости узлов сети решают два протокола:

- Exterior Gateway Protocol (EGP);
- Border Gateway Protocol (BGP).

Для обеспечения маршрутизации мало знать, что данный узел достижим. Необходимо найти и по мере возможности оптимизировать путь к нему. Эту задачу в сетях TCP/IP решает ряд протоколов динамической маршрутизации, которые по внутренней логике можно разделить на два основных класса:

- основанные на подсчете промежуточных ретрансляций (hop count);

- основанные на знании полной топологии ИС (так называемые SPF-протоколы, от Shortest Path First).

Протоколы на основе подсчета промежуточных ретрансляций производят вычисление наиболее коротких путей распространения, определяя число промежуточных маршрутизаторов на пути распространения пакета от данного маршрутизатора до получателя в ИС. На основе подсчета ретрансляций работают такие распространенные протоколы динамической маршрутизации, как Gateway-to-Gateway Protocol (GGP), обеспечивающий обмен информацией между мощными базовыми маршрутизаторами (core, или exterior gateways), подключаемыми непосредственно к магистральным каналам сетей (backbones) или Routing Information Protocol (RIP), решающий ту же задачу для менее значимых маршрутизаторов (Interior gateways).

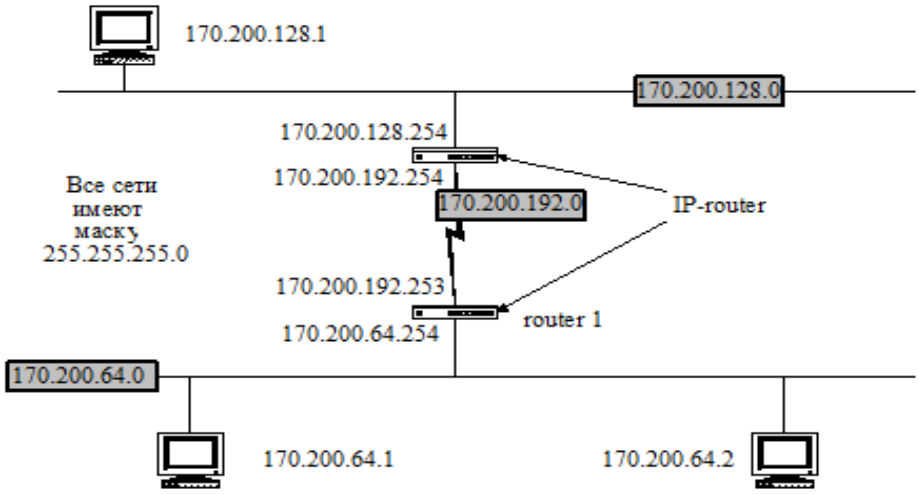
К протоколам второго типа следует отнести протокол HELLO, вычисляющий оптимальный путь на основе измерения задержки распространения пакетов, и развитой протокол OSPF (Open SPF), который содержит ряд дополнительных возможностей. Кроме протоколов RIP и OSPF (NLSP в версии Novell), используются другие протоколы, например IS-IS.

3.6.1. Особенности протокола RIP

IP маршрутизаторы соединяют сети и выборочно переносят IP-дейтаграммы между сетевыми сегментами. Сети, присоединённые непосредственно, сконфигурированы в интерфейсах маршрутизатора наложением на IP-адреса маски подсети. Сети, которые не присоединены непосредственно к маршрутизатору, не конфигурируются в его таблице маршрутизации, но их существование обнаруживается с помощью протокола маршрутизации. Простейшим протоколом маршрутизации, используемым с IP, является RIP.

Таблицы маршрутизации. IP-маршрутизатор, используя RIP, принимает сетевую информацию от соседей, т.е. других машрутизаторов, с которыми имеет общий сетевой номер (расположены на одном сегменте). Пример использования протокола RIP приведен на рис. 3.10. Он использует пакеты RIP-обновлений для построения таблицы маршрутизации. Таблица содержит одну строку для каждой сети, о которой знает маршрутизатор. Строка содержит IP-адрес маршрутизатора, который является следующим и перехватит пакет для направления его в сеть назначения, а также количество сегментов до получателя. При очередном обновлении таблицы маршрутизации в нее могут быть добавлены новые элементы. Если обновление “находит” лучший путь до сети, которая существует в таблице, то маршрутизатор замещает элемент таб-

лицы новой информацией. Новый маршрут до сети, который имеет то же количество сегментов, что и в таблице маршрутизации, не добавляется.



Адрес назначения	кол-во сегментов	Шлюз	Номер адаптера
170.200.64.0	1	170.200.64.254	1
170.200.128.0	2	170.200.192.254	2
170.200.192.0	1	170.200.192.253	2

Рис. 3.10

Каждый маршрутизатор посылает обновление по своему собственному расписанию и эти обновления не синхронизируются с расписаниями других маршрутизаторов. Информация о доступности сети распространяется через один сегмент в единицу времени. RIP и другие дистанционно-векторные протоколы иногда называют “основанными на слухах”.

Локально соединённые сети будут всегда показываться первыми в таблице маршрутизации. Даже после приёма маршрутизатором информации обновления от соседей, он имеет информацию относительно сетей, которые присоединены непосредственно. Протокол RIP требует, чтобы все узлы в сети имели одинаковые адреса подсетей. Это необходимо потому, что пакет RIP-обновления включает в себя только номер сети и количество сегментов (D, V), и маршрутизатор, принимая RIP-обновление, использует собственный адрес подсети для определения объявленного адреса сети.

Изменения в сетевых возможностях. При использовании протоколов дистанционно-векторной маршрутизации, основанных на слухах, информация об уже потерянных связях может держаться в таблицах маршрутизации долгое время, прежде чем до них дойдут “слухи” и пройдут обновления таблиц. Это называется медленным слиянием и является одним из наибольших

недостатков этого типа маршрутизации. С целью уменьшения недостатков этого протокола, в последующие годы в протоколе RIP были сделаны некоторые изменения.

Разделение горизонта. Правило разделения горизонта определяет, что маршрутизатор помнит, из какого сегмента приняты данные маршруты. Когда маршрутизатор обновляет информацию в таблице маршрутизации, он посылает ее во все свои сегменты, но пропускает любые маршрутизаторы, о которых он узнал от своих соседей на сегменте. Важная причина для настройки разделения горизонта - это избежать зацикливания маршрутов. Циклы возникают, когда маршрутизатор опирается на сетевую информацию от соседа, который маршрутизирует сеть через маршрутизатор, принявший обновление. Маршрутизаторы, использующие правило разделения горизонта, исключают размножение маршрутизирующей информации, которая может содержать противоречия.

Отравленный возврат. Отравленный возврат иногда комбинируется с разделённым горизонтом. Отравленный возврат позволяет маршрутизаторам опознавать потерянные связи и соединяться намного быстрее, чем в методе разделенного горизонта. Вместо пропускания маршрутов, узнанных от принятых обновлений соседей, маршруты принимаются от маршрутизатора, пославшего информацию со счётчиком количества сегментов (или ценой), равным бесконечности.

По умолчанию IP позволяет пакетам маршрутизации иметь максимальную цену или “время жизни” в 1 сегмент. Представление сети с количеством сегментов 16 (на один сегмент больше жизни пакета) сигнализирует, что сеть перегружена или недоступна.

Основными недостатками этого протокола являются:

- увеличение трафика ИС при периодической рассылке широковещательных пакетов;
- неоптимальность найденного маршрута и простой имеющихся сетевых путей;
- версия 1 не может маршрутизировать подсети.

Для маршрутизации подсетей разработана и выпущена версия 2 протокола – RIPv2. Структура пакетов протоколов RIP и RIPv2 приведена на рис. 3.11.

8	16	32bits
Command	Version	Unused
Address family identifier		Route tag (only for RIP2; 0 for RIP)
IP address		
Subnet mask (only for RIP2; 0 for RIP)		
Next hop (only for RIP2; 0 for RIP)		
Metric		

Рис. 3.11

3.6.2. Протокол OSPF

Более совершенным является новый протокол маршрутизации OSPF (Open Shortest Path First), который основан на построении каждым маршрутизатором графа связей в сети с учетом их состояний и на выборе кратчайшего пути на графе связей. Маршрутизаторы в этом протоколе обмениваются информацией о состоянии связей только при их изменении, что сокращает объем передаваемого трафика ИС. Этот протокол позволяет находить оптимальные пути в ИС, однако требует проведения более сложных математических операций.

Протокол OSPF содержит ряд дополнительных возможностей, включая:

- вычисление оптимальных маршрутов для различных типов сервиса;
- защиту узлов на основе простой парольной системы;
- выравнивание нагрузки сети (load balancing);
- маршрутизацию в подсетях;
- минимизацию широковещательных вызовов для поиска информации;
- поддержку администрирования виртуальных сетей, связность которых абстрагирована от физической природы сетевых каналов.

Открытие кратчайших путей первыми. Протокол маршрутизации OSPF является альтернативой RIP. Он основан на алгоритме состояния связей и более эффективен, чем RIP в передаче маршрутизирующей информации, а также лучше адаптируется к изменениям топологии сети. OSPF лучше приспособлен к большим, более комплексным IP-сетям с различными скоростями связей.

Обновления состояния связей. Маршрутизация посредством алгоритма состояния связей позволяет лучше использовать доступные линии связи посредством назначения одного маршрутизатора ответственным за посылку обновлений в другие маршрутизаторы. Все маршрутизаторы, использующие OSPF являются членами multicast-группы. Это позволяет одному маршрутиза-

тору заведовать группой маршрутизаторов и одновременно обновлять их таблицы маршрутизации. OSPF-обновления возникают менее часто, чем RIP, обычно каждые 30 минут или когда маршрутизатор обнаруживает изменение в состоянии одной из его связей.

Маршрутизаторы должны содержать согласованную информацию о топологии для построения согласованных рабочих баз данных топологии сети во всех маршрутизаторах. Таблицы маршрутов состояния связей содержат больше информации, чем дистанционно-векторные таблицы маршрутов. Один из недостатков протоколов маршрутизации состояния связей - это интенсивное использование памяти.

Области OSPF. В целях ограничения баз данных топологии ИС очень большие сети можно разбить на области. Каждая область ИС имеет выделенный маршрутизатор, который отвечает за обновление в одной области и может быть административно выбран посредством настройки на высочайший приоритет. База данных о топологии ИС в выделенной области не содержит данные о топологии в других областях.

Области OSPF являются группами IP-подсетей и маршрутизаторов, которые идентифицируются 32-битовыми IP-адресами (на рис. 3.12 IP-адреса 10.10.10.10, 20.20.20.20, 30.30.30.30). Если имеется более чем одна область, то должен быть указан специальный тип области, называемый OSPF область-backbone, которая всегда имеет IP- адрес 0.0.0.0. (OSPF ID). Backbone-область может быть собрана на основе любой комбинации сетей и должна быть разработана для высокой производительности и надёжности. Области OSPF присоединяются к backbone через маршрутизаторы границ областей. Они настроены для межобластной маршрутизации. Области, которые не имеют прямой связи с backbone, настраиваются на виртульные связи через другие области.

Дополнительно к своему IP-адресу каждый маршрутизатор имеет 32-битовый OSPF ID-маршрутизатора, выраженный также в десятично-точечном формате.

Быстрое слияние. Во всех маршрутизаторах состояния связей согласованы с топологией сети. Когда происходит изменение в сетевых возможностях, назначенный маршрутизатор немедленно уведомляет об этом все маршрутизаторы. Уведомление ограничено областью, где это изменение произошло.

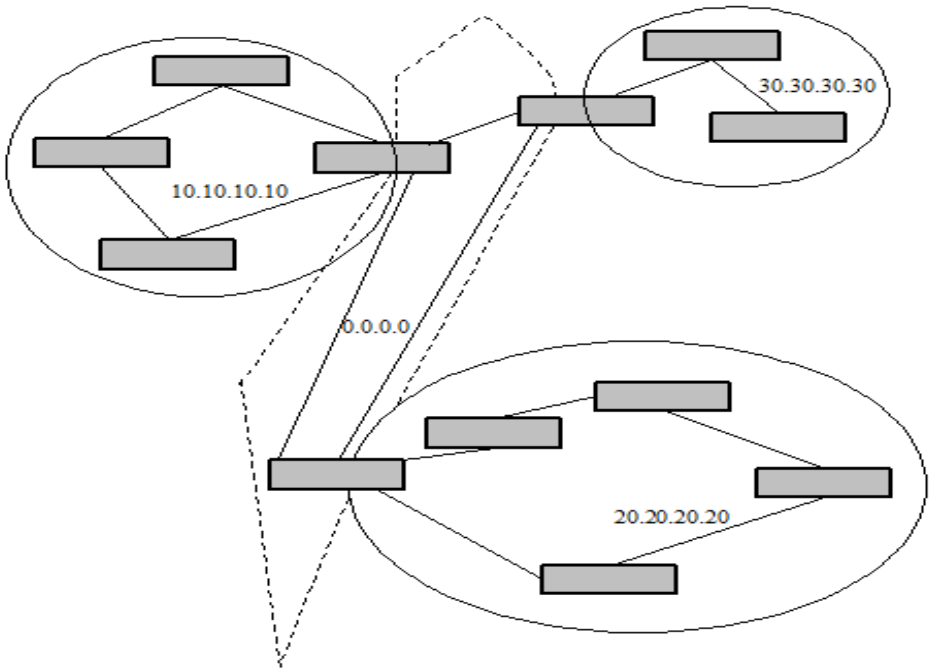


Рис. 3.12

Подлинность. OSPF маршрутизаторы в области могут быть настроены на посылку обновлений состояний связи, которые содержат пароль, запрещающий повреждение таблицы маршрутизации от неизвестных обновлений. Протокол OSPF часто значительно лучше, чем RIP, но он более сложен в настройке.

Различные подсетевые маски. Если сетевые сегменты, которые используют локальные схемы типа Ethernet, Token Ring и FDDI, нуждаются в поддержке нескольких хостов, то широкие области связей типа точка-точка нуждаются в поддержке только двух конечных точек. IP-сети, которые используют RIP как маршрутизирующий протокол, требует и одинаковых масок подсетей на обоих типах сегментов, в результате чего появляется большой блок неиспользуемых хост-адресов. OSPF посылает пакеты обновления состояния связей, которые содержат оба сетевых номера и маски подсетей. Это позволяет подсетевой маске в WAN-связях устанавливать 30 бит для сети/подсети и 2 бита для поля хоста.

На рис. 3.13 порты маршрутизатора, которые общаются с локальным сегментом, связаны с маской подсети, которая содержит 254 хоста (255.255.255.0). Интерфейсы маршрутизатора, которые общаются с WAN-связями, имеют маску подсети которая содержит 2 хоста (255.255.255.252).

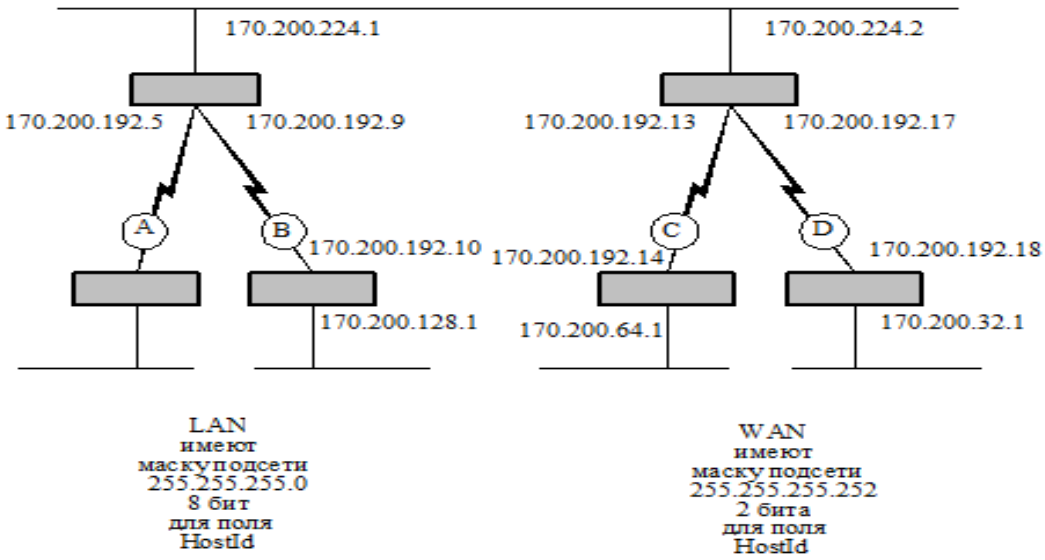


Рис. 3.13

Структура пакетов OSPF приведена на рис. 3.14.

8	16	32bits
Version No.	Packet Type	Packet length
Router ID		
Area ID		
Checksum		AU type
Authentication		

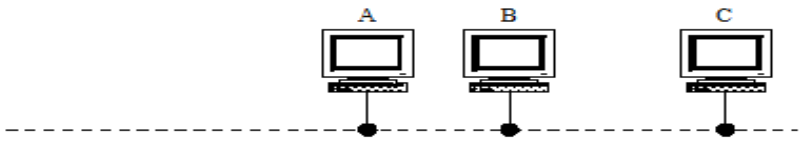
Рис. 3.14

3.6.3. Маршрутизация протокола IP.

Модуль IP является базовым элементом технологии Internet, а центральной частью модуля IP является его таблица маршрутизации. Протокол IP использует эту таблицу при принятии всех решений о маршрутизации IP-пакетов. Содержание таблицы маршрутов определяет администратор сети. Ошибки при установке маршрутов могут заблокировать все передачи. Чтобы успешно администрировать IP-сети, необходимо хорошо представлять, каким образом используется таблица маршрутизации.

Прямая маршрутизация. На рис. 3.15 показана небольшая сеть Internet, состоящая из 3 машин: A, B и C. Каждая машина имеет такой же стек протоколов TCP/IP. Каждый сетевой адаптер этих машин имеет свой Ethernet-адрес. Администратор ИС должен присвоить машинам уникальные IP-адреса, а также присваивает среде Ethernet сетевой IP-номер.

а)



Ethernet 1
IP - сеть "development"

б)

Адрес	Отправитель	Получатель
IP-заголовок	A	B
Ethernet-заголовок	A	B

Рис. 3.15

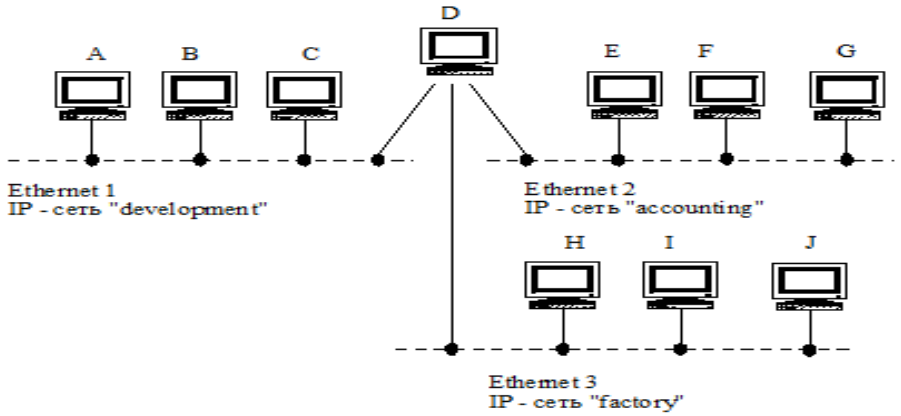
Когда машина A посылает IP-пакет машине B, то заголовок IP-пакета содержит в поле отправителя IP-адрес машины A, а заголовок Ethernet-кадра содержит в поле отправителя Ethernet-адрес A. Кроме этого, IP-заголовок содержит в поле получателя IP-адрес машины B, а Ethernet-заголовок содержит в поле получателя Ethernet-адрес машины B.

В этом простом примере у протокола IP есть излишество, так как он мало что добавляет к услугам, предоставляемым средой Ethernet. Однако протокол IP требует дополнительных затрат времени процессора и пропускной способности среды передачи данных на создание, передачу и обработку IP-заголовка. Когда в машине B модуль IP получает IP-пакет от машины A, он сопоставляет IP-адрес места назначения со своим и, если адреса совпадают, то передает дейтаграмму протоколу верхнего уровня. В данном случае при взаимодействии машины A с машиной B используется прямая маршрутизация.

Косвенная маршрутизация. На рис. 3.16 представлена более реалистичная картина сети Internet. Она состоит из трех сетей Ethernet, на базе которых работают три сети, объединенные машиной D, называемой шлюзом. В каждой IP-сети по четыре машины, каждая машина имеет собственные IP-адрес и Ethernet-адрес.

За исключением D, все машины имеют стандартный стек протоколов TCP/IP. Машина D - шлюз; она соединяет все три сети и, следовательно, имеет три IP-адреса и три Ethernet-адреса. Машина D имеет стек протоколов TCP/IP, в котором вместо двух модулей ARP и двух драйверов, имеется три модуля ARP и три драйвера Ethernet. Обратим внимание на то, что машина D имеет только один модуль IP.

а)



б)

Адрес	Отправитель	Получатель
IP-заголовок	A	E
Ethernet-заголовок	A	D

в)

Адрес	Отправитель	Получатель
IP-заголовок	A	E
Ethernet-заголовок	D	E

Рис. 3.16

Администратор ИС присваивает каждой сети Ethernet уникальный номер, называемый IP-номером сети. Вместо IP-номеров могут использоваться имена сетей. Когда машина A посылает IP-пакет машине B, то процесс передачи идентичен приведенному выше в примере для одной сети. Любое взаимодействие между машинами, подключенными к одной IP-сети, использует прямую маршрутизацию, рассмотренную в предыдущем примере.

Когда машина D взаимодействует с машиной A, то это прямое взаимодействие. Когда машина D взаимодействует с машиной E, это тоже прямое взаимодействие. Когда машина C взаимодействует с машиной H, это прямое взаимодействие. Это так, поскольку каждая пара этих машин принадлежит одной IP-сети. Однако, когда машина A взаимодействует с машинами, включенными в другую IP-сеть, то взаимодействие уже не будет прямым. Машина A должна использовать шлюз D для ретрансляции IP-пакетов в другую IP-сеть. Такое взаимодействие называется “косвенным”.

Маршрутизация IP-пакетов выполняется модулями IP и является прозрачной для модулей TCP, UDP и прикладных процессов.

Если машина A посылает машине E IP-пакет, то IP-адрес и Ethernet-адрес отправителя есть адреса машины A. IP-адрес места назначения является адресом машины E, но поскольку модуль IP в машине A посылает IP-пакет через машину D, Ethernet-адрес места назначения является Ethernet-адресом

машины D. Модуль IP в машине D получает IP-пакет и проверяет IP-адрес места назначения. Определив, что это не его IP-адрес, шлюз D посылает этот IP-пакет прямо к машине E.

Итак, при прямой маршрутизации IP и Ethernet-адреса отправителя соответствуют адресам того, кто послал IP-пакет, а IP- и Ethernet-адреса места назначения соответствуют адресам получателя. При косвенной маршрутизации IP- и Ethernet-адреса не образуют таких пар.

В данном примере сеть Internet является очень простой. Реальные ИС могут быть гораздо сложнее, так как могут содержать несколько шлюзов и несколько типов физических сред передачи. В приведенном примере несколько сетей Ethernet объединяются шлюзом для того, чтобы локализовать широковещательный трафик в каждой сети.

Правила маршрутизации в модуле IP. Рассмотрим правила или алгоритм маршрутизации, используемые модулем IP. Для отправляемых IP-пакетов, поступающих от модулей верхнего уровня, модуль IP должен определить способ доставки - прямой или косвенный - и выбрать сетевой адаптер. Этот выбор делается с помощью таблицы маршрутов.

Для принимаемых IP-пакетов, поступающих от сетевых драйверов, модуль IP должен решить, нужно ли ретранслировать IP-пакет по другой сети или передать его на верхний уровень. Если модуль IP решит, что IP-пакет должен быть ретранслирован, то дальнейшая работа с ним осуществляется также, как с отправляемыми IP-пакетами. Входящий IP-пакет никогда не ретранслируют через тот же сетевой адаптер, через который он был принят. Решение о маршрутизации принимают до того, как IP-пакет передается сетевому драйверу, и до того, как происходит обращение к ARP-таблице.

Выбор адреса. Одно из важнейших решений, которое необходимо принять при установке сети, заключается в выборе способа присвоения IP-адресов вашим узлам. Этот выбор должен учитывать перспективу роста сети. Иначе в дальнейшем вам придется менять адреса. Когда к сети подключено несколько сотен рабочих станций, то изменение адресов становится почти невозможным.

Имена. Пользователям удобнее называть машины именами, а не числами. Например, у машины по имени alpha может быть IP-адрес 223.1.2.1. В маленьких сетях информация о соответствии имен IP-адресам хранится в файлах "hosts" на каждой машине. Конечно, название файла зависит от конкретной реализации. В больших сетях эта информация хранится на сервере и доступна по сети. Несколько строк из файла “hosts" могут выглядеть примерно так:

223.1.2.1 alpha

223.1.2.2 beta

223.1.2.3 gamma

223.1.2.4 delta

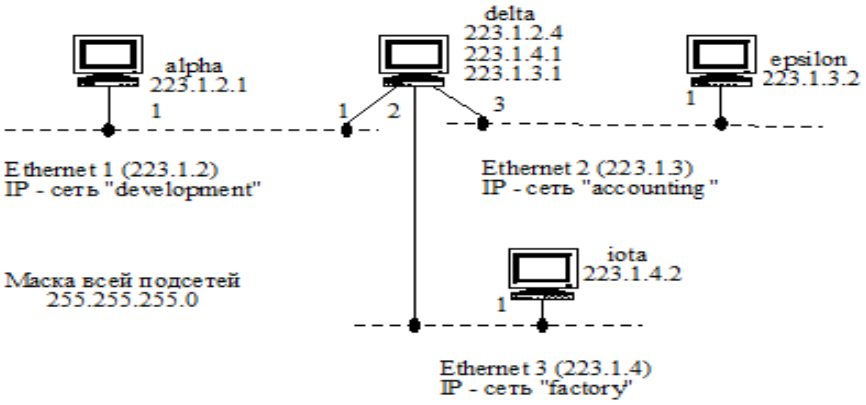
223.1.3.2 epsllon

223.1.4.2 iota

В первом столбце -IP-адрес, во втором - название машины.

В большинстве случаев файлы “hosts" могут быть одинаковы на всех машинах. Заметим, что о машине delta в этом файле есть всего одна запись, хотя она имеет три IP-адреса (рис. 3.17). Машина delta доступна по любому из ее IP-адресов. Какой из них используется, не имеет значения. Когда машина delta получает IP-пакет и проверяет IP-адрес места назначения, то она опознает любой из трех своих IP-адресов.

а)



б)

Сеть	Флаг вида маршрутизации	Шлюз	Номер адаптера
development	прямая	<пусто>	1
accounting	косвенная	devnetrouter	1
factory	косвенная	devnetrouter	1

в)

Сеть	Флаг вида маршрутизации	Шлюз	Номер адаптера
223.1.2	прямая	<пусто>	1
223.1.3	косвенная	223.1.2.4	1
223.1.4	косвенная	223.1.2.4	1

Рис. 3.17

IP-сети также могут иметь имена. Если у вас есть три IP-сети, то файл "networks" может выглядеть примерно так:

223.1.2 development

223.1.3 accounting

223.1.4 factory

В первой колонке - номер IP-сети, во второй - имя сети.

В данном примере alpha - это машина с номером 1 в сети development, beta - машина с номером 2 в сети development и т.д.

Показанный выше файл hosts удовлетворяет потребности пользователей, но администратор сети, возможно, заменит строку, относящуюся к delta на следующее:

```
223.1.2.4   devnetrouter   delta
223.1.3.1   accnetrouter
223.1.4.1   facnetrouter
```

Эти три строки файла hosts сопоставляют каждому IP-адресу машины delta смысловые имена.

Фактически, первый IP-адрес имеет два имени: "devnetrouter" и "delta", которые являются синонимами. На практике имя “delta” используется как общеупотребительное имя машины, а остальные три имени используют только для сопровождения IP-таблицы маршрутизации.

Файлы hosts и networks используются командами администрирования и прикладными программами при работе со смысловыми именами. Они не нужны собственно для работы сети Internet, но облегчают ее использование.

IP-таблица маршрутов. Как модуль IP узнает, какой именно сетевой адаптер нужно использовать для отправления IP-пакета? Модуль IP ищет его в таблице маршрутов. Ключом поиска служит номер IP-сети, выделенный из IP-адреса места назначения IP-пакета. Таблица маршрутов содержит по одной строке для каждого маршрута. Основными столбцами таблицы маршрутов являются:

- номер IP-сети;
- флаг прямой или косвенной маршрутизации;
- IP-адрес шлюза;
- номер сетевого адаптера.

Эта таблица используется модулем IP при обработке каждого отправляемого IP-пакета. В большинстве систем таблица-маршрутов может быть изменена с помощью команды "route”. Содержание таблицы маршрутов определяется администратором ИС, поскольку администратор присваивает машинам IP-адреса.

Особенности прямой маршрутизации. Для того чтобы объяснить, как используют таблицу маршрутов, разберем подробно те случаи маршрутизации, которые встречались ранее. Подобную картину можно увидеть на некоторых UNIX-системах по команде “netstat - r". На рис. 3.17 представлена простая IP-сеть. В этой простой сети у всех машин одинаковые таблицы маршру-

тов. Для сравнения на рис. 3.17,в представлена та же таблица маршрутизации, но вместо имени ИС указан ее номер.

Порядок прямой маршрутизации. Машина alpha посылает IP-пакет машине beta. Этот пакет находится в молуле IP машины alpha, и IP-адрес места назначения равен IP-адресу beta (223.1.2.2). Модуль IP с помощью маски подсети выделяет номер сети из IP-адреса и ищет соответствующую ему строку в таблице маршрутов. В данном случае подходит первая строка.

Остальная информация в найденной строке указывает на то, что машины этой сети доступны напрямую через адаптер с номером 1. С помощью ARP-таблицы IP-адрес машины beta преобразуют в соответствующий Ethernet-адрес, и через адаптер с номером 1 Ethernet-фрейм посылают машине beta.

Если прикладная программа пытается послать данные по IP-адресу, которого нет в сети development, то модуль IP не сможет найти соответствующую запись в таблице маршрутов. В этом случае модуль IP отбрасывает IP-пакет. Некоторые реализации протокола возвращают сообщение об ошибке "Сеть не доступна".

Особенности косвенной маршрутизации. Теперь рассмотрим подробнее более сложный порядок маршрутизации в IP-сети.

Таблица маршрутов у машины alpha приведена на рис. 3.18,б. Таблица на рис. 3.18,в та же, что и таблица на рис. 3.18,б, но с IP-адресами вместо названий.

В столбце "шлюз" таблицы маршрутов на машине alpha указан IP-адрес точки соединения машины delta с сетью development.

а)

б)

Сеть	Флаг вида маршрутизации	Шлюз	Номер адаптера
development	прямая	<пусто>	1

в)

Сеть	Флаг вида маршрутизации	Шлюз	Номер адаптера
223.1.2	прямая	<пусто>	1

Рис. 3.18

Порядок косвенной маршрутизации. Машина alpha посылает IP-пакет машине epsilon (см. рис. 3.17). Этот пакет находится в модуле IP на машине alpha, и IP-адрес места назначения равен IP-адресу машины epsilon (223.1.3.2). Модуль IP выделяет сетевой номер из этого IP-адреса (223.1.3) и ищет соответствующую ему строку в таблице маршрутов. Соответствие находится во второй строке. Запись в этой строке указывает на то, что машины сети с номером 223.1.3 доступны через шлюз devnetrouter. Модуль IP на машине alpha определяет с помощью ARP-таблицы Ethernet - адрес, соответствующий IP-адресу шлюза devnetrouter. Затем IP-пакет, содержащий IP-адрес места назначения epsilon, посылается через сетевой адаптер с номером 1 машине devnetrouter.

IP-пакет принимает адаптер машины delta и передает модулю IP. Там производится проверка IP-адреса места назначения, и, поскольку он не соответствует ни одному из собственных IP-адресов машины delta, шлюз решает ретранслировать IP-пакет.

Модуль IP в машине delta выделяет сетевой номер из IP-адреса места назначения этого IP-пакета (223.1.3) и ищет соответствующую запись в таблице маршрутов. Таблица маршрутов на машине delta приведена на рис. 3.19,а.

а)

Сеть	Флаг вида маршрутизации	Шлюз	Номер адаптера
development	прямая	<пусто>	1
accounting	прямая	<пусто>	3
factory	прямая	<пусто>	2

б)

Сеть	Флаг вида маршрутизации	Шлюз	Номер адаптера
223.1.2	прямая	<пусто>	1
223.1.3	прямая	<пусто>	3
223.1.4	прямая	<пусто>	2

Рис. 3.19

На рис. 3.19,б - та же таблица маршрутов, но с IP-адресами вместо имен. Соответствие находится во второй строке. Теперь модуль IP напрямую посылает IP-пакет машине epsilon через адаптер с номером 3. Пакет содержит IP- и Ethernet-адреса места назначения, равные epsilon.

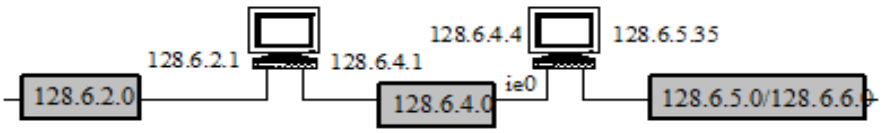
Машина epsilon принимает IP-пакет, и ее модуль IP проверяет IP-адрес места назначения. Обнаружив, что он совпадает с IP-адресом epsilon, модуль передает содержащееся в этом IP-пакете сообщение протокольному модулю верхнего уровня.

Установка маршрутов. До сих пор мы рассматривали то, как используется таблица маршрутов для маршрутизации IP-пакетов. Рассмотрим методы, позволяющие поддерживать корректность таблиц маршрутов.

3.6.4 Статическая и динамическая маршрутизация в ИС

Простейший способ проведения маршрутизации состоит в установке маршрутов при запуске системы с помощью специальных команд. Этот метод можно применять в относительно небольших IP-сетях, в особенности, если их конфигурации не часто меняются. На практике большинство машин автоматически формирует таблицы маршрутов. Например, UNIX добавляет записи об IP-сетях, к которым есть непосредственный доступ. Например, ваш стартовый файл может содержать команды (рис. 3.20):

```
ifconfig  ie0  128.6.4.4    netmask    255.255.255.0
ifconfig  ie1  128.6.5.35   netmask    255.255.255.0
```



а)

```
ifconfig   ie0   128.6.4.4   netmask   255.255.255.0
ifconfig   ie1   128.6.5.35  netmask   255.255.255.0
```

б)

Сеть	Флаг вида маршрутизации	Шлюз	Номер адаптера
128.6.4	прямая	<пусто>	ie0
128.6.5	прямая	<пусто>	ie1

в)

```
route   add   128.6.2.0   128.6.4.1   1
route   add   128.6.6.0   128.6.5.35  0
```

г)

Сеть	Флаг вида маршрутизации	Шлюз	Номер адаптера
128.6.2	косвенная	128.6.4.1	ie0
128.6.6	прямая	<пусто>	ie1

Рис. 3.20

Они объявляют о существовании двух сетевых адаптеров и устанавливают их IP-адреса. Система будет автоматически создавать записи, показанные на рис. 3.20,б. Записи определяют, что IP-пакеты для локальных подсетей 128.6.4 и 128.6.5 должны быть посланы через адаптеры, указанные в этих записях. В стартовом файле могут быть команды, определяющие маршруты доступа к другим IP-сетям. Например:

```
route   add   128,6,2.0   128.6.4.1     1
route   add   128.6.6,0   125.6.5.35    0
```

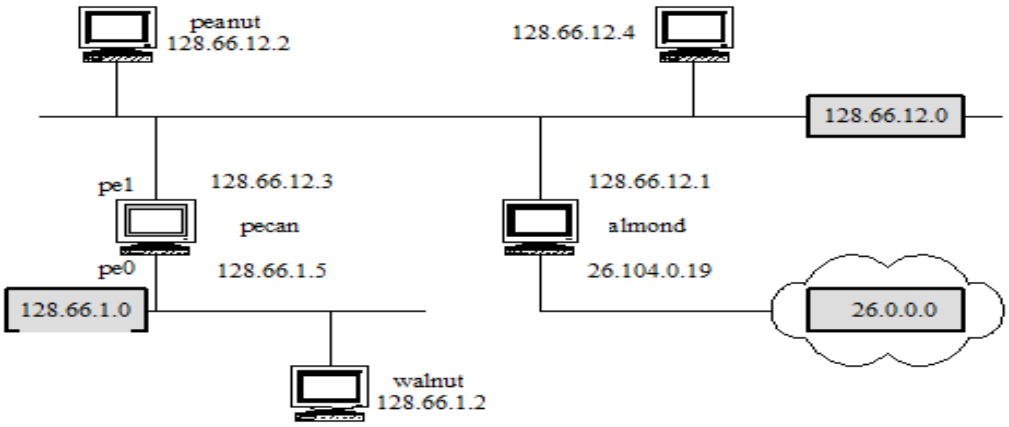
Эти команды показывают, что в таблицу маршрутов должны быть добавлены две записи (рис. 3.20,б). Первый операнд в команде - это IP-адрес сети, которую мы хотим достигнуть. Второй - указывает шлюз, который должен использоваться для доступа к этой IP-сети, а третий - метрика. Метрика показывает, на каком “расстоянии” находится указанная IP-сеть. В данном случае метрика - это количество шлюзов на пути между двумя IP-сетями. Маршруты с метрикой 1 и более определяют первый шлюз на пути к IP-сети. Маршруты с метрикой 0 показывают, что никакой шлюз не нужен - данный маршрут задает дополнительный сетевой номер локальной IP-сети.

Таким образом, команды, приведенные в примере, говорят о том, что для доступа к IP-сети 128.6.2 должен быть использован шлюз 128.6.4.1, а IP-сеть 128.6.6 - это просто дополнительный номер для физической сети, подключенной к адаптеру 128.6.5.35.

Можно определить маршрут по умолчанию, который будет использован в тех случаях, когда IP-адрес места назначения не указан в таблице маршрутов явно. Обычно маршрут по умолчанию указывает IP-адрес шлюза, который имеет достаточно информации для маршрутизации IP-пакетов со всеми возможными адресами назначения.

Если IP-сеть имеет всего один шлюз, тогда все, что нужно сделать, - это установить единственную запись в таблице маршрутов, указав этот шлюз как маршрут по умолчанию. После этого можно не заботиться о формировании маршрутов в других машинах (работа со шлюзом требует более подробного обсуждения - см. рис. 3.21).

Следующие разделы посвящены IP-сетям, где есть несколько шлюзов.



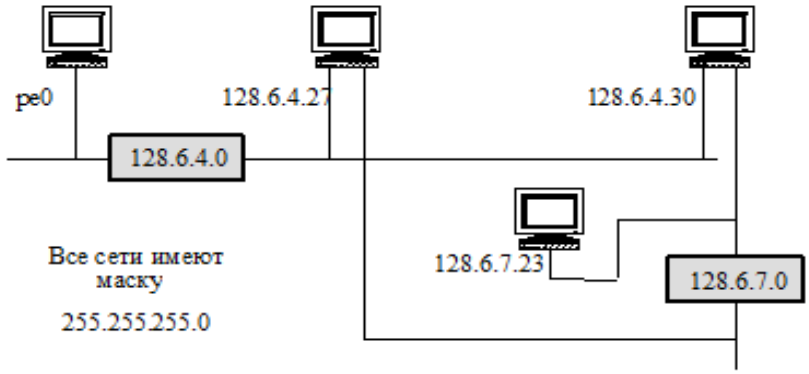
Адрес назначения	Флаг вида маршрутизации	Шлюз	Номер адаптера
127.0.0	прямая	<пусто>	lo0
128.66.1.0	прямая	128.66.1.5	pe0
128.66.12.0	прямая	128.6.12.3	pe1
default	косвенная	128.6.12.1	pe1

Рис. 3.21

Перенаправление маршрутов. Большинство экспертов по сети Internet рекомендуют оставлять решение проблем маршрутизации шлюзам. Плохо иметь на каждой машине большую фиксированную таблицу маршрутов. Дело в том, что при каких-либо изменениях в IP-сети приходится менять информацию во всех машинах. Например, при отключении какого-нибудь канала связи для восстановления нормальной работы нужно ждать, пока кто-то заметит это изменение в конфигурации IP-сети и, администратор внесет исправления во все таблицы маршрутов.

Простейший способ поддержания адекватности маршрутов заключается в том, что изменение таблицы маршрутов каждой машины выполняется по командам только одного шлюза. Этот шлюз должен быть установлен как маршрут по умолчанию. (В ОС UNIX это делается командой “route add default 128.6.4.27 1”, где 128.6.4.27 является IP-адресом шлюза.) Как было описано выше, каждая машина посылает IP-пакет шлюзу по умолчанию в том случае, когда не находит лучшего маршрута. Однако когда в IP-сети есть несколько шлюзов, а таблица маршрутов имеет только одну запись о маршруте по умолчанию, то как использовать другие шлюзы, если это более выгодно? Ответ состоит в том, что большинство шлюзов способны выполнять "перенаправление” IP-пакетов, для которых существуют более выгодные маршруты. "Перенаправление” является специальным типом сообщения протокола ICMP

(Internet Control Message Protocol - протокол межсетевых управляющих сообщений). Сообщение о перенаправлении содержит информацию, которую можно интерпретировать так: "В будущем для IP-адреса XXXX используйте шлюз YYYY, а не меня". Корректные реализации потокола TCP/IP должны использовать сообщения о перенаправлении для добавления записей в таблицу маршрутов. Предположим, таблица маршрутов в начале выглядит как показано на рис. 3.22,а.



а)

Адрес назначения	Флаг вида маршрутизации	Шлюз	Номер адаптера
127.0.0	прямая	<пусто>	lo0
128.6.4	прямая	<пусто>	pe0
default	косвенная	128.6.4.27	pe0

б)

Адрес назначения	Флаг вида маршрутизации	Шлюз	Номер адаптера
128.6.7.23	косвенная	128.6.4.30	pe0

Рис. 3.22

Эта таблица содержит запись о локальной IP-сети 1.28.6,4 и маршрут по умолчанию, указызающий шлюз 128.6.4.27. Допустим, что существует шлюз 128.6.4.30, через который есть лучший путь доступа к IP-сети 128.6.7. Как им воспользоваться? Предположим, что нужно посылать IP-пакеты по IP-адресу 128.6.7.23. Первый IP-пакет пойдет на шлюз 128.6.4.27 по умолчанию, так как это единственный подходящий маршрут, описанный в таблице. Однако, шлюз 128.6.4.27 знает, что существует лучший маршрут, проходящий через шлюз 128.6.4.30. В этом случае шлюз 128.6.4.27 возвращает сообщение перенаправления, где указывает, что IP-пакеты для машины 128.6.7.23 должны посылаться через шлюз 128.6.4.30. Модуль IP на машине-отправителе должен добавить запись в таблицу маршрутов. Все последующие IP-пакеты для машины 128.6.7.23 будут посланы прямо через указанный шлюз.

До сих пор мы рассматривали способы добавления маршрутов в IP-таблицу, а не способы их исключения. Рассмотрим случай, если шлюз будет выключен. Хотелось бы иметь способ возврата к маршруту по умолчанию, после того как какой-либо маршрут разрушен. Однако если шлюз вышел из строя или был выключен, то он уже не может послать сообщение перенаправления. Лучший способ обнаружения неработающих шлюзов основан на выявлении "плохих" маршрутов. Модуль TCP поддерживает различные таймеры, которые помогают ему определить разрыв соединения. Когда случается сбой, то можно пометить маршрут как "плохой" и вернуться к маршруту по умолчанию. Аналогичный метод может использоваться при обработке ошибок шлюза по умолчанию. Если два шлюза отмечены как шлюзы по умолчанию, то машина может использовать их по очереди, переключаясь между ними при возникновении сбоев.

Слежение за маршрутизацией. Заметим, что сообщения о перенаправлении не могут использовать сами шлюзы. Перенаправление - это просто способ оповещения обычной машины о том, что нужно использовать другой шлюз. Сами шлюзы должны иметь полную картину сети Internet и уметь вычислять оптимальные маршруты к каждой подсети. Обычно они поддерживают эту картину, обмениваясь информацией между собой. Для этой цели существуют несколько специальных протоколов маршрутизации, рассмотренных выше. Один из способов, с помощью которого машины могут определять действующие шлюзы, состоит в слежении за обменом сообщениями между ними. Для большинства протоколов маршрутизации существует программное обеспечение, позволяющее обычным машинам осуществлять такое слежение. При этом на машинах поддерживается полная картина положения дел в сети Internet, точно также, как это делается в шлюзах. Программное обеспечение динамически корректирует таблицу маршрутов, что позволяет посылать IP-пакеты по оптимальным маршрутам.

Таким образом, слежение за маршрутизацией в некотором смысле “решает" проблему поддержания корректности таблиц маршрутов. Однако существуют несколько причин, по которым этот метод применять не рекомендуется. Наиболее серьезной проблемой является то, что протоколы маршрутизации пока еще подвергаются частым пересмотрам и изменениям. Появляются новые протоколы маршрутизации. Эти изменения должны учитываться в программном обеспечении всех машин.

Несколько более специальная проблема связана с бездисковыми рабочими станциями. По своей природе бездисковые машины сильно зависят от

сети и от файл-серверов, в загрузке и подкачке программ. На бездисковых машинах опасно использовать программы, следящие за широковещательными передачами в сети. Программы маршрутизации используют в основном широковещательные передачи. Например, каждый шлюз в сети может через каждые W секунд широковещательно передавать свои таблицы маршрутов. Проблема с бездисковыми машинами состоит в том, что программы, которые следят за такими передачами, должны быть загружены через сеть. На занятой машине программы, которые не используются в течение нескольких секунд, обычно откачивают. Когда их вновь активизируют, то они должны быть подкачаны. Как только посылается широковещательное сообщение, все машины в сети активизируют программы, следящие за маршрутизацией. Это приводит к тому, что многие бездисковые станции будут выполнять подкачку в одно и то же время. Поэтому в сети возникнет временная перегрузка. Таким образом, исполнение программ, прослушивающих широковещательные передачи, на бездисковых рабочих станциях очень нежелательно.

Динамическая маршрутизация, как отмечалось выше, поддерживает протоколы RIP, HELLO, EGP, OSPF и т.д. обмена информацией о таблицах маршрутизации узлов сети. Специальный программный процесс (в ОС UNIX - демон gated) порождается командой gated и может быть сконфигурирован для работы по всем протоколам одновременно или для любой их комбинации. Демон gated подстраивает таблицу маршрутизации, минимизирует трафик и меняет его в случае разрывов сети и т.д. Построение таблицы маршрутизации при работе gated происходит следующим образом:

- все локальные интерфейсы должны быть сконфигурированы по ifconfig;
- по route должна быть определена косвенная маршрутизация: default;
- после первых двух шагов можно использовать команду gated с опциями настройки согласно небходимой вам конфигурации для запуска динамической маршрутизации.

4. ТЕХНОЛОГИЯ УДАЛЕННОГО ДОСТУПА

Достаточно трудно строго определить круг всех этих приложений. Сегодня, например, о WWW уже можно говорить, как о традиционном приложении TCP/IP. Поэтому в качестве традиционных рассмотрим “ранние” из них и выполняющие базовые функции: удаленный доступ и управление удаленными процессами, передача файлов, обмен сообщениями между пользователями. Минимальная функциональность, необходимая для передачи информации по сети, обеспечивается тремя нижними уровнями модели OSI. Однако, даже над транспортным уровнем для функционирования приложения приходится надстраивать конструкции более высоких уровней.

4.1. Стандартные ARPA-сервисы

Протокол Telnet

Протокол Telnet позволяет пользователю подключиться к любому удаленному хосту (серверу) и работать с ним со своей (клиентской) машины так, как если бы она была удаленным терминалом этого хоста. Telnet, как протокол, предоставляет механизмы для решения трех основных задач:

- определяет интерфейс, называемый виртуальным сетевым терминалом (Network Virtual Terminal - NVT), который абстрагирует аппаратную реализацию сервера и клиента. И клиент, и сервер “отвязываются” от собственных аппаратных особенностей, договариваясь о свободном формате данных;
- Telnet предусматривает механизм конфигурирования ряда параметров (опций);
- Telnet устанавливает и поддерживает симметричное терминальное соединение. Это означает, что как клиент Telnet, так и сервер Telnet имеют одинаковые права по инициации передачи данных, согласованию опций и т.д.

Взаимодействие локального терминала с операционной системой (ОС) локального и удаленного серверов показано на рис. 4.1. Как видно из рис. 4.1,б, Telnet, сам являясь прикладной программой по отношению к ОС локального сервера, позволяет дистанционно обращаться к сервису удаленной ОС или обеспечивает клавиатуру и монитор для другого приложения, работающего на удаленном сервере.

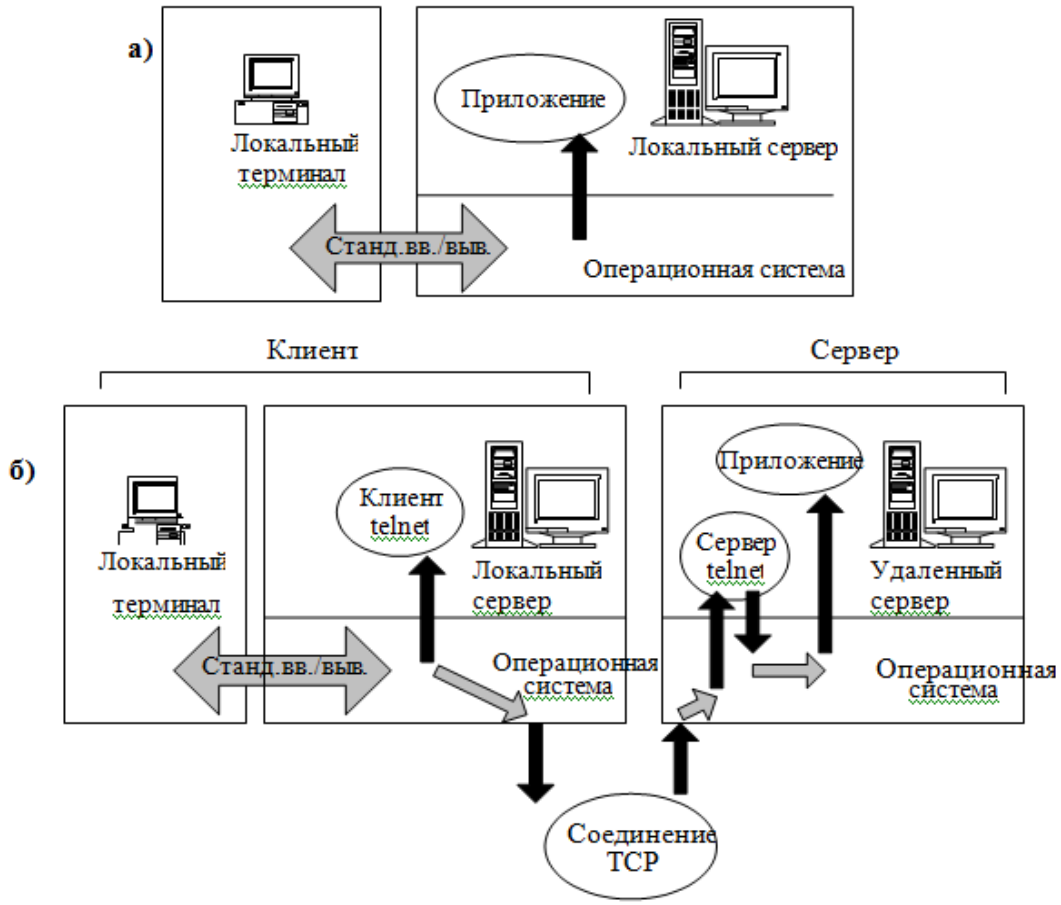


Рис. 4.1

Архитектурное решение о размещении сервиса Telnet на прикладном уровне (а не в ядре ОС) имеет свои преимущества и недостатки. Главное преимущество - реализационная независимость ОС и сервиса Telnet, а недостаток - некоторая потеря в эффективности, которая для преимущественно интерактивных текстовых приложений Telnet неощутима.

Системы “Локальный терминал” и “Локальный сервер”, показанные на рис. 4.1, в современной практике часто могут быть единой аппаратно-программной системой. Но даже в этом случае рассмотрение терминала как отдельного устройства стандартного ввода/вывода достаточно удобно для понимания работы протокола Telnet в целом.

Адаптация к аппаратной гетерогенности ИС

При попытке организации терминальных соединений в реальных ИС, где в качестве оконечного оборудования на различных концах соединения могут применяться существенно различные аппаратные платформы, возникают конфликты, связанные с интерпретацией символьных кодов. Например, одни текстовые терминалы в качестве символа, переводящего строку, используют

специальный код CR (Carriage Return - возврат каретки), другие - код LF (Line Feed - перевод строки), третьи - пару CR-LF.

Решением проблемы служит упомянутый выше интерфейс NVT, являющийся “прослойкой” между локальным терминалом и локальным сервером (рис. 4.2), которая преобразует управляющие символы в приемлемый для удаленного оконечного оборудования вид.

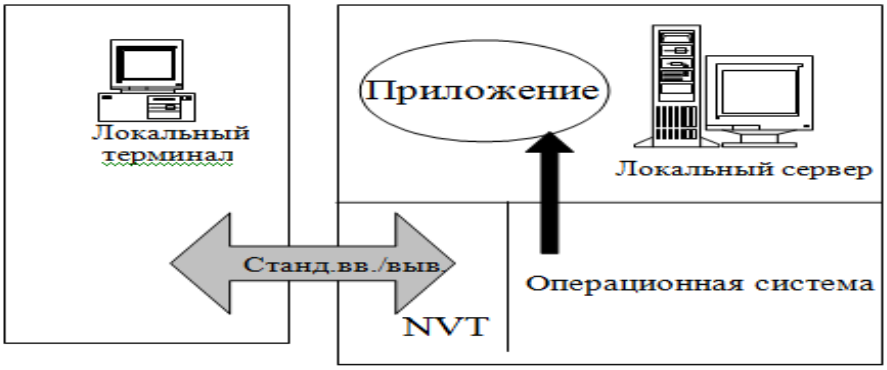


Рис. 4.2

Интерфейс NVT обеспечивает интерпретацию символов управления. Кроме того, в качестве кода перевода строки он формирует последовательность CR-LF вне зависимости от кода, который был сформирован клавиатурой конкретного терминала.

Управление удаленными процессами

Другой функцией NVT является управление удаленным процессом. Эта задача также может конфликтовать с аппаратно-программными системами клиентов и серверов. Пусть вам необходимо прервать процесс на сервере, не разрывая терминального сеанса с ним. Для этого используется стандартный код прерывания процесса Ctrl-C. Однако локальная терминальная система сама воспринимает этот код как код прерывания выполнения задачи. При этом прерывается локальная задача, т.е. оборвется сеанс работы с сервером, а процесс на сервере будет продолжать работать!

Во избежание подобных конфликтов NVT назначает следующие команды управления: IP, AO, AYT, EC, EL, SYNCH, BRK. Для передачи таких команд Telnet использует так называемые escape - последовательности, состоящие из двух кодов. Первым идет код FF (255 - десятичный код), который обычно клавиатурой не генерируется. Этот код называется IAC (Interpret As Command). Вторым следует код команды.

Для передачи команд управления удаленным процессом требуется особый режим. Предположим, что на сервере процесс, с которым работает клиент, перешел в состояние бесконечного цикла, при выполнении которого он не читает входной поток. В этой ситуации входной буфер соединения TCP будет заполнен поступающими данными и новые данные приниматься не будут. Поэтому сколько бы запросов на прерывание процесса ни слал клиент Telnet, они до сервера Telnet не дойдут. Чтобы выйти из такой тупиковой ситуации, команда управления процессом всегда сопровождается командой SYNCH, очищающей буфер от обычных (не срочных) данных и доводящей команду управления процессом до сервера Telnet, который может прекратить существование “зациклившегося” процесса.

Telnet не поддерживает режим блочной передачи и работает только в интерактивном режиме, поэтому команду нельзя запустить как фоновую или из shell-процедуры.

Опции Telnet

Расположение Telnet на прикладном уровне позволяет “развязать” реализации Telnet и операционной системы и, тем самым, обеспечить дополнительную гибкость в модификациях и наращивании функциональности этого протокола. В первую очередь это касается опций Telnet, представляющих собой достаточно интересную и сложную конструкцию, которая позволяет клиенту и серверу “на ходу” реконфигурировать параметры соединения. Набор регулируемых параметров достаточно широк и включает кодировку данных (7 - или 8-битовая), режим передачи (дуплексный или полудуплексный), тип терминала и т.д.

Процедура согласования опций выполнена в открытой манере, характерной для протоколов TCP/IP. Хотя в Telnet инициатива по организации соединения принадлежит клиенту, протокол согласования опций симметричен в том смысле, что сервер, как и клиент, может обладать инициативой в установлении определенных опций. Механизм согласования опций допускает работу на разных концах соединения версий Telnet, снабженных различными наборами опций (вплоть до того, что один из партнеров не владеет никакими опциями и базируется только на NVT).

4.2. Файловый доступ

Протокол FTP

Рассмотрим особенности протокола FTP (File Transfer Protocol). Обеспечение доступа к файлам - одна из старейших и важнейших задач коммуникационных систем, возникшая после первого же опыта соединения двух компьютеров при помощи линии связи. Поэтому этот вид коммуникационного сервиса старше сервиса большинства современных коммуникационных протоколов, в том числе и TCP/IP.

Остановимся на проблемах передачи файлов. Казалось бы, имея такую совершенную транспортную среду, как TCP, передача файлов с компьютера на компьютер - задача тривиальная. Однако простота ее исчезает, как только необходимо решить ее в реальной гетерогенной ИС.

Задачи, которые приходится решать в ИС прикладной системе, следующие:

- разделение доступа к файлам (устранение конфликтов при многопользовательском доступе);
- аутентификация (достоверная идентификация пользователя и его пароля) пользователей;
- авторизация (достоверное выяснение прав и привилегий) пользователей и процессов;
- предоставление данных.

Все эти задачи, за исключением реализации сложных механизмов разделения доступа, более характерные для доступа к базам данных в режиме on-line, приходится решать FTP.

Гетерогенность ИС, как и в случае с Telnet, порождает ряд специфических проблем. О различных представлениях данных в текстах уже говорилось выше. Другой пример - различное представление действительных чисел с плавающей точкой. Даже при корректном распознавании различных форматов таких данных из-за разной разрядности может происходить потеря точности.

Протокол FTP обеспечивает:

- программный доступ к удаленным файлам: для работы программ предоставляется командный интерфейс;
- интерактивный доступ к удаленным файлам: пользователь, вызывая FTP, попадает в интерактивную оболочку, из которой с помощью ряда команд может выполнять достаточно большой набор функций;

- преобразование данных: FTP позволяет клиенту описать формат хранимых данных (например, специфицировать кодировку символов для текстов);
- аутентификацию и авторизацию: FTP проверяет имена пользователей, их пароли и права доступа.

Работа протокола FTP

Сервис протокола FTP, основывающийся на транспорте TCP, поддерживает множество конкурентных FTP-соединений с различными клиентами. В свою очередь, всякое FTP-соединение “внутри” состоит из двух соединений (рис. 4.3): управляющего (control connection) и передачи данных (data transfer connection). Управляющее соединение поддерживается от начала до конца сеанса связи с клиентом (порт 21 в /etc/services). Соединение передачи данных динамически устанавливается на время передачи каждого файла (порт 20 в /etc/services). Обычно оба соединения поддерживаются двумя различными процессами ОС (ftp и ftp-data соответственно). Так происходит в ОС Unix. На ПК все может быть реализовано проще.

Рис. 4.3

Механизм связывания портов различен для клиента и сервера. FTP-сервер работает, как было описано выше, на 21 порту управляющего соединения и на 20 порту передачи данных. FTP-клиент может выбрать и согласовать на локальной клиентской машине номера портов, что позволяет клиенту поддерживать несколько FTP-соединений с одним сервером. (Этот механизм работает для Unix-систем и не всегда поддерживается на ПК).

Система команд протокола FTP

Программный интерфейс FTP выглядит просто. Клиент шлет серверу текстовые команды, состоящие из имени команды и параметров. Список наиболее употребляемых команд, реализующих передачу данных FTP, приведен в таблице 4.1.

Таблица 4.1

Команды, реализующие программный интерфейс FTP		
Команда	Параметры	Назначение
ABOR		Завершает соединение FTP и передачу данных
LIS	filelist	Запрашивает список файлов и каталогов
PASS	password	Передает пароль клиента серверу
PORT	n1,n2,n3,n4, n5,n6	Передает серверу адрес протокола IP (первые четыре байта) и порт клиента (последние два байта)
QUIT		Производит отключение от сервера
RETR	filename	Запрашивает прием файла с сервера
STOR	filename	Запрашивает передачу файла на сервер
SYST		Запрашивает тип ОС сервера
TYPE	type	Назначает тип файла: A-ASCII, I-изображение
UU-	username	Передает имя пользователя на сервер

Ответы сервера - это текстовые строки, для удобства использования в программах начинающиеся с трехцифрового кода ответа, вслед за которым идет текст, раскрывающий значение кода. Первая цифра ответственна за общий тип ответа:

- (1-3) - положительный;
- (4 -5) - отрицательный:
- (1, 2, 4) - промежуточный;
- (3, 5) - окончательный;
- вторая - указывает на причину ошибки;
- третья - конкретизирует событие.

Интерактивный протокол FTP

Для пользователя интерактивный FTP выглядит как отдельная командная оболочка. При вызове протокола FTP с командной строки появляется приглашение ftp>, после которого могут вводиться различные команды.

Для получения информации обо всех (или одной) команде FTP необходимо ввести команду help <имя команды>. Если help вызвана без параметров, то выводится список команд FTP; если указано имя команды, то следует очень краткое пояснение о ее назначении.

На ПК (особенно под Windows) интерактивный протокол FTP реализуется по иному, чаще всего в виде двухоконного интерфейса, отражающего состояние локальной и удаленной машины и выполняющего ограниченный набор команд FTP. Если такое решение не устраивает, то можно воспользоваться Telnet, из которого практически всегда можно вызвать классический протокол FTP.

Анонимный протокол FTP

В Internet существует распространенный сервис, называемый анонимным FTP (Anonymous FTP). Он обеспечивает файловый доступ к серверам с бесплатной публично-доступной или условно-продаваемой информацией. На таком сервере нельзя зарегистрироваться в качестве пользователя (слишком много пользователей пришлось бы регистрировать), поэтому вход осуществляется под условным именем Anonymous, а в качестве пароля используется адрес электронной почты (подлинность которого не всегда контролируется), после чего с этим сервером можно беспрепятственно обмениваться информацией.

Если анонимный доступ на сервер запрещен, то можно попробовать поработать с этим сервером в сеансе Telnet под именем пользователя Guest (если такой есть), который обычно не имеет никаких прав, но и не запрашивает пароля.

Протокол TFTP

Сервис FTP может быть избыточным для ряда прикладных систем, которым необходим простой и нересурсоемкий сервис передачи файлов. К такому сервису можно отнести дистанционную загрузку бездисковых станций, где компактность и нетребовательность к ресурсам файлового протокола имеют определяющее значение.

Соответствующий сервис в семействе TCP/IP поставляет другой, более простой протокол – TFTP (Trivial File Transfer Protocol) - тривиальный протокол передачи файлов. TFTP организован очень просто. Все файлы передаются блоками по 512 байт. Первые два байта (октета) каждого блока содержат код

операции. Самый первый (инициализационный) блок содержит запрос на чтение или на запись. После приема запроса начинается передача данных. Каждая порция данных содержит номер блока данных и далее, до длины 512 байт, данные. На каждый блок данных принимающая сторона отвечает квитанцией. При возникновении ошибки генерируется специальное сообщение. И принимающая, и передающая стороны устанавливают тайм-аут на ожидание следующего блока. Если блок данных или квитанция теряются, то передающая сторона по истечении тайм-аута повторяет передачу блока, а принимающая - квитанции. В качестве транспортного протокола TFTP используется UDP.

4.3. Стандартные Berkeley-сервисы удаленного доступа

Развитием терминального доступа в ряде систем Unix является удаленный доступ Rlogin (remote login). Rlogin отличается от Telnet более тесной связью с ОС. Для Rlogin характерно распознавание прав доступа пользователей (он не требует повторного ввода пароля при последовательном доступе с одного хоста к другому, если пользователь корректно выполнил операцию входа на первый из доступных ему хостов). Кроме того, Rlogin может использовать стандартный ввод/вывод ОС удаленных хостов; интерпретировать стандартные ошибки ОС; распознавать параметры настройки оболочки (environment) как на локальном, так и на удаленном хосте; экспортировать часть этих параметров с локального на удаленный хост и т.д.

Команда rcp - это средство удаленного копирования файлов в систему Unix и из нее. Команда rcp позволяет также выполнять передачу файлов, при которой локальный узел лишь инициализирует передачу данных между удаленными машинами. Использование этой команды требует специальной настройки на локальной и удаленной машинах в специальных файлах суперпользователя, которым необходимо присвоить права *только чтения и только для пользователя root.*

Команды remsh/rexec/. Эти команды являются Berkeley-сервисами, предназначенными для выполнения команд на удаленной системе. Если имя пользователя в команде remsh не определено, то используется текущее имя пользователя, соответствующее действующему идентификатору пользователя. Это имя должно быть зарегистрировано на удаленной системе. Таким образом, ***remsh*** не требует пароля при своем выполнении и указанную команду ищет в каталогах: /bin; /usr/bin; /usr/contrib/bin; /usr/local/bin удаленной машины. Если в remsh не задана команда, то интерпретируются в командной строке опции, как в rlogin. Команда ***rexec*** функционально идентична remsh, но

она требует ввода пароля от пользователя, а не выполняет авторизацию доступа посредством специальных конфигурационных файлов.

Команда rwho. Выводит следующую информацию об Unix-узлах ИС:

- входное имя каждого пользователя, находящегося на машинах в ИС;
- имя машины для каждого активного пользователя;
- терминальную линию пользователя;
- дату и время входа пользователя в систему;
- время бездействия пользователя.

Команда ruptime. Выводит следующую информацию об Unix-узлах ИС:

- имена компьютеров в ИС;
- состояние компьютеров ИС;
- число активных пользователей для каждого компьютера ИС;
- среднее число заданий в очереди каждого компьютера ИС за последние одну, пять и пятнадцать минут.

Особенности удаленного доступа

Удаленный доступ является чрезвычайно мощным инструментом. Практически без всяких территориальных ограничений можно запустить процесс на любом доступном (в административном отношении) хосте; произвести вычисления, получить или передать информацию, дистанционно поработать на этом хосте с приложением, изначально не ориентированным на коммуникации и, возможно, ничего о них даже “не знающем”. Очень важны также и административно-управленческие возможности, предоставляемые протоколами удаленного доступа. Системный оператор может войти на удаленный компьютер, произвести необходимую реконфигурацию, тестирование, настроечные или восстановительные работы. Это дает возможность выделить всего одного системного оператора на 500 хостов. Это особенно актуально в настоящее время, когда стоимостные пропорции изменяются таким образом, что заработная плата высококвалифицированного персонала (к которому принадлежит и системный оператор) становится ощутимой статьей накладных расходов на эксплуатацию автоматизированной системы.

Однако на удаленный доступ можно посмотреть и как на угрозу. Протоколы удаленного доступа в руках инженера являются полезным инструментом, а в руках злоумышленника - грозным оружием. Поэтому их использование должно сопровождаться тщательным анализом риска, связанного с доступом к информации и вычислительным ресурсам сети.

4.4. Сетевая файловая система

Протокол NFS

Развитием файловых служб в TCP/IP является сетевая файловая система NFS (Network File System) - протокол, разработанный фирмой Sun Microsystems. Он обеспечивает прозрачный разделяемый доступ к файлам по сети, доступ в режиме on-line. Отображение протокола NFS на модель OSI приведено на рис. 4.4. Клиентская система NFS реализуется таким образом, что пользователь или приложение практически не различают локальные и удаленные файлы. Когда клиентское приложение обращается к файлу (рис. 4.4), то операционная система сама определяет: к локальному или сетевому файлу обращается клиент, и переадресует управление соответственно локальной или сетевой файловой системе.

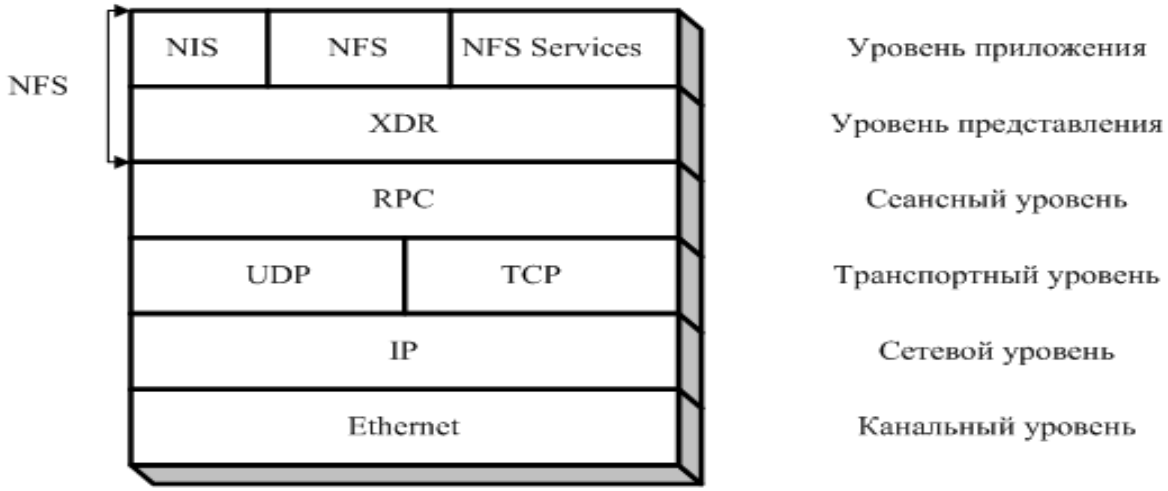


Рис. 4.4

Удаленный вызов процедур и преобразование данных

Сетевая файловая система NFS реализуется одновременно с двумя другими протоколами - удаленного вызова процедур RPC (Remote Procedure Call) и внешнего представления данных - XDR (eXternal Data Representation). Такое функциональное деление позволяет разработчику приложений независимо обращаться к каждому из протоколов и использовать только необходимую функциональность. В сумме же разработчик получает очень мощный инструмент для создания распределенных систем (рис.4.5).

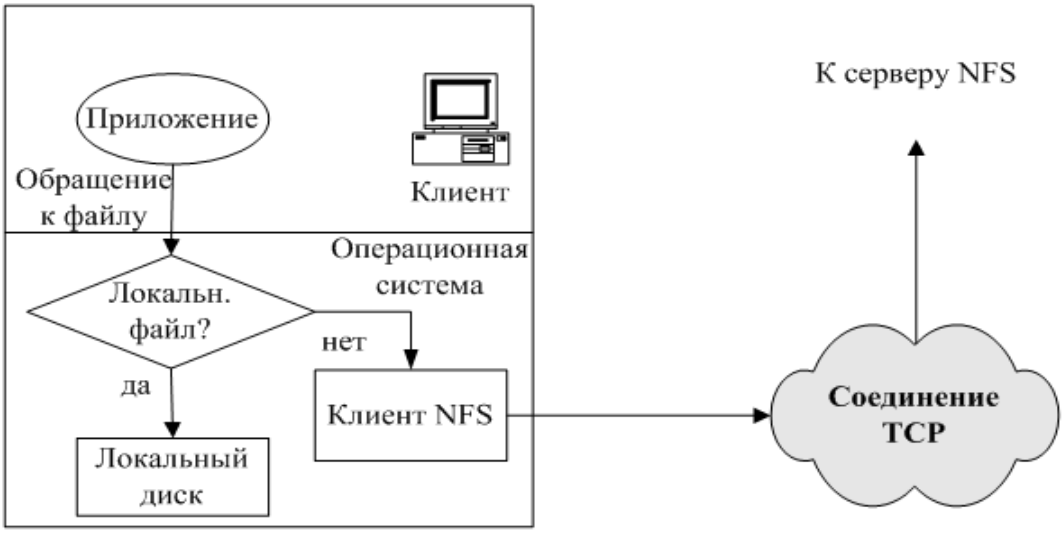


Рис. 4.5

Продукт NFS состоит из нескольких различных частей:

- Network File System (Сетевая файловая система) - распределенная файловая система, разработанная компанией SUN Microsystems. Она стала стандартом распределенных файловых систем и используется другими сервисами.
- Network Information System (Сетевая информационная система) - упрощает администрирование сети. Позволяет иметь в сети только одну копию конфигурационных файлов. Была создана для того, чтобы решить некоторые проблемы, возникшие в связи с использованием NFS.
- Remote Procedure Call (Вызов удаленной процедуры) - сервис для написания распределенных приложений. Некоторые RPC-программы используют основные части NFS.
- External Data Representation (Внешнее представление данных) - обеспечивает передачу данных в машинно-независимом формате. Все запросы преобразуются к этому формату.
- NFS Services (Сервисы NFS) состоят из следующих продуктов:

- Network Lock Manager (lockd) - сервис, поддерживающий блокирование файлов, синхронизацию доступа и разделение файлов в NFS (вызовы lockf и ffcntl);
- Network Status Monitor (statd) - используется вместе с NLM. Позволяет получить статус удаленных систем;
- Remote Execution (REX - rexd и on) - позволяет выполнять команды на удаленных компьютерах. REX являются аналогом сервиса группы Berkeley - remote shell;

- Remote Procedure Call Protocol Compiler (RPCGEN) - используется для автоматизированного создания распределенных приложений. Для программистов позволяет генерировать приложения клиент-сервер;

- Named Pipes;

- DeviceFiles;

- Virtual Home Environment (VHE) - позволяет пользователю иметь одну и ту же домашнюю среду на всех компьютерах сети;

- Export;

- Automounter - позволяет по мере обращения монтировать и размонтировать удаленные файловые системы, если к ним не было обращения в течение определенного времени.

Большинство этих сервисов конфигурируется на этапе инсталляции системы:

- NLM и NSM конфигурируются в загрузочных файлах NFS;
- REX конфигурируется в /etc/inetd.conf;
- RPCGEN обычно является частью операционной системы.

При разработке системы клиент-сервер протокол RPC позволяет определить на клиентской станции некоторые процедуры как удаленные (remote). Эти процедуры должны быть включены в серверное программное обеспечение как доступные для удаленного вызова, что позволяет создавать распределенные приложения, разделив их на два компонента: клиент и сервер. Это также путь к использованию мощности сервера. Далее этого углубляться в подробности сетевого математического обеспечения разработчику не нужно: компиляторы автоматически включат в его приложение код, реализующий протокол RPC и ответственный за взаимодействие распределенных частей системы. Схема вызова удаленной процедуры приведена на рис.4.6.

Рис. 4.6

Протокол XDR также облегчает разработку распределенных приложений для гетерогенных сред. Он берет на себя функцию учета аппаратных особенностей платформ, на которых работает. Например, если клиент и сервер используют представления целых чисел с различным порядком следования младших и старших байтов, то XDR преобразует данные, распространяющиеся в разных направлениях, и разработчику не нужно вручную программировать соответствующий конвертор. Различия между представлением данных двух машин могут быть следующими:

- направление байта;
- работа с числами с плавающей точкой;
- различия представления массивов и строк.

XDR можно представить себе, как универсальный переводчик, который переводит с одного языка на другой (рис. 4.7).

Сетевая информационная система NIS - упрощает администрирование больших сетей, решает некоторые проблемы, связанные с NFS, при помощи создания копий конфигурационных файлов, которые разделяются по сети. К таким файлам относятся: /etc/passwd и /etc/group и т.д.

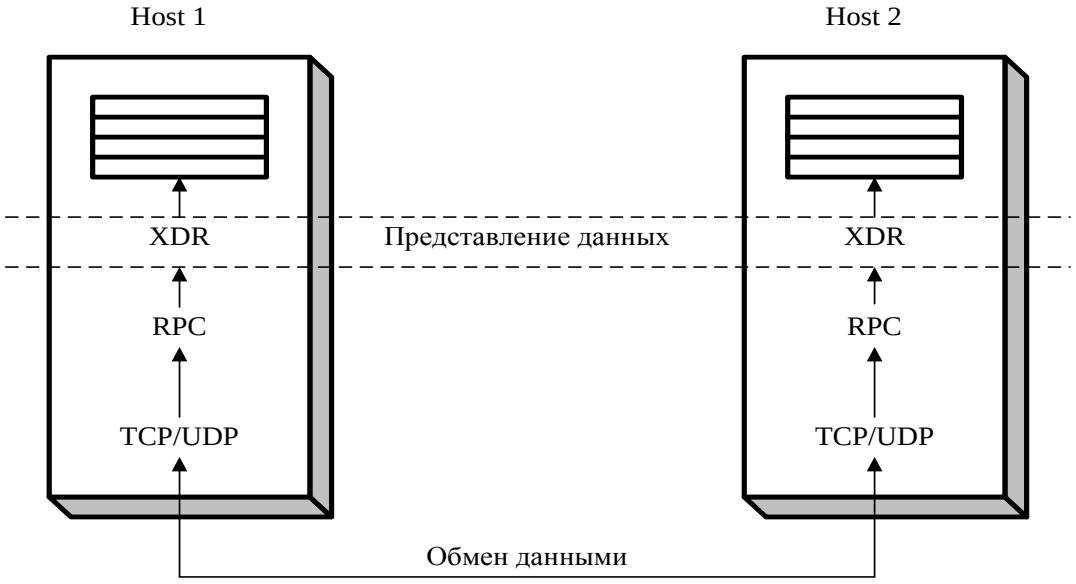


Рис. 4.7

Достоинства и недостатки NFS Services

К числу основных достоинств NFS Services относятся:

- NFS стал фактически промышленным стандартом распределенной файловой системы, поэтому был перенесен в некоторые операционные системы, выпускаемые различными фирмами.

- NFS и NLM позволили перейти от системы разделения времени и сети рабочих станций путем преобразования миникомпьютеров общего назначения в файловые серверы. Это уменьшило стоимость систем и снизило потребность в дисковом пространстве.
- При использовании NFS и NIS администратор системы имеет дело только с одной копией системных файлов, которая используется всеми компьютерами сети.
- Пользователь работает с удаленными файлами точно так же, как и с локальными. Удаленная файловая система монтируется суперпользователем и доступ к ней осуществляется так же, как и к локальной.

К числу основных недостатков NFS Services можно отнести:

- Не все версии NFS поддерживают удаленную загрузку (SUN Network Disc). Это не позволяет разделить диск таким образом, что рабочие станции сети могут использовать его так, как будто это локальный диск каждой из них.
- Администрирование может быть затруднено, так как могут совпадать идентификаторы пользователей и групп на отдельных компьютерах.
- Файлы устройств не для всех версий NFS могут быть экспортированы, смонтированы и использованы из удаленной системы.
- Могут быть ограничения конфигурации NFS:
- Количество связей клиент-сервер;
- Некоторые файлы не могут разделяться;
- Что делать, если сервер выключен?;
- Суперпользовательский доступ.

Ограничения конфигурации NFS:

- Как много клиентов могут иметь доступ на один сервер? Число связей клиент-сервер ограничивается интересами сервера и тем, насколько интенсивно приложения-клиенты эти ресурсы используют.
- Многие системные файлы не должны разделяться по NFS. Это файлы системных протоколов, системы спулинга и файлы, содержащие специфическую системную информацию, такую как сетевой адрес и имя компьютера.
- Если NFS-сервер недоступен, а системы клиенты импортируют домашние каталоги с сервера, то пользователи не смогут войти в систему.
- Когда разделяются системные файлы, доступ суперпользователя является важным фактором. Программы, которые при запуске устанавливают UID пользователя root, будут работать нормально, тогда как любая команда, которой нужен суперпользовательский доступ для выполнения, не будет работать для клиента, если используется старая версия NFS (до 1990 г.).

- Каждая система может быть сконфигурирована как клиент, сервер или как то и другое. Программное обеспечение PC-NFS фирмы SUN Microsystems позволяет ПК быть NFS - клиентом.
- Хорошая практика - выносить пользовательские файлы на отдельный диск. Если сервер выйдет из строя на долгое время, то диск может быть физически перенесен на другой компьютер.

Создание сетевой файловой системы. На сервере необходимо выполнить следующие действия:

- Убедитесь, что NFS установлена в ядро.
- Проверьте, что в файлах загрузки NFS включена возможность быть сервером для других систем-клиентов (отдельно для клиента PC!).
- Раскомментируйте в файле /etc/inetd.conf все строки, которые начинаются с “rpc”.
- Для того чтобы разделить доступ к файловой системе на сервере, в файл /etc/exports добавляется строка: *имя файловой системы - кому разрешен доступ - с какими правами - или доступ разрешен всем.*
- Убедитесь, что идентификаторы пользователей и групп на сервере и клиентах совпадают.

Для того чтобы получить доступ к файловой системе сервера, на клиенте:

- устанавливается программное обеспечение NFS;
- создается точка монтирования;
- выполняется команда монтирования.

Конфигурационные файлы NFS:

/etc/checklist - содержит список файлов и каталогов, которые автоматически монтируются во время загрузки системы.

/etc/exports - содержит список файлов и каталогов, которые клиент может импортировать. Находится только на сервере.

/etc/lnetd.conf -содержит информацию о серверах, которые запускает inetd.

/etc/netgroup - содержит имена сетевых групп, /etc/exports и /etc/passwd могут использовать сетевые группы, определенные в этом файле. Конфигурирование этого файла необязательно.

/etc/rpc - ставит в соответствие именам RPC-програм их номера и другую информацию.

/usr/adm/lnetd.sec – проверяет, разрешено ли использование сервиса для компьютера, запросившего этот сервис. Если имя сервиса не указано в этом файле, то он разрешен для всех.

Демоны NFS. Демоны - это фоновые процессы, которые постоянно работают, ожидая запроса для выполнения задания. Имеются следующие процессы:

biod - сетевой демон, ожидающий запросов на портах и выполняющий следующие действия:

- читает /etc/inetd.conf для выяснения, какой сервер должен обслужить пришедший запрос;
- ожидает и принимает запросы на обслуживание из сети;
- запускает нужный сервер.

inetd - демон, отвечающий на запросы по NFS

pcnfsd - демон, обслуживающий пользователей ПК. Принимает от пользователя ПК входное имя и пароль, а затем возвращает идентификаторы пользователя и группы либо сообщает, что имя и пароль введены неправильно.

portmap - демон, преобразующий номера программ в номера портов. Передает программе inetd следующую информацию:

- список серверов для обслуживания;
- на каких портах ожидать запросы;
- номера RPC-программ и их версии.

Когда клиент делает запрос с заданным номером программы, то сначала он узнает у portmap номер порта, в который посылать запрос.

nfsd - когда программе клиента нужно прочитать или записать в удаленный файл, то она посылает запрос nfsd.

Серверы NFS

Сервер, в отличие от демона, не работает постоянно, а вызывается только по запросу демона.

mountd - отвечает на запросы монтирования. Читает файл /etc/xtab, чтобы определить, какие каталоги и файлы доступны другим системам и пользователям.

rstatd - возвращает статистику по ядру системы. Используется командой rup.

rusersd - возвращает список пользователей, работающих на локальной системе. Используется командой rusers.

rwalld - поддерживает выполнение rwall.

sprayd - поддерживает выполнение spray.

4.5. Технология сетевого управления

Сетевое управление дает возможность получить максимальную эффективность при организации ИС корпорации при минимальных затратах на эксплуатационные расходы (по данным компании Hewlett Packard при использовании современных систем сетевого управления один сетевой администратор может управлять сетью, объединяющей до 500 серверов). Однако, при разрастании ЛВС до сотен (или даже тысяч) рабочих станций поддержание ее работоспособности требует большого опыта и глубоких знаний.

Проектировщики ИС предлагают три специальных протокола, предназначенных для мониторинга (оперативного наблюдения за состоянием всех сетевых ресурсов) и управления информационными сетями - ***SNMP*** (*Simple Network Management Protocol* - Простой протокол для управления вычислительной сетью) для TCP/IP-сетей, и отвечающий модели стандарта OSI протокол ***CMIP*** (*Common Management Information Protocol* - Протокол общего управления информацией) для OSI-сетей и ***CMOT*** (*CMIP over TCP/IP*). Эти протоколы специально разработаны для диагностики работоспособности различных локальных и широкомасштабных сетей как общего доступа, так и корпоративных – масштаба предприятия.

CMIP разрабатывался таким образом, чтобы включить столько возможностей, сколько можно было предвидеть. Вследствие этого данный протокол достаточно сложен и требует больших затрат на то, чтобы реализовать его в сетевом элементе. SNMP, с другой стороны, проектировался таким образом, чтобы выполнять основные простейшие функции. Соответственно он имеет сравнительно небольшой программный код, его легко и быстро можно реализовать в устройствах. CMIP имеет несколько большие возможности в:

- сфере безопасности;
- пересылке больших объемов данных;
- встроенных средствах анализа информации, проводимого посылкой сигнала тревоги устройством.

Однако, поскольку SNMP значительно проще в реализации, он получил значительно более широкую промышленную поддержку. Оба стандарта быстро развиваются, включая в себя преимущества другого. Сегодня уже существуют более простые версии CMIP, а с другой стороны, более интеллектуальные с более развитыми средствами защиты версии SNMP.

Основные принципы сетевого управления

Необходимо иметь возможность дистанционного контроля серверов, мостов, маршрутизаторов и т.д., для чего требуются средства автоматического построения плана ИС, содержащего следующую информацию:

- о кабельных трассах;
- о схемах соединения кабелей;
- о протяженности ИС;
- о стандартах протоколов и оборудования ИС;
- о росте числа рабочих станций и новых коммуникационных технологиях.

Это необходимо для учета многих технических аспектов, касающихся возможных сбоев в ИС.

Управление трафиком. Аппаратный или программный сбой может привести к полной остановке сети или ввести ее в режим резкого увеличения трафика, на который она не рассчитана. В результате перегрузки сеть может и не выдавать сигнала о неисправности, за исключением заметного падения производительности. Система управления должна обнаруживать и сообщать о такого рода ошибках.

Определение статуса устройств ИС. Нередко администратору может понадобиться информация о статусе устройств, таких как рабочие станции, мосты и межсетевые шлюзы и т.д. для тестирования трассы между рабочими станциями, поэтому программное обеспечение должно предоставлять такую информацию.

Управление конфигурацией ИС. Формирование и изменение конфигурации сетевых устройств должны осуществляться просто и понятно с использованием графического представления как физического, так и логического уровней ИС. Система управления конфигурацией позволяет понять, какие ресурсы доступны в сети и каким образом они соединены. При этом рассматриваются не только аппаратные ресурсы, но и используемые программные компоненты, например, тип операционной системы и какие прикладные системы каким пользователям доступны.

Управление контролем доступа и защитой данных. В больших корпоративных ИС, включающих множество ЛВС, функции контроля доступа и защиты данных выполняются средствами управления сетью, так как не могут уже обеспечиваться только сетевой операционной системой. Задача администратора сети в этом случае состоит в установке контроля доступа и процедур доступа на различных уровнях и к различным ресурсам сети. Программное

обеспечение для управления сетью поддерживает функции администратора, как руководителя службы контроля, и даже может регулировать с помощью паролей доступ к прикладным программам.

Учет использования ресурсов. Система учета использования ресурсов количественно определяет использование различных сетевых ресурсов. От каждого пользователя или отдела можно затем взимать плату в соответствии с согласованными критериями. Плата обычно отражает время сеанса связи и объемы получаемых и передаваемых данных.

Сетевое планирование. Жизненный цикл ИС должен быть спланирован, начиная с инсталляции и включая ее обслуживание, расширение и изменение. Соответствующие прикладные программы помогают осуществлять перспективное планирование по принципу “что если”, основываясь на реальных предпосылках. Для прогнозирования будущих потребностей могут быть использованы статистические отчеты о динамике изменения сетевых характеристик и загрузке сети по времени.

Система инвентаризации. Система инвентаризации предполагает создание и поддержку базы данных по всему оборудованию, которое составляет сеть. Описание сетевых компонент могло бы включать указание производителя, тип и соглашение на обслуживание. Некоторые прикладные системы включают CAD-программы, которые могут рисовать чертеж офиса с точным размещением сетевых элементов и кабелей.

Система управления прикладным программным обеспечением. Данные средства позволяют создавать каталог того, какие программные средства используются и кем. Существует также возможность автоматически модифицировать программы. Это дает гарантию того, что каждый пользователь работает с одной и той же версией программы. В результате эти задачи решает минимальный штат поддержки. Становится возможным согласованное и надежное обслуживание отдельных пользователей и лицензирование программ по участкам.

Управление выводом на печать. После объединения локальных сетей проблема управления большинством из доступных принтеров становиться критической: необходимо диагностировать и разрешать проблемы, связанные с ними.

Управление архивированием и резервным копированием. Программы сетевого управления допускают централизованное управление резервным копированием и выполнением процедур, обеспечивающих постоянную защиту корпоративных данных.

Управление коммуникационным оборудованием. Система сетевого управления дает возможность дистанционно настраивать и перенастраивать коммуникационное оборудование: маршрутизаторы, модемы, коммутаторы, концентраторы, мосты и т.д., что позволяет значительно экономить время и штаты для обслуживания протяженных корпоративных ИС.

Управление виртуальными сегментами и виртуальными сетями. Позволяет без перекроссировок и переподключений создавать и изменять состав виртуальных сетей и сегментов согласно потребностям рабочих групп и отделов предприятия.

Управление инсталляцией операционных и прикладных систем. Система сетевого управления может включать в себя модули, позволяющие производить дистанционную установку новых операционных систем, общесистемного и прикладного программного обеспечения, как на серверы, так и на рабочие станции клиентов.

Существует несколько платформ сетевого управления, предлагаемых различными изготовителями. К наиболее полнофункциональным и распространенным относятся системы сетевого управления на базе ОС Unix:

- OpenView фирмы Hewlett Packard (ОС HP-UX);
- SunNetManager фирмы Sun (ОС SunSolaris);
- NetWareManager фирмы IBM (ОС AIX).

Все производители сетевого оборудования обязаны включать в поставку и прикладные модули управления, работающие под управлением вышеуказанных систем. Множество других систем сетевого управления не являются полнофункциональными и не обязательно поддерживаются поставщиками сетевого оборудования, поэтому при их приобретении необходимо проанализировать соответствие всех их возможностей и всех ваших потребностей в управлении сетью с учетом ее развития. Как правило, большинство таких систем относительно дешевы и могут вполне удовлетворить потребности сетевых администраторов небольших локальных сетей.

4.6. Применение протоколов управления

Существует два основных подхода к реализации процесса управления. Первый и, видимо, наиболее мощный по своим возможностям, подход основан на создании и использовании специальных программных средств для управления конкретным сетевым устройством (аналог работы систем АСУТП). Второй подход базируется на работе с некоторыми данными, описывающими сетевое устройство. В этом случае в качестве воздействия ис-

пользуется поток данных, а не поток управления. Поток данных, в отличие от потока управления, позволяет построить более универсальную, хотя и более ограниченную по своим возможностям, модель управления. Основное ее преимущество - независимость не только от операционной среды, но и от конкретной аппаратной реализации управляемого устройства. Этот метод и положен в основу построения систем сетевого управления.

Основные понятия метода управления потоком данных ИС

Вся ИС с точки зрения управления делится на систему управления и объект управления. К системе управления относится совокупность вычислительных средств, предназначенных для формирования управляющих воздействий и анализа информации, на основе которой принимается решение об управлении. Объект управления - это ресурс, которым необходимо управлять.

Система управления ИС состоит из станции управления и набора вспомогательных средств (зонды, анализаторы, приложения и т.д.). Обычно станция управления представляет собой достаточно мощную графическую станцию с установленной на ней системой управления. Важным элементом является описание объектов управления (станция при этом и сама может быть объектом управления).

Встречается также понятие “посредник” (proxy) при наличии нестандартного объекта управления, которому нужно переводить стандартные информационные потоки. Посредниками часто служат компьютеры, работающие как самостоятельные станции управления и периодически обменивающиеся информацией с центральной станцией.

Описание объектов управления

Для описания объекта управления необходимо описать все связанные с ним данные, которые доступны системе управления. Формальное описание объектов осуществляется с помощью языка Abstract Syntax Notation One (ASN.1). Для задания имени объекта достаточно указать его идентификатор (OBJECT IDENTIFIER), представляющий собой последовательность целых чисел. Все идентификаторы организуются в виде листьев дерева. На верхнем уровне присутствуют три узла: iso; ccitt; joint-iso-ccitt.

В поддереве iso присутствует идентификатор org (узел, являющийся корнем для поддеревьев различных организаций), один из подузлов которого, например, - Министерство обороны США (Department of Defence, DOD). “Неофициально” первым узлом в поддереве DOD является Internet. Таким образом, объект “internet” имеет идентификатор “1.3.6.1”:

internet OBJECT

IDENTIFIER ::= {iso(1) org(3) dod(6) 1}.

Документ RFC 1907 определяет набор объектов, которые являются предметом рассмотрения в протоколе SNMP. Часть из них является обязательной для любой реализации.

Для правильной интерпретации объекта одного идентификатора не достаточно. Необходимо указать тип объекта и дать его семантическое (смысловое) определение. Для этого служит информационная база управления (Management Information Base, MIB). В ней описаны все объекты в соответствии с грамматикой языка ASN.1. Наличие текстового представления базы MIB на стороне системы управления позволяет не только правильно интерпретировать получаемые данные, но и давать пояснения об объектах на основе описания, присутствующего в базе. Для объекта нет необходимости в наличии текстового варианта базы MIB.

Под термином “значение переменной” будем понимать значение объекта, имеющего заданный идентификатор (аналог в программировании: имя переменной и ее значение).

Существующие версии протокола SNMP поддерживают работу с базой MIB-II. Использование только объектов, требуемых по спецификации протокола SNMP, не позволяет задействовать некоторые возможности управляемой аппаратуры. Например, маршрутизаторы поддерживают различные сетевые протоколы, отличные от IP, следовательно с помощью стандартных объектов, описанных в базе MIB, невозможно получить информацию, касающуюся этих протоколов. Для разрешения подобного конфликта в дереве объектов существует специальный узел private (internet 4), одним из подузлов которого является узел “предприятия” (enterprises, private 1). Все объекты, которые фирма-производитель желает сделать доступными по протоколу SNMP, описываются в поддеревьях соответствующей фирмы. Таким образом, если системе управления, созданной на основе протокола SNMP, необходимо запросить специфический параметр, то с помощью описания объектов в виде базы MIB можно правильно сформировать запрос и интерпретировать полученный ответ.

Версии протокола SNMP

Существуют две основные версии протокола SNMP:

- собственно SNMP, обычно называемый первой версией (SNMPv1);
- вторая версия SNMP (SNMPv2).

Имеются два основополагающих документа по каждой версии протокола (способа передачи данных) и описание необходимых для него объектов. Подход к управлению в SNMP любой версии базируется на передаче данных, а не на прямом воздействии на управляемое устройство. С помощью передачи данных нельзя перезагрузить компьютер, но можно установить таймер на заданное время, по истечении которого он будет перезагружен или установлена новая версия программного обеспечения.

Протокол SNMPv1. Этот протокол описан в документе RFC 117. Пакет SNMP состоит из трех частей (рис. 4.8):

- Version - версия протокола;
- Community - сообщество;
- SNMP PDU - блок данных.

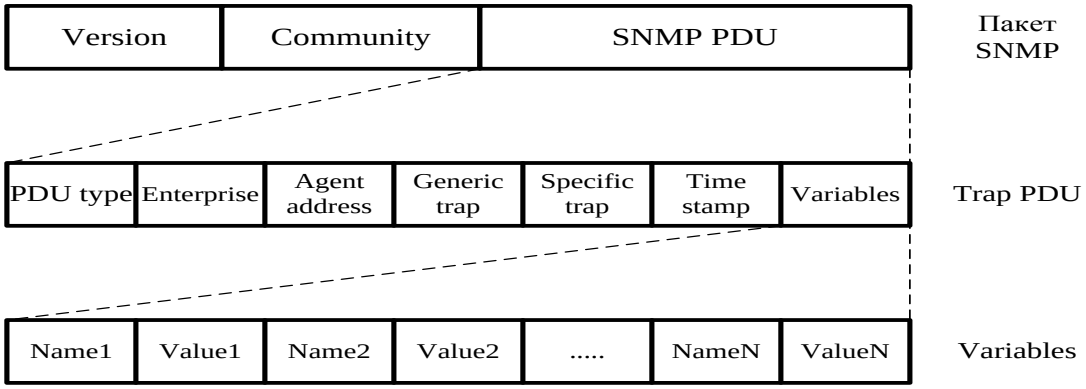


Рис. 4.8

Поле Community применяется для ограничения доступа к управляемой системе. Блок данных PDU начинается с описания его типа (PDU type), который и определяет структуру блока. В общем виде блок данных состоит из нескольких обязательных полей и произвольного числа имен (Name) и значений (Value) переменных.

В протоколе SNMPv1 определены следующие типы блоков данных: Get; GetNext; Set; GetResponse; Trap.

Первые два типа предназначены для запроса информации. Блок Set позволяет изменить значение переменной. Блок GetResponse посылается запрашиваемой стороной в ответ на любой запрос и включает в себя информацию об ошибках и содержательную часть ответа. Блок данных Trap (прерывание) формируется управляемой стороной в силу внутренних причин. Перезагрузка компьютера - типичный пример условия для посылки блока данных Trap.

Данные. Общий вид дерева основных групп объектов, связанных с MIB-II, приведен на рис. 4.9. Обязательным является наличие групп system, interfaces, ip и icmp.

Остальные группы необходимы для устройств, поддерживающих соответствующий протокол. Для получения значения элемента в запросе формируется полный идентификатор нужного объекта, который дополняется значением 0. Таким образом, чтобы получить время работы системы после последней перезагрузки (sysUpTime из группы system), в запросе формируется идентификатор 1.3.6.1.2.1.1.3.0 (iso org dod internet mgmt mib-2 system sysUpTime 0). Некоторые объекты не являются скалярными величинами, а представляют из себя матрицы (таблицы) - например, таблица маршрутизации. Доступ к элементам матрицы в SNMP осуществляется следующим образом: формируется идентификатор, состоящий из идентификатора элемента матрицы, к которому добавляется значение индекса (или индексов).

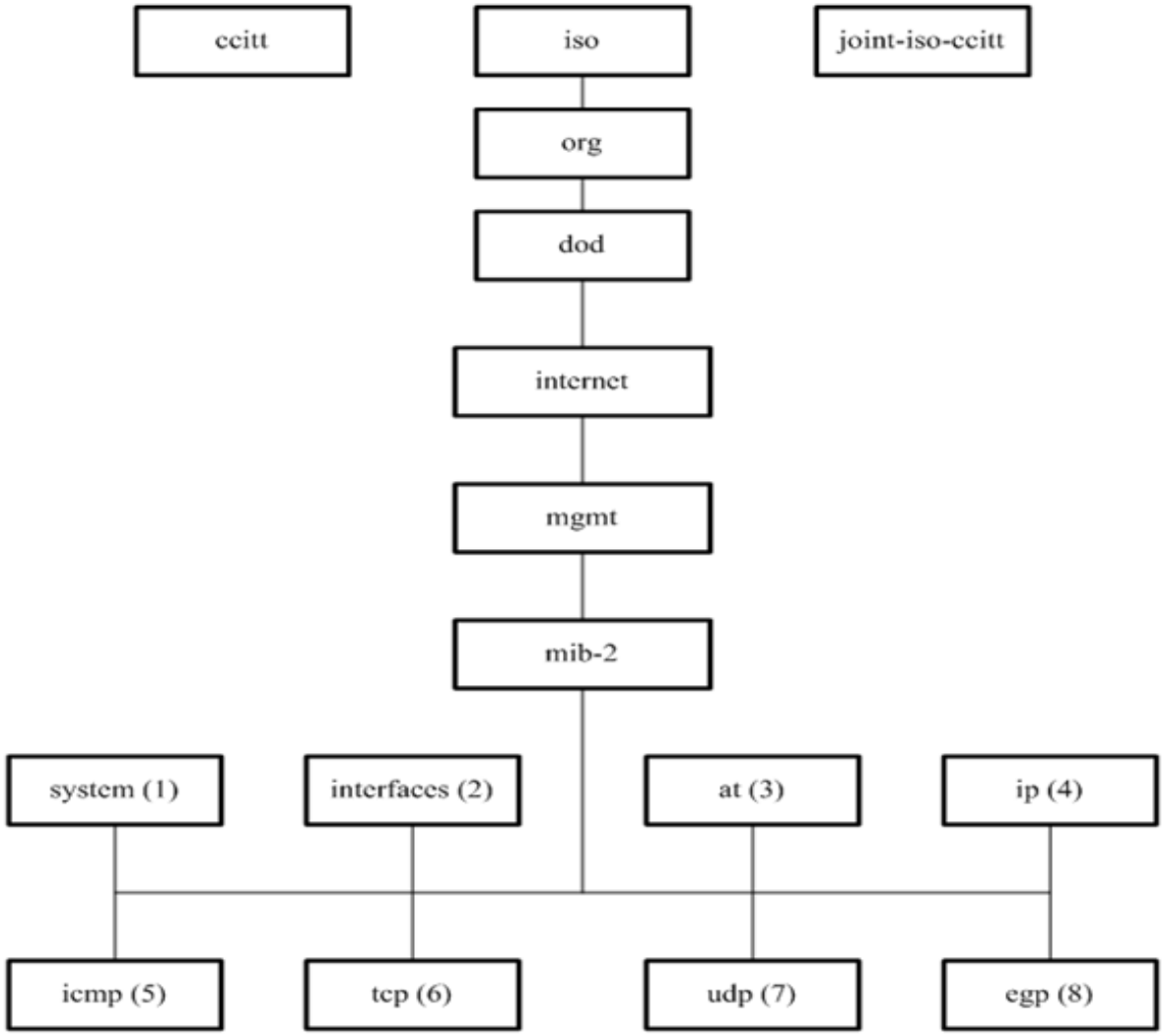


Рис. 4.9

Например, для получения физического адреса второго сетевого адаптера управляемого устройства необходимо запросить значение переменной ifPhysAddress (6) из матрицы ifTable (2) группы interfaces (mib-2 2). Таблица состоит из столбцов ifEntry (1). Полный идентификатор для запроса будет иметь вид: 1.3.6.1.2.2.2.1.6.2 (iso org dod internet mgmt mib-2 interfaces ifTable ifEntry ifPhysAddress 2).

Рассмотрим использование протокола SNMP в группе ip. Группа IP состоит из трех таблиц и ряда скалярных элементов, которые содержат информацию о характеристиках устройства, связанных с протоколом IP в целом. Например, ipInDelivers (ip 9) - число пакетов, успешно переданных на более высокий уровень стека протокола. Таблица ipAddrTable (ip 20) описывает физические интерфейсы управляемого устройства с точки зрения протокола IP:

- максимально возможный для данного интерфейса размер пакета;
- маску подсети;
- значение битов для широковещательных сообщений.

Таблица ipRouteTable (ip 21) - таблица маршрутизации; внося в нее изменения, можно изменять процесс маршрутизации.

Таблица ipNetToMediaTable (ip 22) определяет соответствие физических интерфейсов адресам IP.

Зная конфигурацию ИС и отслеживая загрузку каждого физического интерфейса, можно выравнивать потоки для различных интерфейсов путем оперативного изменения таблицы маршрутизации. Все данные, необходимые для принятия решения, могут быть получены с помощью опроса соответствующих переменных в группе ip.

Если нам необходимо узнать адрес маршрутизатора, используемого по умолчанию (default), то нужно указать элемент таблицы с требуемым индексом, однако индексом в данной таблице является неизвестный адрес IP получателя. Для разрешения подобной ситуации существует запрос GetNext, позволяющий запросить следующий элемент в дереве объектов. В ответе при этом будут переданы не только идентификатор объекта, но и его значение. Следовательно, с помощью запроса GetNext можно считать ряд в таблице (строку массива), не зная значения индекса.

Прерывания. Протокол SNMP функционирует по принципу “запрос-ответ”. Однако в некоторых случаях необходима активная роль управляемого объекта для предоставления ему возможности передать о себе некоторую информацию (сбой, перезагрузка устройства, исчезновение связи и т.д.). Для этого предназначен блок данных Trap (прерывание). Существует всего семь

кодов прерываний (поле Generic trap). Код 7 обозначает прерывание, специфичное для данного производителя аппаратуры. В этом случае специальное поле Specifc trap будет являться расширением кода причины прерывания. При пересылке сообщения типа Trap, помимо кода, передается адрес агента (поле Agent address), пославшего сообщение, время посылки сообщения (поле Time stamp), код производителя аппаратуры (поле Enterprise) и дополнительная информация, состоящая из произвольного числа пар “имя объекта (поле Name) - значение объекта (поле Value)”. Таким образом, может быть передана информация, например, об исчезновении связи по каналу, типе канала, скорости альтернативного канала.

С протоколом SNMP связаны две основные проблемы:

- Слабая защищенность ограничения доступа, особенно к перезаписи переменных, недостаточны. Используемый механизм ограничения - это значение поля Cmmunity в пакете SNMP. Поле Community не зашифровано и передается открыто (причем с каждым пакетом!), поэтому имеется потенциальная возможность подделки управляющего воздействия.
- Большой избыточный поток информации. Для запроса набора переменных необходимо передать весь их список. при управлении большим числом устройств из одного места поток запросов становится существенным. Если принять во внимание, что все известные реализации SNMP базируются на протоколе UDP, не гарантирующем доставку информации, то может возникнуть множество повторных запросов, что еще больше загрузит сеть.

Протокол SNMPv2. Протокол SNMPv2 является не просто следующей версией протокола SNMP. Большое значение на него оказал протокол Secure SNMP и спецификации, определяющие удаленный мониторинг в сети (RMON).

Говоря об SNMP, фактически имеют в виду не только реализацию собственно протокола, но и некоторый набор баз MIB. В настоящее время стандартом де-факто стала поддержка следующих баз MIB: MIB-II; RMON; M2M.

Последняя из указанных баз описывает объекты управления, с помощью которых возможно построение распределенной системы управления. Пример такой системы показан на рис. 4.10 для системы OpenView фирмы Hewlett Packard.

С точки зрения языка описания структура данных в SNMPv2 не претерпела изменений: все объекты представлены листьями в дереве. Однако, состав объектов значительно расширился. В SNMPv2 вошло все дерево MIB-II, а также добавлены объекты, описывающие обмен данными по протоколу SNMP, введены описания посредников, станций управления, средств мониторинга и ограничения доступа.

В протоколе SNMPv2 появился новый тип запроса - массовый (bulk), в котором указываются две группы объектов. При ответе на каждый объект первой группы выдается следующий за ним объект и его значение, а при ответе на каждый объект второй группы - все следующие за ним объекты и их значения в соответствии с лексическим порядком. Под лексическим понимается порядок с точки зрения идентификаторов, выраженных в цифровом виде. Например, объект sysName (1.3.6.1.2.1.1.5) следует за объектом sysUpTime (1.3.6.1.2.1.1.3). Массовый запрос существенно снижает объем передаваемой по сети информации.

Имеются и другие отличия SNMPv2 от SNMPv1, но они не принципиальны.

Ограничение доступа. Рассмотрим модель ограничения доступа, предусмотренную в протоколе SNMPv2. Вводится понятие группы сетевых устройств, характеризуемой их атрибутами. К основным атрибутам относятся адрес IP члена группы, используемые алгоритмы аутентификации и шифрования и ряд других. В дальнейшем вместо перечисления всех атрибутов группы используется просто ее имя.

Необходимая политика безопасности осуществляется следующим образом. Определяются поддеревья дерева объектов, к которым необходимо обеспечить доступ. Узлы, порождающие эти поддеревья, перечисляются в базе данных просмотра (MIB View Database Group), и им присваиваются символические имена. В базе данных привилегий доступа (Access Privileges Database Group) устанавливается соответствие между символическими именами из базы данных просмотра и именами групп. При отправке по сети запроса SNMP с помощью локальных баз данных происходит проверка пославшего запрос узла (характеризуется адресом) на право доступа к запрашиваемой информации. Причем запрос может быть зашифрован. В настоящее время подразумевается, что аутентификация может происходить с помощью алгоритма MD5, а шифрование - DES.

Мониторинг. Центральным элементом системы мониторинга является база Manager-to-Manager MIB (RFC 1451), в которую входит группа snmpM2Mobjects, состоящая их трех таблиц и нескольких скалярных величин. Таблица alarm описывает условия, на которые нужно реагировать. Это описание состоит из имени и сути условия, а также интервала времени, необходимого для наблюдения за некоторым объектом (задается своим идентификатором), чтобы принять решение о возбуждении события. Условия могут быть двух типов: контролируемая величина стала больше или меньше заданного значения. При выходе контролируемого объекта за заданные пороговые ве-

личины будут возникать определенные события. Задавая соответствующий параметр, можно установить, что проверка условия будет относиться не к самой величине, а к ее изменению.

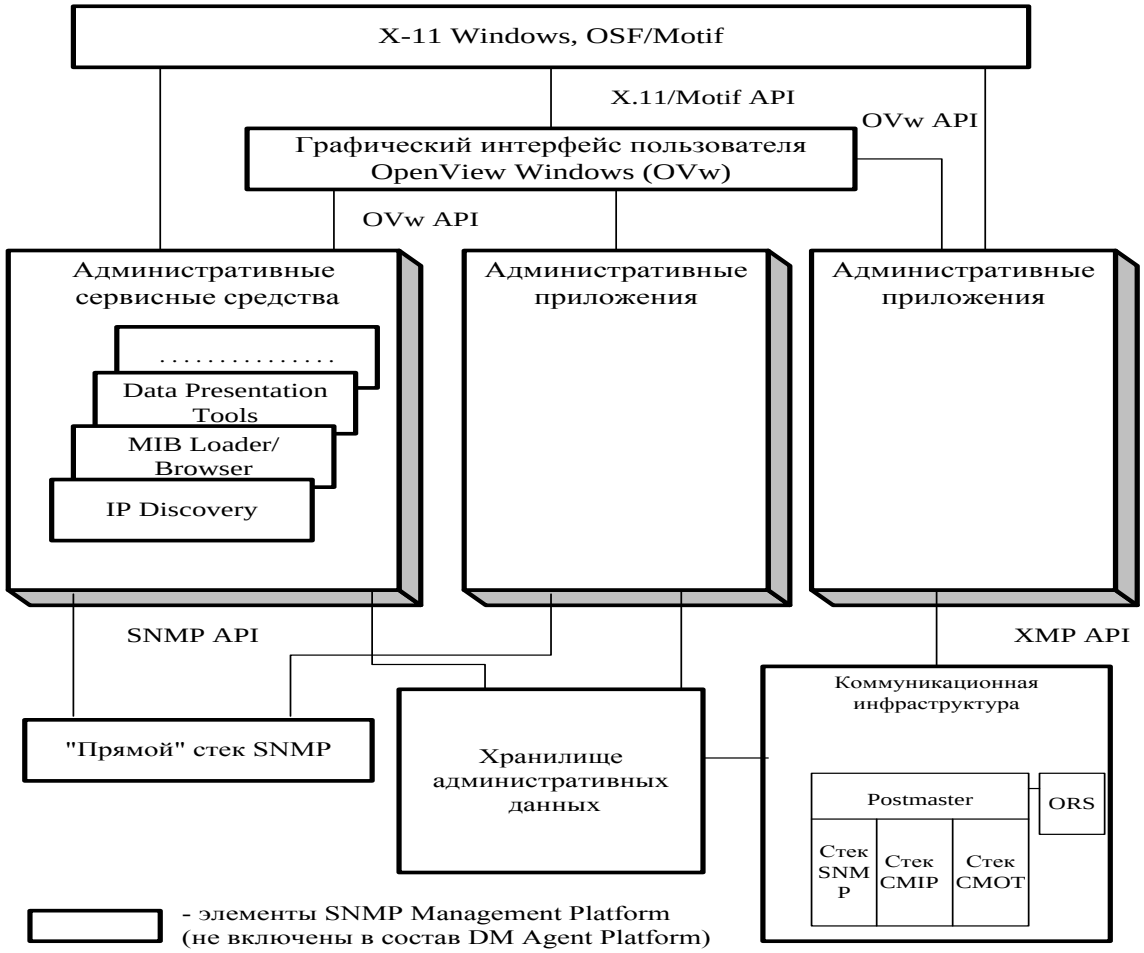


Рис. 4.10

Другие таблицы описывают события, которые надо инициировать при возникновении условий, оговоренных в таблице alarm.

На основе группы snmpM2Mobjects поведение объекта управления легко описать таким образом. что он будет оповещать систему управления о происходящих в нем критических событиях. При этом не требуется постоянно запрашивать объект управления.

Каждый объект управления, поддерживающий SNMPv2, может выступать в двух “обличиях”:

- оставаться объектом управления;
- выполнять функции системы управления.

Пример построения системы управления OpenView приведен на рис. 4.10.

4.7. Электронная почта

Электронная почта является наиболее популярным сетевым приложением. Она способна заменить автоответчиков и обычную бумажную переписку. Программы электронной почты нужно считать программным обеспечением такого же класса, как и программное обеспечение сервера. Его должен устанавливать сетевой администратор, который также заполняет списки пользователей и указывает все подключенные межсетевые мосты (если они имеются).

Доменная система имен DNS

Адресное пространство Internet образуется 32-битовыми IP-адресами, обрабатываемыми сетевыми программами. Для пользователей более удобны символические имена хост-машин, шлюзов и других машин. DNS - Domain Name System - была разработана как распределенная система адресных таблиц сети Internet и введена в действие в 1984 г. Доменная система имен является иерархической структурой имен, выполняющей ряд важных функций:

- организация адресного пространства Internet;
- обеспечение функционирования адресной базы данных, имеющих распределенную структуру;
- делегирование полномочий по адресации и именованию подсетей и их элементов на локальном уровне.

Имя домена - это имя административного объекта сети Internet. Например, имя домена: ccic.icsti.msk.ru относится к узлу ccic, для которого имя вышестоящего объекта: icsti.msk.ru. Следующий по иерархии объект имеет имя: msk.ru.

Последнее имя ru относится к имени домена верхнего уровня и обозначает административный объект - страну (Россия). Доменные имена всех подобных административных объектов - стран зарегистрированы по международному стандарту ISO 3166. Другими важными классами административных имен доменов верхнего уровня являются имена категорий сетей в соответствии с их целевой направленностью.

Имена домена	***Целевая характеристика ИС***
com	Коммерческая
edu	Образовательная
gov	Правительственная (США)
int	Международная организация
mil	Военная (США)
net	Сетевая (поставщики сетевых услуг)
org	Бесприбыльная организация

С 1992 г. начал действовать Европейский сетевой координационный центр RIPE NCC, который обеспечивает следующие услуги:

- хранение и выдача IP-адресов;
- ведение сетевой базы данных, содержащей информацию об IP-сетях, доменах DNS, IP-маршрутизаторах и различную контакт-информацию;
- архив сетевого программного обеспечения;
- архив технической документации;
- интерактивные информационные службы.

Регистрационный центр выдает только сетевую часть адреса, а права выдачи адресов для отдельных хостов сети делегируются администраторам сети, которая подключается к Internet в соответствии с общими правилами DNS.

Образование домена. Иерархия доменов, т.е. адресуемых частей Internet, образует единую систему имен сетей и хост-машин. Корневой, т.е. высший в иерархии домен сети Internet - это DDN NIC (DDN Network Information Center и соответственно для Европы RIPE NCC). Образование нового домена в сети Internet означает, что имя этого домена будет включено в распределенную базу данных адресов, используемых протоколом разрешения адресов при выборе маршрута передачи сообщений.

В DDN NIC и RIPE NCC регистрируются только домены верхнего уровня, большая часть доменов второго уровня и некоторые домены третьего уровня. Домены верхнего уровня обозначают целевые классы сетей (ru, com, mil и т.д.) и классы сетей одной страны. Каждому целевому классу сетей верхнего уровня соответствует домен верхнего уровня. Для каждого из этих доменов определен администратор домена, который обладает полномочиями регистрации сетей, относящихся к этому домену. Вся информация о сетевых доменах хранится в DNS в форме Ресурсных записей (Resource Records, RR). Интерактивная программа распознаватель-сервер обращается к DNS в режиме запрос-ответ и получает требуемые данные о доменах и входящих в них сетях и хост-машинах. Адресная информация находится в отдельных базах данных, размещаемых в нескольких хост-машинах. Все адресное пространство сети Internet делится на адресные доменные зоны, которые могут пересекаться. Каждая такая зона имеет главный сервер и может иметь несколько вторичных серверов с настроенной базой DNS. При регистрации в соответствующем комитете доменного имени, его либо включают в уже существующую зону и базу DNS, либо выдают домен для корпоративной сети. В этом случае за настройку доменной зоны, создание и ведение базы DNS собственной зоны отвечает администратор этой корпоративной сети.

Так как функции системы DNS заключаются в преобразовании доменного имени в IP-адрес и наоборот, то клиенты могут пользоваться доменными именами на своем прикладном уровне, а смена IP-адресов отражается лишь в базе DNS администратором и не влечет за собой изменений в прикладной части. Таким образом, важнейшим свойством доменов является то, что они представляют административные, а не топологические объекты.

Например, два домена

ccic.iscti.msk.ru и abcd.iscti.msk.ru

могут относиться к совершенно разным физическим сетям и не иметь прямого соединения между собой. Если топология сети изменится, то доменные имена и правила маршрутизации при этом не изменятся, поэтому всем пользователям такого приложения, как электронная почта, не надо знать топологию сети и следить за ее изменением - этим занимаются специальные сетевые администраторы, а пользователи работают с неизменными доменными именами.

Настройка базы адресного пространства и основные службы DNS

Прикладная программа обращается к DNS в основном для решения следующих задач:

- по имени домена определить его IP-адрес;
- по алиасу определить имя домена (и наоборот);
- по имени домена определить хосты, которые коммутируют сообщения в запрашиваемый домен;
- по IP-адресу найти имя соответствующего домена.

При решении этих задач службы DNS активно используют Ресурсные записи (Resource Records, RR), в которых хранятся атрибуты имен доменов.

Обслуживание системы DNS в операционной системе Unix осуществляется демоном /etc/named, запуск которого производится во время загрузки системы в соответствии с конфигурационными файлами.

В файле /etc/named.boot описано расположение всех конфигурационных файлов системы DNS. Файлы /etc/named.d/SGP.hosts, /etc/named.d/SGP.rev и /etc/named.d/SGP.local описывают первичный и вторичные DNS-серверы и базу ресурсных записей всех хостов, принадлежащих этой доменной зоне. Например, следующие поля содержат адресную информацию в соответствии с набором протоколов Internet (IN):

- IN A - адресная информация хоста. Для данного домена, именующего хост, указывается 32-битовый IP-адрес хоста.

- IN CNAME - алиас информация. Для данного домена, если он алиас, указывается официальное имя домена.
- IN MX - почтовая коммутационная информация. Для данного домена, именующего хост, указывается: стоимость почты (16-битовое целое число), имя домена для узла, коммутирующего почту в данный хост (почтового сервера).
- IN MINFO - данный хост является почтовым ящиком или содержит списки рассылки почты;
- IN MB - доменное имя почтового ящика;
- IN MG - раздел для почтовой группы;
- IN NS - информация об N-сервере. Для домена, включающего хост, на котором реализован N-сервер, указывается субдомен, для которого этот N-сервер является авторизованным.
- IN PTR - информация об иерархическом указателе.
- IN HINFO - сведения об аппаратной части сервера и установленной на него операционной системе.
- IN WKS - адресная информация хоста и протокол и сервис для обслуживания.
- IN TXT - содержит некоторую строку текста, для данного хоста, которая может быть использована администратором.

Могут быть и другие ресурсные записи, касающиеся вторичных почтовых серверов, имен пользователей и групп пользователей почтовых серверов и т.д.

Настройку системы DNS можно проверить с помощью утилиты nslookup.

Транспортная служба электронной почты

Среди обширного набора протоколов Internet следует выделить четыре основных протокола, на базе которых реализованы прикладные процессы электронной почты Internet:

- SMTP (Simple Mail Transfer Protocol) - простой протокол передачи сообщений, в основном текстового типа, в формате сообщений, соответствующих RFC-822;
- POP (Post Office Protocol) - протокол почтового отделения, обеспечивающий службу почтовых ящиков;
- NNTP (Network News Transfer Protocol) - протокол сетевой передачи новостей;
- DNS (Domen Name System) - доменная система имен, обеспечивающая соответствие (отображение) имен хостов и сетевых адресов.

Отражение основного протокола электронной почты SMTP на модель OSI приведено на рис. 4.11, общая схема взаимосвязи протоколов электронной почты и базовых протоколов Internet приведена на рис. 4.12.

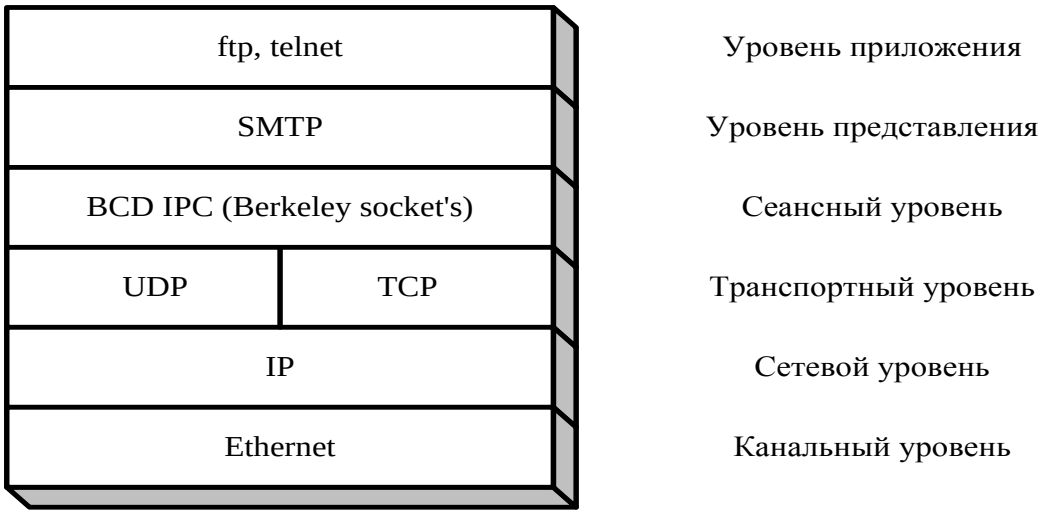


Рис. 4.11

Архитектурный принцип сети Internet состоит в том, что прикладные процессы взаимодействуют, т.е. обмениваются сообщениями, через механизм установления TCP-соединения.

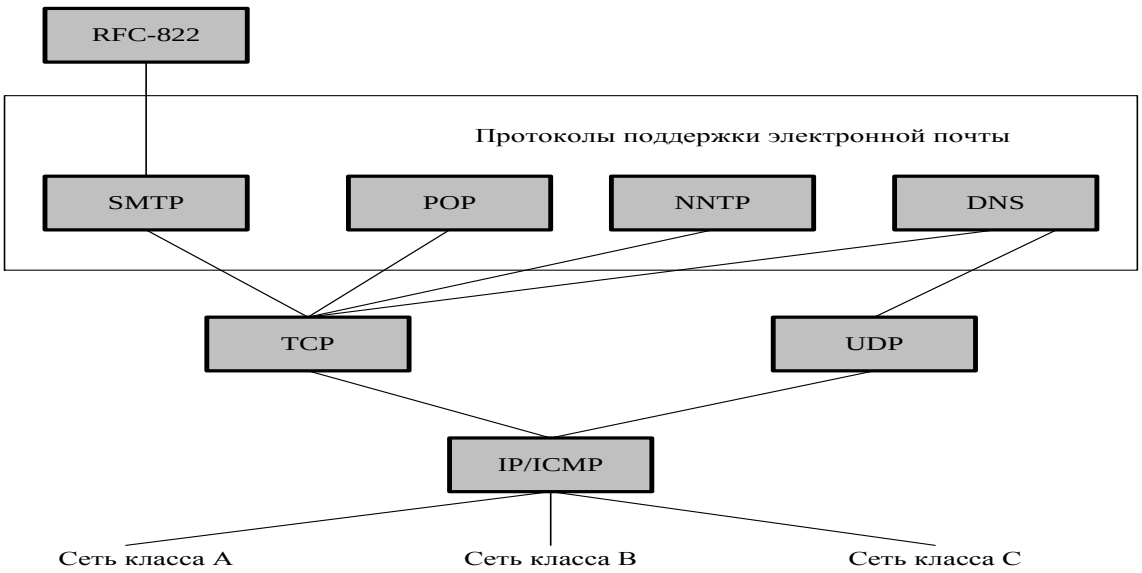


Рис. 4.12

Другой архитектурный принцип, используемый в Internet — для специальных (системных) задач, наоборот, не требует установления соединения: прикладные программы обмениваются дейтаграммами, используя протокол UDP.

Для электронной почты Internet преимущественно используется более надежный протокол TCP, а на уровне взаимодействия прикладных процессов используется модель “клиент-сервер”.

Адресация в Internet

В сети Internet каждый пользователь электронной почты получает свой адрес, записываемый в сообщениях стандартным образом:

local-name@domain

Строка “domain” идентифицирует административную единицу (региональное почтовое отделение, учреждение или администрацию локальной сети), обладающую определенными правами и полномочиями по отношению к своим подчиненным объектам, имеющим, как правило, иерархическое подчинение. Администратор службы DNS отвечает за уникальность всех имен своих локальных объектов и за фактическое использование электронной почты внутри своего домена.

Протокол SMTP (Simple Mail Transfer Protocol)

Основной транспортной службой электронной почты в Internet является подсистема SMTP (Simple Mail Transfer Protocol), реализующая протокол передачи простой почты. SMTP является протоколом коммутационного типа, т.е. с промежуточным хранением и передачей сообщений в отдельных сетевых узлах. Каждое сообщение SMTP поэтому содержит две части: конверт и содержание.

Конверт SMTP состоит из:

- адреса отправителя сообщения (originator address);
- одного или нескольких адресов получателя (recipient address);
- параметров режима доставки сообщения.

Адреса содержат имена доменов в их каноническом виде.

SMTP обеспечивает четыре различных режима доставки:

- доставка сообщения в указанный почтовый ящик получателя;
- доставка и отображение сообщения на каждом терминале получателя, на которых в это время получатель активен:
- доставка и отображение на терминал (если получатель активен) или пересылка сообщения в почтовый ящик получателя;
- доставка сообщения в почтовый ящик получателя и, если получатель активен, отображение сообщения на его терминале.

Как правило, используется при настройке в настоящее время только первый режим.

Списки рассылки (Mailing Lists). Правила адресации SMTP предусматривают использование специального группового адреса или алиаса. Алиас - это адрес или адресная переменная, которая может принимать несколько фактических значений. Значение алиаса - это адрес получателя или адрес почтового ящика. Алиас определяет некоторый список рассылки сообщений или группу адресов. Процедура адресного расширения выполняется программой электронной почты (агентом передачи сообщений) и состоит в генерации списка значений алиаса. Значения этого списка, т.е. адреса получателей, называются подписчиками списка рассылки. Например, если список рассылки называется list, то администратор этого списка именуется как list-request. Если у администратора списка сетевой адрес ***root@ccic.icsti.msk.ru***, то подписчики используют адрес ***root-request@ccic.icsti.msk.ru*** для контактов с администратором этого списка.

Таким образом, список рассылки - это группа адресов, связанная с администратором этого списка. Когда в конверте указывается алиас некоторого списка рассылки в качестве адреса получателя, то применяется процедура list explosion, которая создает конверт с тем же содержанием, но адрес отправителя в этом конверте устанавливается как адрес администратора списка рассылки. Адреса получателей устанавливаются в новом конверте по значениям алиаса, т.е. по адресам подписчиков. Первоначальный (исходный) конверт уничтожается и на его место подставляется новый конверт. После этого сообщение анализируется модулем MTA (Mail Transfer Agent) и обрабатывается стандартными процедурами агента передачи сообщений.

Персональные алиасы. Агенты пользователя обрабатывают алиасы до отправки сообщения. Алиасы удаляются из заголовка конверта и заменяются на адреса подписчиков.

Различные способы расширения алиасов можно сравнить следующим образом:

- системный алиас - MTA заменяет адрес получателя в конверте;
- список рассылки - MTA создает новый конверт, для которого отправителем является администратор почты, а получателями - подписчики;
- персональный алиас - UA (User Agent) расширяет заголовки перед отправкой сообщения.

Протокольные механизмы SMTP. SMTP-сервер постоянно прослушивает порт 25 TCP. Клиент устанавливает TCP-соединение с SMTP-сервером и начинает ожидать подтверждения от сервера. SMTP-сервер подтверждает:

- соединение с клиентом установлено (1);
- состояние локального MTA активное (2).

Если клиенту пришло подтверждение от сервера (1)-(2), то клиент идентифицирует себя и начинает одну или несколько SMTP-транзакций.

Когда клиент завершает сессию связи с SMTP-сервером, то он передает команду завершения и ожидает от сервера подтверждения. Затем клиент завершает TCP-соединение. При этом клиент не должен указывать промежуточные адреса и решать вопросы маршрутизации. Маршрутизация полностью возлагается на службу DNS, а SMTP-сервер, даже если получает в команде маршрутную цепочку адресов, обязан игнорировать их обработку. SMTP-сервер отвечает клиенту стандартными сообщениями вида: код-ответа-текстовая строка.

При взаимодействии с сервером используются протокольные команды HELLO, MAIL FROM, RCPT TO, DATA, NOOP, RSET, QUIT, SEND FROM, SOML FROM, VRFY, EXPN, TURN, HELP, аргументами которых являются имена доменов, текстовые строки или значения полей сообщения, адреса отправителя и получателя.

Сообщение, введенное на PC, обрабатывается UA, который устанавливает SMTP-сессию с локальным сервером, сервер подтверждает получение сообщений командой 250 recipient ok. Взаимодействие «Клиент-сервер» проходит последовательно, без анализа каких-либо ошибок или особых ситуаций.

Протокол NNTP

NNTP (Network News Transfer Protocol) - это специальный сетевой протокол для передачи NEWS-сообщений. Задачами протокола является обеспечение выполнения трех служб:

- распределение новых групп подписчиков для NNTP- сервера;
- обработка новых сообщений, поступающих на NNTP- сервер;
- обработка новых сообщений, рассылаемых клиентами NNTP- сервера.

NNTP-протокол по принятой в семействе TCP-протоколов схеме:

- NNTP-прослушивает порт 119 TCP;
- клиент устанавливает TCP-соединение с NNTP-сервером и ждет подтверждения;

- NNTP-сервер посылает подтверждение клиенту и разрешает послать новое News-сообщение;
- клиент инициирует одну или несколько транзакций с NNTP-сервером;
- клиент завершает TCP-соединение.

NNTP обеспечивает передачу News-сообщений в двух режимах:

- потоковая рассылка новостей;
- просмотр файла новостей удаленным пользователем.

4.8. Прикладные протоколы Internet

Протокол Gopher

Этот протокол разработан в университете штата Миннесота как удобное средство распространения различной электронной информации: новости, обмен компьютерными программами, пересылка сообщений и т.д. Фактически Gohper позволяет организовывать распределенную систему обмена документами. Информация хранится физически в отдельных компьютерных архивах, но через Internet может быть легко найдена, запрошена и доставлена на компьютер пользователя. При выборе клиентом определенного пункта меню сервер посылает детальную информацию об информационных объектах, представляемых этим пунктом меню. Gopher поддерживает следующие типы объектов:

0 - файл;
1 - директория;
2 - телефонный справочник;
3 - ошибка;
4 - файл Bin Xex Macintosh$
5 - DOS двоичный архив;
6 - файл Unix uuen;
7 - сервер Индекс-поиска;
8 - текстовое сообщение сессии Internet;
9 - двоичный файл;
10 - 3270 сессия;
s - звуковой;
g - GIF тип;
M - MIME тип;
h - html (Hyper Text Markup Language) тип;
I - тип изображение;
i - внутристроковый текстовый тип данных.

Взаимодействие с сервером осуществляется на основе TCP/IP, для пересылки информации используется протокол FTP. Использование единой прикладной среды Gopher для интеграции различных сетевых служб создает у пользователя иллюзию работы на едином виртуальном пространстве.

Расширения MIME

На почтовых шлюзах Internet использует стандарт SMTP, поэтому они не рассчитаны на передачу и обработку больших почтовых сообщений. Кроме того, необходимо специальное “кодирование” сообщений для последующей передачи. Эта операция предназначена для гарантированной передачи двоичной (не в текстовом формате) информации ее получателю. Сегодня для передачи текстовых файлов практически любой почтовый шлюз использует 7-битную кодировку. Поэтому, чтобы передать двоичную информацию в виде такого текстового файла, необходимо специальное кодирование. Получили распространение два алгоритма кодирования: BinHex и uuecode, не являющиеся частью MIME (Multipurpose Internet Mail Extensions). Все эти алгоритмы кодирования преобразуют двоичное сообщение в текстовый файл, который может быть послан по каналам электронной почты с использованием протокола SMTP.

В настоящее время все наиболее популярные пакеты электронной почты используют средства кодирования MIME. Сегодня MIME благодаря развитию S/MIME позволяет снабжать объекты MIME электронной цифровой подписью, шифровать их и посылать любые графические файлы, что значительно повышает привлекательность почты Internet для различных типов организаций.

Протокол X.400

X.400 – это набор стандартов ISO/CCITT по электронной почте. Шлюз между Internet-электронной почтой и X.400 приведен на рис.4.13. Во-первых, X.400 принимает во внимание число различных видов соединений; он может обрабатывать простые текстовые сообщения так же, как и двоичные файлы и видео. Во-вторых, X.400 разработан для глобальной коммуникации; он определяет, как отдельные системы должны связываться друг с другом. В-третьих, X.400 имеет поддержку: он был официально подтвержден целым рядом компаний и правительствами различных стран.

Технология X.400. Та часть почты, с которой напрямую работает пользователь, называется UA (User Agent). Сообщение, переданное в UA, передается затем агенту по передаче сообщений MTA (Message Transfer Agent), ко-

торый отвечает за отправку сообщения. В среде LAN UA на рабочих станциях должны посылать свои сообщения в MTA на сервере, выполняющем роль шлюза X.400.

Агент MTA обращается к UA и другим MTA. Сети X.400 не требуют специальной проводки или коммуникационных протоколов, хотя популярная реализация X.400 использует централизованные телекоммуникационные сети с коммутацией пакетов X.25. Одно из самых замечательных свойств X.400 заключается в том, что квитанция о получении сообщения от системы X.400 является юридическим подтверждением электронной доставки документа так же, как для телетекста. Это не так для многих других систем электронной почты.

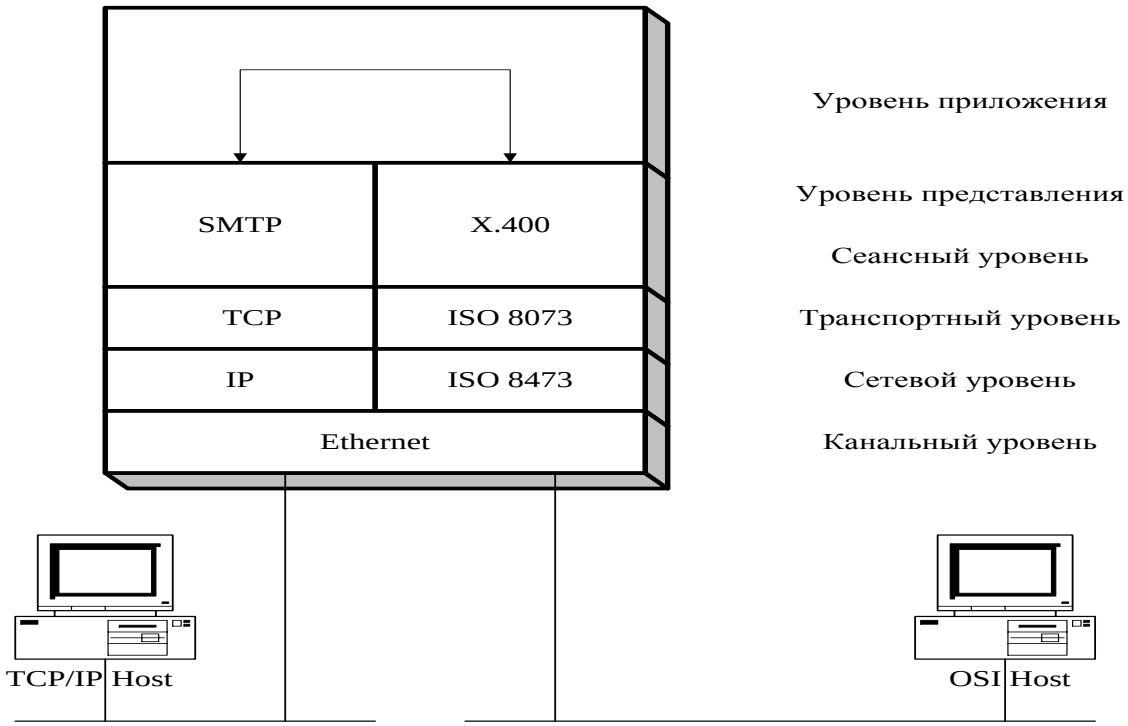


Рис. 4.13

Тело письма может включать текст, изображение факса, закодированные речевые сообщения, видеотекст и специальные форматы, которые позволяют X.400 передавать данные в формате, не определенном в спецификации, такие как графические файлы и электронные таблицы. Недостатком спецификации X.400 является отсутствие удобного справочника пользователя. Если необходимо послать сообщение X.400, то необходимо знать почтовый адрес (имя и организацию) получателя. В настоящее время интенсивно развивается стандарт X.500, который призван решить эту проблему.

Однако X.400 занимает достаточно много памяти: LAN X.400 на ПК может использовать около 600 Кбайт памяти, что значительно больше того, что требуется для электронных почт Internet, поэтому X.400, используя шлюз, может работать с другими почтовыми приложениями клиентов.

Протокол X.500

В соответствии с X.500 стандартом OSI/CCITT пользователи и прикладные процессы существуют подобно программам, и файлы данных могут быть вставлены в хранящийся в компьютере справочник. Справочная база данных может разделяться несколькими системами и, следовательно, является распределенным справочником, поэтому никакие два элемента X.500 не могут быть идентичными.

X.500 стандарт является большим, чем стандарт на базу данных о сетевых пользователях и ресурсах: это еще и стандарт на связывание отдельных X.500 справочников. В идеале можно найти информацию обо всех и обо всем, если заведена запись в X.500 справочнике.

Протокол X.500 быстро развивается и поддерживается сегодня многими разработчиками корпоративных баз данных и больших почтовых систем. Возможно этот стандарт станет основой создания глобального сетевого справочника, тогда пользователь в любой точке мира будет в состоянии связываться с любым другим пользователем.

Протокол LDAP

При администрировании сети на определённом этапе возникает проблема централизованного контроля доступа к файловым системам серверов. Это особенно актуально для крупных сетей (100 и более пользователей). При входе в систему каталогов пользователь должен получать доступ только к «своим» файлам. При этом сама система каталогов может находиться на нескольких серверах. Развитием глобального протокола X.500 для сетей TCP/IP стала служба директорий - LDAP (Lightweight Directory Access Protocol).

Раньше в Unix существовала только одна возможность централизованной аутентификации - службы NIS, работающие через UDP. Протокол LDAP построен по объектно-ориентированному принципу и позволяет с лёгкостью добавлять и изменять объекты при помощи файлов схем. Кроме этого, LDAP обладает большими возможностями в плане структурированности и работы с клиентами различных платформ, т.к. множество приложений поддерживают LDAP. На основе LDAP легко построить гетерогенные сети, для которых информация об объектах хранится в древовидной структуре.

Объекты верхнего уровня являются контейнерными, т.е. могут содержать другие объекты. Объекты нижнего уровня обычно такой особенностью не обладают. Схема LDAP не закрепляет жестко иерархию объектов, дерево может строиться в различном порядке. При работе интернет-дерева может использоваться иерархия определений dc (компонент доменных имен). Древовидная структура позволяет точно определить нахождение пользователя или иного объекта. Работая с LDAP, необходимо указывать полное описание объекта. Для обозначения полного описания (+имени) объекта относительно скелета дерева используется термин DN (distinguished name - назначенное имя, контекст). Например, для имени test.oblect.ru это будет выглядеть следующим образом: dn: dc=test,dc=object,dc=ru.

Модуль LDAP является обычно дополнительным компонентом, поэтому поиск вначале идёт в локальных файлах, а потом уж в LDAP. Ещё одной интересной возможностью LDAP является возможность проверки хоста. Нельзя, чтобы любой человек, занесённый в базу LDAP, мог пользоваться любым хостом, поэтому можно добавить к учётным записям пользователей хосты, с которыми эти пользователи могут работать.

Настройка почтовых серверов для работы с базой LDAP также не представляет особой проблемы.

Реплики - это одна из мощных возможностей системы директорий LDAP. Реплики - это раскопирование базы данных LDAP по нескольким серверам, т.е. фактически само дерево «размазывается» по серверам. Реплика существует между несколькими серверами, один из которых является первичным (primary) (это похоже на службу DNS, только немного для других целей). В таком случае первичный сервер обрабатывает запросы на запись и на чтение, а вторичные - только на чтение. При модификации данных на главном сервере он заходит на все вторичные и синхронизирует их базы со своей.

При использовании протокола LDAP можно для клиентов указать определённый LDAP-сервер и создать основной сервер, который будет синхронизировать все вторичные. При этом любые изменения в базе любого из LDAP-серверов будут раскопированы на все сервера, включая основной.

Если вторичный сервер не сможет связаться с основным, то клиент, подключенный к вторичному серверу, сможет читать информацию, но изменять её не сможет. По такой распределённой схеме можно создавать сколь угодно большую сеть, так как основными являются запросы на чтение, и нагружаться будут вторичные сервера.

Применение LDAP является идеальным решением централизованного управления правами доступа на всех уровнях. Использование реплик делает LDAP достаточно надёжным. Кроме этого, LDAP является довольно быстрым механизмом поиска и, что очень важно, хранит всё в древовидной базе данных.

5. ТЕХНОЛОГИЯ РАБОТЫ TCP/IP

5.1. Драйверы сетевых адаптеров

Первоначально драйверы устройств разрабатывались в расчете лишь на один протокол (TCP/IP, IPX и т.д.). Это означало, что для запуска нескольких сетевых приложений требовалось перезагружать компьютер, чтобы установить или снять те или иные драйверы сетевых адаптеров. В результате начались попытки найти более удачные способы взаимодействия сетевых адаптеров с протоколами локальных сетей. В результате появилось не одно, а сразу три решения этой проблемы, ставшие достаточно популярными.

Сегодня каждый выпускаемый сетевой адаптер снабжается драйверами всех трех спецификаций, поэтому пользователь, устанавливая стек протоколов на своем ПК, может выбрать тот или иной стандарт в зависимости от его дальнейшего использования для работы в ИС.

Спецификация NDIS. Фирмы Microsoft и 3Com в 1989 г. совместно разработали спецификацию NDIS (Network Device Interface Specification), которая определяет способ работы драйвера сетевого адаптера с несколькими сетевыми протоколами (до четырех). Так, благодаря NDIS-драйверам сетевые операционные системы LAN Manager, Windows 3.11, Windows-95 или Windows NT могут работать с тем же драйвером адаптера, что и TCP/IP.

Большинство производителей поддерживают версию NDIS 2.0, называемую также NDIS реального режима (real mode NDIS). Драйверы NDIS 2.0 работают на компьютере с процессором 80286 и выше в среде MS-DOS, являются 16-разрядными, запускаются в реальном режиме и остаются резидентными в основной памяти.

MS Windows, начиная с версии 3.11 for Workgroups, поддерживает NDIS версии 3.0, которую называют NDIS расширенного режима (enhanced mode NDIS). Спецификация NDIS 3.0 появилась в процессе разработки Windows NT. Драйверы NDIS 3.0 работают в системах с процессором 80386 и выше, являются 32-разрядными и используют расширенную (extended) память в защищенном режиме.

Спецификация ODI. Фирма Novell иначе решила проблему работы сетевого адаптера с несколькими протоколами. Спецификация ODI (Open Datalink Interface) организована примерно так же, как и NDIS, однако с совершенно другим программным интерфейсом.

Пакетные драйверы. Третьим и наименее популярным решением проблемы объединения различных сетевых протоколов являются пакетные драйверы - предшественники NDIS и ODI. Фирма FTP Software - один из старейших производителей приложений TCP/IP для MS DOS - не захотела включать свой стек протоколов TCP/IP в драйверы для каждого сетевого адаптера. Чтобы облегчить себе труд, фирма разработала для сетевых драйверов прикладной программный интерфейс под названием Packet Driver Specification.

Выбор спецификации драйвера

Для сетевого администратора разницы между тремя рассмотренными способами нет. Популярные сетевые адаптеры имеют в комплекте поставки все виды драйверов. Тем не менее нужно выбрать драйвер, наиболее подходящий для используемого протокола.

Если одновременно необходимо работать с несколькими сетями, то:

- для сетей NetWare следует начать с ODI;
- для сетей LAN Manager, Windows for Workgroups, Windows NT или Pathworks предпочтителен NDIS;
- в PC, работающих только с TCP/IP, пакетные драйверы, т.к. они занимают меньший объем памяти и легче устанавливаются.

Сочетание некоторых протоколов может вызывать определенные трудности. Например, если пользователь хочет иметь доступ и к Windows for Workgroups (работает с NDIS), и к NetWare (работает с ODI), то придется устанавливать так называемую “прокладку” API, адаптирующую друг к другу разнотипные драйвер – устройства и схему вызовов. В этом случае следует добавить специальный драйвер типа ODINSUP, позволяющий стекам, разработанным в соответствии со спецификациями NDIS и ODI, работать с ODI-драйверами.

Однако добавление нового уровня к стеку протоколов увеличивает вероятность ошибок и нестыковок. Многие адаптирующие “прокладки” API относятся к категории условно-бесплатных продуктов и не отличаются высокой надежностью. Отображение размещения стека протокола TCP/IP на модель OSI приведено на рис. 5.1.

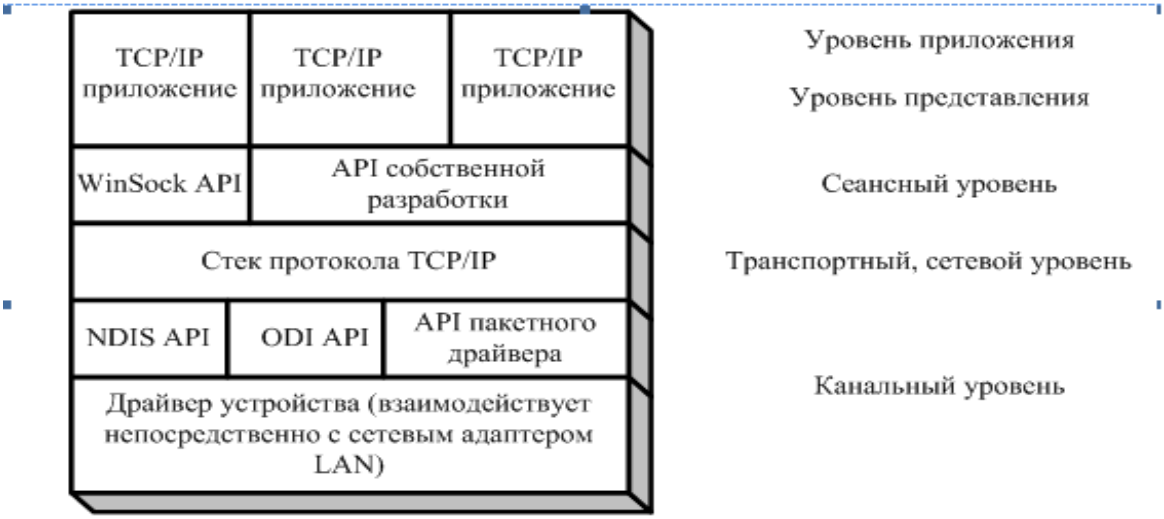


Рис. 5.1

5.2. Установка стека TCP/IP

Наряду с тремя различными взглядами на интеграцию драйверов устройств в MS-DOS и Windows существуют три различных способа размещения стека TCP/IP над этими драйверами (табл. 5.1). Прежде всего производителям приложений и стеков TCP/IP приходится помнить как о ПК с MS-DOS, так и о ПК, в которых поверх MS-DOS функционирует Windows.

В разработанной компанией Microsoft архитектуре MS-DOS наиболее подходящим механизмом для выполнения требуемых функций являются резидентные (TSR) программы. С их помощью можно создать некое подобие х своего рода многозадачной среды MS-DOS: оставить протокол в оперативной памяти и переключать контекст между приложениями.

Резидентные программы TSR имеют существенный недостаток: они должны располагаться в первом мегабайте памяти (в реальном режиме) и резервировать необходимую область памяти при первоначальном запуске системы. Однако эта часть памяти является очень ценным ресурсом, и расходовать его следует экономно, чтобы использовать для других нужд.

TSR-программы могут вызвать замедление работы системы, т.к. их взаимодействие с Windows-приложениями трубует переключения центрального процессора в реальный режим и обратно в защищенный.

По этой причине следующим шагом в развитии технологии стеков TCP/IP стало использование программной конструкции DLL (Dynamic Link Library). Модулями DLL являются 16 или 32-разрядные программные библио-

теки, динамически загружаемые и выгружаемые ядром Windows по запросу приложений.

Сетевой стек на базе DLL целиком находится “во власти” приложений, запущенных под Windows. Windows-задача с тем же или более высоким приоритетом может полностью блокировать работу сетевого сервиса TCP/IP. У DLL имеется преимущество с точки зрения надежности программного обеспечения. Библиотеки DLL работают в самом внешнем кольце (третий уровень) в защищенном режиме системы управления памятью фирмы Intel, которая называется User Mode. Сбойное приложение в User Mode не будет “бродить” по адресам ядра Windows и других приложений. Оно изолировано аппаратными средствами управления.

Третьей распространенной технологией, которую большинство производителей старается включить в новые версии своего программного продукта, является драйвер виртуального устройства VxD (Windows Virtual Device Driver).

Стек TCP/IP на базе драйверов VxD, работает как все драйверы устройств в системе Windows, например, как драйвер мыши он функционирует в наиболее привилегированном режиме управления памятью уровня 0 (Kernel Mode), что дает преимущество по производительности перед стеками на базе DLL. Стек TCP/IP на базе VxD, подобно TSR, постоянно находится в памяти на все время своей работы, а это плохо для систем с ограниченными ресурсами. Поскольку VxD работает в режиме Kernel Mode, то ошибка в драйвере TCP/IP может привести к краху всей системы.

Выбор между TSR, DLL и VxD зависит от рабочей среды, используемой на ПК. Резидентные TSR-программы наилучшим образом подходят для систем, где одни задачи решаются в “чистой” MS-DOS, а другие - в Windows.

Ни DLL, ни VxD стеки не способны поддерживать приложения TCP/IP, работающие вне Windows. Так как библиотеки DLL динамически загружаются и выгружаются, они работают лишь с “чистыми” Windows-приложениями. Для программ, функционирующих в окне DOS внутри Windows, приемлемы как VxD, так и TSR-резидентные программы.

После того, как ответы на вопросы о типе драйвера и стека получены, обратим внимание на пути обеспечения работоспособности приложений, ориентированных на TCP/IP. Большинство производителей сетевого ПО не делают пока различия между стеком протоколов и приложениями. Обычно при покупке стека пользователь получает некоторый набор приложений, который может и не удовлетворить его потребности. Поскольку спектр приложений все больше расширяется, ни один из производителей не способен полностью

его контролировать. По этой причине, возможно, придется использовать приложения нескольких производителей.

Таблица 5.1

Стеки TCP/IP и драйверы устройств

Тип стека	Поддержка систем 80280C	Наивысшая производительность	Возможность работы в среде MS DOC	Поддержка окна DOC в WINDOWS	Минимальное использование обычной памяти DOC
VXD		+		+	+
TSR	+	+	+	+	
DLL	+				+

В связи с этим особое значение приобретает мобильность приложений, ключом к которой в мире Windows является соответствие стандарту программного интерфейса Windows Sockets, или WinSock. Он представляет собой адаптированную для Windows версию популярного сетевого программного интерфейса sockets, разработанного для системы Unix Калифорнийским университетом Беркли.

Подобно тому, как производители стеков TCP/IP организуют интерфейс снизу с драйвером адаптера, используя пакетные драйверы, NDIS или ODI, они могут обеспечивать интерфейс и сверху, с приложениями, используя интерфейс WinSock. В настоящее время почти каждый производитель обеспечивает интерфейс WinSock со своими стеками. Благодаря ему можно, в частности, отключить Windows-приложение от конкретного стека TCP/IP.

ЗАКЛЮЧЕНИЕ

Представленный учебный материал посвящен достаточно активно развивающейся области вычислительной техники – методам, средствам и протоколам доступа к информационным ресурсам. В последнее время все больше и больше новых программных стандартов и средств вычислительной техники внедряются в разработки современных сетей. Поэтому пособие направлено на подробное и точное описание современных стандартов, методов и протоколов, которые появляются в последнее время и используются в различных видах сетей.

Для углубленного изучения учебной дисциплины, кроме приведенной литературы, рекомендуется изучать компьютерные периодические журналы, а также использовать различные информационные сайты в Internet, которые посвящены сетевым технологиям, а также имеются электронные материалы на сайте кафедры Автоматики и вычислительной техники и университетском сайте.

Приведенные в пособии данные могут помочь инженерам в процессе проектирования, эксплуатации и настройки различных типов современных информационных сетей.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Бройко В. Л. Вычислительные сети и телекоммуникации: Учебник для вузов / В. Л. Бройко — СПб.: Питер, 2003.- 213 с.
2. Головин Ю. А. Информационные сети и телекоммуникации: Учебное пособие.Ч.1/ Ю.А.Головин, А.А. Суконщиков— Вологда: ВоГТУ, 2003.- 151с.
3. Суконщиков А. А. Информационные сети и телекоммуникации: Учебное пособие.Ч.2/ А.А. Суконщиков — Вологда: ВоГТУ, 2005. -127 с.
4. Иртегов Д. В. Введение в сетевые технологии: Учебное пособие/ Д. В. Иртегов— СПб.: БХВ-Петербург, 2004.-242с.
5. Колбанев М. О. Модели и методы оценки характеристик обработки информации в интеллектуальных сетях связи / М. О. Колбанев, С. А.Яковлев – СПб: Изд. СПб гос. университета, 2002. -186 с.
6. Куроуз Дж. Компьютерные сети. Многоуровневая архитектура Интернета / Дж. Куроуз, К. Росс. — СПб.: Питер, 2004.-414 с.
7. Головин Ю.А. Информационные сети :учебник для студ. учреждений высш.проф. образования/ Ю.А.Головин, А.А. Суконщиков, С. А.Яковлев. – М.: Академия, 2011. -384 с.
8. Олифер В. Г. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов, 3-е изд./ В. Г. Олифер, Н. А. Олифер — СПб.: Питер, 2007. -202 с.
9. Олифер В. Г. Новые технологии и оборудование IP-сетей / В. Г. Олифер, Н. А. Олифер. - СПб.: Изд. «Питер», 2004. -242 с.
10. Столлингс В. Современные компьютерные сети, 2-е изд./ В. Столлингс . — СПб.: Питер, 2003. -386 с.
11. Танненбаум Э. Компьютерные сети / Э. Танненбаум. — СПб.: Питер, 2003. -424с.
12. Фейт С. TCP/IP: Архитектура, протоколы и реализация. 2-е изд. / С. Фейт. – М: "Лори", 2006. – 342 с.

Оглавление

Учебное издание

Алексей Александрович Суконщиков

МЕТОДЫ, СРЕДСТВА И ПРОТОКОЛЫ ДОСТУПА К СРЕДЕ И УДАЛЕННЫМ ИНФОРМАЦИОННЫМ РЕСУРСАМ

Учебное пособие

Редактор Л.А. Перерукова

Подписано в печать 6.11.2013 г.
Формат 60x90/16. Бумага офисная.
Печать офсетная. Усл. печ. л. 9,2.
Тираж 35 экз. Заказ 472.

Отпечатано: РИО ВоГТУ
160000, г. Вологда, ул. Ленина, 15.