

Д. КОЛИСНИЧЕНКО

САМОУЧИТЕЛЬ СИСТЕМНОГО администратора Linux

Дистрибутивы Fedora 13, Mandriva
2010.1 Spring, openSUSE 11.3,
Ubuntu 10

Установка и настройка
операционной системы

Подробное рассмотрение файловой
системы Linux

Настройка сети и маршрутизации
без конфигураторов

Брандмауэры iptables и ebtables,
chroot-окружение

Настройка серверов: Web, FTP, DNS,
DHCP, почтового и сервера баз данных

Прокси-серверы Squid и SquidGuard

Linux-сервер в Windows-сети: свой
среди чужих

Виртуальные частные сети (VPN)

Создание LiveCD

Сетевой сканер nmap

Система управления доступом Tотоуо

Защита и оптимизация Linux-сервера

Автоматизация задач с помощью bash

Программные RAID-массивы

СИСТЕМНЫЙ
АДМИНИСТРАТОР

Денис Колисниченко

**САМОУЧИТЕЛЬ
системного
администратора
Linux**

Санкт-Петербург

«БХВ-Петербург»

2011

УДК 681.3.06
ББК 32.973.26-018.2
К60

Колисниченко Д. Н.

К60 Самоучитель системного администратора Linux. — СПб.: БХВ-Петербург, 2011. — 544 с.: ил. — (Системный администратор)

ISBN 978-5-9775-0639-7

Описаны основы сетевого взаимодействия, планирование и монтаж сети (Ethernet и Wi-Fi), настройка сети и маршрутизации без конфигураторов. Даны примеры настройки различных типов серверов: Web, FTP, DNS, DHCP, почтового сервера, сервера баз данных. Рассмотрены дистрибутивы Fedora 13, Mandriva 2010.1 Spring, openSUSE 11.3, Ubuntu 10, файловая система Linux, установка и базовая настройка Linux, а также связки Apache + MySQL + PHP. Особое внимание уделено защите сетевых сервисов и оптимизации работы сервера: использованию брандмауэров iptables и ebttables, прокси-серверов Squid и SquidGuard, созданию chroot-окружения, управлению доступом с помощью системы Tomoyo, настройке VPN-сервера, аудиту сети при помощи сетевого сканера nmap. Приведены практические рекомендации по стратегии администрирования и уходу за аппаратными средствами, работе Linux-сервера в Windows-сети, созданию LiveCD, автоматизации задач с помощью bash, использованию программных RAID-массивов.

Для администраторов Linux

УДК 681.3.06
ББК 32.973.26-018.2

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Григорий Добин</i>
Компьютерная верстка	<i>Натальи Караваевой</i>
Корректор	<i>Виктория Пиотровская</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Елены Беляевой</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 03.11.10.

Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 43,86.

Тираж 1800 экз. Заказ №

"БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Санитарно-эпидемиологическое заключение на продукцию № 77.99.60.953.Д.005770.05.09 от 26.05.2009 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12.

Оглавление

Введение.....	1
ЧАСТЬ I. ОСНОВЫ АДМИНИСТРИРОВАНИЯ.....	3
Глава 1. Становимся администратором.....	5
1.1. Краткая история Linux.....	5
1.2. Почему именно Linux?	7
1.3. Основные задачи системного администратора	7
Глава 2. Классификация сетей.....	9
2.1. Краткая история сетей.....	9
2.1.1. 1941–1975 годы.....	9
2.1.2. 1976–1982 годы.....	10
2.1.3. 1983–1989 годы.....	11
2.1.4. 1990–1995 годы.....	12
2.1.5. 1996–1999 годы.....	13
2.1.6. 2000 — наше время.....	14
2.2. Классификация сетей.....	14
2.2.1. По занимаемой территории	14
2.2.2. По топологии	15
2.2.3. По ведомственной принадлежности.....	17
2.2.4. По скорости передачи данных	17
2.2.5. По типу среды передачи данных	17
2.2.6. По способу организации взаимодействия компьютеров.....	17
2.3. Способы передачи данных в сетях.....	17
2.4. Модель OSI.....	19
2.5. Что такое протокол?	21
2.6. Адресация компьютеров	22
2.7. Система DNS	25

Глава 3. Основные сетевые устройства	26
3.1. Активное и пассивное сетевое оборудование	26
3.2. Оборудование, необходимое для построения Ethernet-сети	26
3.3. Оборудование, необходимое для построения сети Wi-Fi	30
3.4. Дополнительные сетевые устройства	31
Глава 4. Планирование сети	34
4.1. Важность планирования	34
4.1.1. Планирование как основа безопасности	35
4.1.2. Построение транспортной системы корпоративной сети	36
4.2. Обеспечение безопасности сети	38
4.2.1. Защита данных, передаваемых по публичным каналам связи	38
4.2.2. Выдача IP-адресов по рабочим местам	39
4.2.3. Привязка IP-адресов к MAC-адресам	39
4.2.4. Антивирусные серверные решения	39
4.2.5. Антивирусные клиентские решения	40
4.2.6. Необходим ли дежурный администратор?	40
4.3. Человеческий фактор	40
4.3.1. Ограничение доступа	40
4.3.2. Как быть с обиженными или уволенными сотрудниками?	40
4.3.3. Принцип "правая рука не знает, что делает левая"	41
4.3.4. Планирование безопасности серверной комнаты/этажа	41
4.4. Отдел системного администрирования и безопасности	42
4.4.1. Подбор персонала	42
4.4.2. Инструктаж отдела IT	42
4.4.3. Распределение задач и сфер ответственности	43
4.4.4. Контроль работы и иерархия	43
4.5. Программы для планирования сети	44
Глава 5. Монтаж Ethernet-сети	45
5.1. Развитие стандарта Ethernet	45
5.1.1. Модификации стандарта Ethernet	45
5.1.2. Стандарты Fast Ethernet (100 Мбит/с)	46
5.1.3. Gigabit Ethernet (1000 Мбит/с)	48
5.1.4. Наше будущее — 10 Gigabit Ethernet	48
5.2. Несколько слов о коллизиях	49
5.3. Монтаж сети	50
5.3.1. Основные компоненты Ethernet-сети	50
5.3.2. Подробнее о витой паре	51
5.3.3. Обжим витой пары	52
5.4. Ограничения при построении сети	55

Глава 6. Основы беспроводной сети. Монтаж беспроводной сети.....	58
6.1. Преимущества и недостатки беспроводной сети.....	58
6.2. Основные принципы работы беспроводной сети	59
6.3. Расширение спектра.....	61
6.4. Wi-Fi.....	62
6.5. Радиочастоты и каналы Wi-Fi.....	65
6.5.1. Стандарты 802.11b и 802.11g	65
6.5.2. Стандарт 802.11a	66
6.6. Режимы работы сети.....	67
6.7. Основные сетевые устройства беспроводной сети.....	68
6.8. Выбор точки доступа.....	69
6.8.1. Поддерживаемые точкой доступа стандарты	70
6.8.2. Область применения и радиус действия точки доступа	70
6.8.3. Антенна точки доступа	71
6.8.4. Алгоритм шифрования	71
6.8.5. Дополнительные функции.....	71
6.9. Настройка беспроводной сети	73
6.9.1. Выбор расположения точки доступа	73
6.9.2. Физическая установка точки доступа.....	75
6.9.3. Практическая настройка беспроводной сети.....	76
6.9.4. Настройка соединения Wi-Fi в Linux	81
ЧАСТЬ II. ЗНАКОМСТВО С LINUX	83
Глава 7. Особенности установки Linux	85
7.1. Системные требования	85
7.2. Параметры ядра.....	86
7.3. Проверка носителей.....	89
7.4. Изменение таблицы разделов	90
7.5. Выбор групп пакетов	95
7.6. Выбор графической среды	97
7.7. Установка пароля root	97
7.8. Создание учетных записей пользователей	99
7.9. Установка Linux по сети.....	100
7.9.1. Немного о загрузке и установке по сети	100
7.9.2. Подготовка загрузочного сервера.....	100
7.9.3. Настройка клиента	102
7.10. Проблемы при установке	102
7.10.1. Проблема с APIC	102
7.10.2. Ошибка: <i>kernel panic: VFS: Unable to mount root fs</i>	103
7.10.3. Проблемы с некоторыми LCD-мониторами	103
7.10.4. Сообщение <i>Probing EDD</i> и зависание системы.....	103

7.10.5. Список известных проблем в Mandriva Linux 2009	103
7.10.6. Не переключается раскладка в Fedora 13	104
7.11. Вход в систему и завершение работы	104
Глава 8. Командная строка Linux	106
8.1. Консоль	106
8.2. Переход в консоль и обратно	106
8.3. Выход из консоли и завершение работы (команды <i>poweroff</i> , <i>halt</i> , <i>reboot</i> , <i>shutdown</i>)	107
8.4. Как работать в консоли	108
8.5. Графические терминалы	109
8.6. Перенаправление ввода/вывода	110
8.7. Команды Linux	111
Глава 9. Файловая система	112
9.1. Файловые системы, поддерживаемые Linux	112
9.1.1. Выбор файловой системы	113
9.1.2. Linux и файловые системы Windows	114
9.1.3. Сменные носители	115
9.2. Особенности файловой системы Linux	115
9.2.1. Имена файлов	115
9.2.2. Файлы и устройства	115
9.2.3. Корневая файловая система и монтирование	116
9.2.4. Стандартные каталоги Linux	117
9.2.5. Ссылки: жесткие и символические	118
9.2.6. Задание прав доступа к файлам и каталогам	119
9.2.7. Специальные права доступа (SUID и SGID)	120
9.3. Монтирование файловых систем	120
9.3.1. Команды <i>mount</i> и <i>umount</i>	120
9.3.2. Файлы устройств и монтирование	121
9.3.3. Опции монтирования файловых систем	124
9.3.4. Монтирование разделов при загрузке	125
9.3.5. Подробно о UUID и файле <i>/etc/fstab</i>	127
9.3.6. Монтирование Flash-дисков	129
9.4. Настройка журнала файловой системы <i>ext3</i>	130
9.5. Файловая система <i>ext4</i>	131
9.5.1. Сравнение <i>ext3</i> и <i>ext4</i>	131
9.5.2. Совместимость с <i>ext3</i>	132
9.5.3. Переход на <i>ext4</i>	132
9.6. Псевдофайловые системы	133
9.6.1. Виртуальная файловая система <i>sysfs</i>	134
9.6.2. Виртуальная файловая система <i>proc</i>	134

9.7. Программы для разметки диска	138
9.7.1. Программа <i>fdisk</i>	138
9.7.2. Программа <i>parted</i>	140
Глава 10. Команды управления пользователями	145
10.1. Многопользовательская система.....	145
10.2. Пользователь <i>root</i>	146
10.2.1. Максимальные полномочия	146
10.2.2. Как работать без <i>root</i>	146
10.2.3. Переход к традиционной учетной записи <i>root</i>	150
10.3. Создание, удаление и модификация пользователей стандартными средствами	152
10.3.1. Команды <i>adduser</i> и <i>passwd</i>	152
10.3.2. Команда <i>usermod</i>	153
10.3.3. Команда <i>userdel</i>	154
10.3.4. Подробно о создании пользователей.....	154
10.4. Группы пользователей.....	155
10.5. Команды квотирования	155
ЧАСТЬ III. НАСТРОЙКА СЕТИ В LINUX.....	159
Глава 11. Настройка локальной сети	161
11.1. Несколько слов о монтаже сети.....	161
11.2. Файлы конфигурации сети в Linux	163
11.3. Настройка сети с помощью конфигуратора.....	165
11.3.1. Настройка сети в Linux Mandriva	166
11.3.2. Настройка сети в Fedora	173
11.3.3. Настройка сети в Debian, Ubuntu и Denix. Конфигураторы <i>nm-connection-editor</i> (NetworkManager) и <i>network-admin</i>	178
11.3.4. Конфигуратор <i>netconfig</i> в Slackware.....	181
11.4. Утилиты для диагностики соединения	181
11.5. Для фанатов, или как настроить сеть вручную	185
11.5.1. Конфигурационные файлы Fedora.....	186
11.5.2. Конфигурационные файлы openSUSE	188
11.5.3. Конфигурационные файлы Debian/Ubuntu	190
11.6. Команда <i>mii-tool</i>	190
11.7. Перед тем как перейти к следующей главе	191
Глава 12. Настройка ADSL-доступа к Интернету	192
12.1. Причина популярности DSL-соединений.....	192
12.2. Физическое подключение ADSL-модема	192

12.3. Настройка DSL-соединения в Fedora	193
12.4. Настройка DSL-соединения в openSUSE.....	195
12.5. Настройка DSL-соединения в Ubuntu	199
12.6. Настройка DSL-соединения в Mandriva.....	203
Глава 13. Подключение к сети Wi-Fi	204
13.1. О настройке Wi-Fi в Linux	204
13.2. Простая настройка (Ubuntu/Denix/Fedora)	204
13.3. "Тяжелый случай"	206
13.4. Возможные осложнения	209
Глава 14. Маршрутизация	210
14.1. Выбор маршрута, или краткое введение в маршрутизацию.....	210
14.2. Таблица маршрутизации ядра. Установка маршрута по умолчанию	211
14.3. Изменение таблицы маршрутизации. Команда <i>route</i>	215
14.4. Включение IPv4-переадресации, или превращение компьютера в шлюз	217
14.5. Протоколы маршрутизации	218
14.5.1. Routing Information Protocol	218
14.5.2. RIP-2.....	218
14.5.3. Open Shortest Path First.....	219
Глава 15. Брандмауэры iptables и ebtables	220
15.1. Что такое брандмауэр.....	220
15.2. Цепочки и правила	221
15.3. Использование брандмауэра iptables	223
15.4. Шлюз своими руками	226
15.5. Брандмауэр ebtables	231
ЧАСТЬ IV. ОПЕРАЦИОННАЯ СИСТЕМА LINUX.....	233
Глава 16. Загрузчики Linux	235
16.1. Базовые загрузчики	235
16.2. Конфигурационные файлы GRUB и GRUB2	236
16.2.1. Конфигурационный файл GRUB	236
16.2.2. Конфигурационный файл GRUB2	237
16.3. Команды установки загрузчиков	242
16.4. Установка тайм-аута выбора операционной системы. Редактирование параметров ядра Linux	242
16.5. Установка собственного фона загрузчика GRUB и GRUB2.....	245
16.6. Постоянные имена и GRUB	246
16.7. Две и более ОС Linux на одном компьютере	246
16.8. Загрузка с ISO-образов	248

Глава 17. Системы инициализации Linux	249
17.1. Начальная загрузка Linux.....	249
17.2. Система инициализации <code>init</code>	250
17.2.1. Файл <code>/etc/inittab</code>	250
17.2.2. Команда <code>init</code>	252
17.2.3. Команда <code>service</code>	252
17.2.4. Редакторы уровней запуска.....	252
17.3. Система инициализации <code>upstart</code>	255
17.3.1. Как работает <code>upstart</code> ?.....	255
17.3.2. Конфигурационные файлы <code>upstart</code>	256
17.4. Система инициализации Slackware	257
Глава 18. Пакеты и управление пакетами	259
18.1. Что такое пакет?.....	259
18.2. Репозитории пакетов	261
18.3. Программы для управления пакетами	262
18.4. Программы <code>rpm</code> и <code>rpmbuild</code> (все Red Hat-совместимые дистрибутивы).....	263
18.5. Графический менеджер пакетов <code>rpm-drake</code> (Mandriva)	264
18.6. Программа <code>urpmi</code>	266
18.6.1. Установка пакетов. Управления источниками пакетов.....	267
18.6.2. Обновление и удаление пакетов	271
18.6.3. Поиск пакета. Получение информации о пакете.....	271
18.7. Программа <code>yum</code>	272
18.7.1. Использование <code>yum</code>	272
18.7.2. Управление источниками пакетов.....	274
18.7.3. Установка пакетов через прокси-сервер	275
18.7.4. Плагины для <code>yum</code>	276
18.8. Графический менеджер пакетов в Fedora — <code>gpk-application</code>	276
18.9. Программы <code>dpkg</code> и <code>apt-get</code> : установка пакетов в Debian/Ubuntu	277
18.9.1. Программа <code>dpkg</code>	277
18.9.2. Программа <code>apt-get</code>	278
18.9.3. Установка RPM-пакетов в Debian/Ubuntu.....	280
18.9.4. Подключение репозитория Medibuntu	280
18.9.5. Графический менеджер Synaptic в Debian/Ubuntu.....	280
18.10. Установка пакетов в Slackware	281
18.10.1. Управление пакетами.....	283
18.10.2. Нет нужного пакета: вам поможет программа <code>rpm2tgz</code>	285
18.10.3. Программа <code>slackpkg</code> : установка пакетов из Интернета.....	286
18.11. Установка программ в openSUSE.....	287
18.11.1. Менеджер пакетов <code>zypper</code>	287
18.11.2. Графический менеджер пакетов openSUSE.....	290

Глава 19. Процессы	294
19.1. Аварийное завершение процесса.....	294
19.2. Программа <i>top</i> — кто больше всех расходует процессорное время?	296
19.3. Изменение приоритета процесса	298
19.4. Перенаправление ввода/вывода.....	299
Глава 20. Протоколирование системы. Журналы	300
20.1. Демоны протоколирования системы.....	300
20.2. Изучаем файлы журналов	302
Глава 21. Резервное копирование.....	305
21.1. Зачем нужно делать резервные копии	305
21.2. Выбор носителя для резервной копии	306
21.3. Правила хранения носителей с резервными копиями	307
21.4. Стратегии создания резервной копии	307
21.5. Программа <i>tar</i>	309
21.6. Сетевое резервное копирование	310
21.7. Запись CD/DVD из консоли	311
21.7.1. Команда <i>dd</i> — создание образа диска	311
21.7.2. Команды <i>cdrecord</i> и <i>dvdrecord</i> — запись образа на болванку	312
21.7.3. Команды очистки перезаписываемых дисков	312
21.7.4. Команда <i>mkisofs</i> — создание ISO-образа	313
21.7.5. Преобразование образов дисков	313
21.7.6. Создание и монтирование файлов с файловой системой	314
Глава 22. Автоматизация выполнения задач.	
Планировщики задач <i>crond</i>, <i>anacron</i>, <i>atd</i>	315
22.1. Планировщик задач — зачем он нужен?	315
22.2. Планировщик <i>crond</i>	315
22.3. Планировщик <i>anacron</i>	317
22.4. Разовое выполнение команд — демон <i>atd</i>	317
ЧАСТЬ V. ЛОКАЛЬНАЯ БЕЗОПАСНОСТЬ LINUX-СЕРВЕРА.....	319
Глава 23. Основные уязвимости	321
23.1. От кого будем защищать сервер? Мотивация взлома	321
23.2. Ваша система взломана	322
23.3. Основные уязвимости Linux-сервера	324

Глава 24. Обеспечение безопасности сервера.....	326
24.1. Защита от "восстановления пароля root"	326
24.2. Защита от перезагрузки	327
24.3. Отключение учетной записи root — нестандартный метод	328
24.4. Отключение учетной записи root средствами KDM.....	331
24.5. Система управления доступом	331
Глава 25. Параметры загрузчика Linux	332
25.1. Установка пароля	332
25.1.1. Загрузчик GRUB2	332
25.1.2. Загрузчик GRUB	333
25.2. Восстановление загрузчика GRUB/GRUB2	334
Глава 26. RAID-массивы	336
26.1. Что такое RAID?.....	336
26.2. Программные RAID-массивы	338
26.3. Создание программных массивов	338
ЧАСТЬ VI. НАСТРОЙКА СЕТЕВЫХ СЛУЖБ.....	341
Глава 27. DNS-сервер	343
27.1. Еще раз о том, что такое DNS.....	343
27.2. Кэширующий сервер DNS.....	344
27.3. Полноценный DNS-сервер	349
27.4. Вторичный DNS-сервер.....	353
27.5. Обновление базы данных корневых серверов	354
Глава 28. DHCP-сервер	357
28.1. Протокол динамической конфигурации узла.....	357
28.2. Конфигурационный файл DHCP-сервера	357
28.3. База данных аренды	359
28.4. Полный листинг конфигурационного файла	359
28.5. Управление сервером DHCP.....	360
28.6. Настройка клиентов	360
Глава 29. Web-сервер. Связка Apache + PHP + MySQL.....	361
29.1. Самый популярный Web-сервер.....	361
29.2. Установка Web-сервера и интерпретатора PHP. Выбор версии.....	361
29.3. Тестирование настроек.....	363

29.4. Файл конфигурации Web-сервера	365
29.4.1. Базовая настройка.....	365
29.4.2. Самые полезные директивы файла конфигурации	365
29.4.3. Директивы <i>Directory</i> , <i>Limit</i> , <i>Location</i> , <i>Files</i>	367
29.5. Управление запуском сервера Apache	369
29.6. Пользовательские каталоги.....	370
29.7. Установка сервера баз данных MySQL	370
29.7.1. Установка сервера	370
29.7.2. Изменение пароля root и добавление пользователей.....	371
29.7.3. Запуск и остан сервера.....	372
29.7.4. Программа MySQL Administrator	372
Глава 30. FTP-сервер.....	374
30.1. Зачем нужен FTP.....	374
30.2. Установка FTP-сервера.....	374
30.3. Конфигурационный файл.....	375
30.4. Настройка реального сервера	379
30.5. Программы ftpwho и ftpcount.....	380
30.6. Конфигуратор gproftpd	381
30.7. Альтернативные FTP-серверы.....	382
Глава 31. Почтовый сервер.....	383
31.1. Что такое Qmail?	383
31.2. Подготовка к установке Qmail.....	383
31.3. Установка Qmail и необходимых дополнений.....	385
31.3.1. Загрузка и установка Qmail	385
31.3.2. Установка ucspi-tcp и daemontools.....	386
31.3.3. Установка EZmlm — средства для создания рассылки	386
31.3.4. Установка Autoresponder — автоответчика.....	386
31.3.5. Установка MailDrop — фильтра для сообщений	386
31.3.6. Установка QmailAdmin — Web-интерфейса для настройки Qmail.....	387
31.4. Настройка после установки и запуск Qmail	387
31.5. Настройка почтовых клиентов	389
31.6. Дополнительная информация	390
Глава 32. Сервис Samba.....	391
32.1. Установка Samba.....	391
32.2. Базовая настройка Samba	391
32.3. Настройка общих ресурсов	392
32.4. Просмотр ресурсов Windows-сети	394

Глава 33. Настройка SSH-сервера	395
33.1. Протокол SSH и SSH-клиент	395
33.2. ssh-сервер	397
Глава 34. Сервер времени	401
34.1. Проблема синхронизации времени	401
34.2. Настройка сервера и Linux-клиентов	401
34.3. Настройка Windows-клиентов	402
Глава 35. Сетевая файловая система NFS	405
35.1. Установка сервера и клиента	405
35.2. Настройка сервера	405
35.3. Монтирование удаленных файловых систем	407
ЧАСТЬ VII. БЕЗОПАСНОСТЬ В СЕТИ	409
Глава 36. Аудит сети с помощью nmap	411
36.1. Что такое nmap?	411
36.2. Где мне взять nmap?	412
36.3. Основы использования nmap	412
Глава 37. Защита сетевых сервисов	414
37.1. Защита Web-сервера	414
37.2. Защита FTP	414
37.3. Защита DNS	415
37.4. Защита Samba	416
37.5. DHCP — привязка к MAC-адресу	416
37.6. Защита от спама: Greylisting и Qmail	419
Глава 38. Оптимизация сервера	421
38.1. Общая оптимизация Linux	421
38.1.1. Оптимизация подкачки	421
38.1.2. Изменение планировщика ввода/вывода	422
38.2. Оптимизация сетевых сервисов	423
38.2.1. Секреты оптимизации Samba	424
38.2.2. Оптимизация ProFTPD	424
38.2.3. Оптимизация Apache	426
38.2.4. Оптимизация SSH	428

Глава 39. Chroot-окружение	429
39.1. Песочница	429
39.2. Пример создания chroot-окружения	429
Глава 40. Управление доступом.....	432
40.1. Что такое Tomoyo?.....	432
40.2. Установка Tomoyo. Готовые LiveCD	432
40.3. Инициализация системы	433
Глава 41. Виртуальные частные сети.....	437
41.1. Для чего нужна виртуальная частная сеть?.....	437
41.2. Необходимое программное обеспечение.....	438
41.3. Канал для передачи данных VPN	438
41.3.1. Соединение сеть-сеть.....	438
41.3.2. Соединение клиент-сеть	439
41.4. Настройка соединения сеть-сеть	439
41.4.1. Установка OpenS/WAN.....	439
41.4.2. Немного терминологии.....	439
41.4.3. Генерирование ключей	440
41.4.4. Конфигурационный файл	440
41.4.5. Установка VPN-соединения	443
41.4.6. Настройка брандмауэра iptables.....	443
41.5. Настройка соединения клиент-сеть.....	444
41.5.1. Редактирование конфигурационных файлов.....	444
41.5.2. Настройка Linux-клиента.....	447
41.5.3. Настройка Windows-клиента.....	449
Глава 42. Прокси-сервер Squid и антивирус ClamAV	454
42.1. Зачем нужен прокси-сервер в локальной сети?	454
42.1.1. Базовая настройка Squid	454
42.1.2. Практические примеры настройки Squid	456
42.1.3. Управление прокси-сервером.....	457
42.1.4. Настройка клиентов	457
42.1.5. Прозрачный прокси-сервер	458
42.1.6. Расширение squidGuard	459
42.2. Антивирусная защита	462
42.2.1. Зачем нужен антивирус в Linux	462
42.2.2. Установка ClamAV.....	463
42.2.3. Проверка файловой системы.....	463
42.2.4. Прозрачная проверка почты	463
42.2.5. Проверка Web-трафика.....	464

ЧАСТЬ VIII. ТЕОРИЯ И ПРАКТИКА СИСТЕМНОГО АДМИНИСТРАТОРА.....	469
Глава 43. Стратегия администрирования	471
43.1. О чем эта глава?	471
43.2. И руководство, и пользователи довольны. Миф или реальность?	472
43.3. Роль главного администратора	474
Глава 44. Уход за "железом"	478
44.1. Обязанности администратора	478
44.2. "Про запас", или обменный фонд	479
44.3. Чистка компьютеров. Профилактика системы охлаждения	480
44.4. Охлаждение компьютеров	481
44.5. Стойки для оборудования	482
44.6. Влажность	483
44.7. Инструмент системного администратора	484
Заключение	487
ПРИЛОЖЕНИЯ	489
Приложение 1. Параметры ядра	491
Приложение 2. Суперсервер xinetd	494
П2.1. Сетевые сервисы и суперсервер	494
П2.2. Конфигурационный файл суперсервера	494
Приложение 3. Команды Linux	496
П3.1. Общие команды	496
П3.1.1. Команда <i>arch</i> — вывод архитектуры компьютера	496
П3.1.2. Команда <i>clear</i> — очистка экрана	496
П3.1.3. Команда <i>date</i>	497
П3.1.4. Команда <i>echo</i>	497
П3.1.5. Команда <i>exit</i> — выход из системы	497
П3.1.6. Команда <i>man</i> — вывод справки	497
П3.1.7. Команда <i>passwd</i> — изменение пароля	497
П3.1.8. Команда <i>startx</i> — запуск графического интерфейса X.Org	497
П3.1.9. Команда <i>uptime</i> — информация о работе системы	498
П3.1.10. Команда <i>users</i> — информация о пользователях	498
П3.1.11. Команды <i>w</i> , <i>who</i> и <i>whoami</i> — информация о пользователях	498
П3.1.12. Команда <i>xf86config</i> — настройка графической подсистемы	499

ПЗ.2. Команды для работы с файлами и каталогами, ссылками, правами доступа.....	499
ПЗ.2.1. Работа с файлами.....	499
ПЗ.2.2. Работа с каталогами.....	501
ПЗ.2.3. Команда <i>ln</i> — создание ссылок, жестких и символических.....	503
ПЗ.2.4. Команда <i>chmod</i> — права доступа к файлам и каталогам.....	503
ПЗ.2.5. Команда <i>chown</i> — смена владельца файла.....	505
ПЗ.2.6. Команда <i>chattr</i> — изменение атрибутов файла, запрет изменения файла.....	505
ПЗ.2.7. Команда <i>mkfs</i> — создание файловой системы.....	505
ПЗ.2.8. Команда <i>fsck</i> — проверка и восстановление файловой системы.....	506
ПЗ.2.9. Команда <i>chroot</i> — смена корневой файловой системы.....	506
ПЗ.2.10. Установка скорости CD/DVD.....	506
ПЗ.2.11. Монтирование каталога к каталогу.....	507
ПЗ.2.12. Команды поиска файлов.....	507
ПЗ.2.13. Создание файла подкачки.....	508
ПЗ.3. Команды для работы с текстом.....	509
ПЗ.3.1. Команда <i>diff</i> — сравнение файлов.....	509
ПЗ.3.2. Команда <i>grep</i> — текстовый фильтр.....	509
ПЗ.3.3. Команды <i>more</i> и <i>less</i> — постраничный вывод.....	510
ПЗ.3.4. Команды <i>head</i> и <i>tail</i> — вывод начала и хвоста файла.....	510
ПЗ.3.5. Команда <i>wc</i> — подсчет слов в файле.....	510
ПЗ.4. Команды для работы с Интернетом.....	510
ПЗ.4.1. Команда <i>ftp</i> — стандартный FTP-клиент.....	510
ПЗ.4.2. Команда <i>lynx</i> — текстовый браузер.....	512
ПЗ.4.3. Команда <i>mail</i> — чтение почты и отправка сообщений.....	512
ПЗ.5. Команды системного администратора.....	512
ПЗ.5.1. Команды <i>free</i> и <i>df</i> — информация о системных ресурсах.....	512
ПЗ.5.2. Команда <i>md5sum</i> — вычисление контрольного кода MD5.....	513
Предметный указатель.....	515

Введение

На этот раз введение не будет длинным. Тому есть две причины. Во-первых, не хочется занимать ваше драгоценное время. Во-вторых, как показывает практика, больше половины читателей считают введение чем-то скучным и вообще его не читают. Зачем же тратить время и бумагу?

Хочется сказать несколько слов об особенностях этой книги, которые выделяют ее среди других книг, посвященных системному администрированию Linux. Я старался написать ее так, чтобы она не стала "еще одной книгой по настройке Linux-сервера". В ней есть все, что нужно знать будущему системному администратору. Так, в первой части книги рассматриваются основы основ: принципы работы компьютерных сетей, адресация в сетях, монтаж сети. Все это подано настолько подробно, чтобы у будущего администратора не возникали вопросы типа "А что такое сетевая маска?" или "Как обжать витую пару?". Могу с уверенностью сказать, что благодаря этой информации вы не только настроите Linux-сервер, но сможете построить локальную сеть.

В остальных частях книги рассматривается настройка Linux (в том числе и настройка сети — как же без нее?), установка программного обеспечения, настройка сетевых служб (Apache, DNS, DHCP, ssh, Squid и т. д.). Особое внимание уделяется безопасности настраиваемого сервера — как локальной, так и безопасности в сети: подробно описываются создание шлюза (маршрутизатора), конфигурирование брандмауэра iptables, а также настройка виртуальной частной сети. Вопросам безопасности в книге действительно уделяется много внимания, а помимо всего прочего мы рассмотрим и сканер nmap — чтобы администратор мог сам просканировать свою сеть на предмет потенциальных уязвимостей.

Материал книги основан на дистрибутивах Fedora 13, Mandriva 2010.1 Spring, openSUSE 11.3, Ubuntu 10.04, Debian 5 и Slackware 13. Учитывая столь обширный список дистрибутивов, могу с уверенностью сказать, что книга подойдет большинству администраторов.

Вот теперь можно с чистой совестью приступить к чтению книги. И даже если вы не новичок, а действующий администратор, рекомендую все-таки прочитать первую часть книги — в ней вы найдете для себя много полезного.



ЧАСТЬ I

Основы администрирования

Первая часть посвящена основам администрирования. Мы рассмотрим краткую историю Linux, основы сетевого взаимодействия, познакомимся с моделью OSI, адресацией в TCP/IP-сетях, с монтажом сетей Ethernet и Wi-Fi. А в следующей части книги поговорим об установке и настройке Linux.

Глава 1



Становимся администратором

1.1. Краткая история Linux

В далеком 1969 году сотрудники фирмы Bell Labs пытались возродить ОС Multics, но превзошли сами себя, и то, что получилось, уже никак не тянуло на обычный "апгрейд" для Multics — это была совершенно новая операционная система, которую назвали UNIX. Интересно, что поначалу UNIX называлась "UNICS", но позже американцы, как они это любят делать, немного упростили название системы.

В начале 70-х годов прошлого века ОС UNIX была существенно доработана. В ее ядро добавили много новых функций, а главное — она была переписана на языке C, что обеспечило легкость переноса этой ОС на другие аппаратные платформы (первоначально UNIX была написана на ассемблере и предназначалась для компьютера PDP-7).

Важно, что с самого рождения UNIX была многопользовательской и многозадачной. Таким образом, идеи, заложенные в представленную в 1995 году Windows 95, оказались, по сути, идеями 20-летней давности — в UNIX все это уже было реализовано 20 лет назад. Да, не было красивого "фантика" — графического интерфейса, — но ведь не это главное в операционной системе.

В начале 1980-х годов появились первые персональные компьютеры фирмы IBM. Однако мощности IBM PC никак не хватало для запуска UNIX. Поэтому в мире персональных компьютеров десять лет царствовала операционная система DOS компании Microsoft. Начиная с 1990-х все изменилось — мощность "персоналок" уже позволяла запускать UNIX. К этому времени (прошло более 20 лет с момента появления первой версии UNIX) разными фирмами, университетами и отдельными энтузиастами было создано много UNIX-подобных операционных систем (IRIX, XENIX, HP-UX, BSD, Minix и др.).

Огромное значение в развитии Linux сыграла одна из UNIX-подобных операционных систем — Minix, которая не была полноценной системой, а создавалась, чтобы демонстрировать основные принципы и устройство настоящих операционных систем. Да, она не была совершенной, но зато ее исходный код (всего 12 тысяч строк) был опубликован в книге А. Таненбаума "Операционные системы". Именно эту книгу и купил Линус Торвалдс (Linus Torvalds).

В 1991 году Линус Торвалдс установил на свой компьютер ОС Minix, но та не оправдала его ожиданий, поэтому он принял решение несколько ее переработать — ведь исходные коды вместе с комментариями были под рукой. Сначала Торвалдс

просто переписал программу эмуляции терминала, а затем фактически взялся за создание собственной операционной системы.

25 августа 1991 года ОС Linux (версия 0.01) была создана. Конечно, это была не та Linux, что есть сейчас, но она уже тогда была лучше Minix, поскольку в ней запускались командный интерпретатор `bash` и компилятор `gcc`. Сообщение о создании новой операционной системы было помещено в группу новостей `comp.os.minix`, там же предлагалось всем желающим протестировать ее.

С этого и началось интенсивное развитие Linux, а к ее разработке в помощь Торвальдсу подключились энтузиасты со всего мира, — ведь ничто так не сокращает расстояния, как Интернет. С момента появления версии 0.01, которой практически нельзя было пользоваться, до создания версии 1.0, пригодной для обычных пользователей, а не программистов, прошло почти три года (она появилась в апреле 1994 года). И уже эта первая версия обладала поддержкой сети (поддерживался протокол TCP/IP), а также графическим интерфейсом X Window, появившимся в Linux еще в 1992 году одновременно с поддержкой TCP/IP.

Первые версии Linux распространялись на обыкновенных дискетах. Комплект состоял из двух дискет: первая содержала ядро, а вторая — корневую файловую систему и необходимые программы. Установить подобную версию Linux на компьютер мог только специалист. Чуть позже появились первые дистрибутивы, которые, помимо того же ядра и корневой файловой системы, включали также программу для установки всего этого на компьютер. Программа установки поставлялась, как правило, на отдельной дискете.

Первые дистрибутивы появились в 1992 году — тогда отдельные энтузиасты или группы энтузиастов выпускали разные дистрибутивы (каждый, естественно, под своим именем). Фактически они отличались друг от друга лишь названием и программой установки. В дальнейшем различия между дистрибутивами стали более существенными.

Самый первый дистрибутив, созданный в Манчестерском компьютерном центре (Manchester Computing Centre, MCC), появился в начале 1992 года и назывался MCC Interim Linux. Чуть позже появился дистрибутив TAMU, разработанный в Техасском университете.

Настоящий прорыв произвел дистрибутив SLS, выпущенный в октябре 1992 года, поскольку именно он содержал поддержку TCP/IP и систему X Window. Впоследствии данный дистрибутив бурно развивался и постепенно трансформировался в один из самых популярных дистрибутивов — Slackware.

Со временем дистрибутивы разрослись до таких размеров, что распространять их на дискетах стало нельзя. Вы можете себе представить дистрибутив на 50 дискетах (дистрибутивы того времени занимали 50–70 Мбайт)? А что делать, если, скажем, дискета № 47 окажется бракованной? Как раз к тому времени лазерные компакт-диски и их приводы немного подешевели, и компания Red Hat стала одной из первых, выпустивших свою разработку на компакт-диске.

Кроме получения на дискетах или компакт-диске, дистрибутив того времени (как, впрочем, и сейчас) можно было бесплатно скачать из Интернета (если не считать стоимости самого Интернета). Но далеко не все могли себе позволить Интернет в online-режиме (тогда online-режимом считалась работа с WWW, а offline — с почтой и новостями Usenet). Да и привод CD-ROM (односкоростной) стоил около

100 долларов. Поэтому в начале 1990-х основными носителями для распространения Linux все же были дискеты. А вот начиная с середины 1990-х Linux постепенно почти полностью перекечевала на компакт-диски.

О дистрибутивах можно говорить еще очень долго. Важно запомнить следующее:

- основные дистрибутивы: Red Hat, Slackware и Debian, все остальные — это производные от них. Например, Mandrake произошел от Red Hat, ALT Linux потом взял за основу Mandrake, а ASPLinux — Red Hat. Потом на смену Red Hat пришел дистрибутив Fedora Core (сейчас просто Fedora), а на смену Mandrake — Mandriva;
- номер версии дистрибутива не совпадает с номером ядра — это принципиально разные вещи.

1.2. Почему именно Linux?

А почему именно Linux? Почему бы не использовать ту же FreeBSD, у которой родства с UNIX намного больше, чем у Linux? На базе FreeBSD, как и на базе Linux, можно построить стабильный сервер. Но у Linux есть одно неоспоримое преимущество — она популярнее. А это значит, что для нее больше русскоязычной документации, на Linux уже обращают внимание производители оборудования (вы без особых проблем найдете драйвер для вашего "железа"), да и Linux более дружелюбна к пользователю. Да, именно к пользователю. Конечно, для администратора сервера это не столь важно, но Linux более универсальна, что позволяет ее использовать как на сервере, так и на рабочих станциях. Получается, что можно установить одну и ту же операционную систему на всех компьютерах сети — следовательно, вам будет проще обслуживать эту сеть, чем "разношерстную" сеть, где компьютеры работают под всевозможными версиями Windows, Mac OS и Linux.

1.3. Основные задачи системного администратора

Сейчас мы рассмотрим основные задачи системного администратора. У нас ведь как бывает: сисадмин и монтажом сети занимается, и обучением пользователей (далеко не все умеют "на кнопки" нажимать). Поэтому сразу скажу: далее приведен список обязанностей администратора Linux-сервера, работающего в идеальных условиях.

- **Установка и настройка программного обеспечения** — после установки самой Linux вам нужно будет установить дополнительное программное обеспечение, например, Web-сервер, FTP-сервер, а затем настроить это программное обеспечение.
- **Управление пользователями** — в обязанности администратора также входит создание, модификация и удаление учетных записей пользователей сервера. Возможно, придется ограничить место на диске, предоставляемое каждому пользователю (эта операция называется *квотированием*).
- **Инсталляция и деинсталляция аппаратных средств** — кому как не администратору подключать новые жесткие диски и подготавливать их для использо-

вания сервером. Причем часто бывает, что устанавливать "железо" (впрочем, как и "софт") придется не только на сервере, но и на рабочих станциях — такова уж судьба сисадмина...

- ❑ **Резервное копирование** — это одна из самых важных задач системного администратора. Часто резервное копирование, к сожалению, не выполняется или выполняется не так, как нужно. В результате — потеря данных. Да, это не интересно, да — это рутинно. Но выполнять эту задачу нужно.
- ❑ **Поиск неисправностей** — время от времени аппаратные средства выходят из строя. Иногда случаются "глюки" в программном обеспечении. Найти и устранить неисправность — задача системного администратора. Сразу предупреждаю: часто найти неисправность сложнее, чем ее устранить.
- ❑ **Защита сети.** Обеспечение безопасности сети и контроль защиты — очень важная задача, ведь вы же не хотите, чтобы ваш сервер взломали? Часто бывает, что "врага" нужно ожидать не извне, а изнутри — это могут быть любопытные либо недовольные пользователи, способные посягнуть на неприступность вашего сервера.
- ❑ **Мониторинг системы** — важно ежедневно просматривать журналы системы. В журналах можно найти много интересной и полезной информации: попытки взлома, ошибки в конфигурации системы и т. д.
- ❑ **Консультации и техническая поддержка пользователей** — чтобы не отвлекать вас от основных задач, желательно, чтобы эту задачу выполнял ваш помощник. Но если вы единственный администратор в компании, то этим придется заниматься лично вам.
- ❑ **Ведение локальной документации** — чтобы вам (или тому человеку, который, возможно, займет впоследствии ваше место) было проще в будущем, следует протоколировать все свои действия: разводку кабелей сети, устанавливаемые программные средства, изменения в конфигурации системы, изменения в схеме сети и т. д.

Вот теперь вы знаете, с чем вам придется столкнуться при выполнении своих обязанностей. Но это только базовый комплект — вполне возможно, что на практике появится еще несколько задач, которые вам придется решать.

ПРОФЕССИОНАЛЬНЫЙ ПРАЗДНИК

Так уж получилось, что данная глава была написана в день системного администратора, поэтому не упомянуть об этом профессиональном празднике я просто не могу. День системного администратора отмечается в последнюю пятницу июля (в 2010 году это 30 июля). Основатель праздника — Тед Кекатос (Ted Kekatos), системный администратор из Чикаго. Именно он посчитал, что раз в году "бойцы невидимого фронта" должны чувствовать благодарность со стороны пользователей. Кстати, в США этот праздник называется День благодарности системному администратору (System Administrator Appreciation Day). Первый раз праздник был отмечен 28 июля 1999 года.

Глава 2



Классификация сетей

2.1. Краткая история сетей

С появлением первых электронно-вычислительных машин (не персональных компьютеров, а именно первых огромных вычислительных машин, которые занимали целые комнаты) возникла проблема переноса данных между ними. С того момента было создано много различных сетей. Сейчас мы вкратце рассмотрим историю сетей, чтобы вы знали, откуда они появились, а потом попробуем классифицировать все имеющиеся виды сетей.

2.1.1. 1941–1975 годы

Первый период развития вычислительных сетей начинается в 1941 году (тогда, если вы помните, появилась первая "большая" ЭВМ) и называется *лабораторным* — в то время сети, как впрочем и ЭВМ, не выходили за пределы лабораторий научных институтов. С самого начала ставилась задача объединения в сеть ЭВМ без привязки к конкретной аппаратуре.

Любопытно

Казалось бы, как давно это было! Но самое интересное, что мы до сих пор используем решения, разработанные в то время. Последовательный интерфейс RS-232C и параллельный интерфейс Centronics (да, тот, который служит для подключения принтеров) используются до сих пор. Интерфейс RS-232C постепенно вытесняется современными последовательными интерфейсами: USB и IEEE 1394 (FireWire), и на некоторых современных компьютерах его больше нет вообще. Однако интерфейс Centronics имеется на каждом современном стационарном компьютере, хотя большинство производителей принтеров уже практически перешло на USB. Наличие "старых" интерфейсов зависит только от производителя материнской платы — как он решит, так и будет. Мой компьютер, на котором я пишу эти строки, был куплен в феврале 2008 года. Тогда я не обратил внимания на наличие/отсутствие старых интерфейсов, но потом выяснилось, что на материнской плате отсутствует RS-232C, но имеется Centronics (LPT), а также USB, IEEE 1394, HDMI (правда, он не имеет никакого отношения к сетевым интерфейсам) и другие современные разъемы, которых не было на более старых компьютерах. С другой стороны, в продаже до сих пор имеются материнские платы с RS-232C, а также предлагаются отдельные PCI-контроллеры, добавляющие два порта RS-232C, если в них возникает острая необходимость.

Интерфейсы RS-232C и Centronics — это, в принципе, хорошо, но они годятся только для связи "точка-точка", то есть для непосредственной связи отправителя

и получателя данных. Понятно, что в сети может быть гораздо больше, чем две ЭВМ, поэтому разработчики сетей на этом не остановились, и в 1974 году компания IBM представила универсальную архитектуру вычислительных сетей: SNA (System Network Architecture). Эта архитектура, помимо всего прочего, поддерживала *адресацию узлов* сети, смысл которой в том, что каждому узлу сети присваивается уникальный адрес, по которому можно обратиться к этому узлу. Сейчас для адресации узлов преимущественно используются протоколы IPv4 и IPv6, о которых мы поговорим далее в этой главе.

2.1.2. 1976–1982 годы

Второй период развития сетей начался в 1976 году, когда сети вышли за пределы лабораторий и начали активно разрабатываться сетевые архитектуры и технологии передачи данных. Тогда и появилось семейство протоколов X.25 — протоколов передачи данных в системах с коммутацией пакетов. Разработка протоколов X.25 стала очень важным событием, поскольку до появления Интернета они были единственными протоколами, используемыми для создания глобальных сетей, — именно X.25-сети связывали тогда весь мир в единое целое. Затем на базе X.25 был создан протокол Frame Relay, а на его базе — технология АТМ. Подробно рассматривать все производные протоколов X.25 мы не будем, поскольку нас сейчас интересуют только ключевые события в развитии сетей (описание истории появления каждого сетевого протокола займет целую книгу, прочитать которую у вас не хватит терпения). Отмечу только, что Frame Relay, как и АТМ, здравствуют и по сей день.

В 1979 году был создан первый модем для персональных (!) компьютеров. Я даже догадываюсь, о чем вы сейчас подумали: какие, мол, персональные компьютеры в 1979 году? Какой модем? Да, Personal Computer (PC) от IBM появился в 1981 году, но это не означает, что до этого не было *персональных компьютеров*. Для работы с первыми ЭВМ обычно требовался целый штат специалистов, а персональный компьютер — это компьютер, предназначенный только для одного человека, для одного пользователя. И настоящие персональные компьютеры, отвечающие этому определению, появились еще до 1980 года — это были компьютеры компании Apple. А словосочетание "Personal Computer" — всего лишь название, правда, весьма удачное, продукта компании IBM. IBM первая ввела термин PC, и с того времени все компьютеры со сходной архитектурой команд считаются PC-совместимыми.

А все современные модемы являются Hayes AT-совместимыми, то есть совместимыми с набором AT-команд управления модемом, разработанным компанией Hayes. Первый модем Micromodem II был выпущен этой компанией в 1979 году. Он развивал скорость в 300 бод (бит/с) и предназначался для компьютеров Apple.

Еще в лабораторном периоде были разработаны *системы с произвольным доступом*. Впервые они были использованы в начале 1970-х годов в сети Alohanet, объединяющей Гавайские острова. Сначала эти системы считались бесперспективными, но в мае 1973 года Боб Меткалф (Bob Metcalfe) усовершенствовал метод CSMA, на котором они были основаны. Усовершенствованный метод назвали CSMA/CD (Carrier-Sense Multiple Access with Collision Detection, множественный доступ с контролем несущей и обнаружением коллизий). Боб Меткалф планировал

использовать этот метод для совместного доступа к сетевым принтерам, но он позже "перерос" в то, что сейчас называется Ethernet-сетью. Тогда сеть CSMA/CD передавала данные по коаксиальному кабелю (как первые Ethernet-сети) со скоростью 2,94 Мбит/с (для того времени это была значительная скорость), а максимальное расстояние передачи данных составляло 1,5 км. В 1978 году Меткалф зарегистрировал компанию 3Com Corporation (наверное, все мы слышали название этой компании), а в 1982 году выпустил первый в мире серийный Ethernet-адаптер для компьютера Apple.

В 1979 году произошло еще одно важное событие — был организован альянс DIX (DEC, Intel, Xerox), результатом деятельности которого стала в 1980 году разработка стандарта Ethernet.

В 1980 году была разработана *модель взаимосвязи открытых систем* (Open System Interconnect, OSI). Эта модель четко определяет семь уровней, которые обеспечивают передачу данных по сети. Модель OSI сугубо теоретическая, но она лежит в основе всех современных сетей. Мы подробно рассмотрим ее чуть позже в этой главе.

2.1.3. 1983–1989 годы

Начиная с 1983 года, в институтах и даже некоторых офисах стали появляться первые локальные сети, связывающие компьютеры толстым коаксиальным кабелем. В то время сетевой адаптер стоил очень дорого (например, для ЭВМ VAX стоимость сетевого адаптера превышала 3 тыс. долларов), поэтому локальную сеть могли себе позволить только самые крупные фирмы. Найти тогда "персоналку" с сетевым адаптером было сложно.

В 1985 году Институтом инженеров по электротехнике и электронике (IEEE) был принят стандарт IEEE 802.3 (10Base-5) — Ethernet-сеть на "толстом" коаксиальном кабеле. В 1989 году был принят стандарт IEEE 802.3a (10Base-2), предусматривающий передачу данных по "тонкому" коаксиальному кабелю. Подробно о стандартах Ethernet мы поговорим чуть позже в этой книге.

Понятно, что Ethernet-сети — не единственный вид локальной сети. В 1988 году IBM превзошла стандарт Ethernet, представив технологию Token Ring с максимальной скоростью передачи данных в 16 Мбит/с (Ethernet предусматривал передачу данных с максимальной скоростью в 10 Мбит/с).

В 1985 году компания StrataCom начала эксплуатацию первых линий T1 со скоростью передачи данных 1,54 Мбит/с. Чуть позже линии T1 стали доступны крупным компаниям и использовались в качестве магистралей для быстрой передачи данных на большие расстояния.

Индивидуальным пользователям в 1980-х годах сети "особо не светили", поскольку сетевое оборудование продолжало стоить весьма дорого. Так, в 1989 году компания Arg Electronics представила высокоскоростной модем (19,2 Кбит/с) стоимостью "всего" 3595 долларов. Интересно, что этот модем был относительно дешевле модемов других производителей, которые, к тому же, не обеспечивали заявленной ими скорости.

Кто мог позволить себе сети ISDN, радовался скорости передачи данных в 128 Кбит/с (сети ISDN BRI) или 1,54 Мбит/с (ISDN PRI). О цене говорить не будем — ISDN-сети стоили неприлично дорого.

Технологии — это, конечно, хорошо. Но сетевые адаптеры и прочее сетевое оборудование без программного обеспечения — просто железки. Чтобы компьютер мог работать в сети, нужна сетевая операционная система. В 1980-х годах сеть поддерживали следующие ОС: UNIX (и ее вариации), Novell Netware, Microsoft LAN Manager (оболочка для OS/2, появившаяся в 1987 году).

В 80-х годах прошлого века появились и первые сотовые сети — да, сотовая телефония! Первая система сотовой телефонной связи Nordic Mobile Telephone System (кто помнит — первые "мобилки", появившиеся у нас в 1990-х годах, поддерживали стандарт NMT) была запущена в Дании, Швеции, Финляндии и Норвегии в 1981 (!) году. В 1983 году заработали две сотовые сети в Северной Америке: AURORA-400 и AMPS.

2.1.4. 1990–1995 годы

В 1990 году произошел очередной "переворот" в Ethernet-сетях — был принят стандарт IEEE 802.3i (10Base-T), предусматривающий передачу данных по витой паре 3-й категории со скоростью 10 Мбит/с. Переворот заключался в том, что Ethernet-сети стали:

- *надежнее* — при использовании коаксиального кабеля все компьютеры подключались к общему кабелю, и если этот кабель обрывался, то вся сеть "падала". В случае с витой парой все компьютеры сети подключаются к центральному устройству сети — Ethernet-концентратору. Если происходит обрыв кабеля, ведущего к какому-нибудь узлу сети, этот узел исчезает из сети, но вся сеть продолжает работать;
- *проще в установке* — монтаж витой пары намного проще, чем коаксиального кабеля, особенно, если речь идет о "толстом" коаксиальном кабеле.

Позднее был принят стандарт IEEE 802.1D, в котором было определено понятие *моста* (bridge), и Ethernet-сети наконец-то стало можно делить на сегменты для локализации трафика. Сегментация сети особо важна для крупных сетей — ведь чем больше узлов, тем меньше производительность сети.

Через три года сети того времени стали напоминать современные — в них активно начали использоваться концентраторы и мосты, появились первые коммутаторы и двухуровневые сети. В двухуровневых сетях компьютеры одной рабочей группы (одного отдела компании) объединялись между собой концентратором, а сами рабочие группы (то есть концентраторы рабочих групп) подключались через мосты к общей корпоративной магистрали. В качестве магистрали обычно использовалось оптоволокно (стандарт 10Base-FL или IEEE 802.3j, принятый в 1993 году). С появлением 10Base-FL на оптоволокне Ethernet-сети выходят за пределы зданий и становятся средством для создания "кампусных" сетей. То есть если раньше Ethernet-сети использовались только для создания локальных сетей, то в 1994–1995 годах стандарт 10Base-FL применялся для связи локальных сетей, находящихся в разных зданиях.

Следующим шагом в создании корпоративных сетей стало изобретение многопортового устройства — центрального коммутатора, в котором были объединены все мосты сети. Такая конфигурация получила название collapsed-backbone ("маги-

страль в точке"). Примерно в это же время родилось понятие *структурированных кабельных сетей* (СКС).

Понятно, что сети росли, и скорости 10 Мбит/с для магистрали стало недостаточно. На тот момент существовала всего одна "быстрая" технология, обеспечивающая передачу данных по оптоволоконному кабелю со скоростью 100 Мбит/с — FDDI (Fiber Distributed Data Interface, распределенный волоконный интерфейс данных). Но в 1992 году компания Grand Junction начала разработку Ethernet-сети, работающей на скорости 100 Мбит/с, и она была стандартизирована в 1995 году (стандарт IEEE 802.3u, сети 100Base-TX, 100Base-T4 и 100Base-FX). В том же 1995 году компания Grand Junction была поглощена компанией Cisco Systems: закон выживания — выживают лишь сильнейшие. После принятия стандартов 100Base-* спрос на технологию FDDI резко пошел вниз, поскольку Ethernet-сети обеспечивали ту же скорость передачи данных, но стоили намного дешевле только за счет среды передачи данных — витая пара стоит намного дешевле, чем оптоволокно. А в 1998 году появились Ethernet-сети, передающие данные со скоростью 1 Гбит/с, но об этом позже.

А что же происходило в мире глобальных сетей? В 1990 году компания US Sprint начала предоставлять услуги объединения точек через Frame Relay по всей территории США. Тогда почти все высокоскоростные магистрали переводились на технологию ATM, но для подключения клиентов использовался Frame Relay. Однако в 1994 году компания Bell Atlantic начинает предлагать подключение клиентов по технологии ATM.

Не стоит забывать и об операционных системах. В 1993 году появилась первая действительно сетевая ОС от Microsoft — Windows NT, а в 1995 году — нашумевшая ОС Windows 95.

2.1.5. 1996–1999 годы

В эти годы ничего революционного в магистральных каналах связи не случилось, если не считать появления сервисов гарантирования качества обслуживания (QoS, Quality of Service). Но нас интересуют технологии, более близкие к пользователю. Можно сказать, что в эти годы (1995–1999) завершилась эра развития аналоговых модемов. В 1998 году был принят стандарт V.90, который используется и по сей день (если не считать его небольшого усовершенствования V.92, появившегося в 2000 году). Судя по всему, телефонные модемы отжили свое. Сегодня все больше и больше провайдеров предоставляют высокоскоростной доступ к Интернету, а обычные аналоговые модемы практически уже не используются.

Зарождение высокоскоростного доступа произошло как раз в 1995–1999 годах, когда появились первые кабельные и ADSL-модемы. Кабельные модемы (они передают данные по сетям операторов кабельного телевидения) преимущественно применялись в США. В Европе получили большее распространение ADSL-модемы, использующие для передачи данных обычный телефонный кабель. К сожалению, в те годы в России о таких модемах только слышали, но никто их практически не видел.

В мире локальных сетей в 1998 году появилась технология 1000Base-X, передающая данные со скоростью 1 Гбит/с по оптоволокну, а в 1999 году — технология 1000Base-T, передающая данные со скоростью 1 Гбит/с по витой паре.

2.1.6. 2000 — наше время

Понятно, что развитие сетей не останавливается, а только начинается. Все еще впереди. Лет через десять все современные технологии будут казаться нам такими же "древними", какими сейчас кажутся решения 20-летней давности.

Из интересного в мире Ethernet можно отметить появление в 2003 году технологий передачи данных со скоростью 10 Гбит/с (10GBase-SR, 10GBase-LR, 10GBase-ER, 10GBase-SW, 10GBase-LW, 10GBase-EW) и технологии PLC, обеспечивающей передачу данных по сети электропитания. В 2003 году это казалось странным, но сейчас — вполне нормально.

Если вы заметили, то в этой краткой истории практически ничего не было сказано о развитии беспроводных сетей. Это сделано умышлено. В *главе 6* мы поговорим о том, как данные передаются "по воздуху", рассмотрим краткую историю беспроводных сетей и существующие беспроводные стандарты.

2.2. Классификация сетей

Сети можно классифицировать по:

- занимаемой территории;
- топологии;
- ведомственной принадлежности;
- скорости передачи данных;
- типу среды передачи данных;
- организации взаимодействия компьютеров.

2.2.1. По занимаемой территории

По занимаемой территории сети могут быть локальными, региональными (они же муниципальные сети) и глобальными:

- локальные* (LAN, Local Area Network) — сети, занимающие небольшую территорию, например, одну комнату или одно здание;
- региональные* (MAN, Metropolitan Area Network) — сети, охватывающие город (отсюда другое название — муниципальные) или даже область;
- глобальные* (WAN, Wide Area Network) — такие сети охватывают территории одного или нескольких государств или даже весь мир. Пример всемирной сети — Интернет.

С локальными и глобальными сетями все понятно, разберемся с сетями региональными. Сеть MAN, как правило, объединяет в единое целое несколько сетей — например, сети двух или более зданий. При этом среда передачи данных сети MAN может быть как проводной, так и беспроводной.

Беспроводная сеть обходится намного дешевле, чем сеть на базе оптоволокна, но она менее надежна и менее безопасна. Тем не менее, беспроводные технологии очень полезны для MAN — не всегда есть возможность проложить кабель. С другой стороны, MAN часто выступает в качестве магистральной сети, поэтому производительности беспроводной сети может оказаться недостаточно.

Сейчас особой необходимости в MAN-сетях нет, поскольку можно организовать *виртуальную частную сеть* (VPN, Virtual Private Network), использующую каналы Интернета для передачи данных. Представим следующую ситуацию: есть организация, главный офис которой находится в Москве, затем эта компания открыла свой филиал в Санкт-Петербурге. Как объединить сети офисов вместе? Вы только представьте себе, сколько кабеля для этого понадобится! Причем витой парой здесь не отделаешься, придется использовать дорогой оптоволоконный кабель — ведь расстояние-то большое. Беспроводные технологии тоже из-за расстояния отпадают. Остается только одно — использовать для передачи данных каналы Интернета. Сеть каждого офиса подключается к Интернету через каналы местного интернет-провайдера, и через Интернет создается виртуальная частная сеть. И дешево, и быстро — ведь высокоскоростное подключение к Интернету в настоящее время вполне доступно. Понятно, что данные будут передаваться по незащищенным каналам, поэтому в виртуальной частной сети используется шифрование всех передаваемых данных. Механизмы VPN позволяют не только объединить две разные сети в единое целое, но и обеспечить безопасность передаваемых данных.

2.2.2. По топологии

Существуют следующие топологии сети:

- *линейная* (рис. 2.1) — подключение по принципу гирлянды: каждый узел сети подключается к следующему узлу сети. В такой сети от узла с номером 1 до узла N будет всегда одинаковый маршрут: через узлы 2, 3, 4, ..., $N - 1$. Понятно, в случае отказа одного из узлов сети, линейная сеть прекратит свое существование. В настоящее время линейные сети практически не используются (если не принимать во внимание нуль-модемное соединение);
- *кольцевая* (рис. 2.2) — каждый узел сети соединен с двумя соседними узлами, все узлы сети образуют кольцо. Кольцевая топология используется технологиями Token Ring, FDDI и некоторыми другими;
- *звездообразная* (рис. 2.3) — в такой сети есть один центральный узел, с которым связан каждый узел сети. Такие сети еще называются *централизованными*. "Падение" центрального узла означает "падение" всей сети. Обычно в качестве центрального узла используется концентратор (hub) или коммутатор (switch). Пример звездообразной сети — Ethernet на базе витой пары;
- *общая шина* (рис. 2.4) — все узлы сети подключаются к единой среде передачи данных, например, к коаксиальному кабелю. Слабое место такой сети — сама среда передачи данных: обрыв кабеля означает сбой всей сети. Пример сети на общей шине — Ethernet на базе коаксиала;
- *древовидная* (рис. 2.5) — топологию этой сети проще представить, чем описать или вникать в определение. В древовидной сети есть более двух конечных узлов и, по крайней мере, два промежуточных узла. В древовидной сети между двумя узлами есть только один путь. Чтобы вникнуть в правильное определение древовидной сети, нужно знать теорию графов, поскольку древовидная сеть — это неориентированный ациклический граф, не содержащий замкнутых путей и позволяющий соединить единственным образом пару узлов;

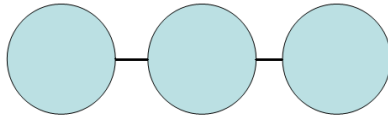


Рис. 2.1. Линейная топология

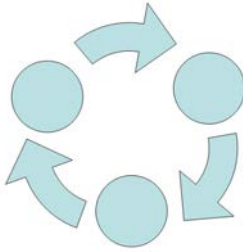


Рис. 2.2. Кольцевая топология

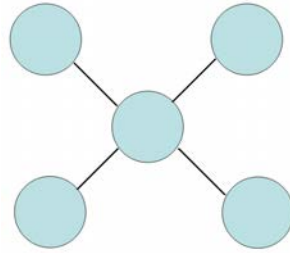


Рис. 2.3. Звезда

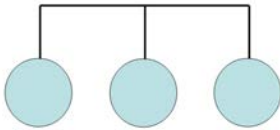


Рис. 2.4. Общая шина

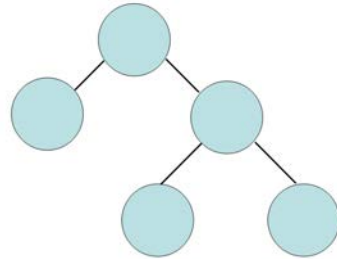


Рис. 2.5. Древоподобная топология

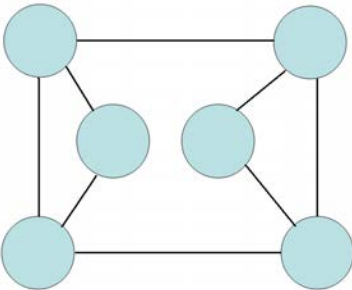


Рис. 2.6. Ячеистая топология

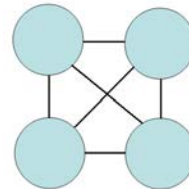


Рис. 2.7. Полносвязная топология

- *ячеистая* (рис. 2.6) — в такой сети есть, по крайней мере, два узла, имеющих два или более пути между ними;
- *полносвязная* (рис. 2.7) — сеть, в которой есть связь между любыми двумя узлами. Это самая надежная топология сети, но она практически никогда не используется, поскольку является самой дорогой и труднообслуживаемой.

2.2.3. По ведомственной принадлежности

По ведомственной принадлежности различают следующие виды сетей:

- *ведомственные* — принадлежат какой-то организации и находятся на ее территории;
- *государственные* — используются в госструктурах.

2.2.4. По скорости передачи данных

По скорости передачи данных сети делятся на низко-, средне- и высокоскоростные. Основным критерий разделения — скорость передачи данных. Понятно, что скорость передачи данных — понятие непостоянное. То, что сегодня считается среднескоростным соединением, завтра будет отнесено к низкоскоростным. Тем не менее, сегодня низкоскоростной сетью считается сеть со скоростью передачи информации до 10 Мбит/с. Среднескоростная сеть передает данные со скоростью до 100 Мбит/с, а высокоскоростные сети передают информацию со скоростью свыше 100 Мбит/с.

2.2.5. По типу среды передачи данных

Казалось бы, тут все просто: сети бывают проводными или беспроводными. Но очень часто "в природе" встречаются *смешанные*, или *гибридные* сети, сочетающие как проводные, так и беспроводные технологии. В этой книге больше внимания будет уделяться именно таким сетям. Представьте, что у вас дома есть компьютер и ADSL-модем. Вы купили ноутбук. Подключать ноутбук по Ethernet-кабелю не очень-то хочется — ноутбук по своей природе мобильное устройство, и хотелось бы использовать его по всей квартире (а если у вас свой дом, то и во дворе). Поэтому вам понадобится *точка доступа*, которую вы подключите к существующей сети. Точка доступа в данном случае будет выполнять функцию моста между беспроводной и проводной сетью. Именно с ее помощью ваш ноутбук сможет подключиться к Интернету. Поэтому у вас дома появится смешанная сеть, созданная своими руками.

2.2.6. По способу организации взаимодействия компьютеров

Сети бывают одноранговыми и типа клиент/сервер. В *одноранговой* сети нет выделенного сервера: каждый клиент сети может выступать как в роли сервера, то есть предоставлять услуги другим узлам сети, так и в роли клиента, то есть пользоваться услугами, которые предоставляют другие узлы сети.

В сети *клиент/сервер* есть выделенный сервер, предоставляющий определенные сетевые услуги (какие именно, зависит от самой сети).

2.3. Способы передачи данных в сетях

Любая сеть данных должна использовать какой-нибудь метод коммутации своих абонентов, то есть сеть должна знать, как отправить данные тому или иному ком-

пьютеру. В современных сетях распространены три основных метода коммутации: коммутация каналов, коммутация сообщений и коммутация пакетов.

Коммутация каналов используется в аналоговых (нецифровых) телефонных сетях. Для передачи компьютерных данных используется *коммутация пакетов*. Разница между этими методами просто огромна. В первом случае (коммутация каналов) для передачи данных нужен физический канал между двумя узлами. Понятно, что прокладывать кабель между каждой парой узлов сети (между каждой парой телефонов) экономически нецелесообразно, поэтому были созданы *коммутаторы* (сейчас мы говорим о телефонных коммутаторах), соединяющие между собой двух разных абонентов сети по их вызову.

В компьютерных сетях такой способ коммутации совершенно не годится, поскольку канал большую часть времени будет простаивать и без пользы занимать ресурсы коммутатора. Кроме того, при отправке большого количества информации по такой сети на коммутатор ляжет огромная нагрузка, поскольку данные будут переданы за один раз.

Метод коммутации пакетов заключается в том, что передаваемые данные разбиваются на части — *пакеты*. Каждый пакет отправляется отдельно, и, что интересно, два разных пакета, отправленные одним отправителем, могут прийти к получателю разными маршрутами. Например, вы отправляете пакеты компьютеру, не принадлежащему вашей сети. Сначала пакеты отправятся провайдеру, затем — какому-нибудь маршрутизатору Интернета, но если этот маршрутизатор окажется недоступен (мало ли чего может случиться), автоматически будет задействован резервный канал, отправляющий данные через другой маршрутизатор. В итоге получится, что первый пакет будет доставлен одним маршрутом, а второй — другим. Однако оба пакета будут доставлены получателю.

К тому же метод коммутации пакетов позволяет использовать физически одну и ту же среду передачи данных (читайте — один и тот же кабель) для (почти) одновременной отправки данных несколькими компьютерами. Рассмотрим ситуацию: у вас в квартире установлено два параллельных телефонных аппарата, и вы разговариваете по одному из них. Если кто-то поднимет трубку второго телефона, то не сможет набрать номер, поскольку среда передачи информации (телефонный кабель) занята.

В случае с коммутацией пакетов такого нет — в сетях с архитектурой "общая шина" (Ethernet) данные отправляются почти одновременно. Например, компьютерам А и Б нужно отправить данные. Допустим, первым получил доступ к общей среде компьютер А. Он отправляет пакет фиксированного размера. Пока компьютер А отправляет пакет, компьютер Б ожидает доступ к среде. После отправки пакета компьютером А компьютер Б сможет получить доступ к общей среде и отправить свой пакет. Компьютер А в это время делает небольшую паузу. Потом компьютер Б должен сделать паузу, за время которой компьютер А успеет передать следующий пакет. Сами понимаете, время ожидания настолько мизерно, что пользователям компьютеров А и Б кажется, что компьютеры отправляют данные одновременно. Если бы в компьютерной сети использовался метод коммутации каналов, то компьютер Б должен был ждать, пока компьютер А не передаст все данные.

Метод *коммутации сообщений* в чистом виде практически нигде не используется, но он послужил прототипом для метода коммутации пакетов.

2.4. Модель OSI

В 80-х годах прошлого века появилась необходимость стандартизировать различные сетевые технологии. Ведь без стандартизации в мире компьютерных сетей воцарился бы хаос: оборудование различных производителей не смогло бы работать вместе. Поэтому международная организация по стандартизации (International Organization for Standardization, IOS) разработала *модель взаимодействия открытых систем* (Open System Interconnection, OSI). Вы также можете встретить другие названия этой модели: *семиуровневая модель OSI*, или просто *модель OSI*. Эта модель предусматривает семь уровней взаимодействия систем:

- Физический.
- Канальный.
- Сетевой.
- Транспортный.
- Сеансовый.
- Представительный.
- Прикладной.

Зачем нужна такая система? Предположим, что нам необходимо заставить вместе работать две сети, использующие разную среду передачи данных, — например, витую пару и радиоволны (беспроводную сеть). Если бы не было модели OSI, то для каждой сети пришлось бы разрабатывать модель взаимодействия, а потом придумывать способ, позволяющий заставить две разные по своей архитектуре сети работать вместе. В случае с моделью OSI не нужно "изобретать велосипед" заново. Следует взять за основу уже имеющуюся сеть (в данном случае Ethernet) и переписать физический уровень. В итоге нам не придется разрабатывать браузеры, почтовые клиенты и другие сетевые приложения для каждой сети — браузеру все равно, какая среда передачи данных используется. Как видите, модель OSI хоть и теоретическая, зато очень полезная. Рассмотрим ее уровни:

- на **физическом уровне** определяются характеристики электрических сигналов, то есть описывается физическая среда данных. На этом уровне и происходит физическая передача данных по кабелю или радиоволнам (в зависимости от типа сети). Пример протокола физического уровня: 1000Base-T — спецификация Ethernet, передающая данные по витой паре 5-й категории (о категориях витой пары мы поговорим позднее) со скоростью 1000 Мбит/с;
- канальный уровень** используется для передачи данных между компьютерами (и другими устройствами, например, сетевыми принтерами) одной сети. Пример протокола канального уровня: PPP (Point-to-Point Protocol). Топология сети (шина, звезда и т. д.) определяется как раз на канальном уровне (ранее мы подробно рассмотрели все используемые топологии сетей). На канальном уровне все передаваемые данные разбиваются на части, называемые *кадрами* (frames). Канальный уровень передает кадры физическому уровню, а тот, в свою очередь, отправляет их в сеть.

На канальном уровне вводится понятие MAC-адреса. *MAC-адрес* — это уникальный аппаратный адрес сетевого устройства (например, сетевого адаптера, точки доступа). Каждому производителю сетевых устройств выделяется свой диапазон MAC-адресов. В мире нет двух сетевых устройств с одинаковыми MAC-адресами;

- теперь рассмотрим **сетевой уровень**. Он используется для объединения нескольких сетей, то есть для организации межсетевого взаимодействия, — ведь протоколы канального уровня могут работать только в пределах одной сети. Канальный уровень не может передать кадр компьютеру, который находится в другой сети. Понятно, что если бы у нас был только канальный уровень и не было сетевого, мы не смогли бы передавать данные между двумя сетями, например, между локальной сетью и Интернетом. Пример протокола сетевого уровня: IP (Internet Protocol). Конечно, IP — это не единственный протокол сетевого уровня, но в этой книге мы будем рассматривать только TCP/IP-сети, поэтому нет смысла упоминать другие протоколы.

При всем своем желании мы не можем построить огромную сеть, охватывающую весь мир (даже если бы это и удалось, не думаю, что такая сеть работала бы быстро). Поэтому Интернет состоит из совокупности различных сетей, которые объединяются в одно целое маршрутизаторами. Расстояние между сетями исчисляется в количестве переходов пакетов (на сетевом уровне передаваемые данные разбиваются именно на пакеты) через маршрутизаторы. Единица такого перехода называется *хопом* (от англ. hop). Количество хопов равно количеству маршрутизаторов между двумя сетями. Например, от моего узла до узла **volia.net** 6 хопов (шесть переходов), что показано на рис. 2.8;

```
denis@denis-desktop:~$ traceroute volia.net
traceroute to volia.net (82.144.192.47), 30 hops max, 40 byte packets
 1  router.shtorm.net (195.62.14.2)  0.330 ms  0.306 ms  0.295 ms
 2  border.shtorm.com (195.62.14.7)  0.523 ms  0.515 ms  0.504 ms
 3  194.44.13.13 (194.44.13.13)  9.435 ms  9.426 ms  9.415 ms
 4  volia-10G-gw.ix.net.ua (195.35.65.221)  9.618 ms  9.613 ms  9.603 ms
 5  v109.TenGig3-8.diamond.volia.net (82.144.193.192)  9.358 ms  9.562 ms  9.554
   ms
 6  tower.volia.net (82.144.192.47)  9.538 ms  9.251 ms  9.240 ms
denis@denis-desktop:~$
```

Рис. 2.8. Количество переходов от моего узла до узла **volia.net**

- **транспортный уровень** отвечает за доставку пакетов получателю. Не секрет, что при передаче по каналам связи данные могут быть повреждены или вовсе потеряны. Гарантирует доставку пакета в первоизданном виде именно транспортный уровень. На этом уровне осуществляются обнаружение и исправление ошибок, восстановление прерванной связи и некоторые дополнительные сервисы вроде срочной доставки (приоритет пакета) и мультиплексирование нескольких соединений. Самым известным и распространенным протоколом транспортного уровня является TCP (Transport Control Protocol);
- **сеансовый уровень** отвечает за установку и за разрыв соединения между компьютерами. На этом уровне также предоставляются средства синхронизации. Сеанс сетевого уровня заключается в установке соединения (при установке стороны, между которыми будут передаваться данные, могут договариваться о дополнительных параметрах связи, например, обмениваться ключами), передаче информации и разрыве соединения.

Многие часто путают сеансы сетевого уровня и сеанс связи. Вы можете установить сеанс связи (например, подключиться к Интернету), но не устанавливать логического соединения, то есть не запустить браузер для соединения с удаленным узлом;

- как было отмечено чуть ранее, на сеансовом уровне стороны могут договориться о дополнительных параметрах, были также упомянуты *ключи*. Однако само шифрование и дешифрование данных осуществляется **представительным уровнем**. Пример протокола этого уровня — SSL (Secure Socket Layer);
- последний (самый высокий) уровень — **прикладной**. На этом уровне работает множество разных протоколов, например, HTTP (Hyper Text Transfer Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol) и т. д.

2.5. Что такое протокол?

В этой главе довольно часто упоминалось слово "протокол". *Протокол* — это правила, определяющие взаимодействие компьютеров в вычислительной сети. Рассмотрим несколько самых важных протоколов.

В основе Интернета лежит протокол TCP/IP (Transmission Control Protocol/Internet Protocol). Чтобы система смогла работать в Интернете, она должна поддерживать протокол TCP/IP. Вообще говоря, TCP/IP — это совокупность двух протоколов. Протокол TCP, как уже было отмечено ранее, отвечает за корректность передачи данных по Интернету (точнее, по любой сети, использующий этот протокол), то есть гарантирует доставку данных по сети. Протокол IP используется для адресации компьютеров сети. Дело здесь в том, что у каждого компьютера сети имеется свой уникальный адрес (IP-адрес) и, чтобы передать данные компьютеру, нужно его IP-адрес знать. Чуть позже мы поговорим о системе доменных имен (DNS, Domain Name System).

Кроме протоколов TCP и IP посетители Интернета работают с теми или иными серверами, использующими следующие протоколы:

- HTTP (Hyper Text Transfer Protocol) — протокол передачи гипертекста. Все Web-серверы Интернета используют протокол HTTP или его безопасную версию HTTPS (HTTP Secure);
- FTP (File Transfer Protocol) — протокол передачи файлов. Используется для обмена файлами между компьютерами. Вы можете подключиться к FTP-серверу и скачать необходимые вам файлы или же, наоборот, закачать свои файлы на сервер, если вы обладаете надлежащими правами доступа к серверу. В Интернете много публичных FTP-серверов, к которым разрешен анонимный доступ. Как правило, с таких серверов скачивать файлы разрешено всем желающим. Иногда, но очень редко, разрешается и запись файлов на публичный сервер. В состав любой операционной системы входит FTP-клиент — программа `ftp`. К тому же многие браузеры можно использовать в качестве FTP-клиента;
- SMTP (Simple Mail Transfer Protocol) — простой протокол передачи почты. Используется для отправки почты (e-mail);

- POP (Post Office Protocol) — протокол, используемый для получения сообщений электронной почты;
- IMAP (Internet Message Access Protocol) — еще один протокол для получения почты, но, в отличие от протокола POP, этот протокол позволяет читать почту без ее загрузки на компьютер пользователя. Протокол IMAP, по сути, намного удобнее, чем POP. Ведь почта хранится на сервере, и вы можете получить доступ к ней с любой точки земного шара, используя любой клиент. К тому же IMAP поддерживает поиск писем на сервере, что позволяет найти нужное письмо без загрузки всех писем на свой компьютер. Однако у IMAP есть один существенный недостаток, благодаря чему до сих пор распространен протокол POP — IMAP требует постоянного соединения с сервером. Если нет соединения с сервером, то вы не прочитаете не только новые сообщения, но и те, которые были получены ранее, поскольку все они хранятся на сервере. Так что в случае с IMAP об автономной работе (без подключения к Интернету) можно забыть.

ПРИМЕЧАНИЕ

На днях полностью "перекочевал" в Linux (после очередного "падения" Windows). Так вот, до сих пор не знаю, как перенести в Linux 20 тыс. сообщений (разбитых на множество папок со своими правилами сортировки) общим объемом около 4 Гбайт. Почта хранилась в программе The Bat!, а я сейчас использую почтовый клиент Mozilla Thunderbird. Программы для конвертации почтового формата The Bat! в формат Thunderbird пока не разработано... Если бы я использовал протокол IMAP, проблема отпала бы автоматически.

2.6. Адресация компьютеров

Для идентификации узлов Интернета используются IP-адреса. IP-адрес представляет собой четыре числа, разделенные точками (или одно 32-разрядное число, которое записывается в виде четырех восьмиразрядных чисел, разделенных точками, — как кому больше нравится). Нужно сразу отметить, что такая идентификация неоднозначная, поскольку IP-адреса могут быть статическими (постоянными) и динамическими. *Постоянные* (статические) IP-адреса обычно назначаются серверам, а *динамические* — обычным пользователям. Так что сегодня определенный динамический IP-адрес может быть назначен одному пользователю, а завтра — другому. Поэтому если в случае с аппаратными MAC-адресами еще можно говорить о какой-то однозначности (и то существуют способы подделки MAC-адресов), то IP-адреса по определению однозначными не являются.

Рассмотрим примеры IP-адресов: 127.0.0.1, 192.168.1.79, 111.33.12.99. Как было сказано ранее, IP-адрес — это одно 32-разрядное число или четыре 8-разрядных числа. Возведем 2 в восьмую степень и получим максимальное значение для каждого из четырех восьмиразрядных чисел — 256. Таким образом, учитывая, что некоторые IP-адреса зарезервированы для служебного использования, протокол IP может адресовать примерно 4,3 млрд узлов. Однако с каждым годом количество узлов во Всемирной паутине увеличивается, поэтому была разработана шестая версия протокола IP — IPv6 (если упоминается просто протокол IP, то, как правило, имеется в виду четвертая версия протокола — IPv4). Новый протокол использует 128-битные адреса (вместо 32-битных), что позволяет увеличить число узлов

до 10^{12} и количество сетей до 10^9 . IPv6-адреса отображаются как 8 групп шестнадцатеричных цифр, разделенных двоеточиями. Вот пример адреса нового поколения: 1628:0d48:12a3:19d7:1f35:5a61:17a0:765d.

ПРИМЕЧАНИЕ

Впрочем, массовый переход на IPv6 (который еще называют IPng — IP Next Generation) пока так и не состоялся, хотя его используют несколько сотен сетей по всему миру. В этой книге мы будем рассматривать только протокол IPv4, поскольку, судя по всему, Интернет не перейдет на IPv6 в ближайшие несколько лет. Интересующиеся могут прочитать об IPv6 по адресу: <http://ru.wikipedia.org/wiki/IPv6>.

IP-адреса выделяются *сетевым информационным центром* (NIC, Network Information Center). Чтобы получить набор IP-адресов для своей сети, вам надо обратиться в этот центр. Но, оказывается, это приходится делать далеко не всем. Существуют специальные IP-адреса, зарезервированные для использования в локальных сетях. Ни один узел глобальной сети (Интернета) не может обладать таким "локальным" адресом. Вот пример локального IP-адреса — 192.168.1.1. В своей локальной сети вы можете использовать любые локальные IP-адреса без согласования с кем бы то ни было. Когда же вы надумаете подключить свою локальную сеть к Интернету, вам понадобится всего один "реальный" IP-адрес — он будет использоваться на маршрутизаторе (шлюзе) доступа к Интернету.

Чтобы узлы локальной сети (которым назначены локальные IP-адреса) смогли "общаться" с узлами Интернета, используется специальная технология *трансляции сетевого адреса* (NAT, Network Address Translation). Маршрутизатор получает пакет от локального узла, адресованный интернет-узлу, и преобразует IP-адрес отправителя, заменяя его своим IP-адресом. При получении ответа от интернет-узла маршрутизатор выполняет обратное преобразование, поэтому нашему локальному узлу "кажется", что он общается непосредственно с интернет-узлом. Если бы маршрутизатор отправил пакет как есть, то есть без преобразования, то его отверг бы любой маршрутизатор Интернета, и пакет так и не был бы доставлен к получателю.

Наверное, вам не терпится узнать, какие IP-адреса можно использовать без согласования с NIC? Об этом говорить пока рано — ведь мы еще ничего не знаем о *классах* сетей. IP-адреса используются не только для адресации отдельных компьютеров, но и целых сетей. Вот, например, IP-адрес сети — 192.168.1.0. Отличительная черта адреса сети — 0 в последнем октете.

Сети поделены на классы в зависимости от их размеров:

- класс А — огромные сети, которые могут содержать 16777216 адресов, IP-адреса сетей лежат в пределах 1.0.0.0 — 126.0.0.0;
- класс В — средние сети, содержат до 65536 адресов. Диапазон адресов — от 128.0.0.0 до 191.255.0.0;
- класс С — маленькие сети, каждая сеть содержит до 256 адресов.

Существуют еще и классы D и E, но класс E не используется, а зарезервирован на будущее (хотя будущее — это IPv6), а класс D зарезервирован для служебного использования (широковещательных рассылок).

Представим ситуацию. Вы хотите стать интернет-провайдером. Тогда вам нужно обратиться в NIC для выделения диапазона IP-адресов по вашей сети. Скажем, вы планируете сеть в 1000 адресов. Понятно, что сети класса С вам будет недостаточно.

Поэтому можно или арендовать четыре сети класса С, или одну класса В. Но, с другой стороны, 65536 адресов для вас — много, и если выделить вам всю сеть класса В, то это приведет к нерациональному использованию адресов. Так что самое время поговорить о *маске сети*. Маска сети определяет, сколько адресов будет использоваться сетью, фактически — маска задает размер сети. Маски полноразмерных сетей классов А, В и С представлены в табл. 2.1.

Таблица 2.1. Маски сетей классов А, В и С

Класс сети	Маска сети
А	255.0.0.0
В	255.255.0.0
С	255.255.255.0

Маска 255.255.255.0 вмещает 256 адресов (в последнем октете IP-адреса могут быть цифры от 0 до 255). Например, если адрес сети 192.168.1.0, а маска 255.255.255.0, то в сети могут быть IP-адреса от 192.168.1.0 до 192.168.1.255. Первый адрес (192.168.1.0) называется IP-адресом сети, последний — зарезервирован для широковещательных рассылок. Следовательно, для узлов сети остаются 254 адреса — от 192.168.1.1 до 192.168.1.254.

А вот пример маски сети на 32 адреса: 255.255.255.224 (255 – 224 = 31 + "нулевой" IP-адрес, итого 32).

Предположим, у нас есть IP-адрес произвольной сети, например, 192.168.1.0. Как узнать, к какому классу она принадлежит? Для этого нужно преобразовать первый октет адреса в двоичное представление. Число 192 в двоичной системе будет выглядеть так: **11000000**. Проанализируем первые биты первого октета. Если первые биты содержат двоичные цифры 110, то перед нами сеть класса С. Теперь сделаем то же самое с сетью 10.0.0.0. Первый октет равен 10, и в двоичной системе он будет выглядеть так: 00001010. Первый бит — 0, поэтому сеть относится к классу А. Опознать класс сети по первым битам первого октета поможет табл. 2.2.

Таблица 2.2. Опознание класса сети

Класс сети	Первые биты
А	0
В	10
С	110
Д	1110
Е	11110

Теперь поговорим о специальных зарезервированных адресах. Адрес 255.255.255.255 является *широковещательным*. Если пакет отправляется по этому адресу, то он будет доставлен всем компьютерам, находящимся с отправителем в одной сети. Можно уточнить сеть, компьютеры которой должны получить широковещатель-

ную рассылку, например, таким образом: 192.168.5.255. Этот адрес означает, что пакет получат все компьютеры сети 192.168.5.0.

Вам также следует знать адрес 127.0.0.1. Этот адрес зарезервирован для обозначения локального компьютера и называется *адресом обратной петли*. Если отправить пакет по этому адресу, то его получит ваш же компьютер, то есть получатель является отправителем, и наоборот. Данный адрес обычно используется для тестирования поддержки сети. Более того, к локальному компьютеру относится любой адрес из сети класса А с адресом 127.0.0.0. Поэтому при реальной настройке сети нельзя использовать IP-адреса, начинающиеся со 127.

А теперь можно рассмотреть IP-адреса сетей, зарезервированные для локального использования. В локальных сетях вы можете использовать следующие адреса сетей:

- 192.168.0.0 — 192.168.255.0 — сети класса С (всего 256 сетей, маска 255.255.255.0);
- 172.16.0.0 — 172.31.0.0 — сети класса В (всего 16 сетей, маска 255.255.0.0);
- 10.0.0.0 — сеть класса А (одна сеть, маска 255.0.0.0).

Обычно в небольших домашних и офисных сетях используются IP-адреса из сети класса С, то есть из диапазона 192.168.0.0 — 192.168.255.0. Но поскольку назначение адресов контролируется только вами, вы можете назначить в своей локальной сети любые адреса, например, адреса из сети 10.0.0.0, даже если у вас в сети всего 5 компьютеров. Так что выбор сети — это дело вкуса. Можете себя почувствовать администратором огромной сети и использовать адреса 10.0.0.0.

2.7. Система DNS

Узлов в Интернете достаточно много, поэтому ни один человек не способен запомнить IP-адреса всех необходимых ему узлов. Да и гораздо легче запомнить символичный адрес, скажем, **www.bhv.ru** или **www.dkws.org.ua**, чем их IP-адреса. Тем более, относительно недавно появилась возможность регистрации доменных имен на русском языке (точнее — на кириллице). Не знаю, приживутся ли такие доменные имена, но то, что они существуют, — это факт.

За преобразование IP-адресов в доменные имена и обратно отвечает *система доменных имен* (DNS, Domain Name System). Когда вы вводите доменное имя в строке браузера, система сначала разрешает это имя в IP-адрес (путем обращения к DNS-серверу), а потом подключается к узлу по полученному IP-адресу.

Не нужно думать, что система DNS появилась недавно. Она более "древняя", чем вы можете предположить. Впервые DNS была представлена в 1984 году. Правда, тогда далеко не все сети перешли на использование DNS-серверов. До этого доменные имена разрешались в IP-адреса с помощью файла *hosts*, в котором содержалась таблица соответствия доменных имен IP-адресам. Понятно, что такой файл нужно постоянно поддерживать в актуальном состоянии. Когда количество узлов увеличилось и поддержка этого файла стала проблемой для администратора сети, вот тогда и началась эра DNS. Кстати, файлом *hosts* можно пользоваться до сих пор. Для обеспечения совместимости его можно использовать даже в самых современных ОС (как в UNIX/Linux, так и в Windows), но, сами понимаете, происходит это очень редко.

Система DNS более подробно будет рассмотрена в *главе 27*. Мы даже настроим собственный DNS-сервер.

Глава 3



Основные сетевые устройства

3.1. Активное и пассивное сетевое оборудование

Для построения компьютерной сети, то есть для организации передачи информации между компьютерами, используется сетевое оборудование. Сетевое оборудование бывает активным и пассивным. *Активным* называется оборудование, обладающее неким "интеллектом" — например, коммутатор (switch), маршрутизатор (router). *Пассивное* сетевое оборудование "интеллектом" не наделено. К пассивному оборудованию относят кабели (например, коаксиальный или витая пара), розетки (RJ45, RG58 и др.), повторитель (repeater), концентратор (hub) и т. д.

Стоп! Если вы хоть немного знакомы с Ethernet-сетями, вы можете запутаться. Ведь концентратор, как и коммутатор, можно использовать в качестве центрального сетевого устройства в Ethernet-сети, почему тогда концентратор — это пассивное устройство, а коммутатор — активное? Дело в том, что концентратор не проявляет никакой интеллектуальной деятельности — он просто получает сигналы и копирует (повторяет) их на все свои порты, равно как и повторитель. Повторитель получает сигнал, усиливает его и повторяет на другой порт. Повторители обычно используются для увеличения дальности передаваемого сигнала. Коммутатор же "знает", к какому порту подключен какой компьютер, поэтому передает полученный сигнал не на все порты, а только на определенный порт, к которому подключен компьютер-назначение.

Различного сетевого оборудования очень много. Мы не будем пытаться объять необъятное, поэтому в этой книге рассмотрим только оборудование, необходимое для построения проводных Ethernet-сетей и беспроводных сетей Wi-Fi.

3.2. Оборудование, необходимое для построения Ethernet-сети

Для организации современной Ethernet-сети (имеются в виду спецификации Fast Ethernet и Gigabit Ethernet) необходим всего один коммутатор (switch). Конечно, если сеть большая, то понадобится несколько коммутаторов, общее количество портов которых сможет обеспечить подключение всех узлов сети. На рис. 3.1 изображен так называемый *промышленный* коммутатор от Linksys.



Рис. 3.1. 16-портовый коммутатор от Linksys

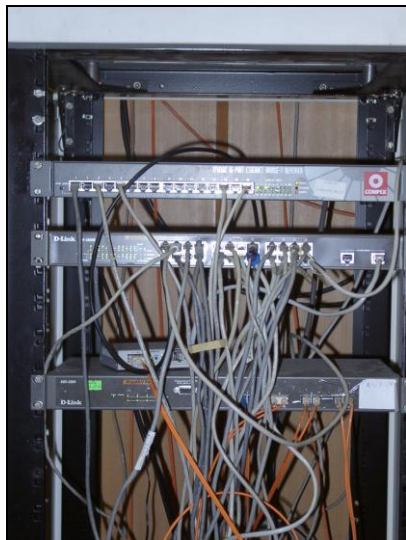


Рис. 3.2. Стойка с коммутаторами



Рис. 3.3. Шкаф с сетевым оборудованием



Рис. 3.4. 8-портовый гигабитный коммутатор от D-Link

Дизайн корпуса промышленного коммутатора обычно не очень эффектен, но сделано это умышленно — чтобы коммутатор можно было поместить в стойку сетевого оборудования. Ведь в больших корпоративных сетях обычно несколько коммутаторов, которые помещаются в специальную стойку (или в специальный шкаф сетевого оборудования, который можно закрыть и тем самым ограничить физический доступ к нему). На рис. 3.2 изображена типичная стойка с коммутаторами.

А на рис. 3.3 показан шкаф с коммутаторами. Такой шкаф может быть большего размера и содержать другое оборудование (например, серверы сети), но главное отличие шкафа от стойки — наличие двери, которая ограничивает доступ к сетевому оборудованию.

Если вы хотите построить небольшую домашнюю или офисную сеть, то можете выбрать коммутатор с более интересным дизайном, который лучше впишется в ваш интерьер. На рис. 3.4 представлен 8-портовый гигабитный коммутатор от D-Link. Вид у него более "дружелюбный", но в стойку его уже не поместишь (впрочем, при организации домашней сети никакой стойки у вас и не будет).

Давайте теперь уточним, почему в современных сетях не стоит использовать концентраторы (hub). Представим, что у нас есть сеть на четыре компьютера. Назовем их А, Б, В и Г. Пусть компьютер А отправляет данные компьютеру Г. Концентратор отправит полученный от компьютера А сигнал на все свои порты — то есть сигнал, отправленный компьютером А, получают все компьютеры сети. Затем каждый компьютер анализирует заголовки пакета, в которых указан компьютер-получатель. Если адрес компьютера совпадает с адресом получателя, компьютер принимает пакет, в противном случае — игнорирует его. Таким образом, использование концентратора приводит к "брожению" по сети паразитного трафика. По сути, концентратор — это обычный многопортовый повторитель (усилитель) сигналов. И чем больше сеть, тем медленнее она работает в случае использования концентратора, поскольку "брожение" паразитного трафика носит лавинообразный характер. Вы только представьте, что в сети не четыре компьютера, а несколько десятков... Поэтому в больших сетях концентраторы существенно снижают производительность сети.

Коммутатор же, в отличие от концентратора, строит специальную таблицу соответствия, позволяющую однозначно узнать, к какому порту какой компьютер подключен (см., например, табл. 3.1).

Таблица 3.1. Таблица соответствия портов коммутатора и адресов компьютеров

Номер порта	Адрес компьютера
1	Б
2	А
3	Г
4	В

Когда компьютер А, подключенный ко второму порту коммутатора, отправляет пакет компьютеру Г, коммутатор знает, что компьютер Г подключен к третьему порту, и отправляет пакет только на третий порт. При этом снижается нагрузка на сеть, потому что компьютеры не получают "лишних" пакетов.

Кроме того, поскольку концентратор отправляет данные каждому компьютеру сети, становится очень простым перехват данных. Существуют специальные программы, переводящие сетевой адаптер компьютера в режим мониторинга, в котором он осуществляет принятие всех данных, даже тех, которые не адресованы этому компьютеру. Поэтому, если в сети используется концентратор, все передаваемые данные становятся общим достоянием — их может перехватить любой компьютер, подключенный к концентратору.

Итак, использование коммутатора позволяет повысить производительность сети и повысить ее безопасность. Ранее сети в основном строились на базе концентраторо-

ров, поскольку их стоимость была существенно ниже стоимости коммутаторов. Со снижением цен на коммутаторы концентраторы практически исчезли с магазинных полок. Однако в некоторых старых сетях они еще используются. Если вам придется обслуживать такую сеть, первым делом замените концентратор на коммутатор — вы сразу почувствуете разницу.

Какой коммутатор применить: Fast Ethernet (100Base-T) или Gigabit Ethernet (1000Base-T)? В первом случае максимальная (теоретическая) скорость передачи данных составляет 100 Мбит/с, во втором случае — 1000 Мбит/с. Коммутаторы Gigabit Ethernet стоят немного дороже (цены приводить не буду, поскольку через год они станут еще доступнее, а через два — о Fast Ethernet забудут, как в свое время забыли о коаксиале и концентраторах).

Учитывая, что сеть строится не на день и не на два, лучше выбрать Gigabit Ethernet. С точки зрения монтажа сети ничего не изменится — даже если вы сейчас установите коммутатор Fast Ethernet, то завтра без проблем сможете заменить его на Gigabit Ethernet. Но нужно помнить следующее: чтобы сеть работала в режиме 1000Base-T, необходимо, чтобы 1000Base-T поддерживали сетевые адаптеры компьютеров. Практически на всех современных материнских платах встроенные сетевые адаптеры уже поддерживают 1000Base-T, но если в вашей сети есть компьютеры, которым 2–3 года, скорее всего, вам придется докупать для них сетевые адаптеры с поддержкой 1000Base-T.

Идем дальше — количество портов. Обычно в продаже есть коммутаторы на 5, 8, 16, 24 порта. Промышленные коммутаторы могут иметь большее число портов, например 32 или 48. Может быть, в скором времени появятся коммутаторы с еще большим числом узлов, но я сомневаюсь. Поскольку обычно один коммутатор обслуживает одну подсеть, я не думаю, что в одной подсети будет больше 48 компьютеров. А если это случится, такую подсеть желательно (из соображений локализации трафика) разделить на несколько подсетей с меньшим числом компьютеров.

Так что для домашней сети покупайте коммутатор, способный подключить все имеющиеся дома компьютеры, — большой запас портов вам вряд ли понадобится. Обычно в домашней сети 2–4 компьютера. В этом случае вам будет достаточно 5-портового коммутатора — 5-й порт пригодится для подключения этого коммутатора к другому коммутатору сети. В коммутаторах с большим числом портов для подключения к другому коммутатору обычно используется один из имеющихся портов (например, порт 1).

Промышленные коммутаторы иногда имеют так называемый *магистральный* порт. Например, 16 портов, работающих в режиме 100Base-T, и один порт, работающий в режиме 1000Base-T, — для подключения к магистральной сети, работающей со скоростью 1000 Мбит/с. Иногда вместо порта 1000Base-T оборудуется оптоволоконный порт, например, 100Base-FB. В этом случае скорость магистральной такой же, как и скорость сети, но расстояние передачи сигнала намного выше (более 2 км), что позволяет использовать оптоволоконный кабель для соединения сетей двух (или более) зданий в одну большую сеть.

В случае с офисной сетью количество портов коммутатора должно в два раза превышать количество компьютеров сети. Например, если в вашей сети четыре компьютера, то нужен 8-портовый коммутатор. Дополнительные четыре порта

могут понадобиться, если придется подключить дополнительные компьютеры, например, ноутбуки ваших клиентов, если у вас пока еще нет для них точки доступа Wi-Fi.

По большому счету, для организации сети больше ничего и не нужно (разумеется, кроме кабеля и коннекторов RJ45, но это уже детали, о которых мы поговорим в *третьей части* книги).

3.3. Оборудование, необходимое для построения сети Wi-Fi

Как и в случае с Ethernet-сетью, нам понадобятся сетевые адаптеры и центральное устройство сети. Только сетевые адаптеры нужны не обычные, а беспроводные. А роль центрального устройства сети будет играть *точка доступа* (access point).

Все современные модели ноутбуков по умолчанию оснащены адаптером Wi-Fi, а вот стационарные (настольные) компьютеры придется дооснастить беспроводными сетевыми адаптерами. Проще всего купить беспроводной адаптер, подключающийся к компьютеру по USB. Есть также адаптеры, выполненные в виде PCI-карты, устанавливаемой в свободный PCI-слот компьютера. Такие адаптеры используются редко, поскольку их установка требует вскрытия корпуса компьютера, что несколько неудобно (особенно, если компьютер еще на гарантии — тогда придется нести его в сервисный центр, а что делать, если таких компьютеров много?).

USB-адаптеры могут быть выполнены в разных корпусах. На рис. 3.5 изображен небольшой беспроводной адаптер, напоминающий по своим размерам флешку. У такого адаптера антенна встроенная, поэтому его можно использовать только, если компьютер находится в зоне уверенного приема. Если же компьютер установлен ближе к "мертвой" зоне, лучше выбрать адаптер, выполненный в виде отдельного устройства (рис. 3.6). Такой адаптер обычно имеет небольшой размер и подключается к компьютеру USB-кабелем (питание адаптер получает тоже по USB). Преимущество этого адаптера заключается в следующем — его можно легко передвинуть в пределах длины USB-кабеля, чтобы попасть в зону уверенного приема сети. Ноутбук можно легко переместить в эту зону — просто взяли и перенесли. Со стационарным компьютером такого не сделаешь — у каждого стационарного компьютера есть свое место. А что делать, если в том месте, где установлен компьютер, не обеспечивается уверенный прием беспроводных сигналов? Не переносить же компьютер? В этой ситуации поможет адаптер, изображенный на рис. 3.6. Иногда перемещение адаптера всего на несколько сантиметров дает весьма ощутимые результаты. Да и антенна у такого адаптера обладает большей чувствительностью, чем встроенная антенна адаптера, изображенного на рис. 3.5. К тому же к подобным адаптерам (с внешней антенной) обычно можно подключить дополнительную антенну с еще большей чувствительностью. Обо всем этом мы поговорим, когда будем строить свою собственную беспроводную сеть. А сейчас перейдем лучше к точке доступа.

ПРИМЕЧАНИЕ

При выборе Wi-Fi-адаптера учитывайте наличие драйверов — особенно, если вы планируете использовать его в Linux. Чтобы не получилось так, что Linux не поддерживает купленный Wi-Fi-адаптер.



Рис. 3.5. USB Wi-Fi-адаптер со встроенной антенной



Рис. 3.6. USB Wi-Fi-адаптер с внешней антенной



Рис. 3.7. Точка доступа от D-Link с тремя антеннами

Точка доступа (рис. 3.7) выполняет в беспроводной сети роль центрального устройства. Казалось бы, все здесь просто: устанавливаем Wi-Fi-адаптеры, подключаем точку доступа, и беспроводная сеть готова — беспроводные клиенты могут обмениваться данными. Однако, если вы планируете купить точку доступа прямо сейчас, не следует покупать первую попавшуюся. Сначала желательно определиться, какие функции точки доступа вам нужны, затем "вычислить" модели точек доступа, обеспечивающие необходимые вам функции, и просмотреть в Интернете отзывы об этих моделях. Только так можно выбрать лучшую точку доступа.

Точка доступа может предоставлять дополнительные функции — например, функции *маршрутизатора*. Предположим, у вас дома есть несколько ноутбуков. К одному ноутбуку подключен ADSL-модем. Как организовать общий доступ к Интернету? Покупается точка доступа, к которой этот ADSL-модем и подключается. Ноутбуки (беспроводные клиенты) будут подключаться к Интернету по Wi-Fi, а точка доступа выступит в роли маршрутизатора.

3.4. Дополнительные сетевые устройства

Представим, что у нас есть два (или более) обычных (настольных) компьютера и одно ADSL-соединение. И нужно обеспечить общий доступ к Интернету. Это можно сделать средствами Windows. Тогда в один компьютер надо будет установить дополнительный сетевой адаптер. Первый сетевой адаптер будет использоваться

для подключения к Интернету, а второй — для подключения к локальной сети (для связи с остальными компьютерами сети). Компьютер с двумя сетевыми адаптерами для остальных компьютеров сети будет выполнять роль *шлюза* (gateway). Преимущество такого решения — дешевизна: ведь мы обеспечили общий доступ к Интернету практически без дополнительных устройств. Недостаток заключается в том, что компьютер-шлюз должен быть постоянно включен, иначе остальные компьютеры не смогут подключиться к Интернету.

Решить эту проблему можно, купив отдельное устройство, называемое *маршрутизатором* (при рассмотрении выбора точки доступа мы это устройство уже упоминали). Маршрутизатор обеспечивает передачу пакетов по заданному маршруту. В нашем случае — от локальных компьютеров к интернет-провайдеру. Таким образом, все компьютеры сети будут подключаться к центральному коммутатору, а он, в свою очередь, — к маршрутизатору. Также к маршрутизатору будет подключен и ADSL-модем.

Маршрутизаторы бывают разные. Некоторые могут выполнять роль коммутатора. Купив такой маршрутизатор, вы сократите количество активного сетевого оборудования (а значит, сэкономите деньги) до двух единиц — маршрутизатора и ADSL-модема. Если же у вас в сети компьютеров немного (2–4), можно подыскать ADSL-модем с функциями маршрутизатора. В этом случае у вас будет всего одна "коробочка" — все компьютеры сети будут подключены к этому устройству, которое, в свою очередь, будет подключено к телефонной сети. Этим вы сэкономите еще больше средств. Поэтому очень важно перед построением сети спланировать сей процесс. Хорошее планирование не только позволяет сэкономить деньги, но и время, впоследствии потраченное на дальнейшую модернизацию сети.

А теперь представим, что в нашей сети есть два (или больше, количество — не принципиально) стационарных компьютера и несколько ноутбуков. Ноутбуки было бы хорошо подключать по Wi-Fi. Стационарные компьютеры принято подключать по Ethernet (хотя бы потому, что не хочется покупать для них беспроводные адаптеры). Так вот, можно купить устройство, которое одновременно является ADSL-модемом, беспроводной точкой доступа и коммутатором. Одним из таких устройств является DSL-2640U от D-Link (далее мы рассмотрим процесс настройки этого устройства). Это устройство (рис. 3.8) позволяет объединить в сеть несколько



Рис. 3.8. ADSL-модем, маршрутизатор, коммутатор и беспроводная точка доступа D-Link DSL-2640U

беспроводных клиентов (это наши ноутбуки) и четыре проводных клиента. Все клиенты (как проводные, так и беспроводные) автоматически настраиваются на доступ к Интернету по совместно используемому ADSL-каналу. Кроме того, это

устройство обладает встроенным брандмауэром, что позволяет защитить вашу сеть от вторжения извне.

Простота настройки сети с помощью такого устройства просто поражает. Все, что вам нужно — это включить устройство, подключить к нему клиентов, запустить программу настройки (как это сделать, написано в руководстве по устройству) и установить базовые параметры сети, а именно: имя пользователя и пароль для ADSL-соединения, идентификатор беспроводной сети (SSID) и выбрать тип шифрования беспроводных соединений. Вот и все — сеть будет работать. Клиентов можно вовсе не настраивать — они будут автоматически настроены по протоколу DHCP (Dynamic Host Configuration Protocol, протокол динамической настройки узла).

Впрочем, у всех комбинированных устройств есть один недостаток — плохая масштабируемость. Если ваша сеть будет расти, добавить новых клиентов в нее будет сложно, а в некоторых случаях вообще невозможно. Тогда придется покупать отдельные устройства. Например, коммутатор, к которому будут подключаться до 48 проводных клиентов, и точку доступа для подключения беспроводных клиентов. В свою очередь, точка доступа и коммутатор будут подключаться к ADSL-маршрутизатору. Хотя в сложных случаях целесообразнее использовать программный (не аппаратный) маршрутизатор — компьютер под управлением UNIX/Linux. Такой компьютер можно использовать в роли маршрутизатора и на нем запустить брандмауэр, DNS-, WWW-, FTP- и почтовый серверы.

Итак, в этой главе мы ознакомились с основными сетевыми устройствами. Следующая глава будет сугубо теоретической. Мы поговорим о том, что должен знать каждый администратор и опытный пользователь любой сети, рассмотрим модель OSI и адресацию в TCP/IP-сети.

Глава 4



Планирование сети

4.1. Важность планирования

Вспомним старую русскую поговорку "семь раз отмерь, а один раз отрежь". Она очень точно подходит к нашему случаю. Конечно, бытует мнение, что пока семь раз будете мерить, кто-то уже отрежет. Согласен, но не сейчас. Сейчас вы планируете сеть, вы — главный, и вам никто не мешает. Очень важно продумать все нюансы, связанные с построением сети. Ведь корпоративная сеть — это очень сложная система, состоящая из тысяч различных компонентов. Это в маленькой домашней сети могут быть два-три компьютера, коммутатор, модем и принтер, подключенный к одному из компьютеров (не думаю, что в домашней сети кто-то организует принт-сервер). А в корпоративной сети могут быть самые разнообразные устройства, которые некоторые домашние пользователи даже ни разу в жизни и не видели. Скажем, кто из обычных домашних пользователей видел настоящий *мейнфрейм*, *кластер* или хотя бы обычный *терминал*, подключаемый к мейнфрейму?

Очень важно ориентироваться во всем этом оборудовании. Ведь жизнь не стоит на месте — все развивается с очень большой скоростью, особенно информационные технологии. Модель маршрутизатора, которая была популярна в прошлом году, уже давно такой не является — на ее место пришла новая, с более совершенными функциями, позволяющими эффективнее использовать всю систему в целом. Поэтому прежде чем закупать оборудование для сети, нужно ознакомиться с возможностями самых последних моделей устройств, а также сравнить устройства других производителей. Вот пример: всю жизнь вы считали, что устройства фирмы ААА (не хочется делать никому никакой рекламы — ни хорошей, ни плохой) — лучшие, но вот всего полгода назад на рынке появилась компания ВВВ, которая начала производство устройств, которые по всем своим характеристикам превосходят устройства компании ААА. Вы привыкли к компании ААА, поэтому всеми правдами и неправдами (мол, устройство от ВВВ еще не проверены временем и т. д.) будете уговаривать себя остановить свой выбор на устройстве от ААА, хотя прекрасно знаете, что устройство от ВВВ явно превосходит его характеристиками. С одной стороны, вы правы — проверенные временем, надежные устройства обеспечивают безотказную работу сети. А с другой стороны — нет, ведь уже через полгода все будут пользоваться принципиально новыми устройствами ВВВ, а вы построили свою сеть на устаревшем оборудовании от ААА.

Интернет внес огромные изменения в корпоративную сеть. Сейчас по каналам Интернета можно передать любую информацию: если раньше преимущественно передавался текст, графика и иногда звук, то сейчас видеоконференции он-лайн — это норма. Кроме того, Интернет можно использовать как компонент корпоративной сети — для передачи корпоративной информации по каналам Интернета: это существенно дешевле, чем прокладывать свои линии связи.

4.1.1. Планирование как основа безопасности

При планировании нужно учитывать еще и *безопасность* сети. Да, это нужно делать именно при планировании, а не после того, как сеть уже построена. Поэтому о безопасности — отдельный разговор.

Небольшой пример уже был приведен ранее: использовать проверенные временем решения или применить новые? Это касается не только оборудования, но и программного обеспечения, которое также является компонентом корпоративной сети, причем очень важным компонентом. Весьма желательно найти золотую середину между проверенными временем решениями и новыми разработками.

В предыдущем разделе мы начали говорить о влиянии Интернета на корпоративную сеть. По данным ISC (www.isc.org) в январе 2010 года (более новых данных пока нет, поэтому будем считать, что это последние данные) в Сети насчитывалось более 732 миллионов (!) узлов (рис. 4.1). Сейчас их еще больше. Посмотреть отчет ISC можно по адресу: <https://www.isc.org/solutions/survey>.

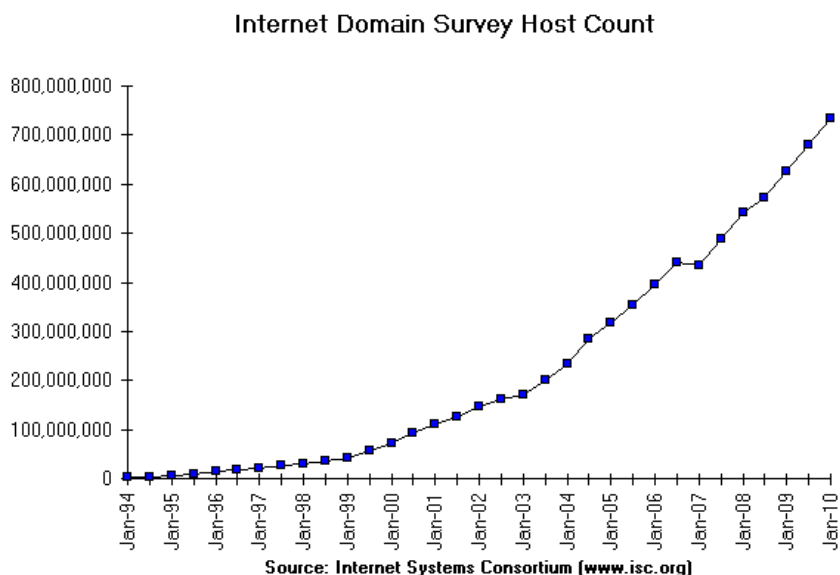


Рис. 4.1. Динамика роста количества узлов (по данным ISC)

Для сравнения: в январе 2009 года было 625 миллионов узлов. За год (с 1 января 2009 года по 1 января 2010 года) добавилось 107 миллионов узлов. Это целая армия