bhv®

АЛЕКСЕЙ СТАРОВОЙТОВ

# СЕТЬ НА LINUX

## ПРОЕКТИРОВАНИЕ, ПРОКЛАДКА, ЭКСПЛУАТАЦИЯ

Основы функционирования сетей

Принципы работы сетевой аппаратуры

Организация локальной сети с нуля

Генерация сервера на основе ASPLinux (Red Hat)

СИСАДМИН
СИСТЕМНЫЙ АДМИНИСТРАТОР

Алексей Старовойтов

# СЕТЬ НА LINUX

## ПРОЕКТИРОВАНИЕ, ПРОКЛАДКА, ЭКСПЛУАТАЦИЯ

Санкт-Петербург
«БХВ-Петербург»
2006

УДК 681.3.06
ББК 32.973.202
С77

**Старовойтов А. А.**

С77 Сеть на Linux: проектирование, прокладка, эксплуатация. — СПб.: БХВ-Петербург, 2005. — 288 с.: ил.

ISBN 5-94157-687-0

Рассмотрены практические вопросы по прокладке сети, организации сервера (Apache, Samba, DNS, DHCP) на основе операционной системы Linux и интеграции этого сервера в сетях Windows. Большое внимание уделено повседневной эксплуатации сети. Излагаются основы функционирования сетей и сетевой аппаратуры. Даются практические рекомендации по проектированию и прокладке сетей, проверке и поиску неисправностей. Приводится пример построения локальной сети небольшой фирмы и методика поиска неисправностей без использования специального оборудования. Рассмотрены вопросы антивирусной защиты сервера. Описанная технология может быть использована не только при прокладке и сопровождении сети небольшой фирмы на основе Linux-сервера, но и для организации домашних сетей.

*Для системных администраторов и опытных пользователей*

УДК 681.3.06
ББК 32.973.202

**Группа подготовки издания:**

| | |
|---|---|
| Главный редактор | *Екатерина Кондукова* |
| Зам. главного редактора | *Евгений Рыбаков* |
| Зав. редакцией | *Григорий Добин* |
| Редактор | *Елена Кошлакова* |
| Компьютерная верстка | *Татьяны Олоновой* |
| Корректор | *Татьяна Кошелева* |
| Дизайн серии | *Инны Тачиной* |
| Оформление обложки | *Елены Беляевой* |
| Зав. производством | *Николай Тверских* |

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 20.10.05.
Формат 70×100 1/16. Печать офсетная. Усл. печ. л. 23,22.
Тираж 3000 экз. Заказ №
"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию
№ 77.99.02.953.Д.006421.11.04 от 11.11.2004 г. выдано Федеральной службой
по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ОАО "Техническая книга"
190005, Санкт-Петербург, Измайловский пр., 29

ISBN 5-94157-687-0

© Старовойтов А. А., 2006
© Оформление, издательство "БХВ-Петербург", 2006

# Оглавление

**Введение** ..... **1**

О чем эта книга ..... 1
Для кого эта книга ..... 1
Какова структура книги ..... 2
Как связаться с автором ..... 2

**ЧАСТЬ I. КРАТКИЕ ОСНОВЫ ФУНКЦИОНИРОВАНИЯ СЕТЕЙ** ..... **3**

**Глава 1. Характеристика протокола TCP/IP** ..... **5**

1.1. Модель OSI ..... 5
1.2. Стек TCP/IP ..... 7

**Глава 2. Технологии локальных сетей** ..... **11**

2.1. Кабели, используемые в локальных сетях ..... 11
Коаксиальный кабель ..... 12
Кабель на основе неэкранированной витой пары ..... 12
Кабель на основе экранированной витой пары ..... 13
Волоконно-оптический кабель ..... 14
2.2. Стандарты сетей ..... 14
2.3. Технология Ethernet ..... 16
2.4. Технология Fast Ethernet ..... 21
2.5. Методика расчета сетей Fast Ethernet на повторителях ..... 23
2.6. Технология Gigabit Ethernet ..... 25

**ЧАСТЬ II. ПРОКЛАДКА КАБЕЛЬНОЙ СИСТЕМЫ** ..... **29**

**Глава 3. Аппаратура локальных сетей** ..... **31**

3.1. Сетевые карты и концентраторы ..... 31
Сетевые адаптеры ..... 31
Концентраторы ..... 32
3.2. Необходимость структуризации сети ..... 33
3.3. Принципы работы мостов (коммутаторов) ..... 35
Прозрачный мост ..... 36
Петли при использовании мостов ..... 37

3.4. Полнодуплексный и полудуплексный протоколы ... 38
Управление при полудуплексной работе ... 40
Управление при полнодуплексной работе ... 40
3.5. Проблема полосы пропускания ... 41
3.6. Технологии коммутации ... 43
Коммутация второго уровня ... 43
Коммутация третьего уровня ... 44
Коммутация четвертого уровня ... 44
3.7. Техническая реализация коммутаторов ... 45
Коммутаторы с разделяемой памятью ... 45
Коммутатор с общей шиной ... 46
Коммутатор на основе коммутационной матрицы ... 47
3.8. Основные характеристики коммутаторов ... 47
Основные характеристики ... 47
3.9. Дополнительные функции коммутаторов ... 49
Расширенная фильтрация трафика ... 50
Поддержка алгоритма Spanning Tree ... 50
Возможность создания виртуальных сетей ... 51
3.10. Методика оценки необходимой производительности коммутатора ... 52

**Глава 4. Проектирование кабельной системы ... 54**

4.1. Логическая структуризация сети и кабельная система ... 54
Необходимость логической структуризации сети ... 54
Структурированная кабельная система ... 55
4.2. Активное оборудование локальных сетей и его связь с СКС ... 57
Горизонтальная система ... 57
Система здания ... 57
Система городка ... 57
4.3. Выбор общей концепции сети ... 58
4.4. Сбор информации о будущей сети ... 59
4.5. Выбор типа кабеля ... 62
Коаксиальные кабели ... 62
Витая пара ... 62
Оптический кабель ... 63
Эксплуатационные характеристики витой пары ... 63
4.6. Расширение пропускной способности сети ... 63
Сеть 100Base-TX с одним сервером на основе концентраторов ... 64
Сеть с несколькими серверами по технологии 100Base-TX на основе коммутаторов ... 65
4.7. Проект сети ... 67
Общие вопросы ... 67
Проектирование топологии сети ... 67
Выбор сетевого оборудования ... 72
Подготовка проектной документации ... 73

**Глава 5. Прокладка сети ... 74**

5.1. Техника безопасности и правила монтажа ... 74
Техника безопасности ... 74
Правила монтажа ... 75
5.2. Используемый инструмент ... 76
5.3. Монтажные шкафы. Типы шкафов. Подготовка и установка монтажного шкафа ... 78
5.4. Кабельный канал, углы, Т-примыкания, заглушки ... 80
5.5. Подготовка и установка розеток ... 82
5.6. Прокладка кабеля ... 83
Последовательность прокладки кабеля через отверстия в стенах ... 84
Прокладка кабеля за потолочными панелями ... 86
5.7. Разделка розеток и патч-панели ... 86
Разделка розетки ... 87
Разделка патч-панели ... 88
5.8. Монтаж патч-кордов ... 90
5.9. Прокладка кабеля по воздуху ... 92

**Глава 6. Сборка сервера ... 94**

6.1. Система клиент-сервер ... 94
6.2. Материнская плата ... 94
6.3. Процессор ... 96
6.3. Оперативная память ... 97
6.4. Жесткие диски и RAID ... 98
IDE/ATA ... 99
SCSI ... 100
SATA ... 100
RAID ... 100
6.5. Подбор конфигурации сервера ... 101
6.6. Источники бесперебойного питания ... 102
6.7. Сборка сервера ... 103

**Глава 7. Тестирование сети и поиск неисправностей ... 108**

7.1. Причины плохой работы сети ... 108
7.2. Как правильно тестировать сеть ... 109
7.3. Проверка правильности разделки контактов ... 110
7.4. Проверка надежности соединения в контактах ... 111
7.5. Проверка методом исключения ... 112
Поочередное отключение узлов от сети ... 112
Прокачка через сетевую карту большого объема информации ... 113
7.6. Поиск неисправного порта коммутатора ... 114
7.7. Мониторинг сети ... 115

**ЧАСТЬ III. LINUX-СЕРВЕР СВОИМИ РУКАМИ ..... 117**

**Глава 8. Установка сервера ..... 119**

8.1. Выбор дистрибутива ..... 119
8.2. Установка сервера ..... 121
8.3. Первый вход в систему и настройка графического режима при помощи утилиты xvidtune ..... 138
Если все сложнее, чем вы думали ..... 141

**Глава 9. Основы администрирования Linux ..... 142**

9.1. Загрузка системы ..... 142
9.2. Пользователи, группы, учетные записи ..... 147
9.3. Управление учетными записями ..... 153
Добавление нового пользователя ..... 153
Модификация существующего пользователя ..... 158
Удаление пользователя ..... 160
9.3. Процессы. Управление процессами. Останов системы ..... 162
Общая характеристика процессов ..... 162
Получение информации о процессах ..... 163
Управление приоритетом процессов ..... 166
Уничтожение процессов ..... 167
Останов системы ..... 168
9.4. Файловая система. Структура каталогов. Работа с файлами ..... 169
Понятие файловой системы ..... 169
Монтирование и демонтирование файловой системы ..... 170
Соглашение об именовании устройств ..... 172
Структура каталогов файловой системы ..... 172
Просмотр информации о файле ..... 173
Типы файлов ..... 175
Права доступа ..... 176
Изменение прав доступа ..... 177
Изменение владельца и группы ..... 178
Установка режима создания файла ..... 178
9.5. Текстовый редактор vi ..... 179
9.6. Файловый менеджер Midnight Commander ..... 181
Настройка прав доступа ..... 182
Настройка владельца и группы ..... 184
9.7. Средства аудита ..... 184
9.8. Действия в случае аварии ..... 189
Составление плана действий на случай аварии ..... 189
Создание аварийной загрузочной дискеты ..... 190
Резервное копирование и восстановление в Linux ..... 191
Создание резервной копии ..... 193
Восстановление данных из копии ..... 193

**Глава 10. Настройка Samba ..... 194**

10.1. Общие сведения о Samba ..... 194
10.2. Одноранговая сеть. Основной файл конфигурации Samba /etc/smb.conf ..... 196
10.3. Samba в качестве PDC ..... 199
10.4. Настройка входа в домен для Windows 98 ..... 202
Настройка входа ..... 202
Устранение проблем в Windows 98 ..... 207
10.5. Настройка входа в домен для Windows NT/2000 ..... 207
Создание учетных записей вручную ..... 209
Настройка входа ..... 209
10.6. Первый запуск Swat ..... 216
10.7. Некоторые переменные Samba ..... 219
10.8. Автоматический запуск Samba ..... 219

**Глава 11. Службы DNS и DHCP ..... 221**

11.1. Краткая характеристика DNS ..... 221
11.2. Как работает служба DNS ..... 222
11.3. Пакет BIND и его компоненты ..... 223
Утилита nslookup ..... 224
Утилита dig ..... 224
Утилита host ..... 225
11.4. Как читать файл настроек BIND ..... 225
Файл named.conf ..... 225
Файлы баз данных зон ..... 227
Запись SOA ..... 227
Запись NS ..... 229
Запись A ..... 229
Запись PTR ..... 229
Запись MX ..... 230
Запись CNAME ..... 230
Запись SRV ..... 230
Запись TXT ..... 230
11.5. Простой пример собственного домена ..... 231
Проверка ..... 233
11.6. Краткая характеристика DHCP ..... 234
11.7. Файл настроек dhcpd и их параметры ..... 235
Глобальные параметры ..... 236
Опция subnet ..... 237
Опция shared network ..... 237
11.8. Установка DHCP-сервера. Связь DNS и DHCP ..... 238
11.9. Запуск служб DNS и DHCP ..... 240
11.10. Настройка клиентской части Windows 98 ..... 241
11.11. Настройка клиентской части Windows 2000 ..... 242

**Глава 12. Запуск Apache и Webmin ..... 244**

12.1. Почему Apache ..... 244
12.2. Основы конфигурирования Apache ..... 245
12.3. Базовые параметры, используемые при настройке Apache ..... 245
12.4. Коды ошибок, выдаваемых сервером ..... 247
12.5. Регистрация ошибок сервера ..... 247
12.6. Настройка автоматического запуска Apache ..... 249
12.7. Самый простой способ организовать Web-сервер в организации ..... 250
Настройка Apache ..... 251
Настройка клиентов ..... 252
Создание информационного наполнения сервера ..... 253
12.8. Краткие сведения о Webmin и запуск ..... 254
12.9. Настройка Webmin ..... 257

**Глава 13. Антивирусная защита ..... 261**

13.1. Установка drwebd и настройка скрипта запуска ..... 262
13.2. Проверка работоспособности drwebd ..... 263
13.3. Установка Samba Spider ..... 264
13.4. Настройка действия антивируса на Samba ..... 265

**Заключение ..... 267**

**ПРИЛОЖЕНИЯ ..... 269**

**Приложение 1. Полезные сочетания клавиш и некоторые команды ..... 271**

**Приложение 2. Аналоги Linux и Windows-программ ..... 274**

Постановка задачи ..... 275
Поиск аналогов используемых программ для Linux ..... 275
Установка программного обеспечения и обучение работе ..... 275

**Приложение 3. Источники информации о Linux ..... 276**

Русскоязычные источники ..... 276
Англоязычные ресурсы ..... 276

**Предметный указатель ..... 277**

# Введение

Побудительным мотивом к написанию этой книги послужил повышенный интерес к операционной системе Linux и компьютерным сетям, построенным под управлением этой операционной системы. В книге собран опыт автора по прокладке сетей и организации серверов на базе Linux. Уникальность подхода состоит в изложении в одной книге двух связанных между собой вопросов:

1. Прокладка сети и ее эксплуатация.
2. Установка, сопровождение и администрирование Linux-сервера.

Большое количество графического материала иллюстрирует повествование, а листинги конфигурационных файлов делают изложение более понятным.

## О чем эта книга

Книга рассматривает практические аспекты в области проектирования и прокладки локальных сетей, а также модернизации существующей сети. Кроме того, предметом книги является создание сервера небольшой организации на основе Linux.

Помимо описания необходимых теоретических основ построения сетей и установки сервера, приводятся рекомендации и примеры, позволяющие быстро перейти от теории к практике. Применение Linux изложено для ASPLinux 7.3 Server, однако подходит для любого из существующих дистрибутивов, в частности, для Red Hat и его клонов. В книге также рассмотрены вопросы повседневной эксплуатации сети и ремонта.

## Для кого эта книга

Книга может быть полезна как пользователям, не имеющим опыта работы с Linux и прокладки сетей, так и желающим расширить свои знания, а также системным администраторам начального уровня.

# Какова структура книги

Книга построена по принципу от простого — к сложному. Начиная с теоретических основ функционирования протоколов и сетей, читатель постепенно осваивает более сложные вопросы, такие как администрирование и настройка сетевых служб. Книга состоит из трех частей.

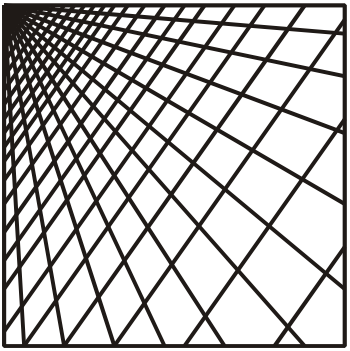
Первая часть описывает функционирование протокола TCP/IP, теоретические принципы построения локальных сетей. Приведены основные определения в области сетевых технологий, основные стандарты построения сетей и их различия.

Вторая часть посвящена прокладке кабельной системы, ее модернизации и тестированию. Рассматривается аппаратура локальных сетей и принципы ее работы. На практическом примере рассмотрено проектирование кабельной системы с использованием витой пары. Рассмотрены инструменты и практические приемы прокладки сети. В заключение второй части даются рекомендации по выбору конфигурации сервера, его сборке. Даны рекомендации по тестированию и поиску неисправности без использования специального оборудования.

Третья часть касается операционной системы Linux. Подробно рассмотрены вопросы инсталляции операционной системы ASPLinux 7.1 Server Edition в контексте решаемых задач, выбор пакетов и настройка графического режима. Основное внимание уделено наиболее частым вопросам начинающих администраторов. Рассмотрены также настройки Linux для интеграции в Windows-сети с использованием программного обеспечения Samba, серверы DNS и DHCP, запуск Apache и Webmin. Отдельное внимание уделено антивирусной защите при помощи программного обеспечения Dr.Web.

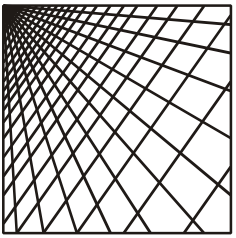
# Как связаться с автором

Официальный сайт автора книги **http://www.Linux-75.narod.ru**. Здесь вы можете получить доступ к новым наработкам по тематике, рассмотренной в книге. При желании можно принять участие в работе сайта.



# ЧАСТЬ I

# Краткие основы функционирования сетей



# Глава 1

# Характеристика протокола TCP/IP

## 1.1. Модель OSI

На основании опыта, полученного при создании сетей, в середине 1980 гг. прошлого века была разработана модель, которая сыграла важную роль в развитии сетевых технологий. Речь идет о модели OSI (Open System Interconnection — взаимодействие открытых систем). Модель OSI определяет стандартные уровни взаимодействия систем и функции каждого из уровней. Иными словами, модель OSI является многоуровневым набором протоколов. При развитии программных или аппаратных средств переписывается не весь протокол заново, а лишь необходимая часть.

Модели OSI разделяет средства взаимодействия на семь уровней: прикладной (его еще называют уровнем приложений, строго говоря, не совсем корректно), представления, сеансовый, транспортный, сетевой, канальный и физический (рис. 1.1).

Каждый уровень выполняет определенные функции и обеспечивает связь со смежным уровнем (представляет вышестоящему уровню определенный сервис и может запросить у нижестоящего уровня сервис для себя). Это позволяет обеспечить независимость высоких уровней от технических деталей более низких.

- Прикладной уровень позволяет пользователям или приложениям получать доступ к сетевым ресурсам: файлам, web-страницам, принтерам, электронной почте.
- Уровень представления обеспечивает интерпретацию данных и их подготовку для прикладного уровня (на принимающем компьютере), а также прием данных от прикладного уровня в формат сеансового уровня (на передающем компьютере). Этот уровень имеет дело с представлением данных, не меняя их сути. Этот уровень также осуществляет защиту данных в сети (шифрование, например SSL) и сжатие данных.

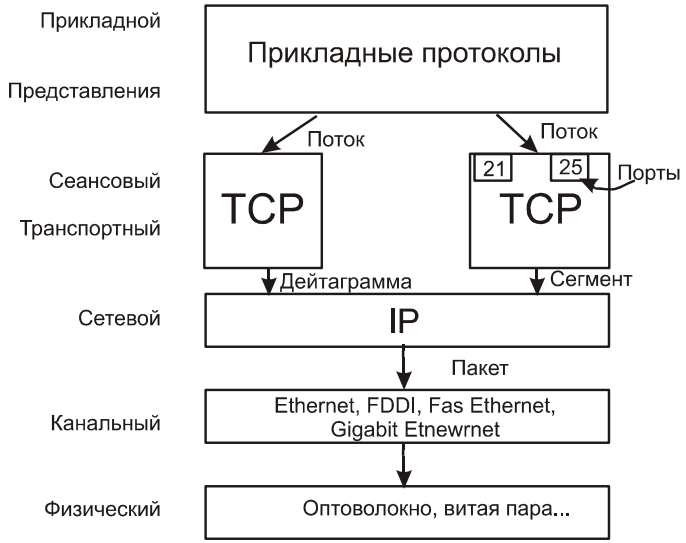



**Рис. 1.1.** Стек TCP/IP и модель OSI

- Сеансовый уровень устанавливает соединение между двумя компьютерами. Управляет этим соединением, восстанавливает и завершает его при необходимости. Этот уровень также отвечает за синхронизацию и содержит дополнительные функции управления.
- Транспортный уровень обеспечивает вышестоящим уровням передачу данных с необходимой степенью надежности. Он отвечает за распознавание и коррекцию ошибок. Также он определяет класс сервиса: срочность, надежность, возможность восстановления. Самый низкий класс сервиса 0, самый высокий 4.
- Сетевой уровень. Функции сетевого уровня достаточно разнообразны, фактически он образует единую систему, объединяющую сети с различными принципами передачи сообщений.

Прикладной, уровень представления, сеансовый, транспортный и сетевой уровни реализуются программно.

- Канальный уровень реализует связь между сетевым и физическим уровнем, выполняет функции проверки доступности среды, проверяет корректность передачи кадров. Функции канального уровня реализуются сетевыми адаптерами и их драйверами.
- Физический уровень определяет электрические, оптические, механические и другие параметры взаимодействия в сети. Он регламентирует тип физический среды передачи, метод кодирования, скорость передачи. Физический уровень реализуется сетевым адаптером и средой передачи (витая пара, оптоволокно).

# 1.2. Стек TCP/IP

Стек протоколов TCP/IP (Transmission Control Protocol/Internet Protocol — Протокол управления передачей/Межсетевой протокол) был разработан более двадцати лет назад по инициативе Министерства обороны США и внедрен в сети ARPANET и MILNET. Из объединения этих сетей и родился Интернет. Большую роль в популяризации TCP/IP сыграла его реализация в ОС UNIX. Сегодня протокол TCP/IP реализован во всех распространенных операционных системах: MS Windows, Novell, Linux.

Соответствие стека протоколов TCP/IP и модели OSI приведено на рис. 1.1. Набор протоколов TCP/IP захватывает транспортный сетевой и сеансовый уровни. Поверх TCP/IP работают протоколы более высокого уровня: Telnet, SNMP, FTP, HTTP.

Протокол TCP примерно соответствует сеансовому и транспортному уровням модели OSI, с его помощью реализуется сеанс связи между двумя компьютерами. Протокол TCP формирует из потока данных сегменты и контролирует прохождение сегмента по сети. К его функциям относится исправление ошибок и отслеживание прохождения пакетов по сети. При необходимости протокол организует повторную передачу потерянных или поврежденных сегментов. Таким способом достигается необходимая надежность, поскольку протокол IP не отвечает за доставку сообщений. При прохождении данных протокол TCP добавляет к ним свою служебную информацию.

Протокол IP (Межсетевой протокол) отвечает за маршрутизацию сегментов TCP. В его функции входит формирование IP-пакетов из данных на его входе, а также межсетевая адресация. По сути, протокол TCP использует IP-протокол в качестве транспортного средства. При прохождении IP пакета между сетями он упаковывается в соответствующие каждой конкретной сети средства транспортировки. Очень важно, что протокол IP рассматривает каждый IP-пакет как независимую единицу. Здесь нет средств контроля доставки пакетов. Если узел передал IP-пакет, то судьба этого пакета его уже не интересует. Все вопросы обеспечения надежности обеспечивает протокол TCP. Именно он организует повторную передачу, если какой-то из IP-пакетов потерялся. Протокол IP, как и протокол TCP, добавляет к данным служебную информацию своего уровня. Затем пакет продвигается по канальному и физическому уровню, где к нему добавляется соответствующая служебная информация.

Полезно будет вспомнить терминологию, применяемую в этой области.

*Поток* — данные, поступающие на вход протоколов TCP (и UDP). *Сегмент* — единица данных протокола TCP. *Дейтограмма* — единица протокола UDP. Единица данных протокола IP называется *пакетом*. *Порт* — уникальный идентификационный номер, который протокол TCP присваивает

приложению, использующему его в качестве транспорта. Все порты с номером меньше 1024 используют определенные приложения. Номера портов больше1024 могут использоваться пользователем произвольно. В табл. 1.1 приведены номера портов некоторых известных служб. Номер порта и IP-адрес образуют *гнездо* (socket).

***Таблица 1.1.*** *Номера некоторых стандартных портов*

| Номер | Служба | Примечание |
|---|---|---|
| 20 | FTP-DATA | Протокол передачи данных FTP, данные |
| 21 | FTP | Протокол передачи данных FTP, управление |
| 23 | TELNET | Telnet |
| 25 | SMTP | Простой протокол передачи сообщений (Передача почты) |
| 53 | DNS | Службы доменных имен |
| 69 | TFTP | Упрощенный протокол передачи данных |
| 70 | GOPHER | Gopher |
| 80 | HTTP | Протокол передачи гипертекста |
| 110 | POP3 | Post Office Protocol (Получение почты) |
| 156 | SQL | Служба SQL |

Чтобы различать узлы, входящие в сеть, существует специальная система адресов. Адреса бывают трех типов:

Локальный адрес — адрес, определяемый технологией данной сети. Если речь идет о локальной сети, то это MAC-адрес (адрес, присвоенный производителем сетевой карте или порту маршрутизатора). Этот адрес уникальный, он назначается сетевой карте производителем. Адрес содержит 6 байтов: 3 байта выделяется централизованно под номер производителя, 3 байта назначается самим производителем под уникальный номер изделия. Локальный адрес имеет следующий вид: A7-B8-C4-11-D5-1F.

IP — это адрес, на основании которого протокол IP осуществляет доставку пакетов. Это адрес представляет собой 4 байта. Поскольку оперировать с двоичными цифрами человеку сложнее, то записывается IP-адрес в десятичной системе с разделением байтов точками. Например, адрес компьютера в локальной сети может выглядеть так: 192.168.10.12, на самом деле это 11000000 10101000 00001010 00001100.

Символьное или доменное имя. Поскольку человеку достаточно сложно запоминать комбинации из четырех групп цифр, то в соответствие IP-адресам

были поставлены символьные имена. С таким типом адресов вы сталкиваетесь в Интернете. Подробнее об этом мы поговорим, когда речь пойдет о службе DNS.

Теперь поговорим об IP-адресах подробнее. Собственно IP-адрес состоит из двух частей: номера сети и номера узла в этой сети. IP назначается во время установки или настройки оборудования администратором сети. Для назначения IP-адресов существуют определенные правила. В сети Интернет выдачей IP-адресов заведуют специальные службы (NIC), и выбирать произвольный адрес нельзя. В локальных сетях все немного проще, в принципе вы можете присваивать любые адреса. Однако если локальная сеть подключена к Интернету (или в обозримом будущем подключение произойдет), то при назначении адресов узлов в локальной сети необходимо придерживаться определенных правил. Для того чтобы их лучше понять, рассмотрим классы IP-адресов.

Как вы знаете, одна часть IP-адреса — это номер сети, а другая — номер узла. То, какая часть относится к номеру сети, а какая к номеру узла, определяется по первым битам в IP-адресе. Существуют 5 классов IP сетей (рис. 1.2).

**Рис. 1.2.** Структура IP-адреса и классы IP сетей

Особыми являются класс D и класс E. Класс D начинается с последовательности 1110 и является групповым адресом — *multicast*. Сообщение, которое содержит адрес класса D, получат все узлы с этим адресом. Класс E начинается с последовательности 11110 и зарезервирован для будущего.

Помимо этих классов, существуют еще так называемые специальные IP-адреса:

- Адрес, состоящий из двоичных нулей, означает адрес узла сгенерировавшего пакет.
- Если в номере сети стоят нули, то считается, что узел назначения принадлежит к той же сети, что и узел, отправивший пакет.
- Пакет, состоящий из одних единиц, рассылается всем узлам в той сети, где находится источник сообщения.

- Если в поле номера узла стоят единицы, такой пакет рассылается всем узлам с указанным номером сети.
- Адрес, начинающийся с 127, предназначен для организации тестирования программ и взаимодействия процессов в рамках одной машины без передачи пакетов в сеть. Этот адрес имеет имя loopback (петля).

Помимо этих ограничений, для назначения IP-адресов в локальной сети существуют еще несколько правил. Для локальных сетей зарезервированы специальные диапазоны адресов:

- в классе A это диапазон 10.0.0.0 – 10.255.255.255;
- в классе B это диапазон 172.16.0.0 – 172.31.0.0;
- в классе C это диапазон 192.168.0.0 – 192.168.225.0.

Для разделения номера узла и номера сети может использоваться маска. Маска — это двоичное число, оно содержит единицы в тех разрядах, которые относятся к адресу сети. Для системы адресации, основанной на классах, маски такие: A — 255.0.0.0, B — 255.255.0.0, C — 255.255.255.0. Используя маски, можно добиться более гибкой системы адресации.

Теперь обратим внимание на то, что у нас есть два объекта: IP-адрес и MAC-адрес (локальный адрес). Эти два адреса не связаны между собой. Для того чтобы IP-пакет попал на нужный локальный узел, необходимо этот пакет перед отправкой в сеть снабдить MAC-адресом, соответствующим IP-адресу назначения. При отсутствии этого адреса у отправителя на помощь приходит ARP-протокол (Address Resolution Protocol — протокол разрешения адреса).

Узел формирует широковещательный ARP-запрос, и все узлы получают его. Затем они сравнивают IP-адрес со своим адресом, при совпадении формируется ответ, в который вкладывается искомый MAC-адрес.

Существует протокол, выполняющий обратную функцию: RARP (Reverse Address Resolution Protocol — протокол обратного разрешения адреса).

С начала 1990-гг. прошлого века началось бурное развитие Интернета, что в конечном итоге привело к дефициту IP-адресов. Одним из выходов из создавшегося положения является внедрение протокола IPv6, в котором для адресации используется уже не $8 \times 4 = 32$ бита, а 128 битов. Также увеличено и число уровней иерархии. Введена двухуровневая иерархия провайдеров и трехуровневая — для абонентов. Это резко повышает количество допустимых адресов и должно снять проблему нехватки свободных адресов.

Совместимость с текущей IPv4-версией обеспечивается ведением специального типа адресов — IPv4 compatible, где в старших 96 разрядах содержаться нули, а младшие 32 соответствуют адресу IPv4. Кроме этого, в протоколе IPv6 реализуются новые технологии защиты.

# Глава 2

# Технологии локальных сетей

## 2.1. Кабели, используемые в локальных сетях

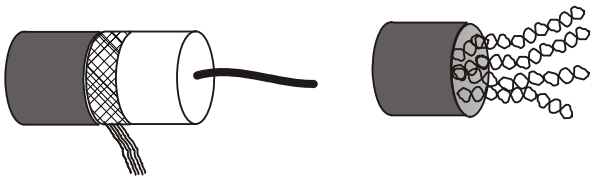
В локальных сетях небольших предприятий на сегодняшний день наиболее используемым стал кабель на основе неэкранированной витой пары UTP. Однако это далеко не единственный кабель, применяемый в современных сетях. Более того, построение сетей исторически началось на других типах кабеля. В некоторых сетях, оставшихся как наследие прошлого, используются коаксиальные кабели. Любой кабель имеет множество характеристик, определяющих возможности его использования. Наиболее важными характеристиками являются:

- активное сопротивление — сопротивление кабеля по постоянному току. Как правило, нормируется на определенную длину, поскольку с увеличением длины кабеля сопротивление растет;
- погонная емкость — это емкость между двумя проводниками или между проводником и оплеткой кабеля. Нормируется тоже на единицу длины кабеля. Погонная емкость приводит к тому, что высокочастотный сигнал затухает по мере продвижения его по кабелю. Погонная емкость является паразитным параметром, и ее стремятся снизить;
- волновое сопротивление (импеданс) представляет собой полное сопротивление кабеля. Как правило, значение сопротивления приводится для определенной частоты, поскольку с ростом частоты оно изменяется;
- затухание — параметр говорит сам за себя. Затухание измеряется в децибелах на единицу длины и частоту сигнала;
- перекрестные наводки. Суть явления в следующем. Если есть два параллельных проводника, и по одному из них идет высокочастотный сигнал, на втором кабеле образуется паразитное напряжение от первого. Величина перекрестных наводок измеряется в децибелах и нормируется на определенную частоту.

Рассмотрим основные типы кабелей, используемых в локальных сетях, и их характеристики.

## Коаксиальный кабель

Коаксиальный кабель используется в самых различных областях техники. Кабель представляет собой центральную медную жилу, которая окружена диэлектриком. Затем идет медная оплетка, которая защищает центральную жилу от внешнего излучения, а также предотвращает излучение во внешнюю среду. Наиболее известный среди обывателей коаксиальный кабель — это телевизионный кабель (волновое сопротивление 75 Ом). В компьютерных сетях применяются кабели RG-8 и RG-11 (так называемый "толстый" коаксиальный кабель), их волновое сопротивление 50 Ом, внешний диаметр 0,5 дюйма. Также используются кабели RG-58 ("тонкий" коаксиальный кабель) — волновое сопротивление 50 Ом, внешний диаметр 0,25 дюйма (рис. 2.1, *а*).



*а* *б*

**Рис. 2.1.** Коаксиальный кабель и витая пара

## Кабель на основе неэкранированной витой пары

UTP (Unshielded Twisted Pair) — неэкранированная витая пара (рис. 2.1, *б*). Представляет собой набор из 4-х пар кабеля, скрученных попарно. Затем эти пары скручены между собой. Витая пара делится на семь категорий (UTP1 – UTP7). Кабели первой и второй категорий сегодня применяются только для телефонных сетей. Кабели третьей, четвертой и пятой категорий широко используются для построений компьютерных сетей. Кабель третьей категории был стандартизирован в 1991 г. для работы на частотах до 16 МГц. Небольшим улучшением кабеля третьей категории послужил кабель четвертой категории, который допускал работу уже на частотах до 20 МГц. Кабель пятой категории разрабатывался специально для работы сетевых высокочастотных протоколов в диапазоне до 100 МГц. Электрические характеристики кабелей UTP3–UTP5 приведены в табл. 2.1.

***Таблица 2.1.*** *Характеристики кабеля UTP3, UTP4, UTP5*

| Параметр | UTP3 | UTP4 | UTP5 |
|---|---|---|---|
| Число пар | 4 | 4 | 4 |
| Волновое сопротивление | 100 ± 15 Ом | 100 ± 15 Ом | 100 ± 15 Ом |

***Таблица 2.1*** *(окончание)*

| Параметр | UTP3 | UTP4 | UTP5 |
|---|---|---|---|
| Затухание (dB на 100 м) | 4 МГц: 5.6<br>10 МГц: 9.8<br>16 МГц: 13.1 | 4 МГц: 4.3<br>10 МГц: 7.2<br>16 МГц: 8.9 | 16 МГц: 8.2<br>31 МГц: 11.7<br>100 МГц: 22 |
| Перекрестное затухание не менее (dB) | 4 МГц: 32<br>10 МГц: 26<br>16 МГц: 23 | 4 МГц: 47<br>10 МГц: 41<br>16 МГц: 38 | 16 МГц: 44<br>31 МГц: 39<br>100 МГц: 32 |

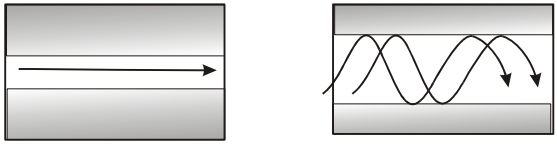
Кабели шестой и седьмой категорий работают на частотах до 200 и 600 МГц, соответственно. Их применение сдерживает относительно высокая цена.

## Кабель на основе экранированной витой пары

Shielded Twisted Pair (STP) — экранированная витая пара (рис. 2.2). Особенностью кабеля является наличие у него экрана. Это защищает его от внешних электромагнитных воздействий, а также снижает уровень излучения во внешнюю среду. Кабели на основе STP бывают девяти типов.

В локальных сетях применяется Type 1. Он представляет собой 2 пары скрученных проводов, экранированных оплеткой. По параметрам кабель Type 1 примерно соответствует кабелю UTP пятой категории. Однако его волновое сопротивление 150 Ом. Поэтому при применении его в качестве физической среды в локальных сетях необходимо обращать внимание на то, чтобы сетевое оборудование могло работать с волновым сопротивлением 150 Ом. В противном случае из-за несовпадения волнового сопротивления кабеля и активных сетевых устройств в местах их соединений будут образовываться отраженные волны. Результатом может стать непредсказуемое поведение сети и возникновение сбоев в ее работе.

Недостатком является необходимость хорошего заземления, а также более сложная по сравнению с UTP прокладка кабеля.



*а* *б*

**Рис. 2.2.** *а* — одномодовый и *б* — многомодовый оптический кабель

## Волоконно-оптический кабель

Волоконно-оптические кабели представляют собой кабель из материала, проводящего свет. (В качестве источников излучения используются либо светодиоды, либо лазеры.) Показатели оптического преломления различны у центральной части и у внешней оболочки. В результате луч света не выходит за пределы центральной части оптического волокна. Различают многомодовые (Multi Mode Fiber — MMF) и одномодовые (Single Mode Fiber — SMF) оптические кабели (рис. 2.2). В многомодовых кабелях используется диаметр внутреннего проводника около 50 мкм. В результате внутри кабеля существует несколько световых лучей. В одномодовых кабелях диаметр внутренней части составляет менее 10 мкм. Это сравнимо с длиной волны используемого светового излучения. В результате световой пучок распространяется вдоль оптического кабеля. Одномодовые кабели имеют более широкую полосу пропускания за счет отсутствия потерь при отражении, однако так как волокно очень тонкое, монтаж одномодового волокна — чрезвычайно трудоемкая процедура, и ей должны заниматься специалисты.

Несмотря на множество преимуществ, у оптического кабеля есть один важный недостаток — более сложный монтаж сети.

# 2.2. Стандарты сетей

Изначально при разработке технологии локальных сетей разработчики основное внимание уделяли созданию простых, надежных и дешевых способов соединения компьютеров в локальную сеть. Поэтому одним из первых было решение об использовании единой разделяемой во времени среды передачи. С физической точки зрения сеть, объединяющая несколько компьютеров, может иметь различную топологию. Наиболее интересны с точки зрения локальных сетей следующие варианты:

- Общая шина — компьютеры подключены к общему кабелю. Применение такой технологии удешевляет прокладку сети. Однако есть серьезный недостаток — низкая надежность. Любой дефект кабеля или разъема приводит к остановке всей сети.
- Звезда — в этом случае каждый компьютер подключается отдельным отрезком кабеля к устройству, находящемуся в центре сети. Это устройство называется концентратор. Фактически концентратор передает сигнал с одного из своих входов на все выходы, за исключением того, с которого пришел этот сигнал. Основное преимущество (по сравнению с общей шиной) — это более высокая надежность, ведь повреждение отдельного кабеля сказывается только на работе одного участка сети и не затрагивает все остальные.

- Кольцо — данные передаются по кольцу в одном направлении. Незначительным недостатком является то, что при применении кольцевой топологии приходится принимать меры по повышению надежности сети, чтобы выход из строя одного узла не повлиял на работу остальных узлов сети.

Схемы физической топологии приведены на рис. 2.3. Построение сети по принципу общей шины или кольца имеет более низкую надежность и более низкую пропускную способность. Это привело к тому, что в современных локальных сетях небольшого размера, как правило, применяется звездообразная топология. Кроме того, появление новых устройств: мостов, коммутаторов, маршрутизаторов — сняло проблему единой разделяемой среды данных.

Быстрые темпы развития сетевых технологий требовали необходимости принятия стандартов. В начале 1980-х гг. прошлого века был разработан и принят стандарт IEEE 802.x. Основой для его создания послужили существовавшие на тот момент внутрифирменные стандарты ведущих производителей компьютерной техники.

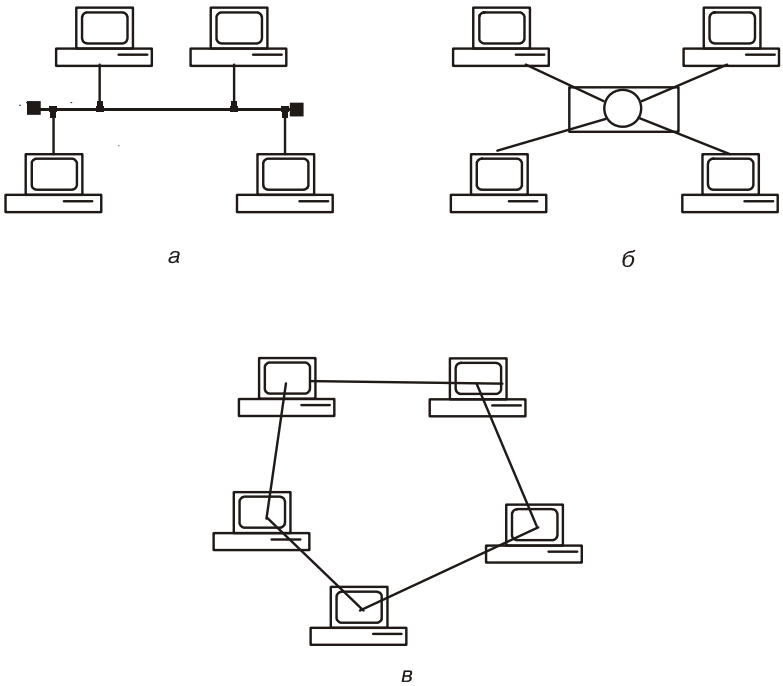



**Рис. 2.3.** Физические топологии сетей: *а* — шина, *б* — звезда, *в* — кольцо

Стандарты IEEE 802.x определяют технологии, касающиеся двух нижних уровней модели OSI. Состав стандартов 802.x приведен в табл. 2.2.

***Таблица 2.2.*** *Основные стандарты 802.x*

| Номер | Наименование |
|---|---|
| 802.1 | Internetworking, объединение сетей |
| 802.2 | Local Link Control — управление логической передачей данных |
| 802.3 | Сети Ethernet |
| 802.4 | Сети Token Bus LAN |
| 802.5 | Сети Token Ring LAN |
| 802.6 | Metropolitan Area Network — сети крупных городов |
| 802.7 | Board Technical Advisor Group — группа консультаций по широкополосной передаче |
| 802.8 | Fiber Optic Techical Advisory Group — группа консультаций по оптоволоконным сетям |
| 802.9 | Integrated Voice and data Networks — интегрированные сети передачи данных и голоса |
| 802.10 | Network Security — сетевая безопасность |
| 802.11 | Wereless LAN — беспроводные сети |

С практической точки зрения наибольший интерес для нас представляют сети Ethernet (802.3) и Fast Ethernet (802.3u). На сегодняшний день это самый распространенный стандарт построения локальных сетей. Причем практически все вновь создаваемые сети строятся по технологии Fast Ethernet. Технология Fast Ethernet отличается от технологии Ethernet только на физическом уровне. Кроме того, в некоторых организациях сети Ethernet еще действуют, поэтому вначале мы рассмотрим технологию Ethernet.

## 2.3. Технология Ethernet

В зависимости от типа физической среды различают следующие виды стандартов сетей Ethernet:

- 10Base-5;
- 10Base-2;
- 10Base-T;
- 10Base-F.

Однако несмотря на различия в типе физической среды все модификации сетей Ethernet, а также сети Fast Ethernet используют единый метод передачи данных — CSMA/CD (Carrier sense multi access / Collision detection — Метод коллективного доступа с опознаванием несущей и обнаружением коллизий).

Суть этого метода состоит в следующем. Для того чтобы получить доступ к единой разделяемой среде, станция должна убедиться, что среда свободна. Для этого станция постоянно прослушивает среду. Признаком свободной среды является отсутствие несущей частоты.

Если среда свободна, то станция может начать передачу. Все остальные станции обнаруживают факт передачи и прослушивают среду на предмет поиска данных для себя. Та станция, адрес которой совпадает с адресом в передаваемом кадре, распознает, что данные предназначаются ей, принимает их, продвигая данные по стеку протоколов вверх.

Если какой-то узел в этот момент хотел начать передачу, но обнаружил, что среда занята, он будет ожидать, пока среда не освободится.

После окончания передачи все узлы выдерживают паузу, она нужна для предотвращения монопольного захвата сетевой среды.

Поскольку сигнал распространяется в общей среде с конечной скоростью, то узлы фиксируют окончание и начало передачи не одновременно. И если две или более станций начнут свою передачу одновременно или почти одновременно (то есть узел начинает передачу в тот момент, когда до него еще не дошли сигналы от другого узла), происходит неизбежное искажение информации. Такая ситуация называется *коллизией* (collision).

Для распознавания коллизии все станции во время передачи сравнивают передаваемые ими данные и данные в общей среде. Различие этих данных является признаком возникновения коллизии.

При обнаружении коллизии все станции прекращают свою передачу, выжидают случайный интервал времени и при условии, что среда свободна, возобновляют свою передачу.

Для надежного распознавания коллизии необходимо, чтобы время передачи кадра минимальной длины было больше времени двойного оборота (Path Delay Value — PDV), так как в худшем случае сигнал должен пройти дважды между наиболее удаленными друг от друга узлами. Посмотрим на рис. 2.4. В момент времени $T$ станция 1 (она расположена на левом конце сети) начинает передачу. Сигнал начинает распространяться по общей среде и в момент времени $T + t$ достигнет правого конца сети. За незначительный момент времени до того, как сигнал дошел до станции 2, она начинает свою передачу, так как для нее общая среда все еще свободна, ведь сигнал от станции 1 до нее еще не дошел. В момент времени $T + t$ эти сигналы столкнутся, и возникнет коллизия. Сигнал коллизии начнет распространяться по сети от станции 2 к станции 1, и ему снова потребуется пройти

всю дистанцию, теперь уже в противоположном направлении. Таким образом, станция 1 получит сигнал коллизии только в момент времени

$T + t + t = T + 2t.$

Вот почему длина кадра должна быть больше времени двойного оборота.

С понятием коллизии также связано понятие домена коллизий (collision domain) — это часть сети, все узлы которой распознают коллизию, независимо от того, в какой части сети коллизия произошла. Домен коллизий представляет собой единую разделяемую среду. Концентратор образует единый домен коллизий. Коммутаторы и маршрутизаторы делят сеть на несколько доменов коллизий (рис. 2.4).

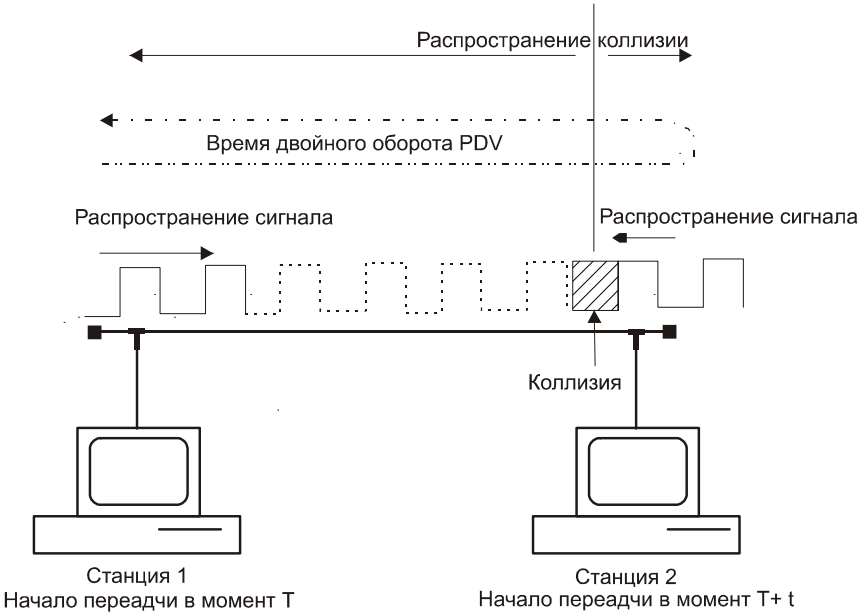



**Рис. 2.4.** Распространение коллизии и время двойного оборота

Стандарты 10Base — это не что иное, как спецификация физической среды передачи. Рассмотрим подробнее эти стандарты.

- 10Base-5 представляет собой коаксиальный кабель диаметром 0,5 дюйма с волновым сопротивлением 50 Ом. Максимальная длина сегмента сети без повторителя до 500 м. Подключение компьютера к кабелю осуществляется при помощи специального устройства — *трансивера* (transiver). Расстояние между трансиверами не менее 2,5 м. Поэтому для удобства на кабеле, как правило, наносятся метки. На концах сети обязательно должны стоять согласующие терминаторы, препятствующие отражению сигнала от концов кабеля. В противном случае в кабельной системе

возникают *стоячие волны*, и некоторые узлы получают сигнал слишком низкого уровня. В результате сеть оказывается неработоспособной. Стандартом допускается не более 4-х повторителей в сети и не более 3-х нагруженных сегментов.

- 10Base-2 представляет собой коаксиальный кабель диаметром 0,25 дюйма с волновым сопротивлением 50 Ом. Станции подключаются к кабелю при помощи специального Т-образного коннектора. Расстояние между станциями не менее 1 м. Поэтому для удобства есть разметка. На концах кабеля также устанавливаются терминаторы. Стандартом допускается не более 4-х повторителей в сети и не более 3-х нагруженных сегментов. Максимальная длина сегмента сети без повторителя до 185 м. Стандарт 10Base-2 очень напоминает 10Base-5.
- 10Base-T представляет собой UTP-кабель (Unshielded Twisted Pair — неэкранированная витая пара). Используется кабель третьей категории. В данном случае категория определяет полосу пропускания кабеля и некоторые другие характеристики. Используется две пары кабеля: одна для передачи от станции к повторителю, вторая для передачи от повторителя к станции. Максимальная длина сегмента сети без повторителя до 100 м. Максимальная скорость передачи данных 10 Мбит/с. Сети на основе UTP-кабеля строятся по звездообразной топологии. В центре звезды находится *концентратор* (hub). Концентратор повторяет сигнал, пришедший на его вход, на все его выходы. Таким образом, реализуется единая разделяемая среда. В случае возникновения коллизия передается на все входы путем посылки специальной jam-последовательности. При соединении концентраторов друг с другом их число между любыми двумя станциями не должно превышать 4-х (правило четырех хабов). При необходимости соединения нескольких хабов логически целесообразно организовать древовидную структуру и использовать более жесткое правило 3-х хабов.
- 10Base-F — физическая среда построена на основе волоконно-оптического кабеля. Оптическое волокно может быть многомодовое с полосой пропускания 800 МГц или многомодовое с полосой пропускания 2—3 ГГц. Оптический кабель на основе многомодового волокна более дешевый и более простой в монтаже. Построение сети аналогично построению сети 10Base-T. Имеет два стандарта спецификации: 10Base-FL, 10Base-FB. Отличаются стандарты мощностью передатчиков, а как следствие, и максимальным расстоянием между элементами сети (до нескольких километров). Сети строятся по звездообразной топологии, в случае соединения нескольких концентраторов топология должна быть древовидной.

При расчете сетей, построенных по технологии Ethernet, важно соблюдать множество ограничений для сетей этого класса (рис. 2.5). Основными

правилами является правило 4-х хабов (как правило, оно вырождается в правило 3-х хабов) и ограничение длины одного сегмента (табл. 2.3). Есть еще условия, которые при построении небольшой сети вам вряд ли придется соблюдать: это ограничение максимального количества узлов (< 1024), а также соблюдение ограничения для времени двойного оборота. Последние ограничения сказываются только в очень больших сетях или сетях, построенных на основе нескольких подсетей различного физического уровня. Методики расчета мы приведем, когда будем рассматривать сети Fast Ethernet. Мы приведем упрощенные методики, если вы захотите более глубоко разобраться в методике расчета сетей, то обращайтесь к соответствующим источникам.

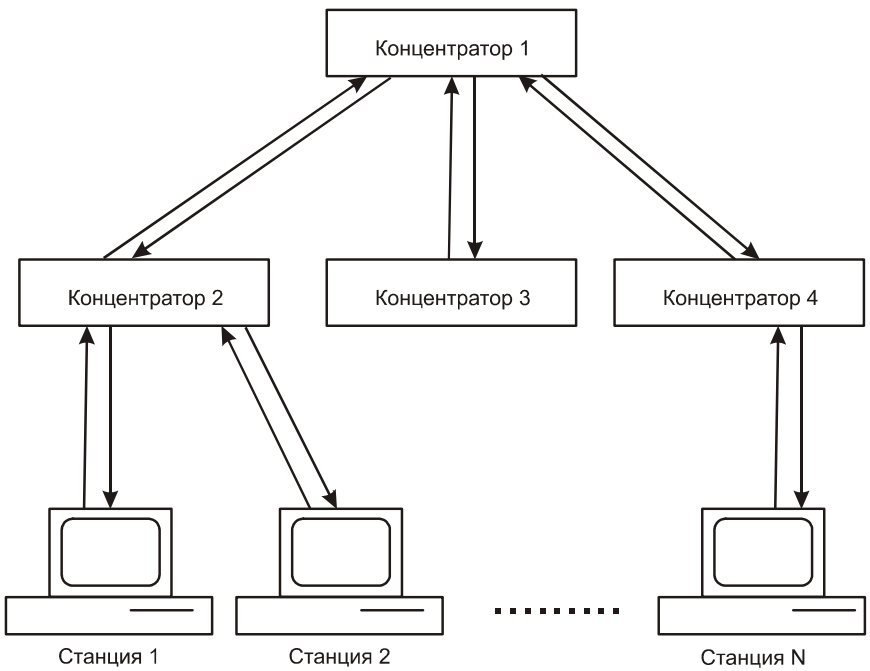



**Рис. 2.5.** Соединение нескольких хабов в сетях 10Base-T

***Таблица 2.3.*** *Сводная характеристика сетей Ethernet*

| | **10Base-5** | **10Base-2** | **10Base-T** | **10Base-F** |
|---|---|---|---|---|
| Тип кабеля | Толстый коаксиальный кабель RG-8/11 | Тонкий коаксиальный кабель RG-58/U или RG-58A/U | UTP3, UTP4, UTP5 | MMF, SMF |

***Таблица 2.3*** *(окончание)*

| | 10Base-5 | 10Base-2 | 10Base-T | 10Base-F |
|---|---|---|---|---|
| Топология | шина | шина | звезда | звезда |
| Максимальное число узлов на сегменте | 100 | 30 | 1024 | 1024 |
| Максимальное количество сегментов | 5 | 5 | 5 (последоват.) | – |
| Максимальная длина сегмента | 500 м | 185 м | 100 м | SMF 5 км<br>MMF 1 км |
| Максимальная длина сети | 2500 м (300 узлов) | 925 м | 500 м | – |
| Минимальное расстояние между точками включения | 2.5 м | 0.5 м | – | – |
| Характер подключения | трансивер | BNC-T-коннектор | RJ-45 | SN-коннектор |

# 2.4. Технология Fast Ethernet

С развитием информационных технологий пропускной способности сети, обеспечиваемой технологией Ethernet, стало не хватать. Сеть стала не успевать передавать данные для компьютеров, скорость работы которых существенно возросла. Сетевая среда стала узким местом, отрицательно сказываясь на производительности системы в целом. В результате научных разработок был создан стандарт Fast Ethernet, расширение Ethernet с пропускной способностью до 100 Мбит/с. Этот стандарт получил название 802.3u. Стандарт Fast Ethernet содержит три типа физической среды: 100Base-TX, 100Base-T4, 100Base-FX. Рассмотрим их подробнее.

- 100Base-TX использует 2 пары кабеля UTP или STP (Shielded Twisted Pair — экранированная витая пара). Максимальная длина сегмента сети 100 м. Стандарт рассчитан на применение сетевой топологии типа звезда. Центром сети является концентратор. Соединение кабеля с портом концентратора или сетевой картой осуществляется при помощи разъема RJ-45.
- 100Base-T4 использует 4 пары кабеля UTP третьей категории. По трем парам идет обмен данными, одна задействована для распознавания коллизий. Максимальная длина сегмента сети 100 м. Этот стандарт был

разработан специально для организации сетей со скоростью 100 Мбит/с при кабеле UTP третьей категории. Стандарт рассчитан на применение топологии звезда. Соединение кабеля с порта концентратора или сетевой картой осуществляется при помощи разъема RJ-45.

- 100Base-FX стандарт определяет построение физической среды на основе многомодового оптического кабеля в полудуплексном (half duplex, одновременно работа возможна только в одном направлении: на передачу или на прием) и полнодуплексном режимах (full duplex, возможна работа сразу в двух направлениях: и на передачу, и на прием). Длина сегмента в полнодуплексном режиме до 2 км, в полудуплексном до 412 м. Используется топология звезды.

Для устройств 100Base-TX/T4 определен следующий протокол договора о скорости и режимах работы:

- 10Base-T использует 2 пары кабеля UTP третьей категории;
- 10Base-T full duplex использует 2 пары кабеля UTP третьей категории;
- 100Base-TX использует 2 пары кабеля UTP пятой категории;
- 100Base-T4 использует 4 пары кабеля UTP третьей категории;
- 100Base-TX full duplex использует 2 пары кабеля UTP пятой или STP Type 1 категории.

Скорость определяется автоматически (auto-negotuiation). Процесс переговоров начинается при включении устройства в сеть. Устройству предлагается работать в самом "верхнем" режиме, если оно не поддерживает, то в ответе указывает тот режим, в котором может работать.

В связи с увеличением задержки повторителями, правило 4-х хабов (табл. 2.4), вырождается в правило 2-х хабов или даже 1-го хаба. Точнее, в одном домене коллизий не может быть более 1-го хаба (I класса) и не более 2-х хабов (II класса). Повторители I и II классов отличаются задержкой в распространении сигнала:

- для класса I задержка составляет 70 битовых интервалов;
- для класса II задержка составляет 46 битовых интервалов.

Если нарушить установленные требования, то суммарные задержки сигнала в кабеле и хабах превысят время двойного оборота, и сеть не сможет работать.

***Таблица 2.4.*** *Сводная характеристика сетей Fast Ethernet*

| | **100Base-TX** | **100Base-T4** | **100Base-FX** |
|---|---|---|---|
| Тип кабеля | UTP5, STP тип 1 | UTP3, UTP4, UTP5 | MMF |
| Топология | звезда | звезда | звезда |

***Таблица 2.4*** *(окончание)*

| | 100Base-TX | 100Base-T4 | 100Base-FX |
|---|---|---|---|
| Максимальное число узлов на сегменте | 1024 | 1024 | 1024 |
| Максимальное количество сегментов | 3 | 3 | |
| Максимальная длина кабеля между концентраторами | 5 м | 5 м | |
| Максимальная длина сегмента | 100 м | 100 м | До 2 км |
| Максимальная длина сети | 205 м | 205 м | |
| Способ подсоединения узла | RJ-45 | RJ-45 | |
| Кол-во используемых пар кабеля | 2 | 4 | |

## 2.5. Методика расчета сетей Fast Ethernet на повторителях

Методика сводится к определению удвоенной задержки в распространении сигнала и сравнению суммарной задержки с минимальной длиной кадра в 512 интервалов. Необходимо также взять запас в несколько интервалов. Для удобства расчета используют табличные значения (табл. 2.5).

Проведем расчет для сети на рис. 2.6. Вообще, расчет можно проводить только для самого плохого случая, но он не всегда явно виден, особенно если значения близки к критической величине. Поэтому расчет лучше проводить для всех возможных ветвей. В нашем случае конфигурация сети взята в учебных целях. Покажем пример расчета.

- Станция 1 — Станция 2:

$$50\ bt + 100 \times 1\ bt + 140\ bt + 100 \times 1{,}112\ bt + 50\ bt = 451{,}2\ bt;$$

- Станция 1 — Станция 3:

$$50\ bt + 100 \times 1\ bt + 140\ bt + 95 \times 1{,}112\ bt + 50\ bt = 445{,}64\ bt;$$

- Станция 1 — Станция 4:

$$50\ bt + 100 \times 1\ bt + 140\ bt + 63 \times 1\ bt + 50\ bt = 403\ bt;$$

- Станция 1 — Станция 5:

$$50\ bt + 100 \times 1\ bt + 140\ bt + 250 \times 1\ bt + 50\ bt = 590\ bt.$$

То есть в данном случае есть явное превышение величины в 512 битовых интервалов, а значит, такая сеть не будет работоспособной. Причина — слишком большое расстояние от Станции 5 до повторителя.

При большом числе станций расчеты могут усложняться, поэтому для автоматизации расчетов лучше применять математические методы с использованием программы MCAD (или аналогичной).

В связи с развитием технологии и удешевлением техники, сети стали строиться на основе коммутаторов, что снимает ограничение на время двойного оборота (табл. 2.5). Остается ограничение только на максимальную длину сегмента сети (рис. 2.6).

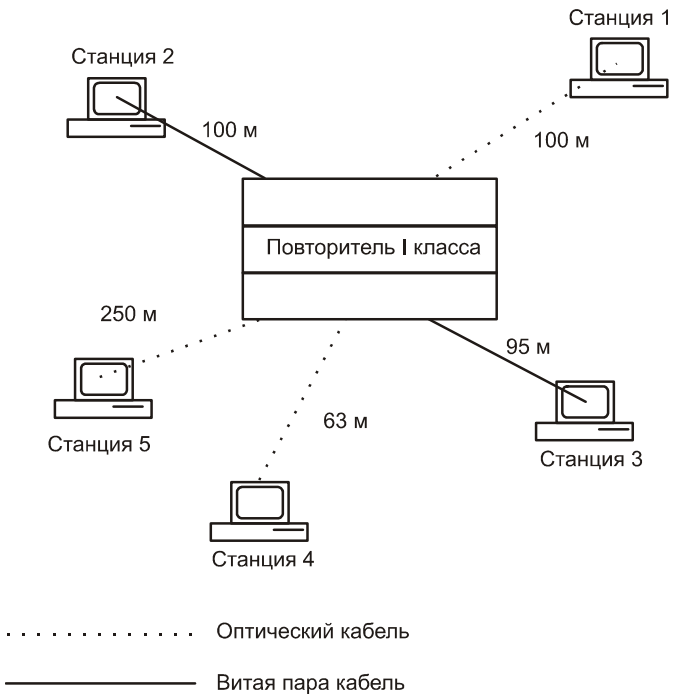



**Рис. 2.6.** Методика расчета сетей Fast Ethernet

***Таблица 2.5.*** *Удвоеные величины задержек, вносимые сетевыми устройствами*

| **Устройство** | **Удвоенная задержка** |
|---|---|
| Кабель UTP 3 | 1,14 bt на 1 м |
| Кабель UTP 4 | 1,14 bt на 1 м |

***Таблица 2.5*** *(окончание)*

| Устройство | Удвоенная задержка |
|---|---|
| Кабель UTP 5 | 1,112 bt на 1 м |
| STP | 1,112 bt на 1 м |
| Оптический кабель | 1,0 bt на 1 м |
| Адаптер TX/FX | 50 bt |
| Адаптера T4 | 69 bt |
| Повторитель класса I | 140 bt |
| Повторитель класса II | 92 bt |

# 2.6. Технология Gigabit Ethernet

Стандарт Fast Ethernet дал некоторую передышку, расширив производительность сети до приемлемого уровня. Однако через несколько лет после принятия стандарта Fast Ethernet стало понятно, что назрела необходимость в разработке более производительного протокола. В 1996 г. Комитет IEEE принял проект стандарта Gigabit Ethernet (802.3 z), который должен был обеспечить пропускную способность 1000 Мбит/с. После принятия стандарта ведущие компании-производители (среди них 3Com, Cisco, Compaq, Intel, Sun) образовали Gigabit Ethernet Alliance. Цель альянса выработка стандарта для производителей оборудования. К 1998 г. к альянсу уже примкнуло более 100 компаний. И в 1998 г. был принят стандарт 802.3z, регламентирующий применение оптоволокна, а также категории UTP 5 на расстояния до 25 м. Стандарт, регламентирующий работу на неэкранированной витой паре на расстоянии до 100 м, был принят через год, поскольку потребовалась разработка специального кода для обеспечения необходимой надежности и помехоустойчивости.

Технология Gigabit Ethernet во многом явилась приемником технологии Fast Ethernet. Сохранились форматы кадров, поддерживаются основные типы кабелей, однако есть и много отличий, изменились методы кодировки и модуляции. Стандарт Gigabit Ethernet определяет возможность работы по следующим типам физической среды (табл. 2.6):

- многомодовое оптоволокно;
- одномодовое оптоволокно;
- экранированная витая пара;
- неэкранированная витая пара пятой категории.

***Таблица 2.6.*** *Сводная характеристика сетей Gigabit Ethernet*

| | **1000Base-LX** | **1000Base-SX** | **1000Base-CX** | **1000Base-T** |
|---|---|---|---|---|
| Тип кабеля | MMF, SMF | MMF | STP | UTP-5 |
| Топология | звезда | звезда | звезда | звезда |
| Максимальное число узлов на сегменте | 2 | 2 | 2 | 2 |
| Максимальная длина сегмента | SMF — 3км, MMF — 550 м | До 550 м | 25 м | 100 м |
| Тип лазера | 1300 нм | 850 нм | – | – |
| Спецификация | | 802.3 z | | 802.3 ab |

Наиболее интересен последний стандарт, поскольку он позволяет использовать уже проложенные кабельные системы на основе кабеля UTP 5. Вместе с тем "втиснуть" пропускную способность 1000 Мбит/с в кабель, который допускает передачу только 100 МБит/с, достаточно сложно, так как электрические параметры кабеля (волновое сопротивление и погонная емкость) приводят к сильному затуханию сигнала с увеличением частоты передачи выше допустимой по номиналу. Кроме того, влияние ближних и дальних перекрестных помех между четырьмя парами кабеля потребовало разработки специальной скремблированной помехоустойчивой передачи и интеллектуального узла распознавания и восстановления сигнала на приеме.

Для достижения скорости 1000 Мбит/с без изменения частоты сигнал передается сразу по 4 витым парам (кабель UTP 5 категории содержит именно 4 витых пары). Значит, по одной паре передается 1000 / 4 = 250 (Мбит/с). Кроме того, для дальнейшего снижения частоты, используется пятиуровневое кодирование сигнала (−2, −1, 0, 1, 2). Причина того, что при использовании пятиуровневого кодирования снижается частота сигнала в кабеле, заключается в следующем. При распространенном четырехуровневом кодировании существует 4 различных комбинации: 00, 01, 10, 11. Это значит, что за один такт передаются 2 бита информации. Получается, что при передаче двух битов информации передатчик меняет сигнал на своем выходе всего один раз, следовательно, уменьшается в два раза частота модуляции сигнала. И теперь частота будет 250 / 2 = 125 МГц, вместо 250 МГц (рис. 2.7). Пятый уровень добавляет в код избыточность, благодаря чему возможна коррекция ошибок. Это дает дополнительный резерв в соотношении сигнал/шум.

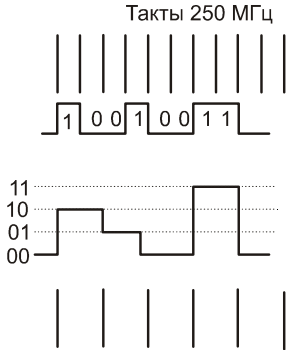



**Рис. 2.7.** Применение четырехуровневого кодирования

Поскольку для передачи используется сразу 4 витые пары, есть сложности с организацией полнодуплексного режима работы. Для отделения своего сигнала приемник вычитает из сигнала, взятого с линии, свой сигнал. Это значит, что в полнодуплексном режиме работы возникновение коллизии (одновременной работы двух узлов) является нормой. На рис. 2.8 изображена работа сети в полнодуплексном режиме.

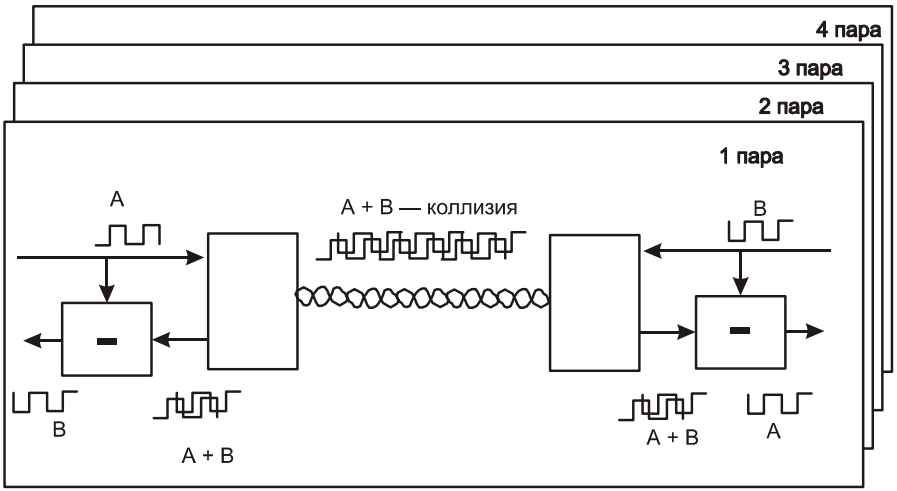
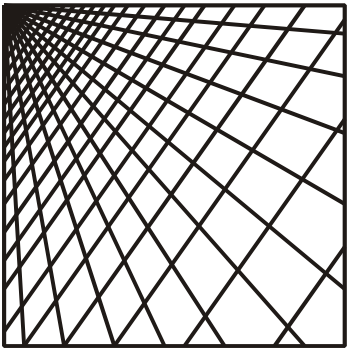



**Рис. 2.8.** Работа сети 1000Base-T

На данный момент технологии Gigabit Ethernet предсказывают большое будущее. На сегодняшний день практически все ведущие производители освоили выпуск сетевого оборудования данного стандарта. Рост рынка Gigabit Ethernet и других высокоскоростных технологий сулит радужные перспективы не только по обилию оборудования, но и по соотношению цена/производительность.



# ЧАСТЬ II

# Прокладка кабельной системы

# Глава 3

# Аппаратура локальных сетей

## 3.1. Сетевые карты и концентраторы

Для построения локальной сети нужны кабельная система, концентраторы и сетевые карты. Поскольку со стандартами кабельных систем мы уже знакомы, рассмотрим подробнее, что из себя представляют сетевые карты и концентраторы и какие они выполняют функции в сетевой системе.

### Сетевые адаптеры

Сетевой адаптер физически представляет собой устройство, вставляемое в один из слотов расширения материнской платы компьютера. Вместе со своим драйвером сетевой адаптер предоставляет операционной системе услугу доступа к физической среде сети. Если обратиться к модели OSI, то сетевой адаптер и его драйвер занимают физический и канальный уровни. Последовательность работы сетевого адаптера для сетей Ehternet на основе протокола TCP/IP заключается в следующем:

- прием пакета IP от сетевого уровня;
- формирование кадра Ethernet: пакет IP инкапсулируется в кадр Ethernet, в кадре заполняется MAC-адрес назначения;
- формирование кодированного сигнала в соответствии с принятым методом кодирования;
- выдача сигналов в кабель.

При приеме происходят следующие действия:

- прием сигнала и его усиление;
- декодирование сигнала;

- проверка правильности передачи кадра по контрольной сумме (некорректно переданный кадр отбрасывается);
- извлечение IP-пакета и его передача наверх по стеку протоколов.

Здесь стоит немного пояснить следующий момент: если Ethernet кадр пришел искаженным и его контрольная сумма не совпадает, то он просто отбрасывается. Данные при этом теряются. За повторную передачу потерянных данных отвечает протокол TCP. Собирая пришедшие к нему TCP сегменты, он обнаружит пропажу одного из сегментов. Номера сегментов должны идти по порядку, и нарушение порядка говорит о том, какой из сегментов пропал. Обнаружив нехватку сегмента, протокол TCP запросит его повторную передачу, таким образом целостность данных будет сохранена.

Сетевые адаптеры сегодня представляют собой сложные устройства. Они, как правило, реализуются на специализированных микросхемах большой степени интеграции. В наиболее дорогих адаптерах применяются собственные процессоры, обрабатывающие поток данных. За счет этого снимается нагрузка с центрального процессора компьютера.

## Концентраторы

Сети Ethernet используют метод доступа CSMA/CD — carrier sense multiply with collision detection (Метод коллективного доступа с опознаванием несущей и обнаружением коллизии). При использовании этого метода все компьютеры делят между собой во времени общую среду — шину. Для создания общей среды служат *концентраторы* (concentrator) или *хабы* (hub). В функции концентратора входит повторение сигнала пришедшего на один из его входов, на все выходы за исключением того, куда пришел сигнал. В результате этого образуется единая, разделяемая во времени среда.

Наиболее ярко функции концентратора проявляются в сетях построенных на витой паре по топологии звезды. Здесь концентратор находится в центре и соединен с каждой станцией отдельным кабелем.

Разъем для подключения компьютера называется портом. Типовые концентраторы имеют, как правило, до 48 портов. Между собой концентраторы должны соединяться на основе кросс-провода либо через специальный кроссированный разъем, который есть на большинстве концентраторов. Дело в том, что сетевой адаптер и концентратор имеют зеркальную разводку контактов, чтобы их можно было соединить прямым проводом. Схема кросс-провода приведена на рис. 3.1. Кросс-провод подобной конструкции можно применять и для непосредственного соединения двух компьютеров, без использования концентратора.

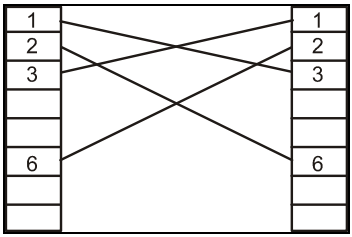



**Рис. 3.1.** Кросс-провод для соединения концентраторов между собой

Помимо основной функции образования общей временной среды, концентраторы могут выполнять много дополнительных функций. Перечислим наиболее полезные из них:

- Поддержка управления по протоколу SNMP (simple network management protocol — простой протокол управления сетью). Последнее очень удобно, так как управлять концентратором стало возможно через сеть, без непосредственного подключения к концентратору. Протокол SNMP относится к стеку протоколов TCP/IP. Для того чтобы в сетях Ethernet была возможность управлять концентратором, необходимо, чтобы у концентратора был IP и MAC-адрес.
- Возможность отключать отдельные порты. Концентратор может сам распознавать некорректно работающие порты и отключать их. Отключение портов может происходить и программно. Для этого необходимо или при помощи SNMP-протокола, или непосредственно подключиться к концентратору и дать команду на отключение конкретного порта.
- Поддержка резервных связей. Для большей надежности сетевой среды некоторые модели концентраторов могут поддерживать резервные связи. В обычном состоянии порты, на которые заведены резервные связи, отключены, поскольку топология сети не допускает образования петель. Однако в случае выхода из строя основной связи включается резервный порт.

## 3.2. Необходимость структуризации сети

Использование общей разделяемой среды при относительно большом числе подключенных узлов приводит к тому, что снижается производительность сети. Причина в том, что в Ethernet-сетях доступ к разделяемой во времени среде носит случайный характер. В каждый отдельно взятый момент на передачу может работать только один узел. В случае если два узла начали передачу одновременно, возникает коллизия, узлы прекращают свою передачу. После этого они выжидают случайный интервал времени и пытаются возобновить передачу. С ростом количества узлов, подключенных к среде, растет и количество передаваемой информации. В идеальном случае оно

равно пропускной способности сети: 10 Мбит/с для Ethernet и 100 Мбит/с для Fast Ethernet. Однако реальная характеристика далека от идеальной.

Для характеристики загрузки сети вводится понятие коэффициента нагрузки сети. Он равен отношению трафика в сети к ее максимальной пропускной способности. Если обозначить его как $p$, то зависимость реальной пропускной способности от коэффициента нагрузки сети будет далека от идеальной (рис. 3.2). Так при $p = 1$ в сети будут в основном одни коллизии. Критической величиной является величина 0,5, когда нагрузка сети составляет половину пропускной способности. При дальнейшем росте коэффициента нагрузки сети начинают расти коллизии, а следовательно, и время задержки в доступе к сети. Значит, с ростом числа узлов будут расти коллизии и время простоя сети. Это приводит к тому, что большая сеть будет работать неэффективно. Количество узлов стараются ограничивать на уровне 20—30. Однако на практике могут потребоваться и более строгие ограничения.

Выходом из создавшейся ситуации может послужить разделение одной разделяемой среды на несколько самостоятельных сред передачи данных. Есть несколько видов устройств, выполняющих такие функции: мосты коммутаторы и маршрутизаторы.

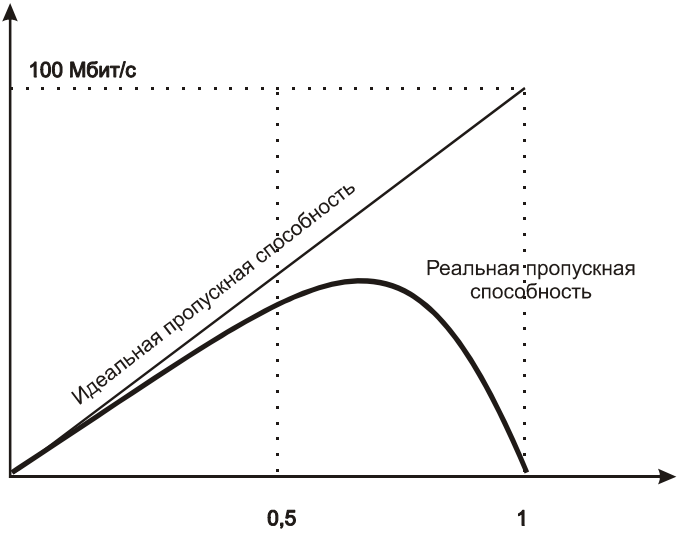



**Рис. 3.2.** Реальная и идеальная пропускная способность

Мост (Bridge), коммутатор (Switch) или маршрутизатор (Router) не стараются поддержать единую разделяемую среду для всей сети. Вместо этого каждая из сетей выступает как конечный узел, который принимает кадр, анализирует адрес, содержащийся в нем, и передает его на один из своих портов. То есть для передачи кадра в сеть нужен доступ к разделяемой среде. В результате такой работы трафик одного сегмента сети изолируется от других сегментов.

В каждом из сегментов, на которые разбита сеть, теперь меньше узлов, соответственно, коэффициент нагрузки сети уменьшается, а значит, повышается эффективность использования общей разделяемой среды.

Мосты, коммутаторы и маршрутизаторы передают кадры с одного порта на другой на основе адресов, помещенных в кадрах. Мосты и коммутаторы анализируют MAC-адреса и соответственно работают на канальном уровне модели OSI. Маршрутизаторы работают с IP-адресами и адресами сетей и работают на сетевом уровне модели OSI.

По своей сути, мост — это частный случай коммутатора: коммутатор с двумя портами. Просто исторически мосты появились первыми и служили для разделения одной общей временной среды на две. Однако логика работы моста и коммутатора одинаковая. Поэтому все сказанное для мостов справедливо для коммутаторов.

Еще одной возможностью коммутаторов является полнодуплексный режим работы. Такой режим возможен только в случае подключения на порт коммутатора одного компьютера. Полнодуплексный режим характерен тем, что данные на прием и передачу передаются по разным парам проводников. Это приводит к отсутствию коллизий. (В сетях GigabitEthernet все немного не так).

Разделение на несколько подсетей, таким образом, дает следующие преимущества:

- снижение задержки доступа к общей разделяемой среде для каждого из сегментов, а следовательно, прохождения IP-пакета от источника до пункта назначения;
- сокращение коллизий в каждом сегменте сети;
- снятие ограничения на рост сети без потери производительности (так что применение коммутаторов делает сеть более масштабируемой);
- снятие ограничения на время двойного оборота (остается только ограничение на максимальный размер сегмента).

Введение технологии коммутации подняло сети на новый уровень. На сегодняшний день наиболее популярным принципом построения сети является микросегментация, которая существенно повышает производительность сети и возможности ее перестроения (*подробнее о микросегментации см. разд. 3.4*).

## 3.3. Принципы работы мостов (коммутаторов)

Мост (коммутатор) обеспечивает связь двух или нескольких сегментов сети. Для каждой из сетей мост (коммутатор) выступает как конечный узел, то есть для передачи кадра в сеть он должен получить доступ к разделяемой среде.

По принципу своего действия мосты (коммутаторы) бывают двух типов: мост с маршрутизацией от источника и прозрачный мост.

Мост с маршрутизацией от источника (Source Routing) был разработан IBM. Такая технология применяется в сетях FDDI и Token Ring. Этот метод маршрутизации требует, чтобы узел-отправитель размещал в пакете информацию о его маршруте.

Мосты второго типа — прозрачные мосты (Transfer Bridge). Этот алгоритм работы лег в основу большинства современных мостов (коммутаторов). Рассмотрим алгоритм его работы подробнее.

## Прозрачный мост

Алгоритм работы прозрачного моста получил свое название по той причине, что конечные узлы в сети ведут себя точно так же, как и при отсутствии моста.

Сеть для прозрачных мостов видится набором MAC-адресов, подключенных к конкретным портам. Построение адресной таблицы идет по следующей схеме:

1. Принимая кадры на свой порт, мост анализирует адрес источника.
   - Если в адресной таблице данного MAC-адреса нет, мост записывает в нее новую строку.
   - Если строка для данного MAC-адреса уже существует, то мост проверяет ее на достоверность:
     - Если запись верна, то мост оставляет адресную таблицу без изменений.
     - Если запись в таблице не соответствует информации из кадра, пришедшего на вход моста, она исправляется.
2. На основании MAC-адреса назначения принимается решение о продвижении кадра.
   - Если адрес назначения лежит в той части сети, что и адрес источника, то кадр отбрасывается, что сокращает межсегментный трафик.
   - Если адрес назначения отсутствует в адресной таблице, мост передает кадры на все порты, кроме порта, с которого пришел. Так будет до тех пор, пока узел с обозначенным MAC-адресом сам не передаст хотя бы один пакет. После того как он выйдет на передачу, MAC-адрес попадет в адресную таблицу моста.
   - Если MAC-адрес присутствует в адресной таблице моста, то мост предает кадр только на тот порт, который сопоставлен этому адресу в таблице. На другие порты кадр не передается.

Записи в адресной таблице моста могут быть двух типов.

- Первый тип — динамический. Такие записи заносятся в адресную таблицу самим мостом и при необходимости могут быть перезаписаны или удалены. Это позволяет легко переключать конечные узлы с одного порта на другой. Мост при этом легко перестраивает адресную таблицу.
- Второй тип — статические записи. Такие записи вносятся администратором при помощи средств управления мостом (коммутатором) и не имеют срока жизни. Они служат для повышения безопасности сети.

Несмотря на те преимущества, которые дают коммутаторы по сегментации трафика в сети, они не защищают сеть от широковещательного шторма (broadcast storm), когда какой-то узел начинает генерировать кадры с достаточно высокой интенсивностью (например по причине неисправности), и эти кадры передаются на все порты коммутатора, попадая во все сегменты сети.

## Петли при использовании мостов

Пожалуй, самым главным ограничением, накладываемым прозрачными мостами на архитектуру сети, является недопустимость петель. Легко это пояснить на практическом примере.

Допустим, есть две сети: сеть 1 и сеть 2. Они соединены между собой двумя мостами. Мосты включены параллельно (рис. 3.3).

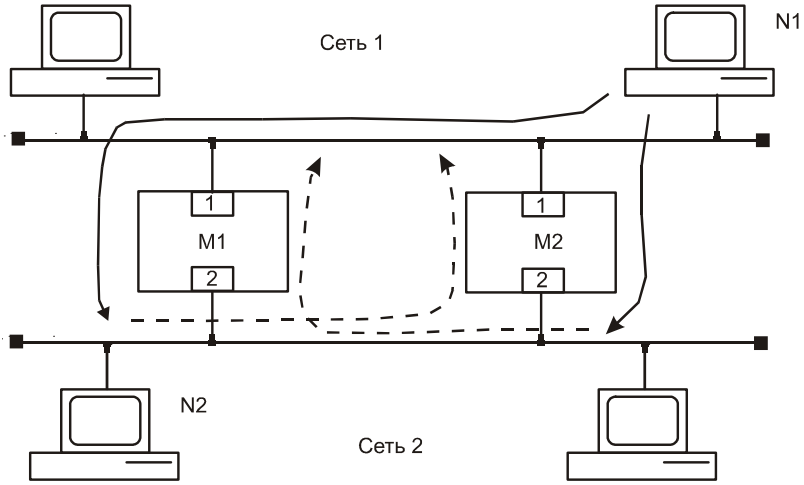



**Рис. 3.3.** Проблема петель при использовании коммутаторов

Допустим, какому-то узлу сети 1 (узел N1) необходимо передать данные узлу, находящемуся в сети 2. Пусть это будет узел N2.

1. В соответствии с логикой работы сети узел N1 получает доступ к общей временной среде и передает свой кадр.
2. Этот кадр попадает на порт 1 коммутаторов M1 и M2.
3. Коммутаторы буферизуют кадр и начинают его обработку. Если узел работает впервые, то данные заносятся в адресную таблицу.

Следующим шагом коммутаторы начинают продвигать кадр в сеть 2.

Допустим, что первым доступ к ней получил коммутатор M2. Он передает кадр в сеть. Теперь этот кадр попадает в буфер второго порта коммутатора M1, и тот начинает его обработку.

1. Коммутатор M1 видит, что кадр пришел из узла N1.
   - В его адресной таблице уже есть такая запись "N1 — порт 1", однако новый кадр более свежий, он исправляет эту запись "N1 — порт 2" и снова передает кадр в сеть 1.
   - Теперь настал черед кадра, буферизованного в порту 1 коммутатора M1. Он передается в сеть 2.
1. Попав на порт 2 коммутатора M2, кадр там буферизуется и обрабатывается. Первым делом выясняется, что в адресной таблице M2 есть запись "N1 — порт 1". Новый пакет говорит об обратном: "N1 — порт 2". Поэтому адресная таблица корректируется, а кадр передается в сеть 1.

Итак, в сеть 1 попадает два кадра. Легко видеть, что в соответствии с логикой работы моста, эти два кадра будут бесконечно циркулировать по сети. Помимо этого, коммутаторы будут постоянно перестраивать свои адресные таблицы.

Поэтому в локальных сетях нельзя образовать активные петли. В малых сетях это решается достаточно успешно. Однако в больших сетях достаточно сложно обеспечить отсутствие петель. Кроме того, желательно было бы иметь резервные линии передачи, на случай выхода из строя основных линий связи. Для реализации резервных связей без нарушения работы всей сети был разработан специальный алгоритм работы, он получил название алгоритм покрывающего дерева (Spanning Tree Algorithm — STA).

## 3.4. Полнодуплексный и полудуплексный протоколы

С точки зрения топологии микросегментация представляет собой подключение к порту коммутатора не целого сегмента сети, состоящего из нескольких узлов, а всего одного компьютера. Доменом коллизий в этом случае является участок, состоящий из порта коммутатора, сетевого адаптера и кабеля, соединяющего их.

Данный участок сети может работать в двух режимах: полудуплексном и полнодуплексном. При полудуплексном режиме один из портов передает, а

другой принимает данные. Одновременная работа портов на передачу вызывает возникновение коллизии.

При полнодуплексном режиме оба порта работают с максимальной скоростью на передачу в обе стороны. Такое возможно благодаря использованию разных пар кабеля на прием и передачу (Ethernet, Fast Ethernet) либо с помощью специальных схем вычитания сигнала (Gigabit Ethernet). Для нормальной работы необходимо, чтобы оба устройства могли работать в этом режиме. Сегодня практически все коммутаторы и сетевые адаптеры имеют такую возможность.

Однако отказ от общей разделяемой среды может привести к потере кадров при передаче через коммутатор. Одна из причин может крыться в неравномерном характере трафика. Большинство сетей построено по технологии клиент-сервер. Сервер является хранилищем практически всей информации, и узлы обращаются к нему за доступом к этой информации. В результате трафик неравномерно распределяется между портами. Порт сервера вынужден пропускать суммарный трафик всех остальных портов (рис. 3.4). Учитывая постоянный рост объемов информации, возможна ситуация, когда в порт сервера направится поток данных, больший, чем тот может обработать. Буфер порта переполнится, что приведет к потере кадров.

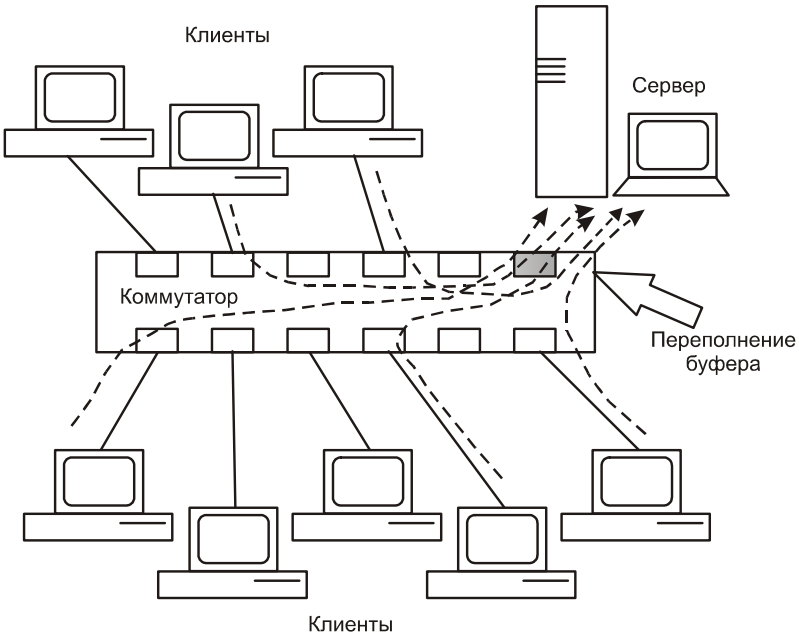



**Рис. 3.4.** Переполнение буфера коммутатора

Поэтому для управления потоками кадров от узлов коммутаторы используют специальные алгоритмы.

## Управление при полудуплексной работе

При полудуплексном режиме работы управление потоком кадров через коммутатор основано на логике работы узлов в разделяемой среде. Как вы знаете, если два узла одновременно пытаются получить доступ к среде, то возникает коллизия, узлы прекращают передачу и выжидают некоторое время. Если один из узлов захватил среду, остальные ждут, когда она освободится. Этот алгоритм работы можно использовать для управления потоком кадров от узлов.

Как правило, выделяют два метода: метод обратного давления (backpressure) и метод агрессивного захвата.

Суть первого состоит в том, что коммутатор следит за наполнением всех своих портов, и если какой-то порт заполняется, то чтобы подавить активность сегмента на выходе порта, коммутатор создает искусственную коллизию.

Второй метод состоит в том, что коммутатор монопольно захватывает шину. Для этого коммутатор сокращает технологические паузы между кадрами, то есть не выжидает положенного по стандарту времени и начинает свою передачу раньше всех узлов в сети.

Таким образом, коммутатор не всегда соблюдает правила работы в сети, однако это позволяет достаточно эффективно управлять поведением узлов и избегать потери кадров вследствие переполнения буфера коммутатора, не изменяя алгоритм работы конечных узлов.

## Управление при полнодуплексной работе

При работе в полнодуплексном режиме применить методы обратного давления и агрессивного захвата невозможно, поскольку и прием, и передача идут по разным физическим каналам. А поскольку для полнодуплексного режима работы пришлось перерабатывать протоколы взаимодействия конечных узлов, то заодно было решено внести туда механизм управления потоком кадров.

Для управления потоком передачи конечного узла в протокол взаимодействия было введено две новых команды: приостановить передачу и возобновить передачу. При получении первой узел должен приостановить передачу до получения второй. Данное положение было закреплено в стандарте 802.3x.

Для сетей Gigabit Ethernet характерна высокая скорость передачи данных. Поэтому приостановка передачи может вызвать переполнение буфера того устройства, которое приостановило свою деятельность. Здесь ситуация похожа на толпу: если идущий впереди резко затормозит, то его просто затопчут. Для того чтобы избежать такой ситуации, применяется более тонкое управление, когда узел не приостанавливает свою передачу, а просто понижает ее интенсивность. Такой механизм позволяет избежать переполнения буферов.

# 3.5. Проблема полосы пропускания

Два предыдущих решения были продиктованы необходимостью управления передачей конечных узлов в сетях любого типа. Однако в большинстве случаев современная сеть характеризуется неравномерностью трафика между узлами в сети. В сети, как правило, есть выделенный сервер. Это может быть узел с серверной операционной системой или обычный узел, предоставляющий определенные сервисы другим узлам (одноранговая сеть).

Основной поток данных в такой сети направлен от конечных узлов к серверу и наоборот. Между узлами обмена практически нет. Это значит, что применение только управления потоком кадров приведет к тому, что пропускная способность канала сервер-коммутатор будет делиться между всеми компьютерами в сети. При достаточно большом числе узлов это приведет к снижению интенсивности обмена данными между отдельно взятым узлом и сервером.

Допустим, узлов 10. И подключены все узлы и сервер по технологии Fast Ethernet (рис. 3.5). Если интенсивность работы узлов низкая, и они не мешают друг другу, то каждый узел получает полосу пропускания 100 Мбит/с. Если интенсивность узлов начинает расти, то начинается конкуренция узлов между собой. В предельном случае каждому узлу достанется 100 / 10 = 10 Мбит/с, а за счет неравномерности работы, технических ограничений и управления передачей — гораздо меньше.

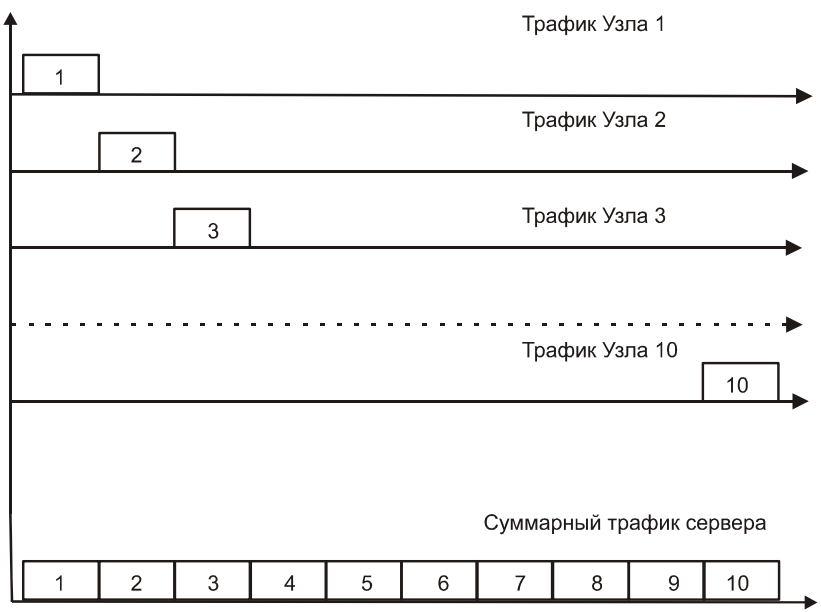



**Рис. 3.5.** Пропускная способность в клиент-серверных технологиях

Выходом из этого положения является создание коммутаторов на одном или нескольких портах по технологии Gigabit Ethernet. В нашем случае при замене коммутатора каждый узел получит канал пропускной способностью: 1000 / 10 = 100 Мбит/с. Однако с ростом числа узлов пропускная способность канала снова начинает снижаться. Рекомендуется использование двух портов со скоростью 1000 Мбит/с и распределение узлов между портами. Многие современные модели коммутаторов имеют несколько выходов для подключения высокоскоростных сетевых карт серверов.

Например, неуправляемый коммутатор второго уровня DES-1026G имеет 24 порта 10/100Base-T и 2 порта 10/100/1000Base-T, с функцией автоматического определения скорости.

DES-1200M представляет собой модульное шасси, на которое могут устанавливаться дополнительные модули, например DES-121T, обеспечивающий Gigabit Ethernet 100/1000Base-T.

Другим примером может послужить коммутатор 3 Com SuperStak 3 4226T, который имеет 24 порта 10Base-T/100Base-TX и 2 порта 10Base-T/100Base-TX/1000Base-T.

Коммутатор 3 Com SuperStak 3 4228G также имеет имеет 24 порта 10Base-T/100Base-TX и 2 порта 10Base-T/100Base-TX/1000Base-T. К тому же, еще у него есть слоты расширения для установки дополнительных модулей (рис. 3.6).

Подобные решения обеспечивает экономичное подключение к Gigabit Ethernet, устраняя узкие места в сети при подключении серверов и магистрали сети.

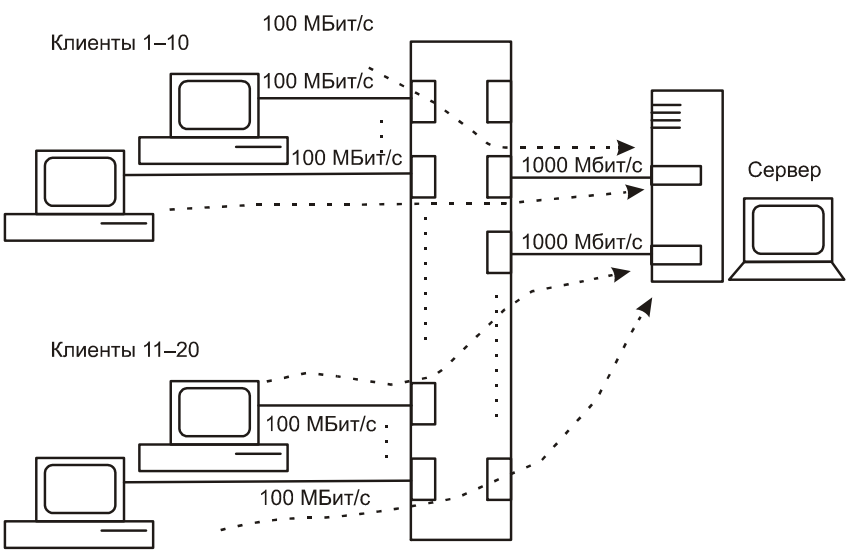



**Рис. 3.6.** Решение проблемы перегрузки портов

# 3.6. Технологии коммутации

Применяемые в коммутаторах технологии можно классифицировать с точки зрения модели OSI. Можно выделить три типа коммутации: коммутация второго уровня, коммутация третьего уровня и коммутация четвертого уровня. Каждый из типов коммутаторов работает на своем уровне и прозрачен для протоколов, работающих на более высоком уровне модели OSI.

## Коммутация второго уровня

Коммутаторы второго уровня — это аппаратные коммутаторы. Они строятся на основе специальных микроконтроллеров ASIC (Application Specific Integrated Circuits). Характерной особенностью коммутаторов второго уровня является высокая производительность (Gigabit Ethernet), поскольку данные не подвергаются обработке, а недостатком — то, что они не защищают сеть от широковещательного шторма. Как правило, применяются они для коммутации рабочих групп и для сегментации сети.

Например: DES-1016 R+ — коммутатор второго уровня. Его характеристики таковы:

- 16 портов 10/100 Мбит/с;
- поддерживает стандарты:
  - IEEE 802.3 10Base-T Ethernet,
  - IEEE 802.3u 100Base-TX Fast Ethernet;
- поддержка полного дуплекса на любом порту;
- автоматическая коррекция полярности подключения портов;
- автоопределение MDI/MDIX;
- таблица MAC-адресов — 4 Кбайт на устройство;
- пропускная способность при фильтрации пакетов Ethernet 14,880 pps (packet per second — пакетов в секунду) на порт;
- размер RAM-буфера 256 Кбайт на 8 портов.

Можно обратиться к другой фирме, например 3Com, и там мы найдем семейство коммутаторов 3 Com SuperStack 3 Switch 4200. Для данного устройства характерны:

- производительность 6 млн pps (4226T) — 10 млн pps (4250T);
- поддержка алгоритма STA (*описываемого в разд. 3.9*);
- автоопределение скорости и соединения коммутатор-коммутатор:
- возможность управления по протоколам SNMP, MIB, MIB II.

## Коммутация третьего уровня

Коммутация третьего уровня основана на обработке сетевой информации в пакете. Технически коммутаторы третьего уровня — это также аппаратные коммутаторы на основе ASIC-микроконтроллеров. Отличие коммутатора третьего уровня от маршрутизатора состоит в том, что маршрутизатор имеет программную реализацию, а коммутатор аппаратную, что определяет более высокую скорость работы и более низкую стоимость.

Обработка пакетов в коммутаторе третьего уровня включает следующее:

- определение пути на основе сетевых адресов (для протокола TCP/IP это IP-адреса);
- проверка целостности заголовков третьего уровня на основе контрольной суммы;
- проверка TTL или времени жизни пакета.

Помимо этого, при обработке пакетов осуществляется управление безопасностью, контроль статистики трафика (пакетов).

Примерами коммутаторов третьего уровня можно назвать:

- DGS-3324Sri;
- DGS-3308;
- DES-3350 SR;
- 3Com Switch 4924;
- 3Com Switch 4950;
- 3Com Switch 40Ч0.

Устройства такого плана имеют большое число (12—24) портов 10/100/1000 Мбит/с с автоматическим определением скорости, поддерживают основные протоколы маршрутизации и протоколы управления сетью. Имеют встроенную поддержку VLAN, STA, поддерживают классификацию трафика на основе IP-адресов и номеров портов TCP.

## Коммутация четвертого уровня

Коммутаторы четвертого уровня могут работать с информацией четвертого, транспортного уровня. Это позволяет управлять пакетами не только на основе MAC или IP-адресов, но и на основе данных дейтаграмм TCP и UDP. При маршрутизации могут учитываться номер порта или тип протокола.

Это позволяет определять качество сервиса: например, для мультимедийных потоков данных предоставляется более высокий приоритет, чем для протокола электронной почты POP3.

Умение использовать информацию четвертого уровня позволяет вести гибкую статистику трафика. Коммутаторы четвертого уровня — аппаратные.

# 3.7. Техническая реализация коммутаторов

Сегодня в специализированных магазинах существует большой выбор коммутаторов. Различаются они по функциональным возможностям: число портов, скорость фильтрации, скорость маршрутизации, пропускная способность и др. Чем сложнее и эффективнее коммутатор, тем выше его цена. По архитектуре различают три варианта коммутаторов:

- коммутатор с разделяемой памятью;
- коммутатор с общей шиной;
- коммутатор на основе коммутационной матрицы.

## Коммутаторы с разделяемой памятью

Основой коммутаторов этого типа является общая для всех портов память (Shared memory). Схема управления коммутатором организует в общей памяти несколько очередей с пакетами, по одной на каждый порт (рис. 3.6).

Анализируя адрес назначения, схема управления записывает пакет в соответствующую очередь. Пакеты в каждой очереди продвигаются по принципу: первый вошел — первый вышел (First Input First Output — FIFO). По мере продвижения в очереди пакет через коммутатор попадает на выход интересующего его порта. Хотя в коммутаторах с разделяемой памятью и используют быстродействующую память, буферизация данных перед их выдачей на выход коммутатора приводит к возникновению задержек.

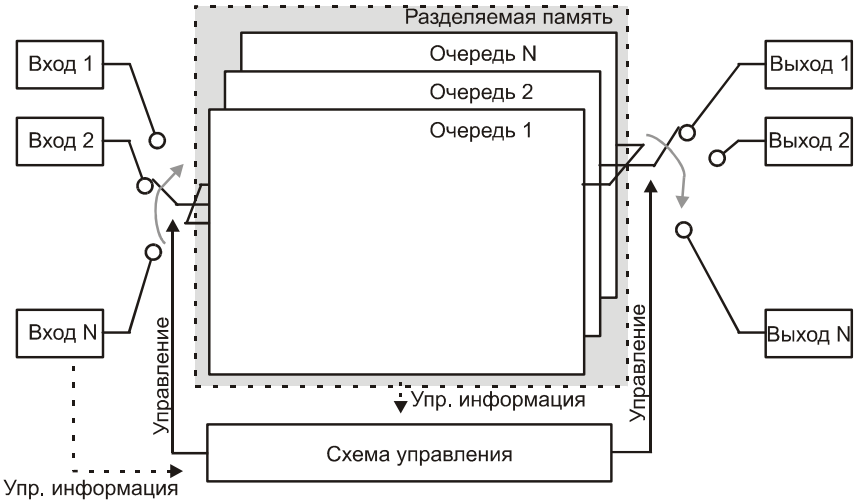



**Рис. 3.7.** Коммутатор с разделяемой памятью

# Коммутатор с общей шиной

Коммутаторы с общей шиной (backplane) используют для продвижения пакетов между портами высокоскоростную шину, в режиме разделения времени (рис. 3.7).

Входными устройствами данные преобразуются к формату, пригодному для передачи по высокоскоростной шине. Кадры для передачи по высокоскоростной шине разбиваются на небольшие куски, снабжаются адресом порта назначения. Затем они передаются по очереди по шине. В результате получается, что все порты могут работать одновременно, и нет необходимости буферизации кадра. Такая работа называется псевдопараллельной. Для того чтобы избежать перегрузки шины, ее производительность должна равняться производительности всех портов вместе взятых.

Так как шина может обеспечить одновременную передачу данных от всех портов, то такие коммутаторы называют неблокирующими (non-blocking).

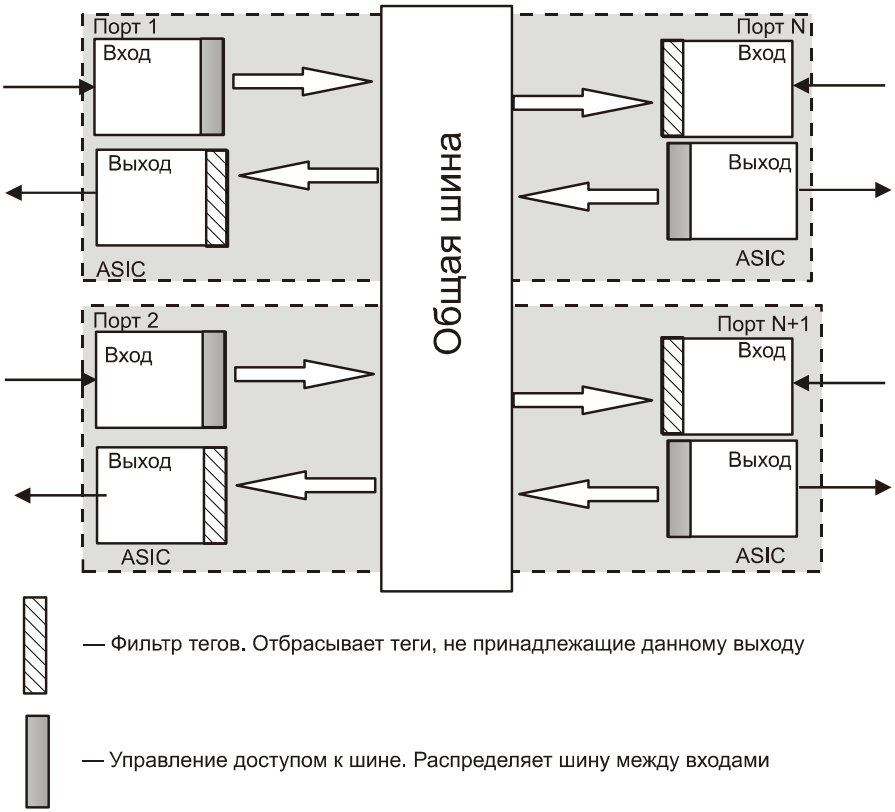



**Рис. 3.8.** Коммутатор с общей шиной

## Коммутатор на основе коммутационной матрицы

Коммутационная матрица (cross-bar) представляет собой основной и самый быстрый способ взаимодействия передачи данных от одного порта к другому (рис. 3.9). Здесь К — элемент, который осуществляет коммутацию между двумя сигналами на входе. Основные достоинства такой схемы коммутации — это высокая скорость и регулярная структура коммутационной матрицы. Однако у коммутационной матрицы есть серьезный недостаток — сложность реализации возрастает пропорционально количеству портов коммутатора. На практике это приводит к тому, что реализовать по данной схеме можно только коммутатор с ограниченным числом портов. Еще одним недостатком является отсутствие буферизации данных.

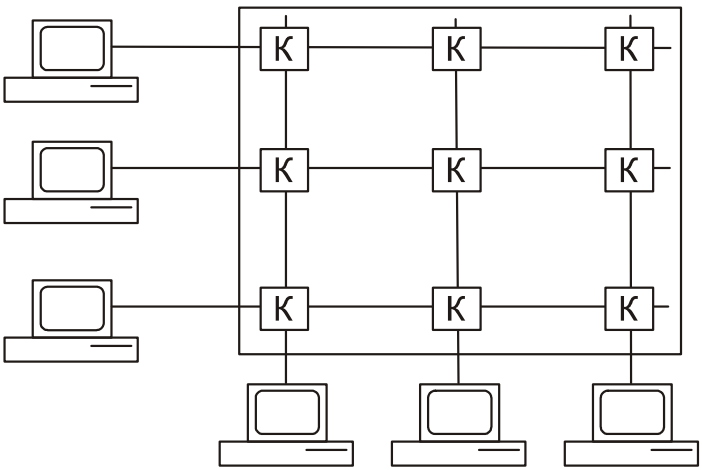



**Рис. 3.9.** Коммутатор на основе коммутационной матрицы

Современные коммутаторы на основе специальных микроконтроллеров ASIC комбинируют в себе несколько архитектур.

# 3.8. Основные характеристики коммутаторов

Один из самых важных параметров при выборе коммутатора — это производительность. От правильного выбора уровня производительности коммутатора зависит нормальная работа вашей сети.

## Основные характеристики

Основными характеристиками коммутатора являются следующие:

- Скорость фильтрации (filtering) представляет собой скорость, с которой выполняется прием кадра в буфер, просмотр адресной таблицы с целью

нахождения порта назначения, уничтожение кадра в буфере (так как адресат находится на том же порте, что и источник).

Скорость фильтрации измеряется в кадрах в секунду. Данные приводятся для кадров минимальной длины, так как именно они создают наиболее плохой режим для коммутатора. В самом деле, чем короче кадр, тем меньше полезной информации помещается в нем. Решение же о продвижении кадра приходится принимать по каждому пришедшему кадру. Поэтому при равных объемах переданной полезной информации у кадров минимальной длины коммутатор чаще просматривает адресную таблицу и принимает решение о продвижении или отбросе кадра.

- Скорость продвижения (forwarding) представляет собой скорость, с которой выполняется прием кадра в буфер, просмотр адресной таблицы с целью нахождения порта назначения, передача кадра в другой сегмент сети на основе данных адресной таблицы.

  Как и скорость фильтрации, этот параметр измеряется в кадрах в секунду. Измерения проводятся для кадров минимальной длины.

- Пропускная способность (throughput). Измеряется количеством переданной полезной информации в единицу времени. Под полезной информацией подразумевается информация, переносимая в теле того кадра, на уровне которого работает коммутатор. Для коммутаторов второго уровня это информация в теле кадра Ethernet. Для коммутатора третьего уровня это информация, переносимая в теле IP-пакета.

- Задержка передачи — время с момента прихода кадра на вход коммутатора до его появления на одном из выходов. Задержка складывается из буферизации кадра на входе, поиска маршрута продвижения в адресной таблице и собственно продвижения кадра на выход коммутатора.

- Размер адресной таблицы — максимальное число соответствий между MAC-адресом и портом. Недостаточный размер адресной таблицы служит причиной дополнительных задержек: если адресная таблица заполнена, а коммутатор встречает новый адрес, то он начинает удаление одного из старых адресов из таблицы и внесение туда нового, а на это тратится время.

- Объем буфера порта. Буфер служит для временного хранения данных, если их невозможно передать на другой порт, сглаживает пульсации трафика. Однако если буфер недостаточно большой, то кадры будут теряться, что вызовет необходимость их повторной передачи и потерю времени.

В технических описаниях (примером служит табл. 3.1) могут приводиться как все параметры, так и их часть, но и этой части вполне достаточно для того, чтобы оценить производительность коммутатора.

***Таблица 3.1.*** *Характеристики коммутатора фирмы 3Com*

| **Число портов** | **48 10/100 Ports Plus 2 10/100/1000** |
|---|---|
| Тип интерфейса | 10Base-T/100Base-TX/1000Base-T с разъемами RJ45 |
| Выбор скорости: | автоматический на каждом порту |
| Выбор режима MDI/MDIX | автоматический на каждом порту |
| Управление | конфигурирование системы, отображение подключенных устройств, выдача сообщений о неисправностях и отчетов |
| Режимы связи | поддерживает режимы Full-duplex и Half-duplex с автосогласованием |
| Объединение в стек | в стек можно объединять до 4-х устройств через гигабитные порты и RJ45-RJ45 (при этом используется один IP-адрес) |
| Дополнительные модули | нет |
| Индикаторы | сетевого трафика, состояния соединения, скорости работы |
| Конструктив | настольное использование или монтаж в шкаф |
| Поддержка коммутации второго уровня | поддержка стандартных функций второго уровня |
| Скорость коммутации | 13,6 Гбит/с |
| Скорость передачи данных | до 10,1 млн пакетов в секунду |
| Количество MAC-адресов | 8000 |

# 3.9. Дополнительные функции коммутаторов

Из дополнительных функций коммутатора интерес представляют расширенная фильтрация трафика, поддержка алгоритма Spanning Tree и создание виртуальных сетей.

## Расширенная фильтрация трафика

Наряду со стандартными условиями фильтрации трафика, коммутатор может предоставлять пользователю возможность создавать свои фильтры на пути кадров. Фильтрация производится на основе адресных полей, содержащихся в кадре. Это может быть MAC-адрес (для коммутатора второго уровня) или IP-адрес (для коммутатора третьего уровня). Возможна также фильтрация по определенным полям, содержащимся внутри кадра. Для того чтобы задать такую фильтрацию, необходимо указать смещение этого поля относительно начала кадра и условие фильтрации при помощи операций логического умножения (AND) или логического сложения (OR).

## Поддержка алгоритма Spanning Tree

Как вы помните, для нормальной работы необходимо отсутствие в сетевой архитектуре петель. Если в небольших сетях это не является ограничением, то в сложных сетях со множеством связей необходимо наличие резервных путей для повышения надежности работы сети. Также крайне желательно, чтобы обнаружение вышедших из строя связей было автоматизировано.

С этой целью был разработан специальный протокол взаимодействия коммутаторов. Он был закреплен в стандарте 802.1D. и получил название Spanning Tree Algorithm (STA) — алгоритм покрывающего дерева. Суть алгоритма в том, что в сети создаются резервные связи. Коммутаторы на основе обмена служебными пакетами изучают топологию сети и выбирают оптимальную древовидную конфигурацию сети. Резервные связи, образующие петли, отключаются путем блокировки соответствующих портов коммутатора. Таким образом, активные петли отсутствуют, и сеть имеет нормальную древовидную архитектуру.

Сеть постоянно тестируется служебными пакетами. Если обнаруживается потерянная связь, то коммутаторы начинают строить оптимальную конфигурацию заново.

Конфигурация строится в несколько этапов. Вначале определяется корневой коммутатор, от которого будет строиться дерево, с минимальным MAC-адресом. На втором этапе для каждого коммутатора определяется корневой порт, имеющий кратчайшее расстояние до корневого коммутатора. Затем определяются порты, через которые подключены другие коммутаторы по кратчайшему пути. Все остальные порты блокируются.

Для построения алгоритма и тестирования исправности дерева используется специальный пакет данных Bridge Protocol Data Unit (BPDU) — протокол блока данных моста. Пакет BPDU изображен на рис. 3.10.

**Рис. 3.10.** BPDU-пакет

Идентификатор BPDU необходим для определения версии протокола BPDU. В случае использования разных версий может образоваться активная петля.

Тип BPDU. Возможно два типа. Первый — это заявка на то, чтобы стать корнем. Второй — это требование проведения реконфигурации.

Флаги — флаг подтверждения изменения конфигурации.

## Возможность создания виртуальных сетей

С точки зрения коммутатора виртуальная сеть — это несколько разделенных информационных потоков.

Основным отличием виртуальных сетей от фильтрации на основе пользовательских фильтров является то, что сеть, построенная на основе технологии VLAN (Virtual LAN — виртуальных локальных сетей), локализует широковещательные сообщения в рамках одной виртуальной сети. Иными словами, виртуальная сеть образует широковещательный домен (broadcast domain). Технология создания виртуальных сетей закреплена в стандарте 802.1Q.

С точки зрения администратора создание виртуальной сети заключается в назначении порту коммутатора номера той или иной сети.

Можно назвать следующие основные преимущества VLAN:

- повышение производительности в каждой сети за счет локализации трафика;
- возможность гибкого изменения структуры сети программными средствами;
- изоляция сетей друг от друга на гибкой основе.

Технология VLAN широко используется для построения крупных сетей, так как позволяет создавать полностью изолированные сети на основе программной коммутации. Это создает возможности гибкой перестройки сети без использования физической перекоммутации (изменение коммутации на кросс-панелях или панелях концентраторов и коммутаторов) (рис. 3.11).

Отмечается также защита от широковещательного шторма. В случае, если по какой-либо причине одна из сетей "затоплена" широковещательными сообщениями, работа остальных виртуальных сетей нарушена не будет.

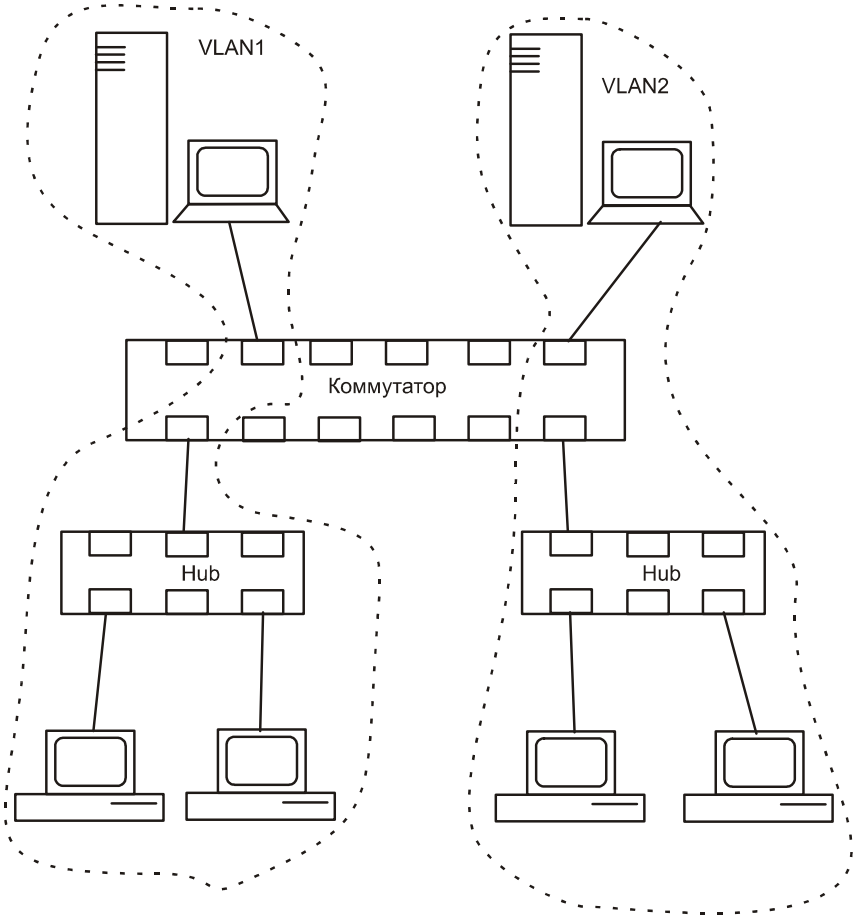



**Рис. 3.11.** Построение виртуальных сетей на базе коммутатора

# 3.10. Методика оценки необходимой производительности коммутатора

Исходными данными для расчета необходимой производительности коммутатора являются данные о трафике между узлами сети. Если сеть уже существует, то такие данные можно получить путем пассивного слежения за сетью при помощи специальных программ. Если сеть только проектируется, то источником информации может послужить подобная сеть. В самом худшем случае собираются данные по трафикам отдельных приложений и суммируются. На данном этапе необходимо уже четко представлять себе, на

каком узле какое приложение будет работать и какой трафик оно будет генерировать.

Производительность будем оценивать в байтах в секунду. Как вы знаете, производительность коммутаторов оценивается в кадрах в секунду. Причем оценка идет для кадра минимальной длины (64 байтов). Из них данные занимают именно 46 байтов. Поэтому для перевода байт/с в кадр/с необходимо разделить на число байтов в одном кадре минимальной длины, а именно на 46.

Обозначим трафик от узла *i* к узлу *j* через *Pij*. Тогда на коммутатор с архитектурой "общая шина" накладываются следующие ограничения:

- Общая производительность коммутатора должна быть больше суммарной интенсивности всего трафика $> \sum P_{ij}$ .

  В противном случае коммутатор не будет справляться с потоком кадров, и они будут отбрасываться.

- Производительность каждого порта (пусть это порт с номером *k*) коммутатора должна быть не меньше трафика через этот порт:
  - для полудуплексного общая производительность $> \sum P_{ik} + \sum P_{kj}$ ;
  - для полнодуплексного общая производительность $> 2 \times (\sum P_{ik} + \sum P_{kj})$ .

- Производительность общей шины должна быть больше средней интенсивности суммарного трафика $> \sum P_{ij}$ .

- Производительность процессора каждого порта не меньше интенсивности трафика через порт.

Помимо приведенных, на производительность коммутаторов влияет размер адресной таблицы: недостаточность служит причиной дополнительных задержек, тратится дополнительное время на обновление.

Важную роль играет и объем буфера порта для временного хранения данных. Буфер сглаживает пульсации трафика. Если он будет маленьким, то кадры потеряются, что вызовет необходимость их повторной передачи.

# Глава 4

# Проектирование кабельной системы

В предыдущих главах мы рассмотрели протокол TCP/IP и его функционирование, изучили теоретические принципы построения IP-сетей, основные стандарты и их технические характеристики. В этой главе будут рассмотрены основы проектирования кабельной системы сети.

## 4.1. Логическая структуризация сети и кабельная система

Нормальная работа сети зависти от того, насколько грамотно проведен этап проектирования сети, ее логической структуры и кабельной системы. Так что построение структурированной кабельной системы (СКС) — объективная необходимость на сегодняшний день.

### Необходимость логической структуризации сети

Допустим, на базе нескольких локальных сетей, возникших в разное время и предназначенных для решения различных задач (бухгалтерский учет, отдел продаж, конструкторское бюро, кадровая служба), в рамках организации возникла единая система. Если объединение пошло по одной из элементарных топологий (общая шина или звезда), то для такой системы можно выделить несколько отрицательных моментов:

- Различия проявляются не только на программном уровне, но и на физическом. Например, в кадровой службе сеть может быть на коаксиальном кабеле, а в конструкторском бюро на витой паре.
- Для такой сети характерна неоднородность информационных потоков (конструкторская документация, управленческие данные и др.). Причем

наиболее интенсивный обмен данными идет между узлами одного структурного подразделения. Взаимодействия между подразделениями организации (рабочими группами), конечно же, существуют. Однако соотношение трафика внутри рабочей группы и за ее пределами примерно пятикратное.

- Сложность администрирования и обеспечения безопасности такой сети, к которой все имеют доступ, возрастает.
- Ограничение по количеству узлов в сети определяется ее технологией (для Ethernet это 1024 узла). Если для небольшой организации оно вряд ли представляет угрозу, то для крупных компаний создает большую проблему.
- Физическое ограничение на протяженность сети, вызванное использованием метода доступа к общей временной среде CSMA/CD, а также ограничение на интенсивность трафика, играет важную роль. (Как вы помните, с увеличением трафика растут коллизии и снижается эффективность работы сети).

Применение современных технологий хранения данных (на серверах) практически не изменит сложившуюся картину, поскольку с ростом требований к вычислительным возможностям компьютерной техники свои серверы появились практически у каждого подразделения.

Все сказанное говорит о том, что единая разделяемая среда для построения крупных сетей непригодна. Необходимо логически структурировать сеть с использованием коммутаторов, маршрутизаторов или мостов.

## Структурированная кабельная система

Кабельная система является фундаментом сети. От правильности ее построения зависит не только нормальное функционирование сети, но и удобство в работе. Простой пример: допустим, в бухгалтерии приобрели новый компьютер с целью автоматизировать одну из операций. Поставили на рабочее место, а подключить некуда, в проекте кабельной системы не была заложена избыточность.

С целью стандартизации кабельных систем в 1995 г. был принят стандарт EIA/TIA 568A, определивший понятие *структурированной кабельной системы* (Structured Cabling System — SCS). В стандарте описаны основные требования и технические характеристики, применяемые для СКС. Впоследствии этот стандарт был дополнен.

Структурированная кабельная система имеет иерархическое строение и состоит из следующих уровней (рис. 4.1):

- горизонтальная система;
- система здания;
- система городка.

Горизонтальная подсистема соответствует отдельным этажам здания. Она представляет собой кроссировочный шкаф, кабельную систему на этаже и розетки пользователей.

Система здания объединяет в единое целое системы этажей с головной аппаратурой здания.

Соответственно, система городка объединяет в единое целое системы зданий.

Рассмотрим подробнее преимущества предлагаемой схемы.

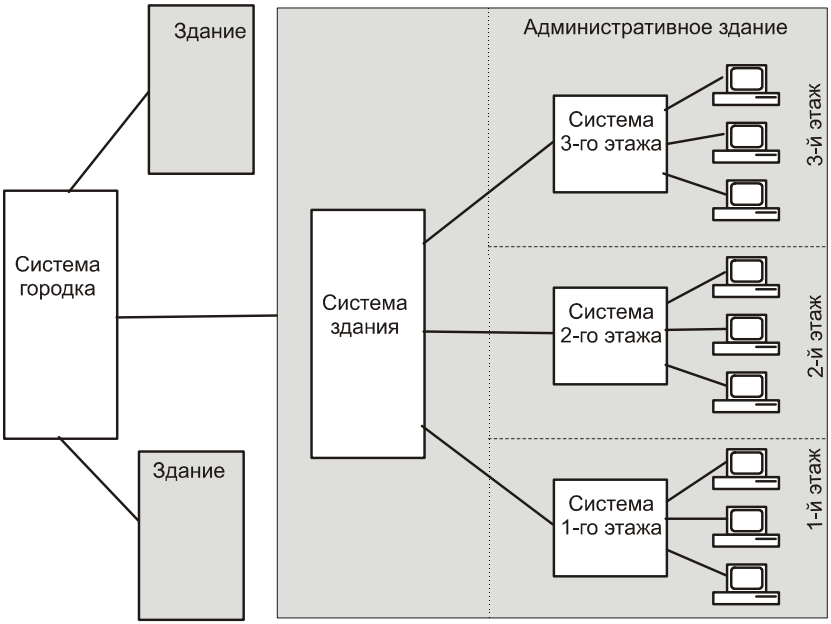



**Рис. 4.1.** Схема структурированной кабельной системы

- СКС как среда состоит из вполне конкретных элементов передачи данных (разъемы, кабели шкафы), а также из элементов, не имеющих непосредственного отношения к передаче данных (кабельные каналы, лестницы), назначение каждого элемента определено и прописано стандартом. То, что на каждом уровне определены выполняемые функции, позволяет при необходимости легко расширить сеть.
- Универсальность состоит в том, что система поддерживает работу разнородного оборудования различных производителей для передачи данных различного типа: компьютеры, телефонная сеть, служебные сигналы.
- СКС обычно монтируется с расчетом на 10—15 лет работы. Это значит, что в течение этих лет она должна прослужить без существенных измене-

ний. Поэтому на этапе проектирования рассчитывается потребность с существенным запасом — избыточная. В простейшем случае это возможность подключения телефона и компьютера на каждом рабочем месте.

- Надежность состоит в использовании качественного оборудования и совместимости оборудования различных производителей между собой.

# 4.2. Активное оборудование локальных сетей и его связь с СКС

Можно выделить следующие требования к сетевому оборудованию.

## Горизонтальная система

Начнем с горизонтальной системы. Основные задачи этого уровня:

- создание отдельных доменов коллизий;
- управление доступом и политикой сети;
- подключение к системе здания.

Уровень доступа является ближайшим к пользователю. Активное сетевое оборудование этого уровня должно поддерживать подключение отдельных пользователей к сети. Применяются здесь коммутаторы со скоростью подключения 10/100 Мбит/с, с одним или несколькими портами на 1000 Мбит/с.

## Система здания

Следующим уровнем является система здания. Коммутаторы этого уровня служат местом концентрации горизонтальной системы. Они должны справляться с большим потоком данных. В функции этого уровня входят:

- объединение низкоскоростных каналов горизонтальной подсистемы в высокоскоростные каналы;
- переход от одной физической среды распространения сигнала к другой;
- обеспечение безопасности сети и качества обслуживания.

## Система городка

Активное сетевое оборудование, находящееся на этом уровне, должно надежно и быстро передавать большие объемы данных. Для этого уровня очень важен фактор отказоустойчивости, поскольку сбой на верхнем уровне иерархии может привести к остановке всей системы, поскольку этот уровень координирует их работу. Трафик здесь отфильтрован уровнем здания.

Основным требованием, предъявляемым к этому оборудованию, является его высокая производительность.

## 4.3. Выбор общей концепции сети

Полноценная структурированная кабельная система предназначена для крупных объединений пользователей с числом пользователей от 50, с характерным покрытием крупных территорий. Оптимальность применения структурированной кабельной системы тем выше, чем выше число пользователей сети и больше ее диаметр (расстояние между ее двумя наиболее удаленными узлами).

Для небольших организаций (с числом сотрудников до 100) применение полноценной структурированной кабельной системы может оказаться экономически нецелесообразным по следующим причинам:

- отсутствие необходимости в подобных решениях даже в ближайшем будущем;
- высокая стоимость внедрения СКС, определенная спецификой ее построения.

Сеть для организации ценна не сама по себе, а как средство решения определенных задач. Средства, потраченные на внедрение локальной сети, не должны быть слишком большими, поскольку это затруднит окупаемость сети. С другой стороны, они не должны быть слишком малыми, поскольку сеть должна быть надежной, избыточной и в меру масштабируемой.

Вначале стоит разобраться с концепцией сети как таковой. Даже если организация располагается на разных этажах и в разных зданиях, кабельная система будет строиться как горизонтальная система (или, как ее еще называют, система этажа). Это значит, что будет единый коммутационный центр, один или несколько коммутаторов второго уровня, монтажные шкафы и кроссировочные панели. Смысла строить систему здания для объединения разбросанных групп пользователей нет. На этом оборудовании можно не экономить, оно относительно легко демонтируется, если приходится менять помещение. С общей концепцией построения мы определились, теперь выберем вариант прокладки кабеля.

Есть три варианта решения задачи: открытая прокладка кабеля вдоль стен, прокладка кабельной системы в недорогих кабельных каналах, прокладка кабельной системы ЛВС совместно с другими коммуникациями.

Первый вариант (открытой проводки) дешев и прост в монтаже. Кабели прокладываются открытым методом. Недостатком этого способа является незащищенность кабеля от внешних воздействий и эстетическая непривле-

кательность. Он приемлем только на короткий период времени, например сеть нужна срочно, но скоро планируется капитальный ремонт. В этом случае полноценную сеть лучше проложить после ремонта. Вторым примером в пользу такой сети может стать случай, когда помещение арендуется на небольшой срок, и нет необходимости прокладывать полноценную кабельную систему.

Вариант второй — золотая середина. Прокладка осуществляется в кабельных каналах без учета электропроводки: только кабель ЛВС и телефонные кабели. В этом случае кабель защищен, и сеть выглядит привлекательнее.

Третий вариант — полноценная кабельная система. В едином канале все телекоммуникации: кабели ЛВС, телефонные, телевизионные, сигнализация и др. Для такой системы необходим дорогой кабельный канал с желобами для электропроводки и специальными посадочными местами для розеток. Такая система достаточно дорогая не столько за счет стоимости оборудования, сколько за счет прокладки всего кабельного хозяйства заново. Бесспорным преимуществом является структурность и универсальность такой системы. Кроме, того, кабельный канал такого типа выглядит очень эффектно.

Второй и особенно третий варианты должны планироваться с избыточностью. То есть на каждое рабочее место должен быть проведен кабель ЛВС и телефонный кабель. Помимо этого, должны быть избыточные точки входа в ЛВС и телефонные розетки, на тот случай, если придется установить еще один телефон или компьютер. Вполне возможно, что запасные кабели не будут пока подключены к коммутатору или мини-АТС.

При проектировании и прокладке сети лучше ориентировать себя на качественный кабель. В дешевых кабелях, как правило, используется очень неудачная цветовая маркировка (видимо, экономят на краске). Отличить один проводник от другого в пучке бывает очень сложно, а порой приходиться вскрывать кабель, чтобы посмотреть, что это за цвет. Помимо этого, электрические характеристики таких кабелей оставляют желать лучшего.

## 4.4. Сбор информации о будущей сети

На данный момент мы определились с общей концепцией сети. Теперь приступаем к первому этапу проектирования сети, это этап сбора информации. Наиболее важными вопросами, на которые надо ответить на этом этапе, являются следующие:

- количество и расположение рабочих мест. Учитывается, как падает свет, достаточно ли пространства для работы, не мешают ли двери, близость электрических розеток и др. Если есть малейшие сомнения, то лучше

перепланировать размещение рабочего места до того, как вы проложите сеть;

- организация баз данных и возможности их переноса. Любой специалист создает те или иные наработки. У проектировщика это может быть набор конструкторской и справочной документации. У бухгалтера это, как правило, специализированная программа бухгалтерского учета со своей базой данных. В канцелярии это может быть набор документов. Основной задачей данного этапа является предотвращение потери баз данных при дальнейшей развертывании сети;
- перспективы и направления роста в ближайшее время. В каждом подразделении необходимо выяснить, хватает ли рабочих мест, и есть ли необходимость в их увеличении. Удовлетворяет ли существующее оборудование и используемое программное обеспечение или требуется его замена;
- геометрические параметры помещений. Лучше составить поэтажную карту расположения помещений. Особое внимание уделить толщине стен, расположению капитальных стен. На плане наносятся все, даже небольшие углы (потом проще будет проектировать прокладку кабельного канала). Необходимо, по возможности, выяснить места прохождения коммуникаций внутри стен: электропроводки, канализации и др. Подобная документация должна быть у владельца здания;
- в случае наличия прежней локальной сети необходимо выяснить ее тип и возможность использования. Например, если старая сеть была проложена с кабелем UTP третьей категории, возможно использование отдельных элементов, если это не скажется на надежности.

Логическим завершением данного этапа должно стать документирование на текущий момент. Вы должны иметь план здания с размещением рабочих мест. На плане должны быть изображены пунктиром и будущие или проектируемые рабочие места. Кроме того, необходимо иметь карту имеющегося оборудования и используемого программного обеспечения (табл. 4.1).

Составленный нами план и карта имеющегося оборудования и используемого программного обеспечения послужат основой для следующих этапов: выбора сетевого кабеля, проектирования топологии сети, проверочного расчета сети на соответствие стандартам, проектирования логической структуры сервера. Табл. 4.1 является лишь примером и может быть расширена по вашему желанию.

**Примечание**

Этот и все последующие шаги по проектированию сети должны заверяться подписью заказчика или руководителя, для разрешения конфликтных ситуаций.

***Таблица 4.1.*** *Текущее состояние сети*

| Отдел | № помещения | № рабочего места, задачи | Состав оборудования | Необходимая замена | Используемое ПО | Необх. нового ПО | Необх. установки РМ |
|---|---|---|---|---|---|---|---|
| 1 | 11 | РМ-1<br>Оптимизация перевозок | PI-600,<br>128 Мбайт,<br>20 Гбайт,<br>32 Video | Добавить сетевую карту | M-CAD | Нет | 1 РМ с MCAD |
| | 12 | РМ-2<br>Оптимизация производственных планов | PII—633,<br>128 Мбайт,<br>20 Гбайт,<br>16 Video | Добавить сетевую карту | M-CAD<br>Word | Нет | |
| | 14 | РМ-3<br>Ведение документации | PII-600,<br>128 Мбайт,<br>20 Гбайт,<br>16 Video, | Добавить сетевую карту | Word<br>Excel | Нет | |
| 2 | 05 | РМ-4<br>Учет зарплаты | P4-2400,<br>512 Mb, 120Gb,<br>128 Video | Не требуется | По собственной разработке | 1С Зарплата и кадры | Нет |
| | 06 | РМ-5<br>Складской учет | P4-2400,<br>512 Мбайт,<br>120 Гбайт,<br>128 Video | Не требуется | 1С Тоговля и склад | – | |
| | 07 | РМ-6<br>Бухучет | P4-2400,<br>512 Мбайт,<br>120 Гбайт,<br>128 Video | Не требуется | 1С Бухгалтерия | – | |

# 4.5. Выбор типа кабеля

Основным основанием для выбора типа кабеля является пропускная способность вашей сети. Дополнительными являются:

- физические условия эксплуатации сети;
- наличие уже установленного оборудования и проложенной кабельной системы;
- финансовая сторона вопроса.

Рассмотрим все теоретически возможные варианты прокладки сети: тонкий коаксиальный кабель, толстый коаксиальный кабель, витая пара третьей категории, витая пара пятой категории, оптоволокно.

## Коаксиальные кабели

Достоинством коаксиальных кабелей с точки зрения прокладки является, то, что все узлы подключаются к единому кабелю, недостатком — их низкая пропускная способность (до 10 Мбит/с), что недостаточно для большинства современных приложений. Такая сеть образует единый домен коллизий, что также плохо сказывается на общей производительности сети. Существенным недостатком является также сложность поиска неисправностей в сети. Еще одним важным недостатком является, то, что такое сетевое оборудование уже сложно найти в продаже.

## Витая пара

Основным достоинством является высокая пропускная способность до 100 Мбит/с. При применении коммутаторов сеть разбивается на несколько доменов коллизий, что увеличивает общую пропускную способность и надежность работы. Достоинством является и легкость поиска неисправностей. Логика поиска здесь проста: если вышел из строя какой-то узел, то дело либо в нем, либо в порте коммутатора. Если сеть не работает полностью то, скорее всего, вышел из строя коммутатор.

К недостаткам можно отнести бóльший, по сравнению с сетью на коаксиальном кабеле, объем работ по прокладке кабеля. Причина кроется в топологии сети. Сети на витой паре строятся по топологии звезды. В центре звезды устройство, образующее единую сетевую среду, — концентратор (коммутатор). Каждый узел подключен к концентратору отдельным кабелем. Таким образом, сколько устройств, столько и кабелей.

Пропускная способность кабеля пятой категории на порядок выше — до 1000 Мбит/с, это ваш запас на будущее. Разница в стоимости кабелей треть-

ей и пятой категории не существенна. А вот стоимость повторной покупки кабеля и повторных работ по его прокладке уже заставляет задуматься.

## Оптический кабель

Преимуществом является высокая пропускная способность. Однако существенным недостатком является сложность прокладки, так как оптоволоконный кабель достаточно хрупкий. Кроме того, его монтаж значительно сложнее. Оптоволоконный кабель проигрывает UTP-кабелю по цене. А если посмотреть на запас, который дает витая пара по скорости (1000 Мбит/с), то предпочтение отдается кабелю на витой паре.

Однако оптоволоконный кабель может пригодиться для прокладки линии между двумя удаленными зданиями, так как позволяет передавать данные без промежуточного усиления на расстояние до нескольких километров.

## Эксплуатационные характеристики витой пары

Поскольку мы остановились на витой паре, то рассмотрим основные эксплуатационные характеристики этого кабеля:

- минимальный радиус изгиба не менее 4–5 диаметров кабеля;
- температура при монтаже кабеля:
  - от –20 до +60є С для кабеля в поливинилхлоридной оболочке;
  - от –35 до +200є С для кабеля в тефлоновой оболочке;
- допустимый рабочий диапазон температур:
  - от –35 до +60є С для кабеля в поливинилхлоридной оболочке;
  - от –55 до +200є С для кабеля в тефлоновой оболочке;
- возможность применения на открытом воздухе:
  - запрещено для кабеля в поливинилхлоридной оболочке;
  - разрешено для кабеля в тефлоновой оболочке.

Стоит также отметить, что кабель в тефлоновой оболочке более устойчив к влаге и брызгам по сравнению с кабелем в поливинилхлоридной оболочке.

# 4.6. Расширение пропускной способности сети

Материал этого раздела относится скорее к расширению пропускной способности уже существующей сети, нежели к проектированию новой сети.

Несмотря на то, что речь идет о скоростях 100 Мбит/с и 1000 Мбит/с, советы имеют силу и для сетей 10 Мбит/с. По мерее развития техники и программного обеспечения сеть, проложенная несколько лет назад, начинает отставать от требований, предъявляемых к ней. Конечно, возникающие проблемы различны для каждой конкретной ситуации, однако их можно свести к нескольким типовым ситуациям.

## Сеть 100Base-TX с одним сервером на основе концентраторов

Типичной проблемой таких сетей является высокий уровень трафика в сети. Как только уровень трафика приближается к 50 % пропускной способности, возрастает уровень коллизий. Как следствие, возрастает время доступа к ресурсам (рис. 4.2).

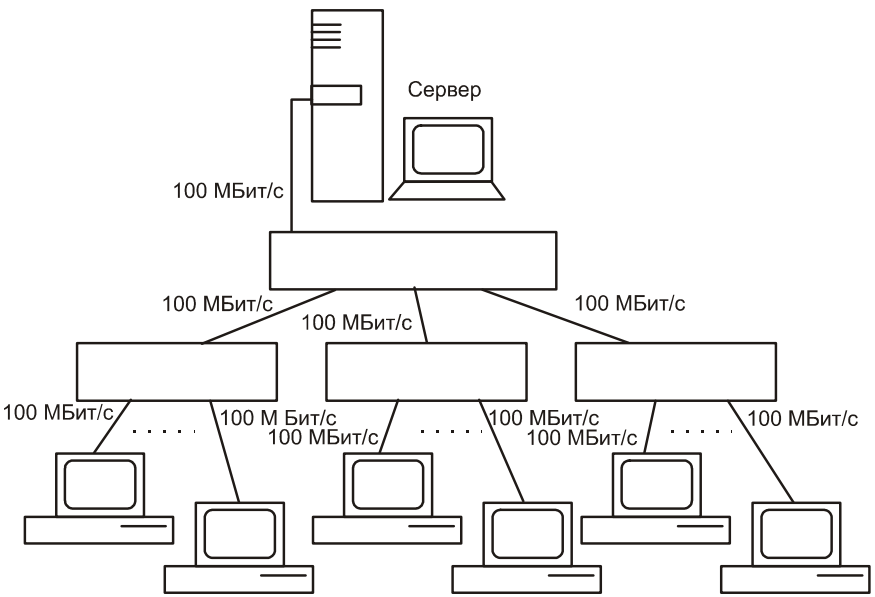



**Рис. 4.2.** Исходная сеть

Решением данной проблемы является сегментация сети, это позволит увеличить пропускную способность на одного пользователя. Для этого концентраторы расстыковываются, в сервер устанавливаются новые сетевые карты по числу концентраторов, затем каждый концентратор подключается к сетевой карте на сервере (на скорости 100 Мбит/с).

После преобразований сеть примет вид, как на рис. 4.3.

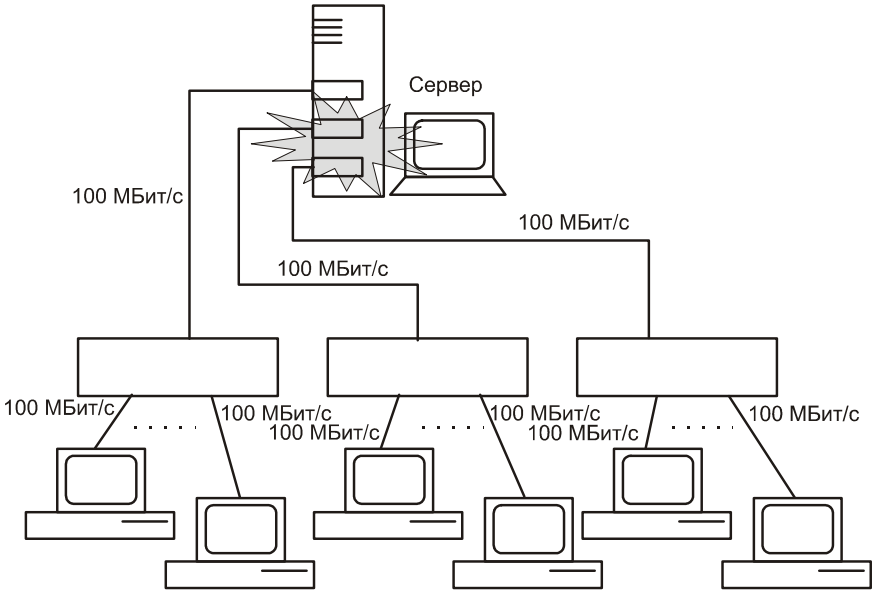



**Рис. 4.3.** Сеть после первой модернизации

## Сеть с несколькими серверами по технологии 100Base-TX на основе коммутаторов

Симптомы "заболевания" у этой сети будут аналогичные: высокий уровень трафика в сети. Как только уровень трафика приближается к 50 % пропускной способности сети, возрастает уровень коллизий. Исходная схема сети приведена на рис. 4.4.

Решение проблемы состоит в сегментации сети. В нашем случае это означает следующие действия:

- Разбираем сеть.
- В серверы ставим сетевые карты 1000 Мбит/с.
- Устанавливаем коммутатор 1000 Мбит/с.
- На два канала этого коммутатора подключаем сервер.
- Клиентов подключаем к коммутаторам 100 Мбит/с.
- Коммутаторы клиентов подключаем к центральному коммутатору на скорости 1000 Мбит/с.

Полученная сеть приведена на рис. 4.5.

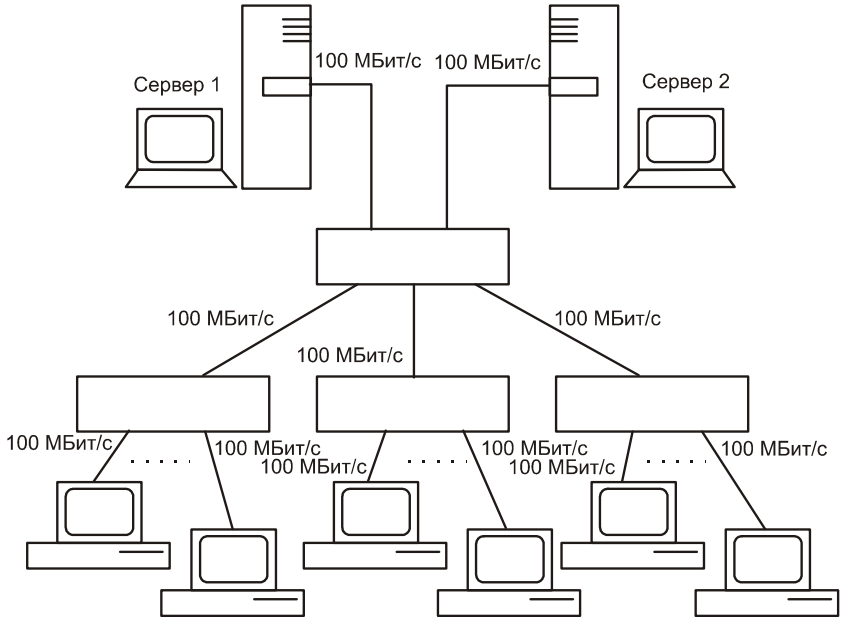



**Рис. 4.4.** Исходная сеть с несколькими серверами

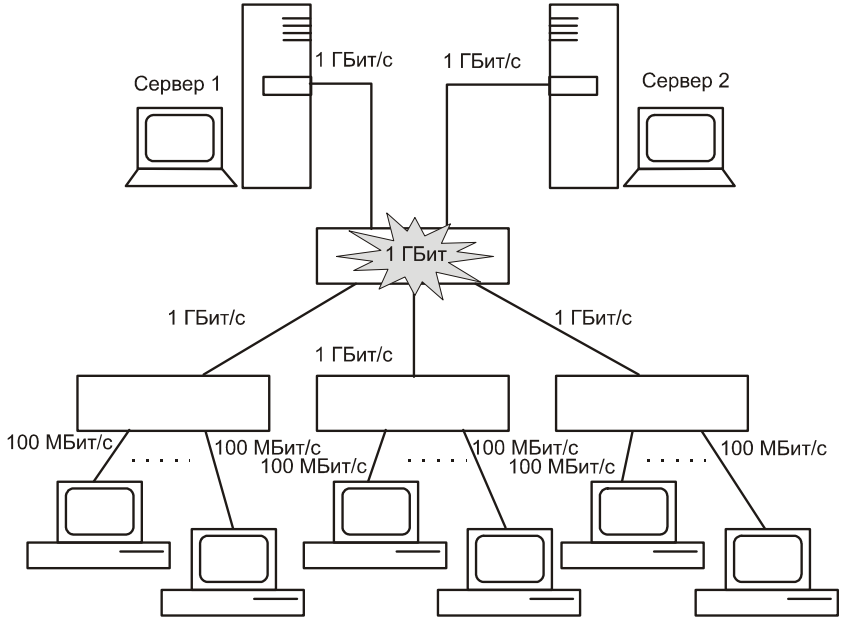



**Рис. 4.5.** Сеть с несколькими серверами после преобразования

Для небольшого числа клиентов (до 40–45) и двух серверов можно обойтись одним центральным коммутатором. В этом случае все входы к клиентам будут работать на скорости 100 Мбит/с. А входы серверов — на скорости 1000 Мбит/с.

# 4.7. Проект сети

## Общие вопросы

Теперь мы знаем, что при построении небольшой сети целесообразно применять витую пару пятой категории. Применение оптоволокна оправдано только для связи между удаленными зданиями.

Сеть должна быть избыточной, поэтому на каждые 3–4 м помещения лучше заложить отдельный UTP-кабель.

Также нужно учесть, что длина одного сегмента сети для кабеля на витой паре не может превышать 100 м. Это значит, что реально в кабельном канале длина кабеля не должна превышать 90 м. Запас 10 м необходим для соединений розетка-компьютер, плюс запас в кроссировочной панели и монтажном шкафу.

При проектировании сети очень важно учесть наличие заземления, общего для всех компьютеров сети. В противном случае между разными контурами заземления, не имеющими общего заземляющего элемента (штырь, вбитый в землю), возникает разность потенциалов. Поскольку единственным проводящим элементом между этими устройствами является кабель локальной сети, то потенциалы уравниваются путем протекания электрического тока через UTP-кабель. Ситуация может усугубиться, если в одном из заземлений произошло короткое замыкание. Подобные эффекты уравнивания потенциалов через сетевой кабель приводят не только к сбоям в работе сети, но и к выходу из строя оборудования.

При проектировании сети стоит особо обращать внимание на наличие общего заземления у всей сети. Особенно, если помещения располагаются в двух или более зданиях.

Мы знаем, что сеть будет состоять из коммутаторов 100 Мбит/с и сервера 1000 Мбит/с. При такой организации сеть работает в режиме микросегментации. Это значит, что снимаются ограничения на время двойного оборота и остаются только ограничения на максимальную длину сегмента.

## Проектирование топологии сети

Для примера проектирования была взята не самая маленькая сеть. Как правило, сети у организаций значительно меньше. Хотя данное помещение и существует, объемы кабельной системы значительно меньше предложенных в проекте. Реально, включая сервер, задействовано 46 узлов. Разводка

сделана на 50 розеток. Схема помещения и расположения рабочих мест приведена на рис. 4.6. Итак, приступим к проектированию.

Пусть на данный момент у организации насчитывается 28 компьютеров. Расположение по отделам показано на рис. 4.6. В ближайшее время планируется закупка еще 5 компьютеров с установкой их менеджерам участков и в отдел кадров.

Здание находится в собственности предприятия, в ближайшие 5—10 лет менять местонахождение не планируется. В то же время парк используемой компьютерной техники может сильно увеличиться. Поэтому целесообразно заложить избыточность в кабельной системе.

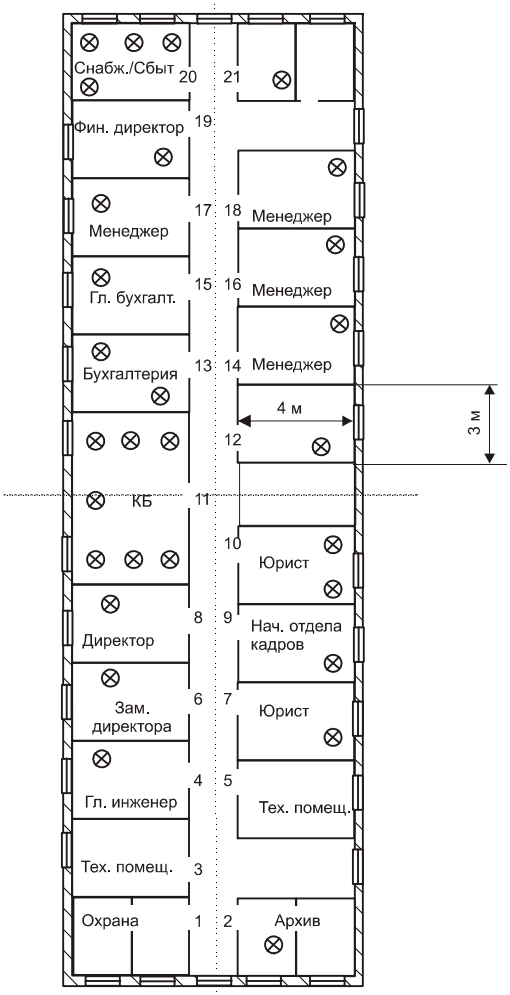



**Рис. 4.6.** Схема помещений организации

Количество клиентов в сети в ближайшее время будет:

$$28 + 5 = 33.$$

Добавьте еще два порта: один на сервер и один — на компьютер администратора. Если их не было в планах, то их надо туда включить.

Теперь, опираясь на рассмотренные в предыдущем разделе проблемы, можно нарисовать следующую структуру сети (рим. 4.7):

- ❒ два коммутатора на 24 порта 100 Мбит/с и по одному свободному порту 1000МБит/с на каждом коммутаторе;
- ❒ один сервер с двумя сетевыми картами по 1000 Мбит/с;
- ❒ серверы подключены к коммутаторам на скорости 1000 Мбит/с.

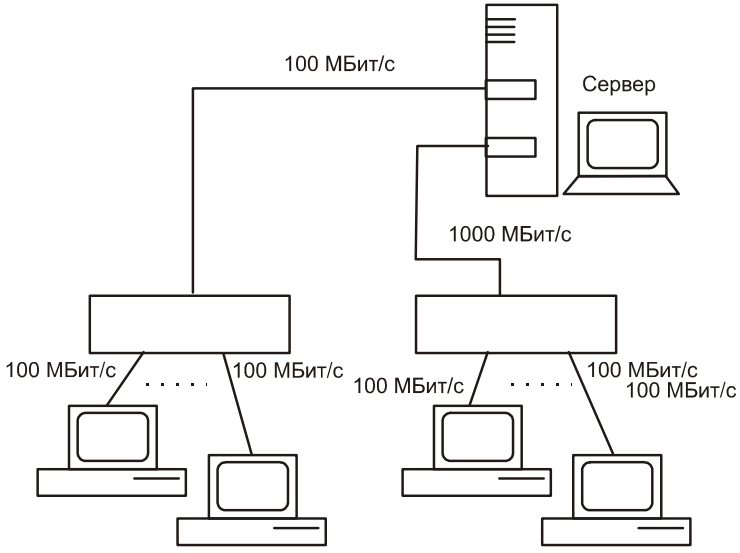



**Рис. 4.7.** Общая структура будущей сети

Поскольку длина здания составляет 40 м, а предельная длина одного сегмента для UTP кабеля 90 м (10 м запас), то разместить сервер можно в любом месте. Однако целесообразнее создать серверное помещение ближе к центру здания. Поскольку пользователи более или менее равномерно распределены по зданию, то такое размещение позволит сократить протяженность кабелей.

Ближе всего к центру расположено конструкторское бюро, однако, площадь его помещения слишком большая. Директора и отдел кадров "подвинуть сложно", а вот менеджеров попроще. Поэтому, серверная комната у нас в помещении № 12.

Избыточность будем закладывать, исходя из расчета: одна розетка ЛВС и одна телефонная розетка на 3 м$^2$ помещения. (Например, в помещении № 20 такая нагрузка уже есть) Поскольку все помещения по площади примерно одинаковые, то расчет упростится. Исключение составляет конструкторское бюро. Проведем расчет необходимого количества точек подключения:

$$20 \times (12/3) + 1 \times (32/3) = 90.$$

Получилось 90: это не значит, что нам понадобятся дополнительные коммутаторы. Мы лишь подведем провода к кросс-панелям, а к коммутаторам подключим только используемые порты.

Необходимое число панелей:

$$90 / 24 = 4 \text{ панели.}$$

Шкаф понадобится на 8 посадочных мест или чуть больше:

4 панели и 4 коммутатора.

Если сервер хотите поставить в шкафу, то необходимо предусмотреть место и для него. (Более подробно шкафы и их характеристики описаны *в главе 5*).

В реальных проектах приходится выбирать между избыточностью сети и ее стоимостью. Так что рассчитанные цифры, скорее всего, придется скорректировать в меньшую сторону.

Количество патч-кордов:

- соединение компьютер-розетка: $1 \times 33 + 2$ запас = 35 длиной 2 м;
- кроссировка в шкафу: $1 \times 33 + 2$ запас = 35 длиной 0,5–1 м.

При необходимости патч-корды можно сделать самому.

Как правило, кабельную систему стараются прокладывать на высоте около метра от пола. Если здание на данном уровне имеет много углов для батарей отопления и других целей (как в нашем случае), то целесообразнее кабельный канал проложить под потолком. Прокладка вдоль пола не является лучшим решением, поскольку во время уборки канал может подвергаться постороннему воздействию. Конечно, прокладка под потолком усложнит работу, но впоследствии это окупится.

Кабель поведем, как показано на рисунке 4.8, *а*, такая схема сэкономит кабель и сократит сечение необходимого кабельного канала. Если вести по периметру здания ко всем розеткам по очереди, это приведет к дополнительным расходам (рис. 4.8, *б*).

Расчет площади кабельного канала проведем для крыла кабинетов 14, 16, 18, 21. Для аналогичных крыльев методика расчета будет аналогичной. Исходными данными для расчетов служат геометрические параметры, количество компьютеров и размещение розеток.

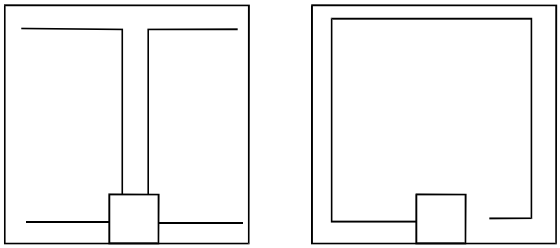



**Рис. 4.8.** Схема разводки кабеля

Розетки разместим равномерно, по две на каждой из стен на расстоянии 1 м и 3 м от внешней стены здания. Такая схема наиболее рациональна.

Количество розеток, расположенных в этих кабинетах (рис. 4.9):

$$4 \times 3 + 2 = 14.$$

Можно взять канал 80 × 40 мм, внутренним сечением 2620.

В случае, если в кабельном канале проходят проводники одного сечения, можно воспользоваться таблицами пересчета, показывающими, сколько проводников данного сечения смогут поместиться в нашем кабельном канале.

От основного канала отходят отводы к розеткам. Количество проводов в каждом сечении представлено на рис. 4.9. Здесь основной кабельный канал получился не очень большого сечения, поэтому его диаметр нет необходимости уменьшать по мере того, как большая часть отойдет на розетки. Если же сечение кабельного канала большое, то его можно уменьшать, по мере того, как количество жил кабеля будет сокращаться.

Также, забегая вперед, необходимо сказать, что для стыковки и сращивания

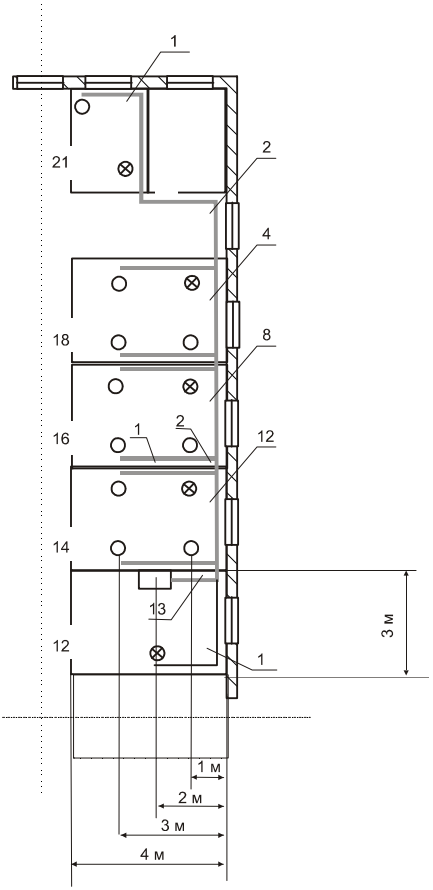



**Рис. 4.9.** Часть проекта кабельной системы

канала используются специальные элементы. Это углы и примыкания. Углы бывают внутренними, внешними и плоскими. Также есть Т-образные примыкания (рис. 4.10). Фотографии их можно увидеть в следующем разделе.

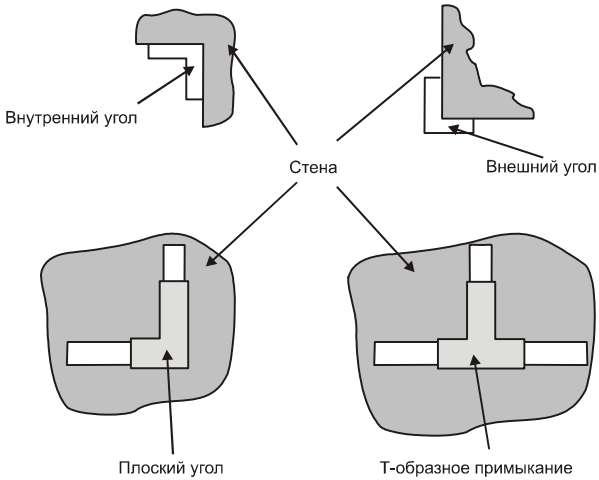



**Рис. 4.10.** Соединения кабельного канала

Считаем, что высота потолков 3 м, а розетки будут крепиться на высоте 1 м. Сервер расположен, как показано на рисунке.

Канал идет по прямой линии вдоль стены, так что нет необходимости огибать углы.

## Выбор сетевого оборудования

Выбор сетевого оборудование производится на основе спроектированной нами структуры сети и расчетов.

- Два коммутатора на 24 порта 100 Мбит/с + 1 свободному порту 1000 Мбит/с на каждом коммутаторе; например, это может быть коммутатор DES-1026G (D-Link).
- Четыре кроссировочные панели на установочный размер 19 дюймов 1U с контактами на внешней стороне RJ 45 и IDC на задней, например EuroLan 19 дюймов 24 Ч RJ45.
- Кроссировочный шкаф на 4U (кросс-панели) + 2U (коммутаторы) + 2U (коммутаторы в будущем) + 4U (запас) = 12U. Это может быть настенный шкаф.
- Сетевые карты должны быть одной фирмы, это позволит избежать проблем в дальнейшем. Лучше если они будут той же фирмы, что и комму-

таторы. Несмотря на то, что все протоколы канального уровня стандартизированы, бывают случаи, когда сетевые карты разных производителей начинают конфликтовать. Связано это, скорее всего, с тем, что некоторые производители добавляют недокументированные функции в свои устройства.

Перечень необходимого для закупки оборудования прилагается к плану сети и заверяется подписью заказчика.

## Подготовка проектной документации

Основной частью проектной документации является схема сети и спецификация на закупку оборудования и материалов.

На схеме должно быть отображено:

- собственно расположение помещений;
- расположение сервера и монтажного шкафа;
- места прокладки кабеля и количество проводов в каждом сечении (витая пара и телефонный кабель);
- тип кабельного канала, используемого в сечении;
- расположение розеток для подключения компьютеров с их сквозной нумерацией (впоследствии эта нумерация будет использоваться для маркировки кабелей);
- другая необходимая вам информация.

# Глава 5

# Прокладка сети

Основным документом, на основании которого будут вестись работы по прокладке сети, является схема. На схеме должно быть показано размещение розеток, план прокладки кабеля, сечение кабельного канала, расположение монтажного шкафа и др. Не лишним будет сказать, что проект, который вы составили, должен быть заверен подписью заказчика, чтобы потом все шишки не посыпались на вас.

Ранее мы также решили, что оптимальной будет прокладка сети на основе витой пары, поэтому все сказанное далее относится к ней.

## 5.1. Техника безопасности и правила монтажа

### Техника безопасности

Самым важным моментом в любых работах является техника безопасности. При такой ответственной процедуре, как прокладка сети, вопросы безопасности перекрывают несколько областей: электробезопасность, безопасность при проведении монтажных работ, техника безопасности при проведении высотных работ. Подробное описание каждого из выделенных моментов может занять не одну книгу. Здесь мы остановимся лишь на некоторых.

- Электробезопасность:
  - при проведении работ необходимо, по возможности, обесточить то место, где работы будут проводиться. Лучше это сделать с представителем электросетей;
  - используйте исправный инструмент. Изоляция на плоскогубцах, отвертках, кусачках не должна быть повреждена. Электрический инструмент, который вы используете, должен быть исправен. Если есть возможность, то используйте низковольтные инструменты. Например, переноска, которая будет светить на чердаке, должна питаться от на-

пряжения 36 В. Пониженное напряжение снижает риск смертельного исхода при поражении электрическим током;

- не работайте в одиночку. Если кто-то попал под действие электрического тока, первым делом необходимо его освободить от поражающего фактора: выключить рубильник, перерубить провод (инструмент, которым вы будете перерубать, должен быть изолирован), отдернуть пострадавшего за одежду.

- Безопасность при работах на высоте. С этим вы столкнетесь, если вам предстоит прокладка кабеля между крышами задний:
  - старайтесь работать дальше от края крыши;
  - при работе возле края крыши используйте монтажный пояс и страховку. Нельзя проводить работы в условиях плохой видимости (туман, сумерки, ночью, снег). Нельзя проводить работы во время дождя или после него (крыша скользкая, да и удовольствия мало).
- Техника безопасности при проведении монтажных работ. Здесь хотелось бы остановиться на работе с помощью лестниц.
  - перед началом работ лестницы должны быть проверены на отсутствие трещин и видимых повреждений;
  - не работайте с верхних ступенек;
  - лучше использовать страхующего, который поддержит лестницу;
  - при прокладке кабельного канала удобнее его придерживать с двух сторон, поэтому используйте вторую лестницу либо придерживающие приспособления.

Это только основные правила. Однако соблюдение их поможет уберечь вас от несчастного случая. Еще одним хорошим правилом является следующее: если вам кажется, что чего-то делать не стоит, то этого действительно не стоит делать. Перед любым действием всегда подумайте, безопасно ли это. Возможно, то же самое можно сделать и по-другому, с меньшей опасностью для жизни.

## Правила монтажа

Перед тем, как пойти дальше, необходимо усвоить некоторые правила монтажа кабельных систем. Поскольку у нас сеть на основе UTP-кабеля, то рассмотрим эти правила для них.

- Необходимо избегать лишней нагрузки на кабель, вызванной перекручиванием при протяжке или монтаже кабеля, его провисанием, туго затянутыми хомутиками.

- Чтобы избежать деформации и разрыва кабеля, нагрузка на 4-парный кабель не должна превышать 1 кг (110 Н).
- Радиус изгиба кабеля на витой паре не должен быть менее 4 диаметров кабеля.

## 5.2. Используемый инструмент

Монтаж кабельной системы — достаточно трудоемкая операция. Помимо обычных инструментов, таких как:

- молоток;
- дрель с перфоратором (для углублений в стенах);
- перфоратор (необходим для отверстий в стенах);
- сверла (лучше с победитовыми наконечниками);
- нож;
- ножницы;
- отвертка;
- плоскогубцы;
- кусачки,

вам потребуется также целый ряд специальных инструментов, без которых невозможна работа с UTP-кабелем:

- Монтажные клещи предназначены для монтажа разъемов прессовкой. Как правило, инструмент предназначен для прессовки разъемом RJ45 (это наша кабельная система) и RJ11 (такие разъемы используются в телефонии). Помимо собственно обжимной функции, монтажные клещи выполняют функции по резке кабеля. В некоторых моделях предусмотрен специальный нож для снятия внешней и внутренней изоляции (рис. 5.1).
- Инструмент для снятия внешней и внутренней изоляции (рис. 5.2). Основное его назначение — это снятие изоляции. Несмотря на то, что в некоторых клещах есть такая функция, работать с этим инструментом намного удобнее, поскольку он специально предназначен для этого. А в монтажных клещах это дополнительная функция.
- Еще один инструмент, без которого вам не обойтись, это ударный инструмент для разделки контактов. Бывает он без сменных головок (рассчитан только на один тип плинтов) и более продвинутый и дорогой вариант со сменными головками (рис. 5.3). Последний позволяет работать с различными типами плинтов (66, 110, KRONE). Сменные насадки могут быть с ножом для обрезки кабеля или без. Первый вариант для нас

более удобен, так как сразу отрезает оставшийся после вдавливания лишнюю часть кабеля.

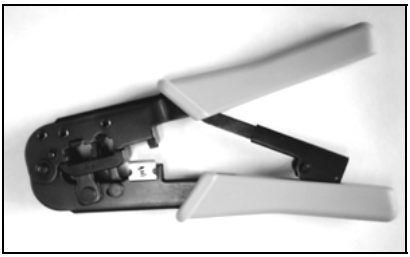


**Рис. 5.1.** Монтажные клещи

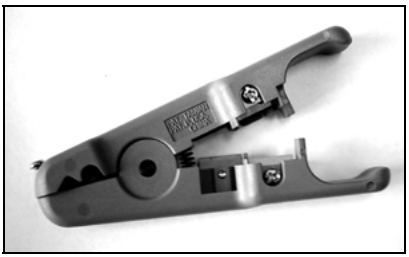


**Рис. 5.2.** Инструмент для снятия изоляции

**Рис. 5.3.** Ударный инструмент со сменной насадкой

- Последний из наиболее важных — это инструмент для проверки целостности цепей. Он позволяет проверить правильность обжимки и отсутствие повреждения кабеля (рис. 5.4).

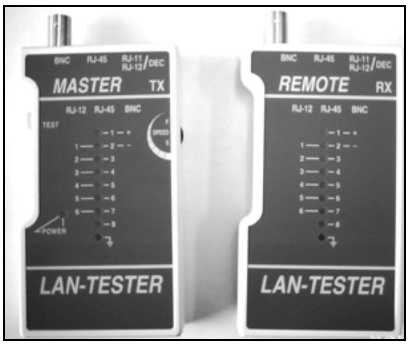



**Рис. 5.4.** Инструмент проверки целостности цепей

Существует еще целый ряд инструментов, применяемых при монтаже кабельных систем, однако, так как мы прокладываем сеть один раз, без них можно обойтись.

## 5.3. Монтажные шкафы. Типы шкафов. Подготовка и установка монтажного шкафа

Можно сказать, что монтажный шкаф в будущем станет центром вашей сети. Сюда будут стекаться кабели со всех концов здания, здесь они будут разведены на кросс-панели. Здесь также будут установлены коммутаторы.

Внутреннюю высоту шкафа (высоту его рабочего пространства) принято измерять в *юнитах* (обозначается U): один юнит составляет около 4,5 см. Внешние габариты шкафа проставляются в миллиметрах.

Также для шкафов применяется IP XX-код, характеризующий степень его защиты. XX — это две цифры. Кодировка этих цифр приведена в табл. 5.1.

***Таблица 5.1.*** *Коды защиты шкафов (IP)*

| 1 цифра | Расшифровка | 2 цифра | Расшифровка |
|---|---|---|---|
| 0 | Нет защиты | 0 | Нет защиты |
| 1 | Защита от твердых внешних объектов диаметром > 50 мм | 1 | Защита от вертикально падающих капель |
| 2 | Защита от твердых внешних объектов диаметром > 12,5 мм | 2 | Зашита от вертикально падающих капель при наклоне 15° в любую сторону |
| 3 | Защита от твердых внешних объектов диаметром > 2,5 мм | 3 | Защита от распыляемой воды |
| 4 | Защита от твердых внешних объектов диаметром > 1 мм | 4 | Защита от брызг |
| 5 | Пылезащитный | 5 | Защита от струй воды |
| | | 6 | Защита от мощной струи воды |
| | | 7 | Защита от временного погружения в воду |
| | | 8 | Защита от продолжительного погружения в воду |

Дорогие шкафы предназначены для размещения в них не только коммутационного оборудования, но и серверов, и источников бесперебойного питания. Шкафы для небольших сетей выпускаются размером поменьше и предусмотрены только для коммутационного оборудования. Зато стоят они намного дешевле.

Монтажные шкафы (в зависимости от типа) бывают настенные и напольные. Настенные шкафы (рис. 5.5, *а*) крепятся на стене на уровне человеческого роста, чтобы нижний край шкафа был бы на уроне около 1,5 м от уровня пола.

*а* *б*

**Рис. 5.5.** Настенный монтажный шкаф и кронштейны

Шкаф крепится с использованием специальных кронштейнов (рис. 5.5, *б*). В стене для этого проделываются отверстия. Они должны быть достаточной глубины, чтобы удержать вес шкафа, 10–15 см будет достаточно. Схема сборки шкафа приведена на рис. 5.6.

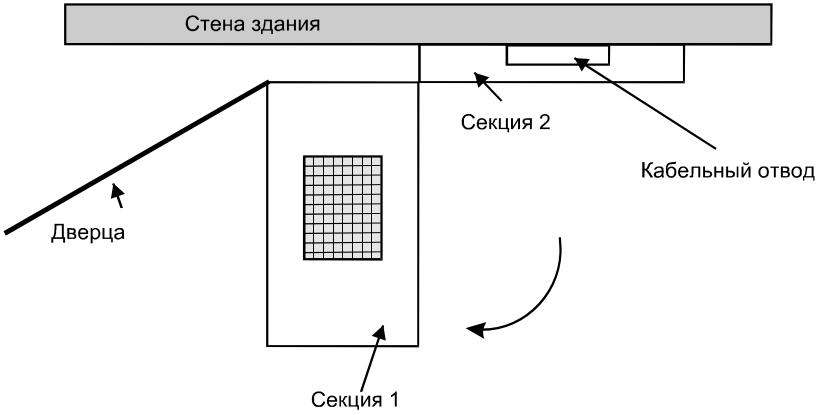



**Рис. 5.6.** Схема сборки шкафа

# 5.4. Кабельный канал, углы, Т-примыкания, заглушки

Итак, установка монтажного шкафа позади, самое время приступить к прокладке канала. Кабельный канал послужит несущей конструкцией вашей будущей кабельной системы. Кабельный канал и принадлежности приведены на рис. 5.7:

- Т-образное примыкание (1);
- внешний изменяемый угол (2);
- плоский угол (3);
- кабельный канал (без крышки) (4);
- внутренний изменяемый угол (5);
- заглушка (6);
- розетка (без крышки) (7).

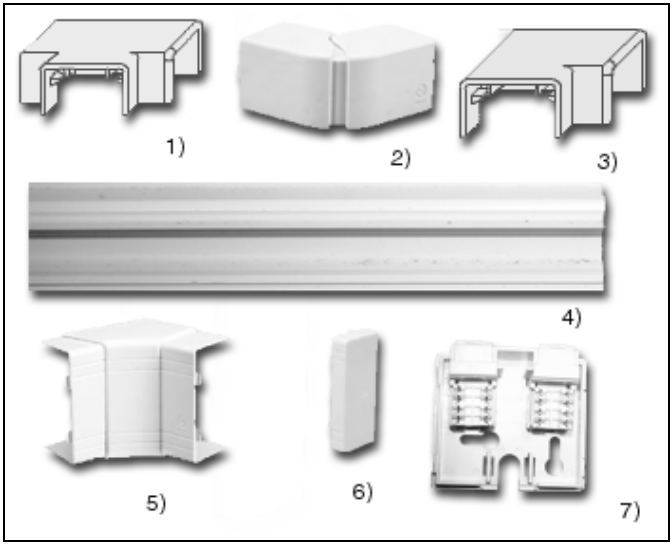



**Рис. 5.7.** Кирпичики каналообразования

Основным документом для прокладки кабельного канала служит документация проекта. Прокладку лучше начинать с канала большего диаметра, переходя к более мелким диаметрам.

**Примечание**

*Перед тем как приступить к работам, необходимо выяснить, есть ли скрытая проводка и где она проходит. Для обнаружения скрытой проводки используются специальные приборы — датчики электромагнитного поля.*

*Если их нет, можно пользоваться вторичными признаками: размещением розеток и распределительных коробок.*

1. Сверлим отверстия между помещениями. Делается это при помощи перфоратора. В некоторых зданиях капитальная стена при примыкании к внешней стене может иметь утолщение, если оно слишком большое, его лучше обогнуть (рис. 5.8).

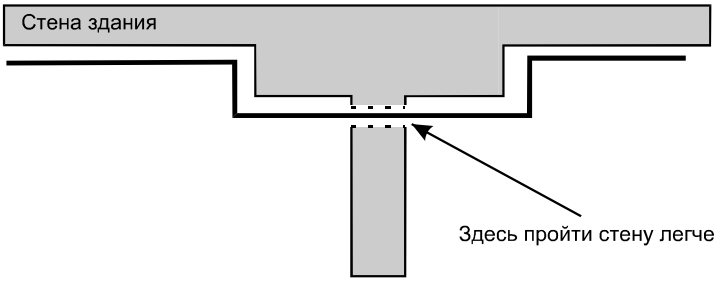



**Рис. 5.8.** Капитальная стена

2. Следующий этап — компоновка кабельного канала. Здесь вы должны прикинуть, какими кусками нарезать канал. Затем на канале делается разметка отверстий для крепежа канала к стене. Отверстия размечаются примерно через 0,7—1 м в зависимости от длины канала. Если полотно канала меньше, то должно быть минимум два отверстия.
3. В размеченных дырках сверлим отверстия в кабельном канале. Сверлить нужно без нажима, аккуратно. Пластик в кабельном канале достаточно мягкий, и его легко продавить.
4. Теперь разметим на стене, на какой высоте пойдет кабельный канал. Обычно он размещается на высоте около 80 см от пола. Однако, если профиль стены сложный (батареи отопления, проемы), то кабель можно поднять и выше, как это сделано в нашем случае. Размечайте аккуратно: если канал будет закреплен криво, то это будет плохо смотреться. Здесь же через отверстия в кабельном канале делаем засечки на стенах. В этих местах мы будем сверлить отверстия для крепления канала.
5. Отверстия для крепления канала должны быть около 5 см. Затем в эти отверстия вставляется пробка.

Канал крепится к стене путем ввинчивания шурупа либо самореза в пробку (рис. 5.9, *а*). Лучше использовать саморезы с большой площадью шляпки. Если площадь шляпки небольшая, можно проложить шайбу.

Между каналом и стеной можно проложить специальную клейкую таблетку, она позволит каналу надежнее держаться. Если стена деревянная или из мягкого материала, можно ввинчивать саморезы непосредственно в стену (рис. 5.9, *б*).

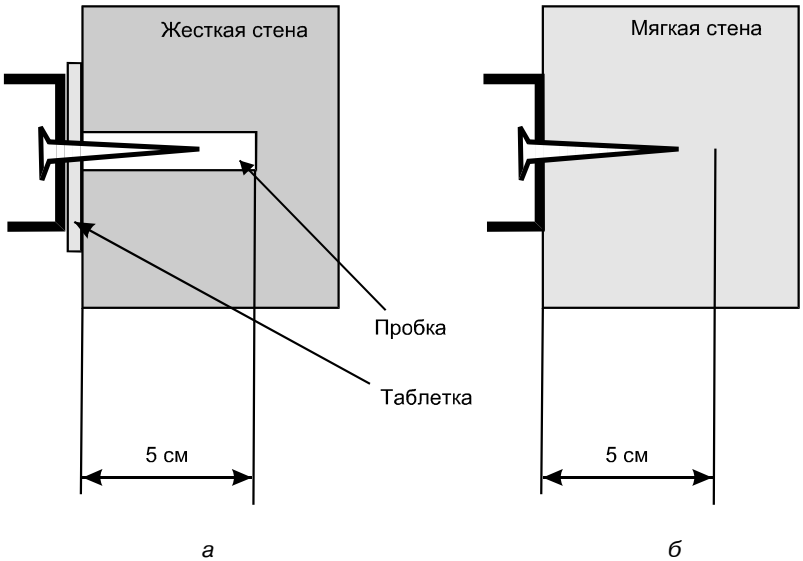



**Рис. 5.9.** Крепление канала

# 5.5. Подготовка и установка розеток

После того как кабельный канал закреплен, переходим к установке розеток (рис. 5.10). Для крепления розетка разбирается. С нее снимается крышка, вынимается пакетик с винтами, в некоторых модификациях розеток надо будет снять плату с разъемом RJ45.

На рисунке изображены:

- собственно розетка (1);
- таблетка для крепления к стене (2);
- крышка розетки (3);
- винты (4);
- хомутик (5).

Для того чтобы вид был аккуратным, необходимо, чтобы розетка крепилась встык с кабельным каналом. Не допускается наличие зазора между кабельным каналом и розеткой. Чтобы избежать этих зазоров, приложите розетку впритык к кабельному каналу и нанесите засечку на стену. Двойные, а также некоторые одинарные розетки могут крепиться двумя винтами. Тогда, соответственно, засечек делается две.

По нанесенной засечке заготавливаем отверстие (около 5 см), вставляем туда пробку. На стену наклеиваем таблетку (она идет в комплекте с розет-

кой). Наклеить таблетку здесь обязательно, так как площадь крепления у розетки маленькая. Клейкая таблетка создаст дополнительное сцепление со стеной.

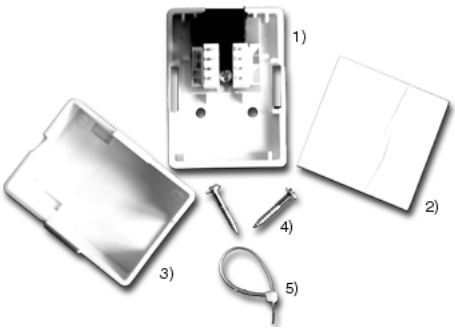



**Рис. 5.10.** Розетка RJ45

Затем на таблетку крепится розетка и завинчивается шуруп. Нижняя часть розетки должна сидеть плотно и не болтаться. Затем устанавливаем на место плату с разъемом RJ45 и крышку.

## 5.6. Прокладка кабеля

Кабель, как правило, поставляется в бухтах по 300 или по 100 м. Бухты по 300 м упакованы в картонные коробки (рис. 5.11). Бухты по 100 м пакуются в пластиковую оболочку.

**Рис. 5.11.** Бухта кабеля

Раскладка кабеля — достаточно ответственная операция. И в случае большого объема работ ее лучше разбить на части. Что касается нашего случая, то работу лучше проводить в четыре этапа, по количеству ответвлений

кабельного канала от коммуникационного центра. Каждый этап представляет собой законченную операцию с закрытием кабельного канала крышкой.

Прокладывать кабель лучше вдвоем. Вначале кабель выкладывается вдоль стены, к которой прикреплен кабельный канал. Прокладка идет по одной жиле.

Начинаем прокладку от монтажного шкафа. Постепенно выматываем кабель из бухты и выкладываем его вдоль стены. При вытаскивании кабеля из бухты надо быть осторожным и не прилагать усилий. Случается, что кабель в бухте путается, точнее сказать, образуются петли, которые задерживают выход кабеля в горловину (рис. 5.12). При применении силы его можно повредить. Также при прокладке кабеля важно избегать сильных изгибов. Изгиб с радиусом менее 4 диаметров кабеля может привести к повреждению кабеля.

**Рис. 5.12.** Место образования петель

По мере того, как кабель будет прокладываться, вы непременно подойдете к стене, разделяющей помещение. В ней уже должно быть заготовлено отверстие.

## Последовательность прокладки кабеля через отверстия в стенах

Прокладка кабеля через отверстия в стенах — разговор особый. Конечно, если стена тонкая, а отверстие достаточно большое, то затруднений это вызвать не должно. Но если отверстие достаточно протяженное, да к тому же еще и малого диаметра, то протянуть кабель не удастся. Необходимо вначале туда протянуть жесткую проволоку. Медная проволока здесь плохо подходит, так как она мягкая.

1. Надо взять стальную проволоку нужной длины. К одному концу проволоки при помощи изоленты, ниток или шпагата прикрепить кабель (рис. 5.13).

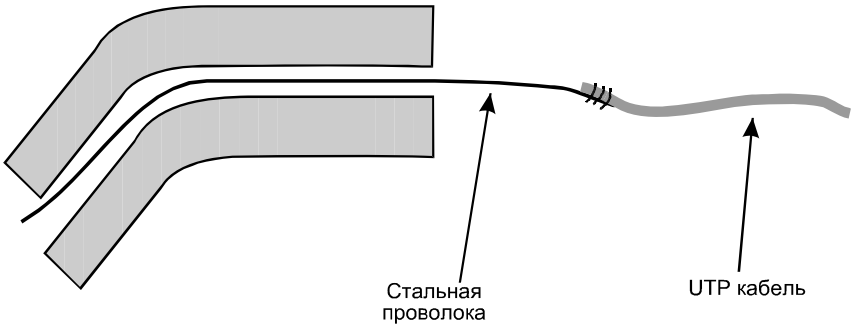



**Рис. 5.13.** Прохождение тонких отверстий

2. Мы вначале проталкиваем жесткую проволоку (при необходимости ее можно слегка изогнуть). Следом за собой проволока протянет и кабель. Протяжка кабеля сквозь отверстия должна производиться вдвоем. Один работник выдает кабель, а второй выбирает его с другой стороны стены.
3. После того, как жила кабеля протянута, она с запасом отрезается и маркируется с обоих концов специальными наклейками (можно подписать маркером или на бумаге и приклеить скотчем). Нумерация кабелей во всей кабельной системе должна быть сквозная, повторение номеров не допускается.
4. После того как весь кабель выложен вдоль стен, его стягивают кабельными стяжками (по 2, 4 или 8) в зависимости от сечения кабеля и числа кабельных жил в данном месте. На полу кабельные стяжки сильно не затягиваются. Окончательная затяжка пройдет уже после того, как мы уложим кабель в кабельный канал.
5. Теперь, когда кабель стянут, приступаем к укладке в кабельный канал. Для того чтобы кабель не выпадал из канала, после того, как вы его отпустите, необходимо поставить держатели. Для этого из крышки кабельного канала нарезаются держатели шириной 3–5 см. Они будут поддерживать кабель до того момента, когда вы закроете кабельный канал крышкой.
6. После того как кабель уложен в канал, устраните излишнее провисание кабеля и затяните стяжки. При затягивании стяжек не прикладывайте чрезмерных усилий. Стяжки должны быть затянуты плотно, но не сильно. Если их затянуть сильно, то можно повредить кабель.
7. Теперь аккуратно закрываем кабельный канал крышкой. Начинать надо с одного края, постепенно идя к другому. По мере того, как крышка установлена, снимайте держатели, которые вы использовали.

## Прокладка кабеля за потолочными панелями

В некоторых случаях приходится прокладывать кабель за потолочными панелями. При этом необходимо помнить, что ни в коем случае нельзя укладывать кабель непосредственно на панели, что может вызвать их повреждение. Смысла укладывать кабель в кабельный канал тоже нет.

Обычно кабель вешается на крюках. Для удобства работы используется специальный телескопический шест. Если его нет, то можно воспользоваться телескопической удочкой, на ее тонком окончании необходимо закрепить крюк. Шест (удочка) заносится за панель в сложенном состоянии и там раскладывается. Использование шеста значительно облегчает работу, поскольку потолок теперь не приходится вскрывать полностью.

# 5.7. Разделка розеток и патч-панели

Перед тем как приступить к разделке розеток и патч-панели, необходимо определиться со схемой разводки. Она должна быть единой во всей сети.

Существуют две схемы расположения проводов: EIA-T568A и EIA-T568B (рис. 5.14). Системы отличаются положением красного и зеленого кабелей. Поскольку все пары равноправны, то и оба эти стандарта тоже равноправны. Однако, если в сети не будет единого стандарта и часть контактов будет разделана по одному варианту, а часть по второму, то сеть не будет работать, и придется заново переделывать разделку контактов.

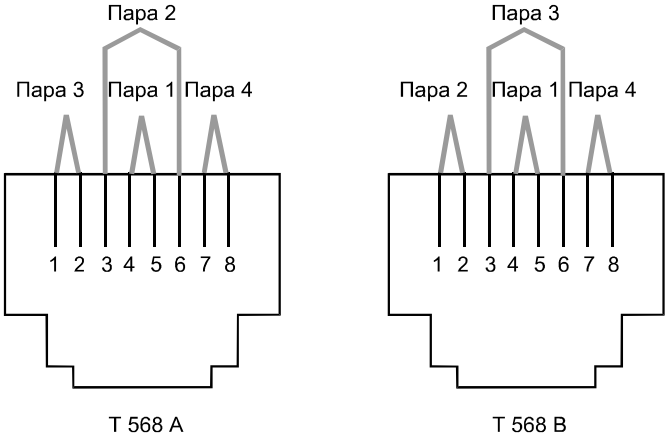



**Рис. 5.14.** Расположения проводов EIA-T568A и EIA-T568B

На розетках и патч-панелях варианты расположения проводов EIA-T568A и EIA-T568B обозначаются соответственно A и B. Итак, выбираем вариант разделки, пусть это будет B.

# Разделка розетки

Розетка нами уже прикреплена на стену, кабель уложен в кабельный канал, подведен к розетке и промаркирован. Осталось только разделать контакты в розетке.

Для разделки розеток нам понадобится устройство для зачистки кабеля, ударный инструмент для разделки контактов, отвертка, кусачки.

1. Снимаем изоляцию. Для этого инструментом для зачистки кабеля выполняем кольцевой надрез и снимаем срезанную часть изоляции.
2. Затем при помощи кусачек или шелковой нити в оболочке кабеля выполняем разрез и заворачиваем оболочку кабеля вниз.
3. Закрепляем кабель в розетке при помощи хомутика.
4. Теперь обратите внимание на цветовую маркировку клемм в розетке. Видите буквы А и В? Это те самые стандарты разделки EIA-T568A и EIA-T568B. Поскольку мы выбрали, что сеть у нас будет разводиться по стандарту В, то в соответствии с цветовой маркировкой, нанесенной под буквой В, разводим кабель. Кабель слегка утапливаем в углублении контактов.
5. Следующий шаг — непосредственно опрессовка. Для этого используем инструмент для разделки контактов. Он может быть без ножа или с ножом. Если инструмент с ножом, то нож должен располагаться во внешнюю сторону розетки. Аккуратно вставляем инструмент в направляющие и надавливаем под прямым углом к поверхности розетки. Инструмент издаст характерный щелчок. Повторяем операцию для всех контактов.
6. Если инструмент был без ножа или не все концы контактов нормально отрезались, их нужно отрезать теперь.
7. Теперь осталось только закрыть розетку.
8. На рис. 5.14 изображены некоторые этапы разделки розетки.

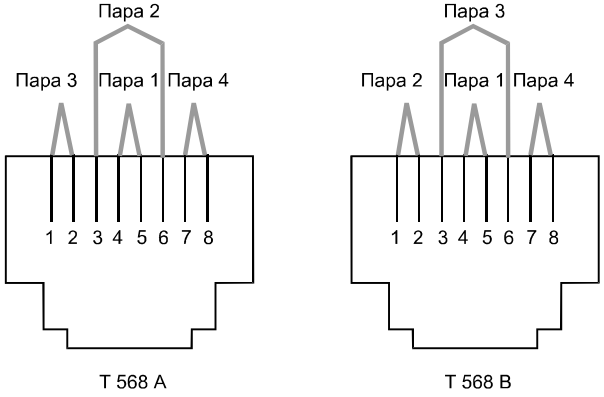
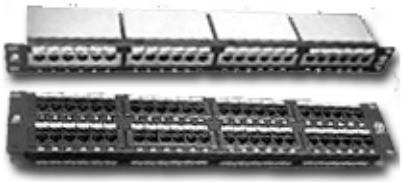



**Рис. 5.14.** Разделка розетки

# Разделка патч-панели

Разделка патч-панели во многом аналогична разделке розеток. Только повозиться придется подольше. Расположение контактов в патч-панели (а также их физическая конфигурация) зависит от производителя. Вообще патч-панели делятся на два больших класса: экранированные (рис. 5.15, *а*) и неэкранированные (рис. 5.15, *б*). В первом случае патч-панель закрыта защитным экраном.

Для разделки розеток нам понадобится устройство для зачистки кабеля, ударный инструмент для разделки контактов, отвертка, кусачки.



*а*

*б*

**Рис. 5.16.** Экранированная и неэкранированная патч-панели

Порядок разделки следующий:

1. Если это была экранированная патч-панель, снимаем защитный экран.
2. Приспособления для фиксации кабеля к патч-панели могут быть самой разной модификации. Это могут быть клипсы (в экранированных панелях) (рис. 5.17, *а*) и зажимы различной формы (в неэкранированных) (рис. 5.17, *б*).

   Снимаем элементы фиксации кабеля, если это необходимо.

*а* *б*

**Рис. 5.17.** Элементы крепления кабеля в панелях

3. Отрезаем кабель до нужной длины и зачищаем его, как это было с розетками. Кабель надо отрезать так, чтобы остался достаточно большой свободный ход.

4. Очищаем кабель от верхней изоляции. Для этого, как и раньше с розетками, делаем круговой надрез с помощью инструмента для зачистки кабеля.
5. Если патч-панель закрытая, то при помощи шелковой нити внутри кабеля или кусачек делаем вдоль изоляции разрез и отводим внешнюю изоляцию вниз. Если это открытая патч-панель, то внешнюю изоляцию просто срезаем.
6. Заводим подготовленный кабель к нужному гнезду патч-панели и закрепляем его кабельной стяжкой.

   Здесь лучше соблюдать определенные правила подключения. Например, первый кабель на первое гнездо патч-панели. Затем берется кабель со следующим порядковым номером и заводится в следующее гнездо патч-панели. Крайне нежелательно хватать кабели вразнобой и также их крепить к патч-панели, поскольку потом возникнут трудности с маркировкой.
7. В соответствии с выбранным стандартом (в нашем случае B) распределяем жилы кабеля в углубления на гребенке контактов. Жилы слегка притапливаем. Контактные группы могут различаться в зависимости от патч-панели (рис. 5.17).

Опрессовываем кабель при помощи инструмента для разделки контактов. Инструмент вставляем в направляющие и надавливаем под прямым углом к поверхности. Надавливать нужно плавно, нормируя усилие. Инструмент издаст характерный щелчок.

Повторяем операцию для всех контактов в гребенке. После чего отрезаем концы контактов, если они остались.

Повторяем операцию для оставшихся разъемов патч-панели.

Закрываем, если это была экранированная патч-панель.

Переносим маркировку с кабеля на лицевую сторону. Сделать это можно маркером, краской или специальными клеящимися номерками (рис. 5.18).

**Рис. 5.18.** Нумерация на патч-панели

Закрепляем патч-панель в монтажном шкафу. Кабель жгутируется и закрепляется к поверхности монтажного шкафа.

После того как монтаж первой патч-панели закончен, можно приступить к разделке оставшихся кабелей.

В последнюю очередь в монтажный шкаф устанавливаются коммутаторы. В принципе порядок установки патч-панелей и коммутаторов может быть любым. Но и здесь лучше придерживаться определенной последовательности. Например, патч-панели устанавливаются в верхнюю часть шкафа, а коммутаторы в нижнюю.

## 5.8. Монтаж патч-кордов

Наличие патч-панели позволяет оперативно перебросить тот или иной компьютер на другой вход коммутатора. Но для того, чтобы соединить разъем коммутатора и разъем патч-панели, необходим проводник. Этот проводник — патч-корд. Он же соединяет розетку RJ45 на стене с разъемом сетевой карты компьютера. Патч-корды нормируются по длине: 0,5; 1; 1,5; 2; 3; 5; 10; 15 м.

Патч-корд представляет собой проводник (витую пару) с двумя разъемами RJ45 на концах. Для коммутации компьютер-коммутатор используется прямой патч-корд. Для коммутации компьютер-компьютер или коммутатор-коммутатор используется кроссовый патч-корд. Справедливости ради, стоит заметить, что многие из современных коммутаторов могут распознавать соединение коммутатор-коммутатор, так что их можно соединить и прямым патч-кордом.

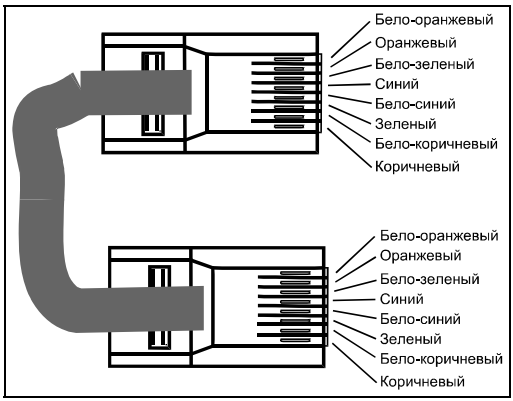



**Рис. 5.19.** Прямой патч-корд

Изготовить патч-корд можно и самому. Для этого нужно сделать следующее:

1. Снять изоляцию с конца кабеля. При помощи инструмента для зачистки делается кольцевой надрез и снимается верхняя часть кабеля (3—4 см).
2. Расплести жилы кабеля и выровнять их.

3. Выстроить жилы кабеля в следующей последовательности:
   - бело-оранжевый;
   - оранжевый;
   - бело-зеленый;
   - синий;
   - бело-синий;
   - зеленый;
   - бело-коричневый;
   - коричневый.

Теперь нужно выровнять и расплющить жилы. Приставьте рядом коннектор RJ45 и прикиньте, насколько нужно подкоротить жилы для того, чтобы оплетка кабеля зашла под фиксатор.

Отрезать их под прямым углом, оставив нужную длину. Вставляем кабель в разъем. Каждая жила должна попасть в свой паз и дойти до упора. Это хорошо видно, так как разъем прозрачный. Если какая-то жила не дошла или оплетка кабеля не попала под защелку, процесс повторите сначала.

Вставить коннектор RJ45 в гнездо монтажных клещей и произвести опрессовку. Кромки контактов прорежут изоляцию и обеспечат надежный контакт. А фиксатор закрепит кабель.

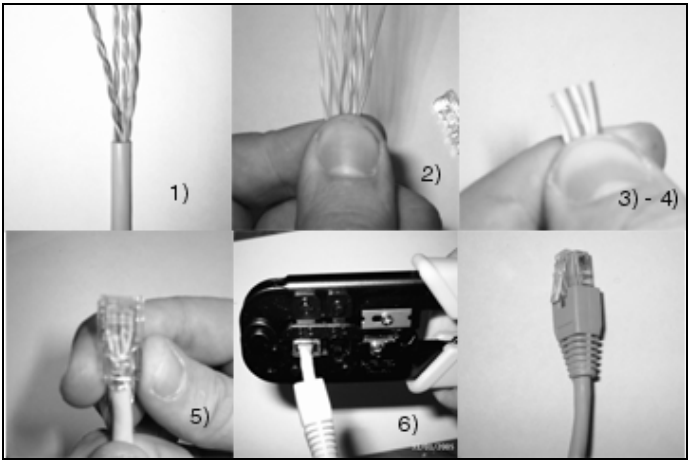



**Рис. 5.20.** Подготовка патч-корда

Компьютер-компьютер или коммутатор-коммутатор соединяются при помощи кроссового патч-корда (кросс-провода).

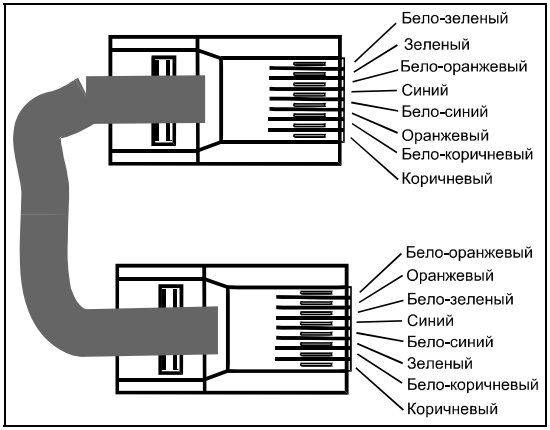



**Рис. 5.21.** Кроссовое соединение

Обжим производится точно так же, но вот схема разводки проводов другая. Здесь зеленая и оранжевые пары меняются местами.

# 5.9. Прокладка кабеля по воздуху

Иногда приходится прокладывать кабель не только внутри одного здания, но и между зданиями. Например, офисы могут быть расположены в близлежащих зданиях.

Уже упоминалось, что безопасность — важнейший вопрос при проведении работ по прокладке кабельной системы. Особенно это важно при прокладке кабеля между зданиями, поскольку приходится работать на высоте.

При прокладке кабеля по воздуху необходимо предусмотреть, чтобы он проходил на достаточной высоте. Это защитит его от вандалов и случайного разрыва проезжающим транспортом. Минимальная высота закрепления должна быть 4—5 м. Это примерно высота потолка второго этажа.

Продумайте прокладку кабеля так, чтобы он:

- проходил подальше от линий электропередачи. Во-первых, этого делать нельзя по требованиям безопасности, во-вторых, вызовет наводки в кабеле и сбой в работе сети;
- проходил подальше от деревьев и возвышенных объектов. С дерева ваш кабель можно прекрасно срезать. Кроме того, ветка или дерево может сломаться и повредить кабель.

Для того чтобы на кабель не создавались дополнительные нагрузки под действием собственного веса, его необходимо прикрепить к струне. Струна представляет собой стальную проволоку. Кабель крепится к ней через каждые 50—70 см. Для этого он приматывается к струне тонкой проволокой, например цветной проволокой из телефонного кабеля.

После того как кабель подготовлен, следует перебросить его с одной крыши на другую. Наиболее безопасный вариант следующий:

1. С обеих крыш при помощи груза спускаются концы шпагата (можно сложить его в несколько раз, если шпагат тонкий).
2. Затем эти концы внизу связываются вместе и поднимаются так, чтобы шпагат был не сильно натянут.
3. На том здании, откуда ведется кабель, необходимо кабель за стальную струну прикрепить к шпагату.
4. Затем аккуратно без рывков и сильного натяжения кабель перетягивается с одной крыши на другую.
5. После этого необходимо закрепить кабель на краях крыши. Крепление производится только за струну, чтобы основная нагрузка легла на нее.
6. Кабель не должен быть сильно натянут. Во-первых, это сделать не так-то просто, во-вторых, при понижении температуры тела, как известно, сжимаются, поэтому если кабель натянут сильно, в мороз он может лопнуть.

Есть и второй вариант прокладки — с помощью лука (арбалета) и стрел. Правда, если вы не Вильгельм Телль, можете попасть кому-нибудь в глаз или куда еще. Велик риск травмы, и приятного в этом мало. Так что стоит подумать, перед тем как натягивать тетиву.

# Глава 6

# Сборка сервера

## 6.1. Система клиент-сервер

Система клиент-сервер предполагает наличие в сети особого узла —сервера. Сервер предоставляет свои ресурсы (дисковое пространство, web-сервис и др.) остальным узлам и управляет сетью через специальные сервисы. В общем случае сервер разрешает или не разрешает доступ к тому или иному файлу или каталогу, разрешает символьные имена в сети (DNS-служба), конфигурирует компьютер для работы в сети (DHCP-сервис) и многое другое. Поскольку на сервер возложено достаточно много обязанностей и обращаются к нему сразу несколько узлов, то ясно, что это должен быть самый мощный компьютер в сети. В этой главе мы рассмотрим, из чего состоит сервер, как подобрать оптимальную конфигурацию сервера за приемлемые деньги и, наконец, как собрать сервер своими руками.

## 6.2. Материнская плата

Материнская плата (MB, Mother Board) — один из самых важных компонентов компьютера, она объединяет все остальные компоненты в единую согласованно работающую систему (рис. 6.1). Наряду с процессором *(см. разд. 6.2)*, материнская плата определяет общую производительность компьютера.

У материнской платы много характеристик, но одна из них является определяющей. Это Chipset (дословный перевод — набор микросхем), на основе которого она реализована. Chipset определяет возможности материнской платы: тип поддерживаемого процессора, тип, объем и количество модулей памяти, жестких дисков, видеокарт и др. Однако даже на одном чипсете материнские платы разных моделей могут сильно отличаться.

На материнской плате присутствует несколько типов разъемов:

- Разъемы шины PCI (Peripheral Component Interconnect — соединение внешних компонентов) служат для установки периферийных устройств, таких как модем, звуковая карта, сетевая карта и др. Шина PCI — самая популярная шина в настоящее время. Тактовая частота ее работы 33/ 66 МГц, скорость передачи данных соответственно до 133 Мбит/с и 266 Мбит/с.
- Шина AGP (Accelerator Graphics Port — ускоренный графический порт). Само название говорит о том, что эта шина предназначена для подключения видеоадаптера к материнской плате. Необходимость ввода нового порта продиктована увеличением потока данных между процессором и видеокартой в современных приложениях. При работе в обычном режиме пропускная способность этого порта практически не используется.
- USB (Universal Serial Bus — универсальная последовательная шина) представляет собой интерфейс для подключения внешних устройств. Допускается подключение до 127 устройств к одному USB-каналу. Как правило, на плате бывает 2—4 разъема USB. Можно купить отдельно и установить дополнительный контроллер USB в PCI-слот. Появится еще 4—6 портов.

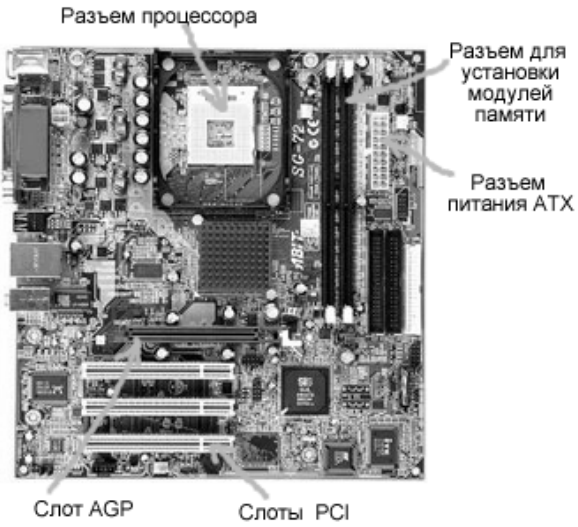



**Рис. 6.1.** Материнская плата

Помимо всего прочего, современные материнские платы могут содержать в себе очень много дополнительных возможностей, таких как встроенная звуковая и видеокарта, сетевая карта.

Материнские платы со встроенными контроллерами (звук, видео, сетевая карта) используются для клиентских компьютеров невысокого ценового диапазона. Наличие "всего в одном" полезно сказывается на цене и легкости сборки и установки.

Еще одним важным параметром, который характеризует материнскую плату как устройство, является форм-фактор. Форм-фактор определяет размеры материнской платы, вид разъемов питания и размещение некоторых ее элементов. Для конечного пользователя при покупке важны только размеры, поскольку они обусловлены размером корпуса компьютера, в который материнская плата устанавливается.

Первые компьютеры имели форм-фактор AT, современные компьютеры имеют форм-фактор ATX. В настоящее время осуществляется плавный переход на стандарт BTX. Все стандарты плохо совместимы между собой. В свое время были переходные модели AT/ATX, сегодня появляются переходные ATX/BTX. Но это не стоит внимания. На данный момент технология BTX еще плохо отработана, и при выборе оборудования внимание стоит обращать на ATX-комплектующие.

## 6.3. Процессор

Процессор — это сердце компьютера (рис. 6.2). Назначение процессора состоит в выполнении математических и логических операций. Если вы хотите вычислить 2 + 2, то выполнением операции сложения занимается именно процессор. На сегодняшний момент на рынке процессоров представлены два основных игрока:

- компания Intel, процессоры Pentium (верхний сегмент рынка) и процессоры Celeron (сегмент, рассчитанный на дешевые компьютеры);
- компания AMD, процессоры Athlon, Athlon XP (верхний сегмент рынка), Duron, Sempron (сегмент, рассчитанный на дешевые компьютеры).

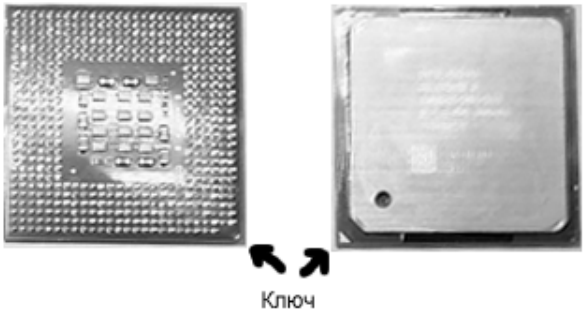



**Рис. 6.2.** Процессор

Основными характеристиками процессора являются его ядро и тактовая частота. Ядро представляет архитектуру построения процессора: объем кэш-памяти, длину конвейера и другие тонкости. Тактовая частота определяет скорость выполнения операции в процессоре.

У многих сложилось впечатление, что на производительность влияет только частота. На самом деле это не так. Архитектура процессора тоже много значит. Например, процессоры Athlon имеют маркировкой не реальную частоту, а так называемые рейтинги. Рейтинг представляет собой сравнение процессора Athlon с процессором Pentium. Если производительность одинакова, то для Athlon устанавливается тактовая частота Pentium, хотя его реальная тактовая частота работы ниже. Выигрыш происходит за счет архитектуры. Не все так однозначно, во многих приложениях Athlon отстает от заявленных характеристик.

Кроме того, у Athlon по сравнению с Pentium больше тепловыделение, а значит, создаются проблемы теплоотвода. Да и к тому же разница в цене этих процессоров уже не так заметна, как раньше.

## 6.3. Оперативная память

Оперативная память (ОЗУ — оперативное запоминающее устройство) предназначена для хранения используемых в настоящий момент данных (просто данных и фрагментов программного кода).

Модуль памяти — это прямоугольная пластина с микросхемами памяти и разъемом для подключения к материнской плате (рис. 6.3). Число микросхем памяти на модуле может быть различно и определяется как размером одной микросхемы, так и емкостью модуля в целом.

Модули памяти отличаются по напряжению питания и алгоритму работы. Стандартными на сегодняшний день являются небуферизованные модули с напряжением питания 3,3 В.

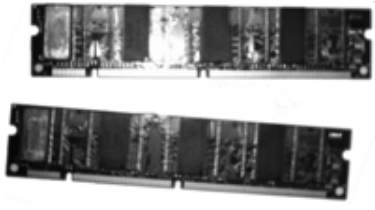


**Рис. 6.3.** Модуль памяти

В подавляющем большинстве компьютеров сегодня применяется DRAM (Dynamic RAM). Память этого типа имеет несколько разновидностей. Однако в модулях памяти применяются следующие типы памяти:

- SDRAM;
- Rambus;

- DDR SDRAM;
- DIMM DDR SDRAM.

Эти типы памяти отличаются не только скоростью работы, но и алгоритмом, поэтому разъемы для них электрически и механически не совместимы.

Для характеристики производительности DIMM DDR SDRAM используется ее пропускная способность — до 3,2 Гбайт/с. Чем выше пропускная способность, тем лучше.

## 6.4. Жесткие диски и RAID

Жесткий диск (HDD — Hard Disk Drive) представляет собой основное хранилище информации на компьютере (рис. 6.4). В отличие от оперативной памяти (ОЗУ), данные с жесткого диска не пропадают после выключения питания.

**Рис. 6.4.** Жесткий диск

Наиболее важными показателями диска являются его емкость и скорость работы.

Емкость определяет, сколько данных вы можете записать на жесткий диск. Так как запись в жестком диске ведется на магнитные пластины, то емкость определяется количеством пластин и емкостью одной пластины. При маркировке выдается суммарная емкость всех пластин. Современные HDD имеют емкость в сотни Гбайт.

Второй показатель — это скорость работы HDD. А точнее, скорость передачи данных с поверхности диска к процессору либо непосредственно в память. Это комплексный параметр, который определяется следующими факторами:

- объем кэш-памяти жесткого диска;
- скорость оборотов шпинделя, на котором закреплены магнитные пластины;

- интерфейс подключения к материнской плате;
- размещение данных по поверхности жесткого диска.

Кэш-память представляет собой буфер, через этот буфер проходит обмен между собственно диском и процессором либо памятью. Чем больше кэш, тем больше производительность жесткого диска (до определенных пределов). Кроме того, чем больше кэш-память HDD, тем выше его цена.

Скорость оборотов шпинделя измеряется в оборотах в минуту. Чем быстрее вращается шпиндель, тем быстрее считывающая головка получит доступ к нужному на диске месту. Однако большая скорость оборотов означает больший нагрев.

Жесткие диски отличаются также интерфейсом подключения. Он может быть самый разный, вплоть до USB. Рассмотрим некоторые интерфейсы.

## IDE/ATA

IDE/ATA допускает подключение до двух устройств на один канал. Устройства именуются master и slave. Наименование условное и служит для того, чтобы как-то различать устройства между собой. На современных материнских платах обычно существует два разъема IDE. Однако есть модели материнских плат, поддерживающие до четырех разъемов (2 IDE и 2 E-IDE). Как правило, на разъемах E-IDE есть функция организации аппаратного RAID-массива.

В настоящее время существует несколько протоколов работы по интерфейсу ATA (табл. 6.1):

***Таблица 6.1.*** *Протоколы IDE/ATA*

| Протокол | Скорость передачи данных |
|---|---|
| UDMA 33 | 33,3 Мбит/с |
| UDMA 66 | 66,6 Мбит/с |
| UDMA 100 | 100 Мбит/с |
| UDMA 133 | 133,3 Мбит/с |

Если на одном канале два устройства с разной скоростью, то работа идет на скорости наименьшего устройства. Например, типичной ошибкой является установка жесткого диска на один шлейф с CD-ROM. CD-ROM работает на скорости 33,3 Мбит/с, и ваш "шустрый" HDD, который мог бы работать на 100, вынужден работать на 33,3 Мбит/с. Более того, они разделяют среду передачи во времени, так что если активно работают два устройства, скорость падает еще в два раза.

**Внимание!**

*Никогда не устанавливайте CD-ROM и HDD на один шлейф.*

Плюсами является хорошее соотношение цена/производительность. Главным недостатком жестких дисков с интерфейсом IDE/ATA является их низкая скорость по сравнению со SCSI-дисками.

## SCSI

Еще одним интерфейсом является SCSI. Стандарт SCSI разработан для высокопроизводительных жестких дисков. Увеличение эффективной пропускной способности в SCSI было достигнуто увеличением тактовых частот и изменением протоколов. На одной шине может находиться несколько устройств, при этом потеря в скорости будет минимальной.

Главным достоинством SCSI является высокая производительность и масштабируемость. Минусы — высокая цена и сложность настройки.

## SATA

Еще один вариант — это жесткие диски SATA. Главное отличие — это более высокая пропускная способность (150 Мбит/с). На одном шлейфе может быть только одно устройство, поэтому делить пропускную способность кабеля ни с кем не придется. Кроме того, количество проводов в кабеле сокращено с 40 (80) IDE-кабеля до 4. Компенсация в потере лишних проводов осуществляется увеличением тактовой частоты передачи. Несмотря на то, что скорость повысилась до 150 Мбит/с, реально IDE/ATA и SATA по производительности отличаются не сильно. Сказываются ограничения внутренней архитектуры жестких дисков: скорость оборотов шпинделя, объем кэш-памяти и др. Справедливости ради, стоит сказать, что и в цене разница несущественная.

## RAID

С жесткими дисками также связано понятие RAID — Redundant Array of Index Disk, или матрица независимых дисковых накопителей с избыточностью. Массивы RAID используются для увеличения скорости и надежности жестких дисков. Существует несколько вариантов RAID:

- RAID 0 — используются два жестких диска. Запись осуществляется одновременно на два диска, причем часть данных записывается на один диск, а часть на другой, в результате возрастает эффективная скорость работы.
- RAID 1 — используются два жестких диска. При работе данные записываются на оба диска одновременно, дублируя друг друга. Такая комбинация повышает надежность хранения данных, поскольку при выходе из строя одного диска, данные на втором диске сохранятся.

- RAID 0 + 1 представляет собой комбинацию RAID 0 и RAID 1. При этом используется четыре жестких диска, два — в режиме RAID 0, два — в режиме RAID 1. Таким образом, достигается высокая надежность и скорость работы.

## 6.5. Подбор конфигурации сервера

Если процессор — сердце компьютера, то сервер — сердце локальной сети. Производительность сервера определяется теми задачами, которые на него возлагаются. Если это файловый сервер (сервер, предоставляющий свое дисковое пространство), то требования к нему относительно невысокие. Если это сервер приложений (сервер, на котором выполняются запущенные пользователем программы), то здесь уже надо подумать о более дорогом варианте. Например, каждый пользователь в Apache (Http-сервер) при подключении отнимает около 20 Мбайт оперативной памяти. Вроде бы немного, а если этих пользователей 10, то уже 200 Мбайт. Помимо этого, есть еще и другие задачи.

Перегрузка сервера выльется в то, что пользователи будут получать данные с задержкой, и нормальной эту работу будет назвать сложно. Экономия в этом месте выйдет боком. Поэтому, по возможности, на сервере деньги экономить не стоит (в разумных пределах).

Говоря о конфигурации сервера, прежде всего, имеют в виду тактовую частоту процессора, объем оперативной памяти и жесткие диски. Минимальной отправной точкой должна стать конфигурация P-III 600 МГц и 256 Мбайт оперативной памяти. Машина подобного плана может справиться с 6—9 пользователями, а возможно, и больше. При выборе соотношения цен необходимо исходить из того, что основная часть стоимости придется на материнскую плату, процессор и оперативную память.

Жесткий диск выбирается, исходя из хранимого объема информации, причем рассчитанные показатели лучше взять с коэффициентом запаса 1,5—2.

В меньшей степени стоит обращать внимание на видеокарту, поскольку нагрузка на этот участок у сервера минимальна. Вполне допустима видеокарта от 32 Мбайт оперативной памяти и даже меньше.

Какой выбрать процессор, от Intel или AMD — дело вкуса. В спорах о производительности этих процессоров было сломано немало копий. Во многом выбор определяется личными предпочтениями. Остановимся на процессорах Intel.

Форм-фактор выбираем ATX, на данный момент он хорошо отработан в плане технологии, а значит — надежен.

Будем просматривать процессоры самого нижнего диапазона в направлении повышения цены: когда увеличение стоимости относительно тактовой частоты станет неприемлемым, можно остановиться.

Конечно, это весьма условная методика, и возможно, ее придется подкорректировать. При корректировке обязательно обращаем внимание на такие параметры, как кэш и частота шины процессора.

Приведенный далее пример отражает ценовую ситуацию на момент написания книги, однако методика выбора не зависит от цены.

Выбираем Pentium 4: 2.8 ГГЦ Socket 478, 1024 кэш, частота шины 800, box (коробочная версия, процессор поставляется с вентилятором). Переходим к выбору материнской платы. Для создания сервера начального уровня подойдут материнские платы на основе чипсетов 865 и 875. Здесь при выборе лучше ориентироваться на именитых производителей: ASUS, Abit, Chiantech и др. Наиболее хорошо для этого подходит материнская плата ASUS P4800—E Deluxe i875 S478.

При выборе оперативной памяти обратите внимание на ее объем и ее пропускную способность. Примерно оценить объем памяти можно из соотношения на 25 пользователей 512 Мбайт памяти. Для нашего случая достаточно будет 1 Гбайт. Выбираем два модуля по 512 Мбайт: PC-3200 от Samsung.

При выборе жестких дисков первостепенным фактором является объем хранимой на сервере информации. Объема в 120 Гбайт достаточно, хотя могут быть и исключения. Выбираем два диска Seagate 120 Гбайт ST3160023A. У этого диска кэш 8 Мбайт. Два диска мы выбрали неслучайно. Помните RAID 1 с параллельной записью для надежности? На сервере надежность очень важна. Помимо этого, необходимо взять внешний RAID-контроллер, подойти может практически любой.

Переходим к выбору корпуса. Несмотря на то, что подбирать конфигурацию мы начинали с процессора, корпус тоже немаловажная деталь. Особое внимание уделяем качеству блока питания и системы охлаждения. Обязательно должны быть дополнительные вентиляторы. Корпус должен быть полноразмерный ATX. К выбору также надо подойти и с эстетических позиций. Выбор оставляю за вами.

Следует не забыть также:

- ❒ две сетевые карты на 1000 Мбит/с;
- ❒ дисковод флоппи-дисков;
- ❒ CD-RW, видеокарты;
- ❒ клавиатуру и мышь.

## 6.6. Источники бесперебойного питания

Весьма немаловажным вопросом является обеспечение надежного питания сервера (ИБП), поскольку сбой работы вызовет прямую потерю данных. Если же не поставить источник бесперебойного питания на Switch, то сер-

вер работать будет, но пользователи к нему доступ не получат. А если сбой произойдет во время передачи данных, то они все равно будут потеряны.

Источники бесперебойного питания нормируются по емкости, которая измеряется в VA. Чем больше этот показатель, тем больше может ИБП поддерживать работоспособность системы при сбоях электричества. Вполне достаточно 15–20 минут на то, чтобы выйти из приложений и сохраниться. Статья на ИБП в смете расходов проекта тоже должна присутствовать.

# 6.7. Сборка сервера

Несмотря на то, что сборка сервера звучит громко, ничего сложного в ней нет. Если вы хотя бы раз собирали компьютер, то вам вообще можно не читать главу.

Первое, что надо помнить при сборке, — все должно делаться без усилий. Это касается винтов: они должны входить без усилий и в тоже время не болтаться. Вся периферия (CD_RW, FDD, HDD) должна крепиться на 4 винта.

При установке комплектующих, если есть возможность, лучше их максимально разнести, это облегчит тепловой режим их работы.

В *разделе 6.5* мы подобрали достаточно мощный сервер. В этом разделе мы продемонстрируем сборку на примере сервера более простой конфигурации: процессор P III 1200, 512 Мбайт ОЗУ, 40 Гбайт жесткий диск. Сервер такой конфигурации вполне может обслужить до 15–20 пользователей. Итак, приступим:

1. Снимаем с корпуса боковые крышки. Они могут быть либо на винтах (отвинчиваем), либо на замках (открываем) (рис. 6.5).

**Рис. 6.5.** Корпус со снятыми боковыми крышками

2. Вынимаем из корпуса фурнитуру: винты, шнур питания и пр.
3. Вынимаем из упаковки материнскую плату и устанавливаем ее в корпус (рис. 6.6). При установке материнской платы необходимо обратить внимание на то, что для большинства моделей придется сменить стандартную планку для разъемов на прилагаемую с материнской платой. Также необходимо помнить о том, что материнская плата должна быть закреплена на все винты.

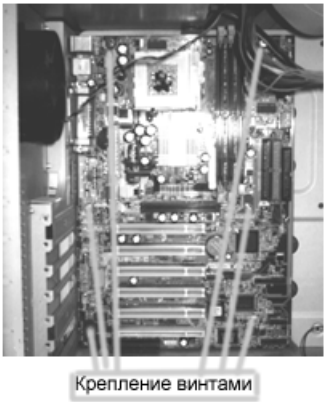



**Рис. 6.6.** Корпус с установленной материнской платой

4. Устанавливаем жесткий диск. Жесткий диск устанавливается на специальное посадочное место. Закрепляем на 4 винта (рис. 6.7).

**Рис. 6.7.** Установка жесткого диска

5. Устанавливаем дисковод флоппи-дисков и привод CD-RW. Их также закрепляем на 4 винта.
6. Устанавливаем процессор и вентилятор на процессор (рис. 6.8):
   - Для установки процессора поднимаем рычажок, слегка отведя его в сторону.

- Устанавливаем процессор. При этом надо следить за правильностью установки в посадочные гнезда, для этого на процессоре и в посадочном гнезде (Socket) есть направляющие.
- Смазываем процессор термопастой и устанавливаем вентилятор.
- Подключаем вентилятор к материнской плате (рис. 6.9).

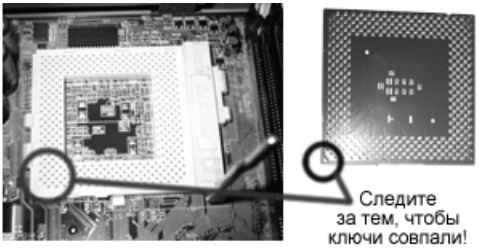



**Рис. 6.8.** Установка процессора

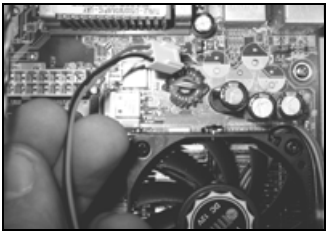


**Рис. 6.9.** Подключение вентилятора

7. Устанавливаем сетевую и видеокарту (рис. 6.10). Сетевая карта устанавливается в IDE-разъем, видеокарта устанавливается в AGP-разъем.

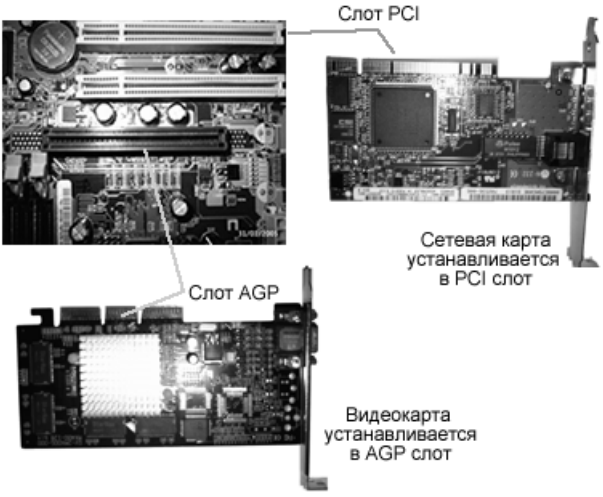



**Рис. 6.10.** Установка сетевой и видеокарты

8. Подключаем к материнской палате разъемы питания ATX (если у вас Pentium 4, то этих разъемов 2 или 3) и кабели управления и индикации (Power, Reset, Hdd led, Power led) (рис. 6.11). Для правильности подключения этих кабелей воспользуйтесь руководством для материнской платы. При подключении питания обращайте внимание на направляющие. Если перепутаете, то материнская плата выйдет из строя.

**Рис. 6.11.** Подключение разъемов

9. Подключаем дисковод флоппи-дисков, CD-RW и жесткий диск к материнской плате при помощи кабелей. Для того чтобы не напутать с подключением кабелей, нужно помнить следующее: ножки в разъемах на материнской палате и на устройствах пронумерованы (рис. 6.12). Кабель имеет одну жилу, которая отличается по цвету, именно она должна быть подключена к первой ноге.

Справедливости ради, стоит сказать, что большинство серьезных производителей выпускают кабели со специальными ключами, благодаря которым подключить кабель неправильно очень сложно.

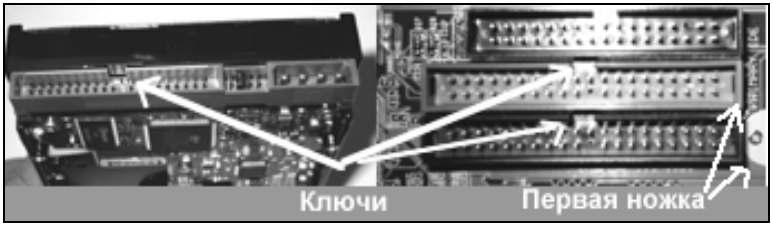



**Рис. 6.12.** Особенности подключения IDE

10. Подключаем питание к внешним устройствам. Здесь также необходимо помнить о ключах. Если вы, приложив усилие, деформируете ключи и

перепутаете правильность подключения, то можете сказать вашему жесткому диску или любому другому устройству: "До свидания!".

11. Если не было дополнительного вентилятора, его нужно установить (рис. 6.13).

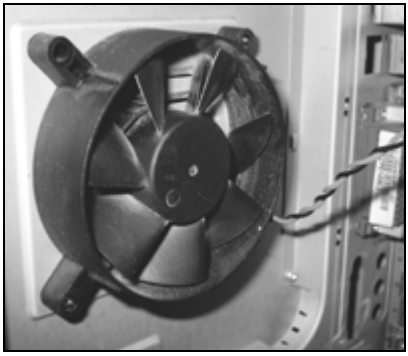


**Рис. 6.13.** Дополнительный вентилятор

12. Закрываем корпус. Сервер собран.

# Глава 7

# Тестирование сети и поиск неисправностей

## 7.1. Причины плохой работы сети

Когда сеть работает хорошо, она незаметна для пользователя. Однако такое бывает не всегда. Сбои бывают однократные, периодические и постоянные. Надо выделить время и провести тестирование сети с целью выявления причин, вызвавших сбой. Лучше не доводить до ситуации, когда нормальная работа становится затруднительной либо полностью невозможной. А это значит, что все шишки — на администратора, и проблемой заниматься все равно придется.

Прежде всего, необходимо представить, какого рода дефекты могут возникать в сети:

- дефекты кабельной системы — кабеля, розеток, наводки от электрических приборов и машин;
- дефекты активного сетевого оборудования (коммутаторов) и сетевых карт;
- неправильно выбранная архитектура сети либо нехватка ресурсов;
- перегруженность сервера;
- ошибки программного обеспечения.

Выявить сразу все ошибки бывает сложно, так как многие из них имеют схожие симптомы. В этой главе мы рассмотрим вопрос выявления ошибок первой и второй группы.

Приводимые методы могут применяться комплексно и не обязательно в изложенной последовательности. Возможно, что для выявления неисправности будет достаточно одного метода, а возможно — и всех не хватит. Необходимо знать, какой метод что проверяет, и на основании поведения сети выбирать. Если сразу сказать сложно, то лучше просто по очереди проверить сеть всеми методами.

# 7.2. Как правильно тестировать сеть

Ответ на этот вопрос однозначный — полноценное тестирование сети возможно только при помощи специализированного оборудования. Можно выделить две категории такого оборудования:

- анализаторы физического уровня;
- анализаторы более высоких уровней модели OSI.

Кабельная система тестируется только анализаторами физического уровня. Наиболее известный из приборов проверки сетей — это кабельный сканер (рис. 7.1). В условиях, когда вы профессионально не занимаетесь прокладкой сетей и их последующим тестированием, такое оборудование вам вряд ли понадобится.

**Рис. 7.1.** Кабельные сканеры

Рассмотрим тестирование сети при помощи подручных и недорогих средств. Это может быть LAN-tester любой модификации, поддерживающий тестирование кабелей, обжатых под разъем RJ-45. Например, LAN-tester L-100 (рис. 5.4). Еще одним аппаратом, который может помочь, является обычный тестер с функцией звукового тестирования кабеля (рис. 7.2). Смысл ее в том, что через тестируемый кабель прогоняется сигнал звуковой частоты, и по характеру звучания оценивается качество канала. Об этом мы еще поговорим далее.

**Рис. 7.2.** Тестер

# 7.3. Проверка правильности разделки контактов

Проверить правильность разделки контактов можно визуально либо при помощи тестера.

Визуальный метод заключается в последовательной проверке обжатия всех розеток, патч-панелей и патч-кордов. Достоинство метода — относительная простота и дешевизна. Недостаток — громоздкость (приходится разбирать много оборудования) и большая вероятность ошибки: проверить одну две розетки несложно, но по мере увеличения числа проверенных розеток, внимательность падает.

Проверка при помощи LAN-tester диагностирует правильность соединения, но не оценивает качественные характеристики канала. Прибор состоит из двух частей: master и remote. При последовательном присоединении проводников по индикации на обеих частях можно судить о правильности разделки кабеля. Однако для использования в нашем случае метод придется немного модифицировать. Причина в том, что, с одной стороны, у нас кабель обжат в розетку, а с другой — в патч-панель. Выход из этой ситуации простой, дополнительно понадобятся три патч-корда: один подлиннее (от 3 до 10 м) и два по 2 м. Обратите внимание, что все патч-корды должны быть прямыми. В противном случае можно запутаться.

Выполним следующие действия.

Тем кабелем, что длиннее, мы соединяем розетки на рабочих местах.

Патч-корды покороче вставляем в соответствующие этим розеткам гнезда на патч-панели.

Свободные концы вставляем в тестер и запускаем тест. Схема приведена на рис. 7.3.

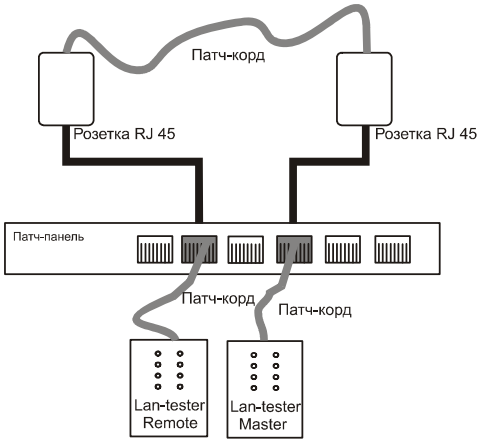



**Рис. 7.3.** Проверка при помощи LAN-tester

Проверяя попарно все розетки, можно выявить те, где неправильно разделаны контакты в розетке или патч-панели. Необходимо проверить и то, и другое. Вскрывайте первой розетку, это проще сделать. После исправления ошибки снова проверьте систему.

## 7.4. Проверка надежности соединения в контактах

LAN-tester проверяет только правильность разводки контактов, не проверяя качество соединения в контактах. На практике может быть такой случай, когда вследствие плохого обжатия или дефекта кабеля, механических дефектов розетки или патч-панели имеется неустойчивый контакт. Выявить такой контакт можно при помощи тестера. Поскольку попасть в разъем патч-панели тестером достаточно сложно, да и контактные группы можно погнуть, необходимо заготовить две одинаковые панели. С одной стороны этой панели будет разъем RJ-45, со второй стороны — контакты (или контактные площадки) с подписанными номерами (рис. 7.4).

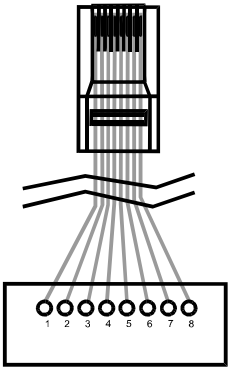



**Рис. 7.4.** Схема панели

Для проверки необходимо сделать следующее:

1. Как и в предыдущем случае, соединяем две розетки длинным патч-кордом.
2. В соответствующие разъемы вставляем наши самодельные панели.
3. Тестер переводим в режим звуковой проверки кабеля.
4. Касаясь по очереди одноименных контактов, по качеству звукового сигнала оцениваем качество контактной пары.

Проверяя попарно все контактные группы, можно выявить те, где качество контакта плохое. Это выразится в искаженном звуковом сигнале. Необходимо повторно обжать данные контактные группы.

Наличие плохого контакта может быть вызвано неправильной вставкой коннекторов в разъемы или дефектами разъемов. Если после повторного обжатия все повторилось, проверяйте уже непосредственно кабель, минуя розетки.

Подобным методом можно выявить плохие соединения. Хотя проверка идет на звуковой частоте, а частота сети значительно выше, некоторые неисправности все же выявляются.

Недостатком данного метода является громоздкость операций. Поэтому его применение ограничено. Схема испытаний приведена на рис. 7.5.

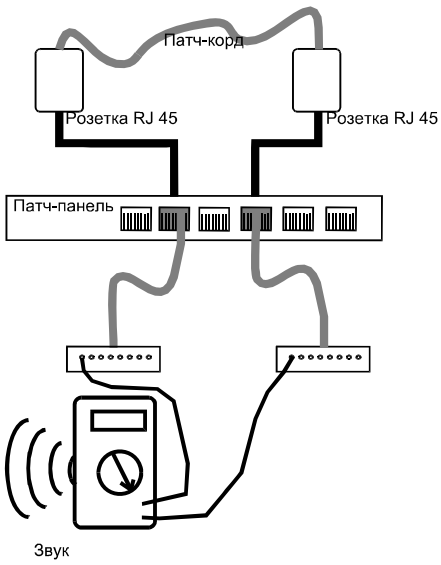



**Рис. 7.5.** Схема проверки на слух

# 7.5. Проверка методом исключения

Вкратце суть данного метода состоит в анализе поведения сети и выявления некорректно работающего узла или участка сети. Пожалуй, это один из первых методов, к которому нужно прибегнуть, если проблемы начинают возникать в уже работающей сети. Этот метод подразумевает несколько приемов.

## Поочередное отключение узлов от сети

Суть заключается в том, что вы по очереди отключаете узлы и наблюдаете работу сети. Под отключением понимается отключение от розетки локальной сети. Если после отключения какого-то узла восстанавливается нормальная работоспособность, то значит, проблемы были в этом узле.

Первое, что стоит сделать, это заменить сетевую карту, патч-корд и проверить качество соединения в розетках.

О сетевых картах стоит казать особо. Несмотря на то, что все они подчиняются единому стандарту, некоторые производители добавляют расширенные протоколы, которые призваны улучшить работу. На деле все выходит с точностью до наоборот. Недокументированные протоколы начинают конфликтовать друг с другом, в результате сеть начинает работать со сбоями. Для такого случая характерно возникновение трудно прогнозируемых сбоев. Бывает, месяц работает нормально, а бывает, по несколько раз в день вылетает. Поэтому лучше выбирать сетевые карты одного производителя.

## Прокачка через сетевую карту большого объема информации

Суть данного метода в том, что через сетевую карту прогоняются большие объемы данных.

Последовательность действий:

1. Берем большой объем данных. Идеально подходит DVD-фильм.
2. Архивируем его и перекачиваем на другой компьютер, а затем назад. Перекачку можно осуществлять любым файловым менеджером или непосредственно в Windows. На рис. 7.6 показана перекачка файлов на сервер и обратно в файловом менеджере Total Commander.

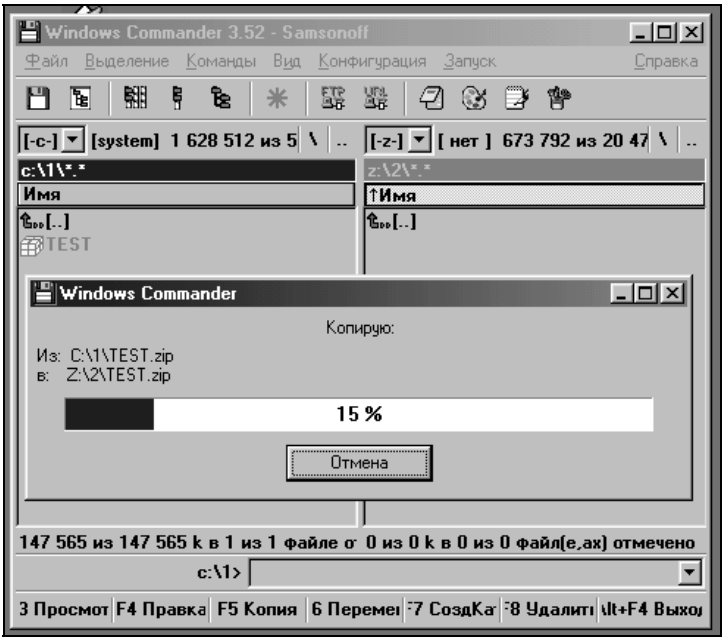



**Рис. 7.6.** Перекачка файлов в Total Commander

3. Повторяем это действие несколько раз.
4. Разархивируем файл и проверяем его на наличие CRC-ошибок. При нормальной работе сети их быть не должно.

Поскольку сбои в передаче данных могут быть вызваны не только работой сетевой карты, то для чистоты эксперимента все повторяется после замены на заведомо исправный образец.

# 7.6. Поиск неисправного порта коммутатора

Причиной некорректной работы сети может быть коммутатор. Если коммутатор выходит из строя, то останавливается вся сеть. Итак, если все компьютеры в сети перестали "видеть" друг друга, то, скорее всего, не работает коммутатор. На практике чаще выходит из строя один из портов коммутатора. При этом остальная часть сети этого даже не замечает. Сеть пропадает только для одного компьютера.

Если один из узлов не видит сети, а сетевая карта заведомо исправная, то, возможно, не работает порт коммутатора.

Для проверки, так ли это, необходимо переключить компьютер на заведомо исправный порт. Если работоспособность после такой манипуляции восстановилась, то причина была в выходе из строя порта коммутатора.

Небольшая хитрость состоит в том, что, если заранее заготовить кросс-кабель, то при помощи этого кабеля непосредственно на патч-панели можно будет коммутировать компьютеры друг с другом (рис. 7.7). В этом случае можно будет провести проверку двух узлов независимо от остальной части сети. И в дальнейшей диагностике опираться на них, как на заведомо исправные узлы.

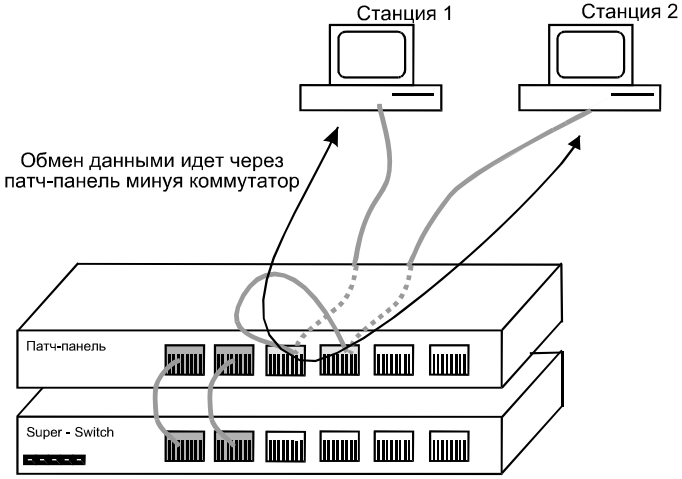
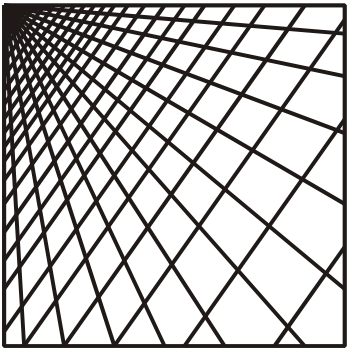



**Рис. 7.7.** Проверка узлов на себя

# 7.7. Мониторинг сети

Важной составляющей частью повседневной работы является мониторинг состояния сети. Мониторинг осуществляется как на основе данных активного сетевого оборудования, так и на основе встроенных программ диагностики сети.

При этом необходимо помнить, что коллизии в сети в среднем не должны превышать 20 %. А в пике (но не более, чем на минуту с последующим спадом) допускается уровень коллизий до 40–50 %. Причина в том, что с ростом числа коллизий пропускная способность канала снижается.



# ЧАСТЬ III

# Linux-сервер своими руками

# Глава 8

# Установка сервера

## 8.1. Выбор дистрибутива

Сегодня сложно найти человека, который не слышал о Linux. И то, что Linux отвоевывает у Microsoft позиции в серверах для небольших организаций, ни у кого не вызывает сомнения. Однако если вы остановились на Linux как на операционной системе для вашего сервера, вам предстоит еще решить, на каком собственно дистрибутиве Linux остановиться. В табл. 8.1 приведено несколько дистрибутивов, это далеко не полный перечень, но он должен дать представление о том, что у вас очень широкий выбор.

***Таблица 8.1.*** *Некоторые дистрибутивы Linux*

| Дистрибутив | Описание |
|---|---|
| Red Hat | Самый известный и широко распространенный дистрибутив, ставший стандартом. Наверное, это и самый дорогой продукт. Правда, вы платите за техническую поддержку, так как дистрибутив можно найти или скачать бесплатно. Горячая линия технической поддержки работает в России по бесплатной линии. Сайт **www.redhat.ru** |
| Caldera | Компания Caldera (**www.caldera.com**) предлагает продукт двух типов. Один ориентирован на использование в рабочих станциях (eDesktop), другой в серверах (eServer). Отличительной особенностью является минимализм входящих пакетов при сохранении функциональности. Данный продукт можно рекомендовать в корпоративном применении |
| Mandrake | Мандрейк является клоном Red Hat. Позиционируется больше как настольная система, **www.linux-mandrake.com** |
| ASPLinux | Продукты отечественной компании ASPLinux основаны на дистрибутиве Red Hat. Сотрудники ASPLinux переработали, перевели и адаптировали большинство исходных пакетов, **www.ASPLinux.ru** |

Сразу стоит оговориться, что практически любой дистрибутив можно перенастроить под решение любых задач, начиная от простой настольной системы и заканчивая полноценным сервером. Как правило, в поставку входят все необходимые для этого пакеты. Но выбор сделать надо, и тут я бы порекомендовал поддержать отечественного производителя. Во-первых, русифицирована большая часть программного обеспечения. Во-вторых, компания знает нужды нашего рынка. Например, в дистрибутиве ASPLinux одна из возможных конфигураций называется "1С Сервер" и предназначена для организации сервера баз данных 1С. Этот пример наглядно показывает, на кого стоит ориентироваться.

В принципе, как уже говорилось, можно взять любой дитрибутив. Однако, если вы недостаточно уверены в себе или просто хотите использовать готовое решение, то лучше остановиться на дистрибутиве ASPLinux 7.3 Server Edition. Как сказано в описании продукта, это стабильное решение для построения надежно защищенной сети малого и среднего предприятия, включающей почтовые и web-серверы, печать и базы данных. Централизованная настройка большинства сервисов и программ, входящих в дистрибутив, существенно уменьшает сложность их установки, использования и администрирования. При установке можно выбрать один из нескольких вариантов:

- конфигурация маршрутизатора и межсетевого экрана — не требовательна к ресурсам, предназначена для интеграции независимых сегментов сети. Для управления доступом и сбором статистики можно воспользоваться как стандартными средствами ОС Linux, так и специализированным пакетом **NetAMS**, обеспечивающим гибкую настройку параметров доступа, а также возможность генерации отчетов об использовании сетевых ресурсов;
- конфигурация сервера рабочей группы, включающая набор сервисов, необходимых для организации работы небольшого офиса, а именно систему электронной почты, печати, файл-сервер и др. Включает службы ftp, http, mail, dns, dhcp, samba;
- конфигурация сервера данных для работы с программами семейства 1С — готовое решение, позволяющее работать в программах семейства "1С:Предприятие" с использованием ОС ASPLinux на сервере данных. Конфигурация включает сервер электронного ключа защиты HASP, используемого системами 1С. На рабочих станциях необходимо установить утилиту для терминальной работы, например rdesktop (свободно распространяемая) или winconnect (коммерческая);
- конфигурация сервера данных с возможностью доступа по протоколам ftp, nfs, smb и другим достаточным для пользователей, использующих практически любую ОС на рабочих станциях.

Наиболее лаконичным для нас будет выбор сервера рабочей группы, тем более что выбранную конфигурацию всегда можно оптимизировать, добавив необходимые пакеты или убрав лишние. Об этом уже во время установки.

Хочется также склонить вас к покупке данного продукта, а не простому переписыванию его у друзей. Сделав покупку, вы получаете техническую поддержку, а также подробную документацию в виде книг, что очень полезно.

Оговоримся также, что установку мы делаем на сервер. Поэтому установка будет идти на "чистый" (нет других ОС) жесткий диск, и операционная система будет единственной.

## 8.2. Установка сервера

Мы определились, с дистрибутивом (ASPLinux 7.3 Server Edition) и вариантом установки — сервер рабочей группы. Теперь приступим к установке.

1. В BIOS сервера необходимо поставить загрузку с CD-ROM. О том, как это сделать для каждой конкретной материнской платы, смотрите документацию, поставляемую вместе с материнской платой.
2. Затем вставляем первый установочный диск и перезагружаем компьютер. После перезагрузки будущий сервер начнет загружать программу установки с CD-ROM (рис. 8.1).

```
mice: PS/2 mouse device common for all mice
md: md driver 0.90.0 MAX_MD_DEVS=256, MD_SB_DISKS=27
md: Autodetecting RAID arrays.
md: autorun ...
md: ... autorun DONE.
NET4: Linux TCP/IP 1.0 for NET4.0
IP Protocols: ICMP, UDP, TCP
IP: routing cache hash table of 512 buckets, 4Kbytes
TCP: Hash tables configured (established 4096 bind 4096)
NET4: Unix domain sockets 1.0/SMP for Linux NET4.0.
RAMDISK: Compressed image found at block 0
Freeing initrd memory: 1624k freed
EXT2-fs warning: checktime reached, running e2fsck is recommended
VFS: Mounted root (ext2 filesystem).
Freeing unused kernel memory: 136k freed
init started:  BusyBox v0.51 (2003.02.12-05:45+0000) multi-call binary
gzipped pcitabe NOT found
/bin/probe - no PCMCIA detected.
found 0 SCSI cards
Total memory detected: 61M
Searching ASPLinux CDROM...
Starting install...
Running detect script...
```

**Рис. 8.1.** Начальная загрузка программы установки

3. Если программа установки сможет корректно определить тип видеокарты, то установка пойдет в графическом режиме, если нет, то в псевдографическом. Окна этих режимов имеют одинаковую смысловую

нагрузку, а шаги аналогичны, поэтому остановим свое повествование на графическом режиме. Первым окном будет окно выбора языка установки. По умолчанию выпадает английский язык. Выбираем русский и жмем кнопку **Далее**. Хочется также сказать, что выбор языка установки никак вас не ограничивает в дальнейшем выборе языка системы (рис. 8.2). Выбор языка системы осуществляется в конце установки. Его можно выбрать любым, независимо от выбранного языка установки.

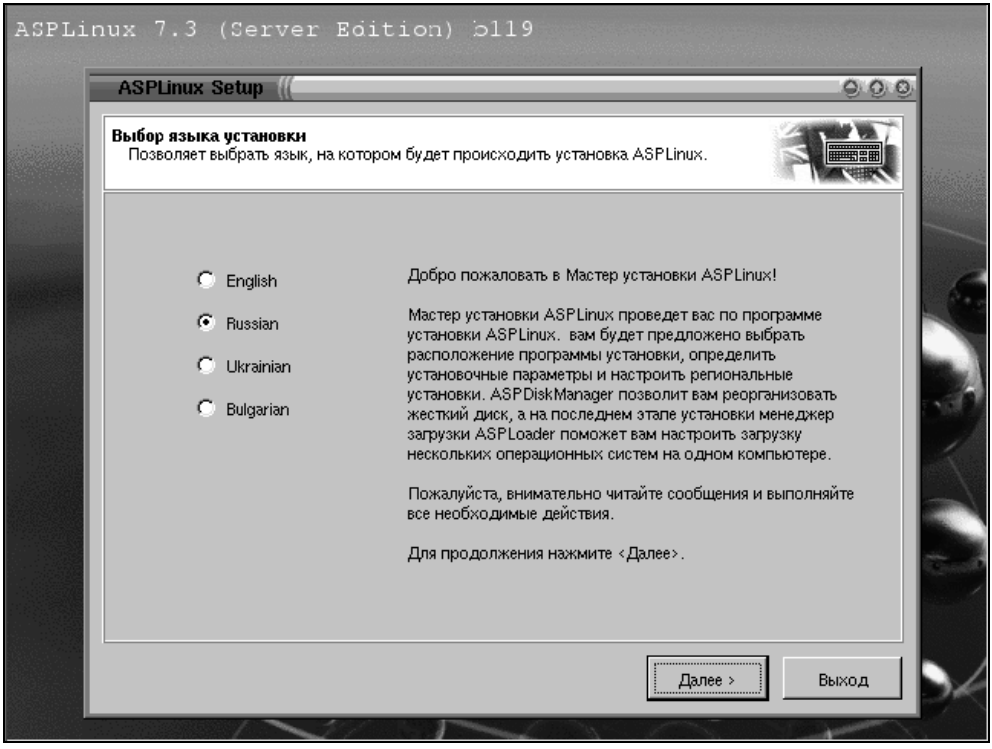



**Рис. 8.2.** Выбор языка системы

4. Следующее окно это выбор типа мыши (рис. 8.3). Как правило, мышь определяется правильно. Даже если выбранный системой тип мыши не соответствует действительности его можно изменить. И нажав кнопку **Применить** зафиксировать сделанные изменения.

   Также стоит сказать о том, что в Linux приятнее работать с трехкнопочной мышкой, поскольку некоторые функции доступны только по нажатию третьей кнопки мыши. Поэтому, если у вас двухкнопочная мышь, выберите эмуляцию трехкнопочной. В этом случае третья кнопка будет работать по одновременному нажатию левой и правой кнопки мыши.

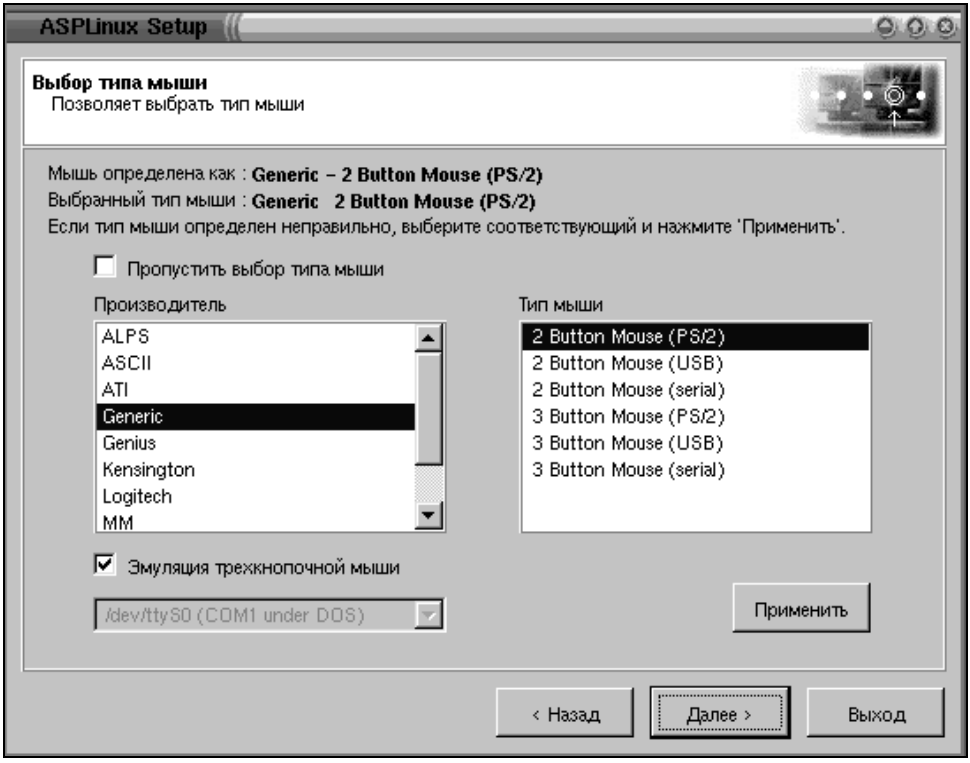



**Рис. 8.3.** Выбор типа мыши

Полезным будет поставить эмуляцию даже в случае верного определения типа трехкнопочной мыши, поскольку бывают случаи, что третья кнопка не определяется системой. В этом случае вам и поможет эмуляция этой самой кнопки.

5. Вам предстоит выбрать из двух типов установки (рис. 8.4). Быстрая установка предполагает автоматическое конфигурирование и установку предопределенного набора пакетов: минимальное вмешательство пользователя, что не совсем удобно.

   Гораздо более привлекательным является второй вариант — выборочная установка. Ветка выборочной установки позволяет пользователю принять непосредственное участие в процессе установки системы и управлять дисковыми разделами. Второй вариант, помимо большей гибкости, что наиболее важно, имеет и большую образовательную пользу, поскольку на этапе установки вы начинаете знакомиться с особенностями организации и работы Linux.

Следующим шагом является выбор типа носителя (рис. 8.5). У нас альтернатив нет, поскольку это будет единственная машина с Linux в сети, и образов Linux у нас нет. Устанавливать будем с CD-ROM.

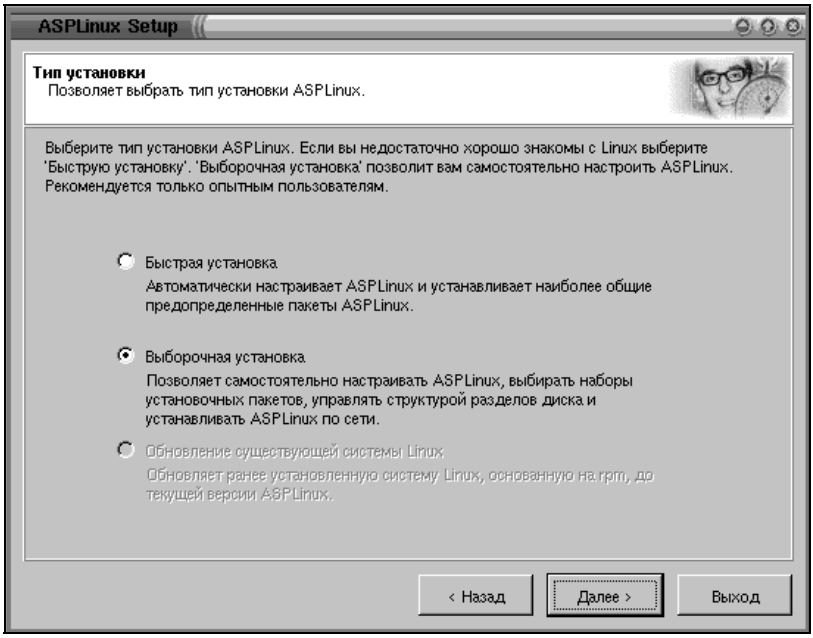



**Рис. 8.4.** Выбор типа установки

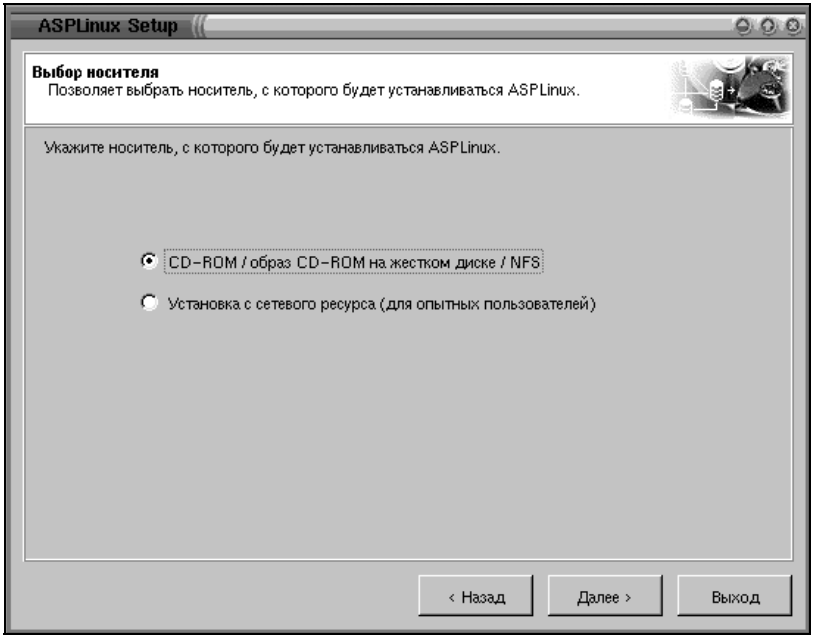



**Рис. 8.5.** Выбор типа носителя

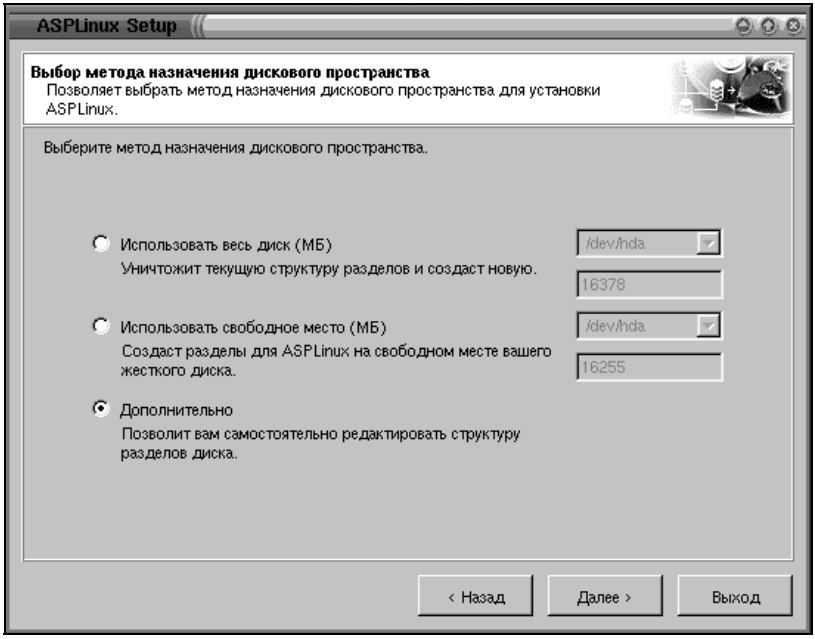



**Рис. 8.6.** Выбор метода назначения дискового пространства

6. В окне на рис. 8.6 вам предстоит выбрать, как будет использоваться жесткий диск. Первый вариант стирает все данные на диске. Поэтому, если вы используете пробную установку на машине с уже имеющейся системой, то ни в коем случае не выбирайте этот пункт. Linux с дисками работает быстро, глазом моргнуть не успеете, как потеряете информацию. Наиболее приемлемым является выбор дополнительного режима, который позволит вам самостоятельно редактировать дисковое пространство для своих нужд. При планировании дискового пространства, необходимо помнить следующие моменты:

- В большинстве случаев достаточно выделения трех разделов:
  - корневого раздела объемом около 2 Гбайт;
  - раздела файла подкачки swap (его размер равен удвоенному размеру ОЗУ, на машинах с большим объемом ОЗУ это правило не играет роли);
  - домашних каталогов пользователей.
- Логичным является выбор файловой системы Ext2 как наиболее надежной и прошедшей проверку временем.
- Для каждой отдельной задачи лучше выделить отдельный раздел жесткого диска. Например, для бухгалтерии /buch, для конструкторского бюро /kb.

7. В предыдущем окне мы выбрали **Дополнительно**, теперь у нас появится окно редактирования разделов (рис. 8.7). Для начала редактирования выбираем диск в окне выбора диска. Наименование дисков и разделов мы подробнее рассмотрим в *главе 9*. Здесь остановимся кратко:
   - hda — master на первом канале IDE;
   - hdb — slave на первом канале IDE;
   - hdc — master на втором канале IDE;
   - hdd — slave на втором канале IDE.

   Соответственно, если жесткий диск один, то выбираем /dev/hda. Так же будет в случае, если у вас аппаратный рейд-контроллер. В этом случае, несмотря на то, что дисков два, для системы они будут выглядеть одним диском.

   Для создания раздела необходимо выделить строку с надписью **Free** и нажать кнопку **Создать**, которая станет активной. Для редактирования или удаления раздела выберем его и нажмем соответствующую кнопку. При необходимости возможно создание программного RAID-массива.

**Рис. 8.7.** Окно ASPDiskmanager

8. Первым делом создаем корневой раздел (рис. 8.8). Размер выберем с запасом 4 Гбайт. Файловую систему выбираем Ext2. В поле точки монтирования выбираем / (корневой каталог) и нажимаем **OK**. Действия повторяем для всех остальных разделов.

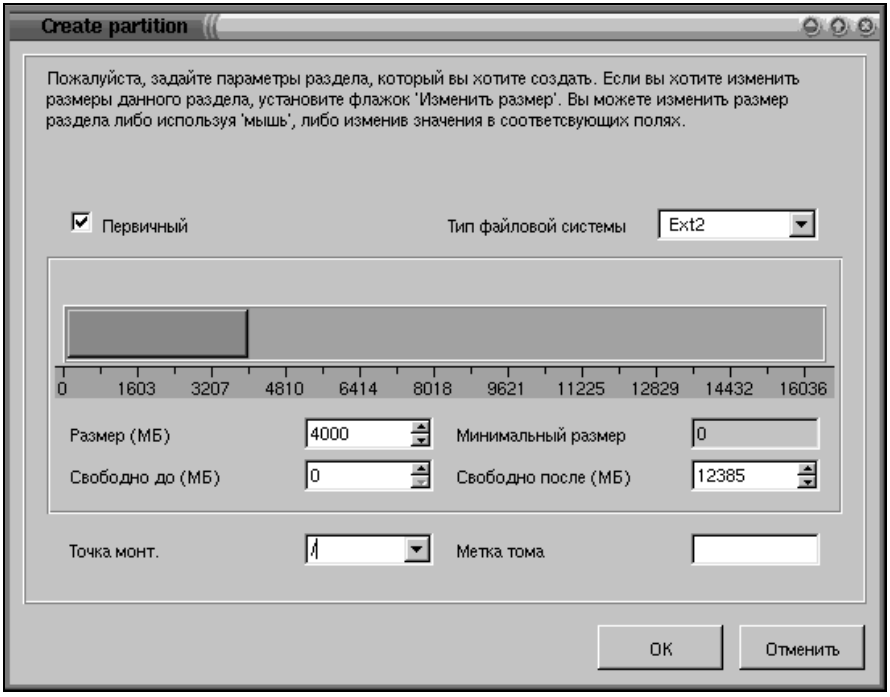



**Рис. 8.8.** Создание корневого раздела

Если операционная система Linux является единственной на диске, то предпочтительнее разделы создавать как первичные (количество их ограничено четырьмя). Размеры дисков для нужд пользователей определяются решаемыми задачами. При проектировании раздела необходимо предусмотреть запас на будущее. Например, если объем используемых данных до 1 ГБайт, то берем 4 Гбайт (с учетом роста). При больших объемах информации объем раздела увеличивайте, но уже не с таким большим запасом.

9. Особо стоит отметить создание раздела Swap — он не имеет точки монтирования (рим. 8.9). При небольших объемах ОЗУ его величина должна составлять удвоенную величину оперативной памяти. При больших объемах эта формула не действует и можно ограничиться 500 Мбайт. Для создания Swap-раздела необходимо указать его размер, как мы это делали ранее. В типе файловой системы выбираем тип **Swap**. Теперь все готово, жмем **OK**.

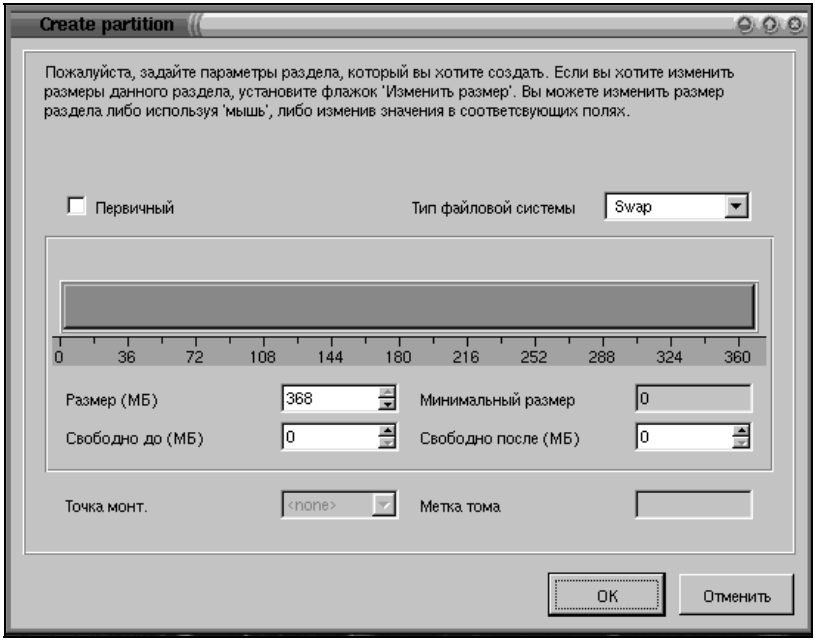



**Рис. 8.9.** Создание Swap-раздела

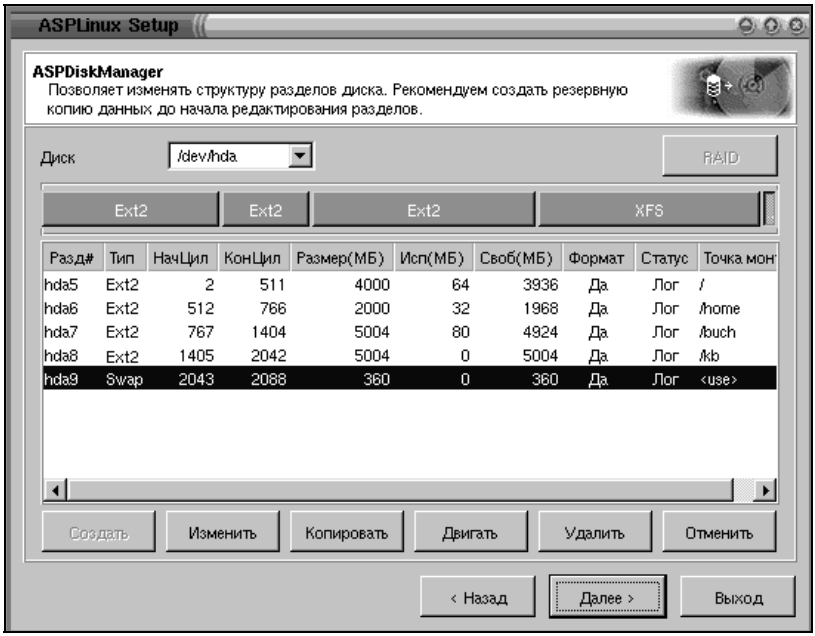



**Рис. 8.10.** Создание разделов закончено

После окончания создания разделов вы увидите примерно следующую картину (рис. 8.10). Еще раз стоит оговориться, что величина и количество разделов будут определяться вашими задачами и конфигурацией сервера. И не стоит стремиться подводить свой диск под эти цифры. Здесь под корневой раздел отведено 4 Гбайт, под домашние каталоги пользователей выделено 2 Гбайт, под бухгалтерию и деятельность конструкторского бюро выделено по 5 Гбайт. Остаток отведен под Swap.

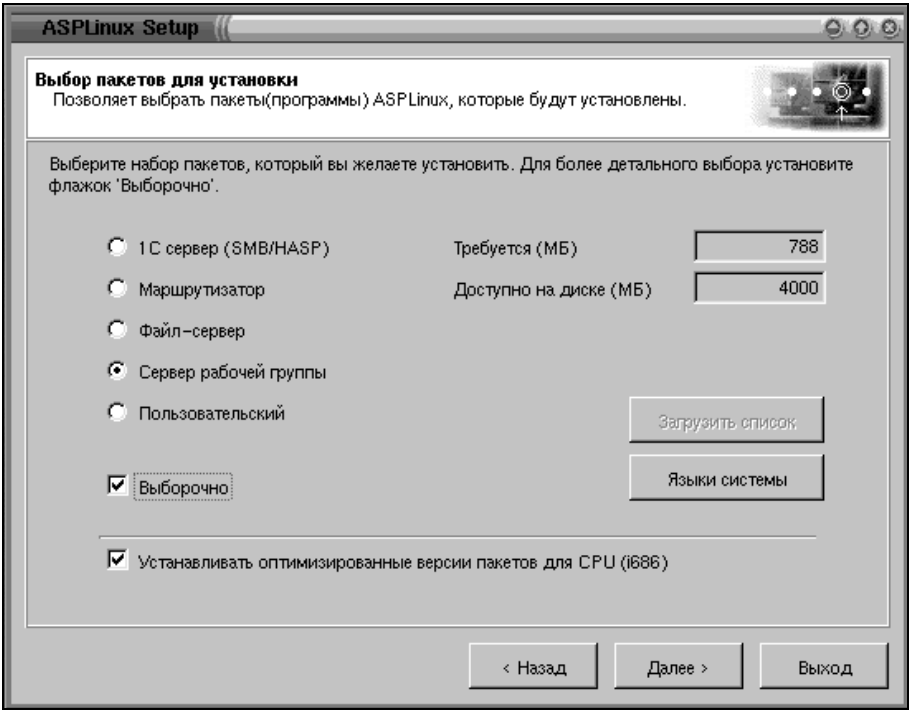



**Рис. 8.11.** Выбор пакетов для установки

10. Следующим окном является выбор типа установки (рис. 8.11). Выбираем сервер рабочей группы, он включает набор сервисов, необходимых для организации работы небольшого офиса, а именно: систему электронной почты, систему печати, файл-сервер, web-сервер и др. В принципе, добавлять понадобится только web-браузер Mozilla. И возможно пакет поддержки ключа HASP (необходимо, если HASP будет стоять на сервере). HASP — это такой ключ, с помощью которого программы, в частности продукты компании 1С, защищаются от незаконного копирования (рис. 8.12).

    В принципе HASP может стоять на любой машине в сети. Само собой, она должна быть включена, когда кто-то работает с программой "1С:Бухгалтерия" или другой, защищенной ключом.

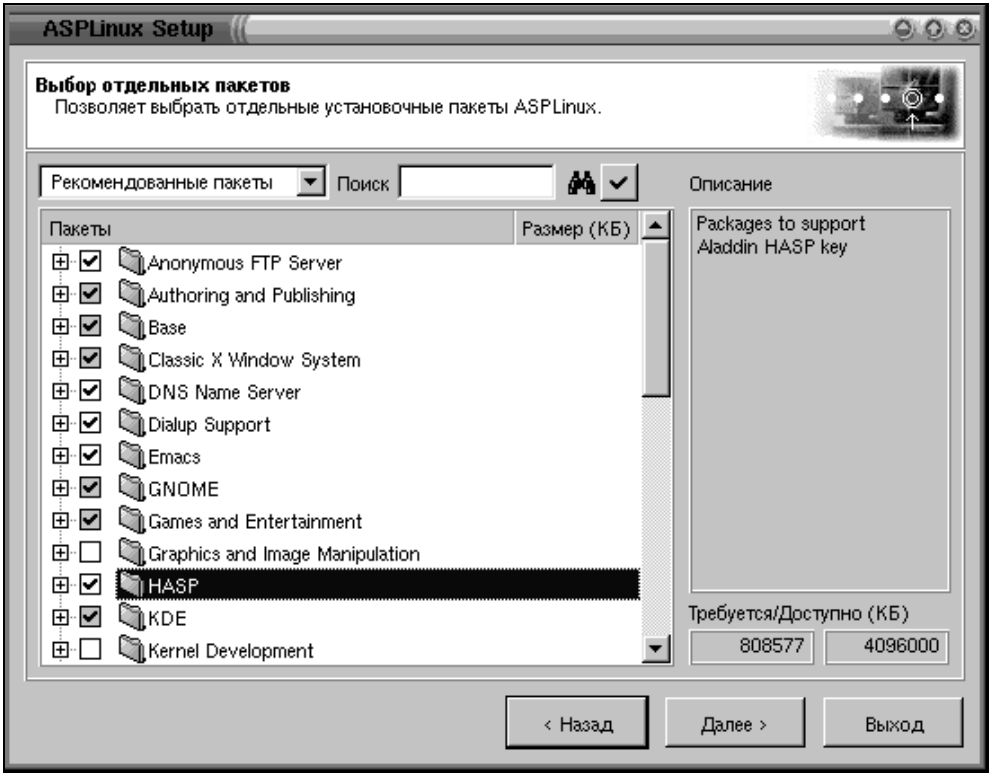



**Рис. 8.12.** Подключение пакета поддержки HASP

**Примечание**

Необходимо сказать, что продукты компании 1С необходимо вовремя обновлять. Обновления проводятся в соответствии с прилагаемыми к программному обеспечению инструкциями. Также необходимо резервировать базы данных (о том, как это делать, *см. в главе 9*). В противном случае можно наткнуться на проблемы в самое неподходящее время. Вообще бухгалтерии нужно уделять самое пристальное внимание. Следите за базами данных, регулярно проводите их сервисное обслуживание. Если база давно не индексировалась, то поведение программы может быть очень странным, что сильно затруднит работу бухгалтеру, а вам будет стоить лишних нервов. Так что не пускайте это дело на самотек.

11. По умолчанию Mozilla не входит в набор пакетов для сервера рабочей группы (рис. 8.13). Понадобится он вам для администрирования с использованием утилиты Webmin непосредственно с сервера. Для того чтобы выбрать пакет, необходимо войти в **Messaging and Web Tools** и выбрать все пакеты, в названии которых присутствует слово mozilla. Все необходимые пакеты выбраны, остается нажать кнопку **Далее**.

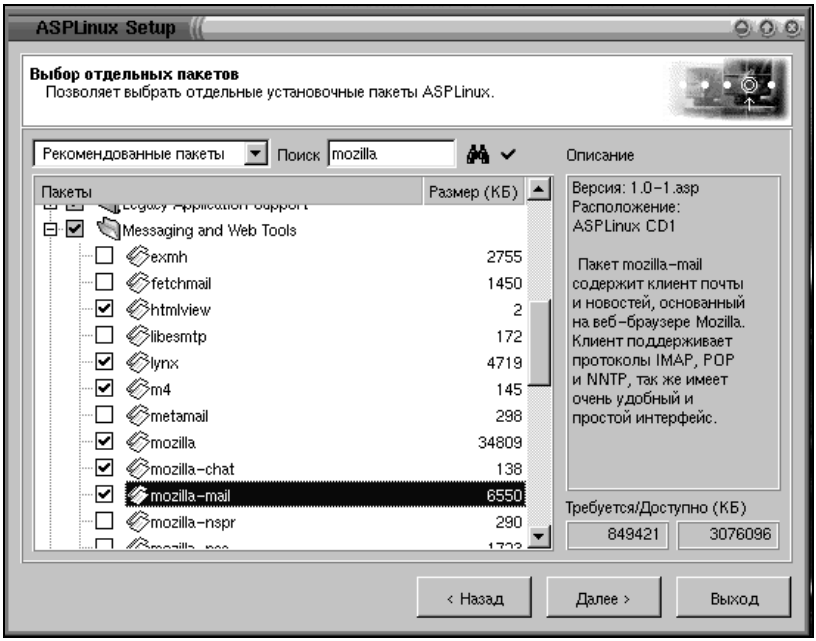



**Рис. 8.13.** Выбор пакета mozilla

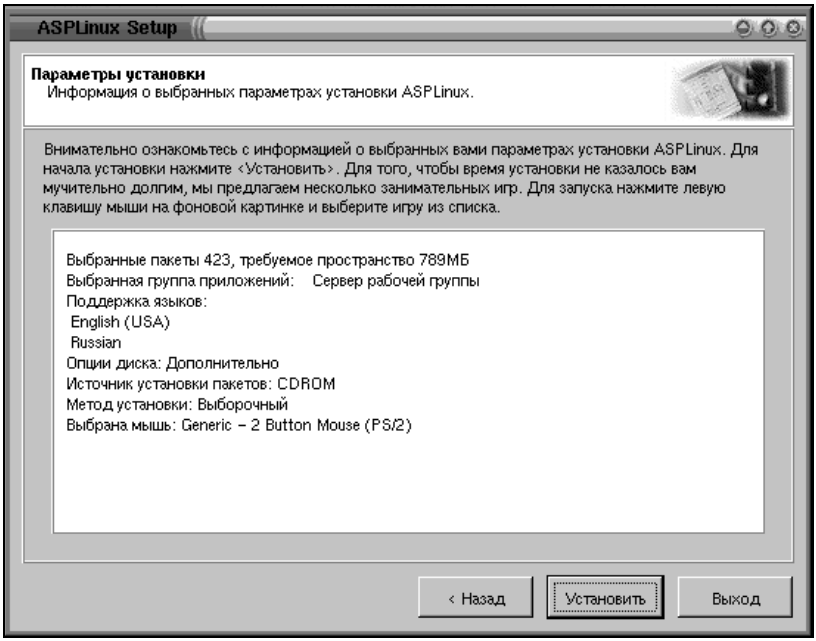



**Рис. 8.14.** Информация о выбранных параметрах установки

12. В следующем окне вам выдадут информацию о выбранном типе пакетов и параметрах установки (рис. 8.14). Проверьте информацию и нажмите кнопку **Установить**.

    Однако, если добавляли или убирали какие-то пакеты по своему усмотрению, то могло случиться так, что вы нарушили взаимозависимость пакетов. Ничего страшного, программа установки все проверит за вас. Если нарушение зависимости все же имело место, то у вас перед этим окном может появиться окно, говорящее о нарушении зависимости пакетов. Кнопка **Разрешить** позволяет программе установки автоматически пополнить список пакетов недостающими пакетами.

13. Следующим будет окно, отображающее процесс установки. Внизу видна полоска, отображающая ход установки, в центре экрана отображаются устанавливаемые пакеты. В зависимости от выбранной конфигурации, может понадобиться вставить второй диск.

14. Об окончании процесса установки пакетов вас уведомит информационное окно с соответствующей надписью (рис. 8.15). Правда, окончание процесса установки пакетов не означает окончание процесса установки как такового, придется еще немного поработать.

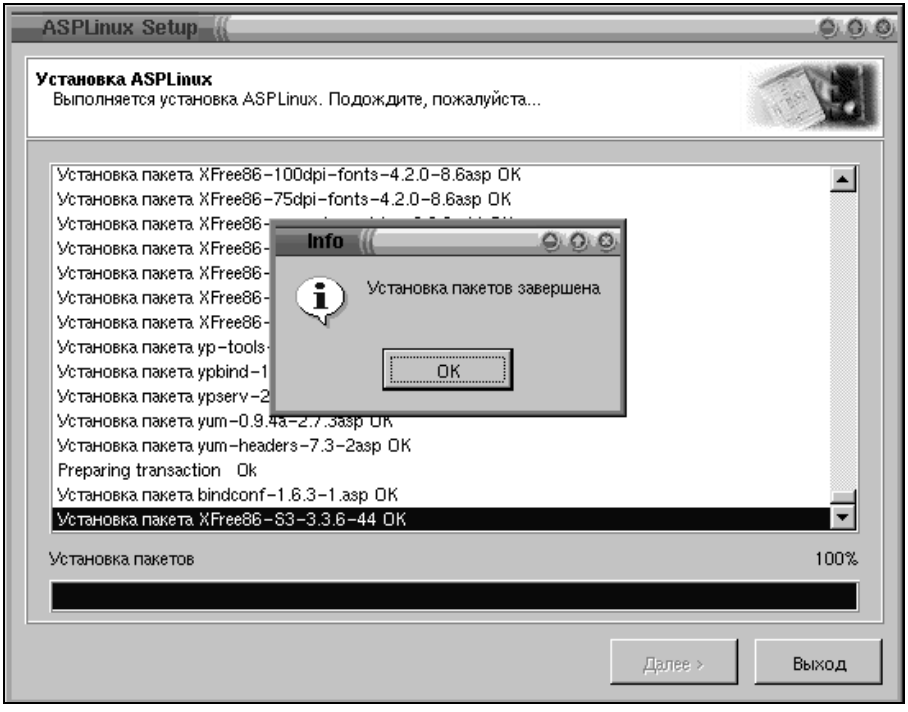



**Рис. 8.15.** Окончание процесса установки пакетов

**Рис. 8.16.** Выбор загрузчика

15. Переходим к окну выбора загрузчика операционной системы. Загрузчик управляет нормальным запуском операционной системы. Если вы не обладаете достаточным опытом, то лучше установить переключатели, как указано на рис. 8.16.

**Примечание**

ASPLoader намного удобнее и приятнее в работе многих других загрузчиков. В частности, он выглядит намного приятнее Lilo, используемого в Red Hat по умолчанию.

16. Скорее всего, параметры сетевой карты определятся автоматически, как на рис. 8.17. Если этого не произошло, то необходимо из раскрывающегося списка выбрать драйвер, указать вручную **Порт IO** и прерывание **IRQ** и нажать кнопку **Добавить**. Если установлено несколько сетевых карт, необходимо настроить их все.
17. На рис. 8.18 производится настройка сети. В выпадающем списке выбираем по очереди все сетевые карты и настраиваем их. Для сервера параметры сетевых карт должны указываться явным образом, а не через DHCP-сервер. Поэтому флажок **Настроить с помощью DHCP** снимаем и явным образом указываем IP-адрес и маску сети. Если у вас возникли вопросы, почему IP-адрес начинается с 192.168.0.5, то вернитесь в первую часть книги (*см. разд. 1.2*).

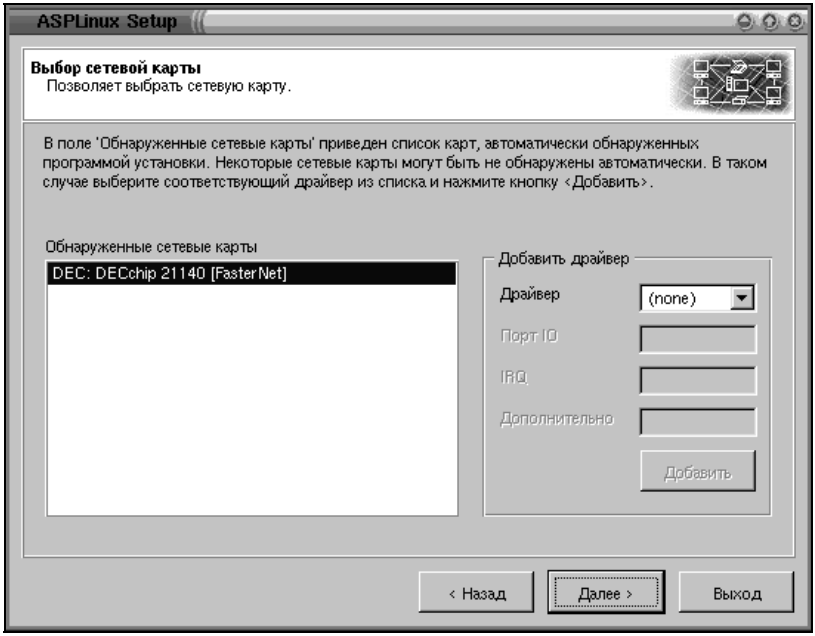



**Рис. 8.17.** Выбор сетевой карты

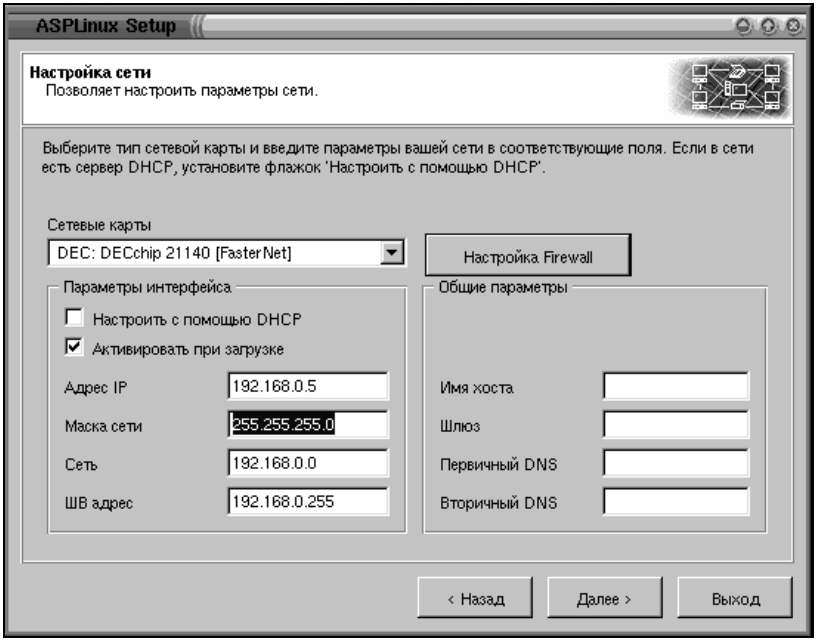



**Рис. 8.18.** Настройка параметров сети

18. Следующим окном (рис. 8.19) будет окно выбора монитора. Автоматически определяются большинство мониторов. Однако, если этого не произошло (в этом случае написано **Unknown or Laptop monitor**), вы можете выбрать монитор вручную, не обязательно именно вашу модель. Можете выбрать близкую по классу, значение горизонтальной и вертикальной частот разверток можно подкорректировать в выпадающем списке.

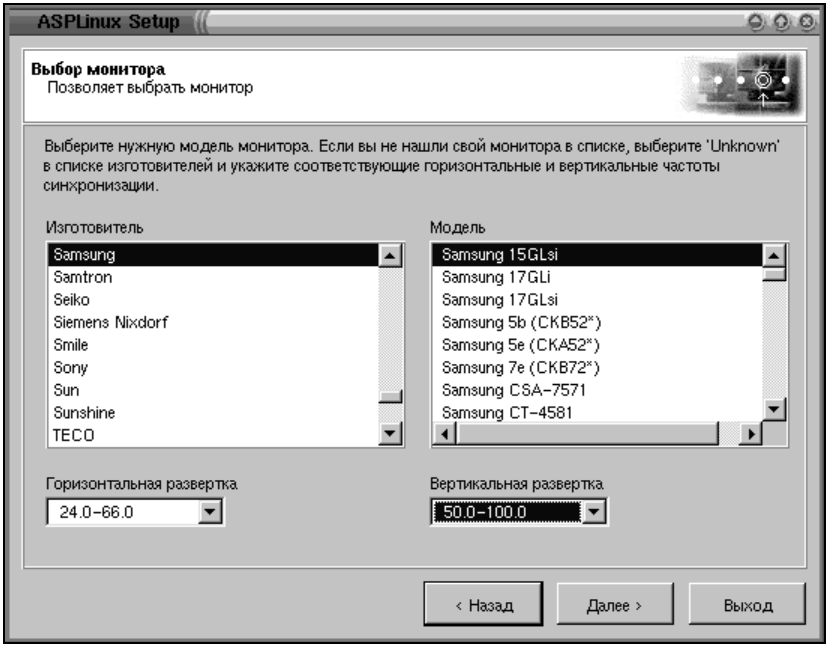



**Рис. 8.19.** Выбор типа монитора

19. Следующим шагом является выбор видеокарты (рис. 8.20). Правильность выбора видеокарты можно проверить, нажав кнопку **Тестировать**. Если все выбрано правильно, вы увидите следующее диалоговое окно (рис. 8.21).
20. Подтвердите правильность выбора. Небольшое смещение изображения и искажение размеров мы подправим позднее, не обращайте на это пока внимания. Если вы ошиблись, то, скорее всего, увидите черный экран. Флажок **Использовать графический вход** лучше снять, это избавит от проблем, если ошиблись с настройками видеокарты, а войти в графический режим несложно.
21. На рис. 8.22 представлен экран выбора модели клавиатуры, языка, раскладки клавиатуры и переключения ее. После установки необходимых значений жмем кнопку **Далее** и переходим к настройке даты времени и часового пояса.

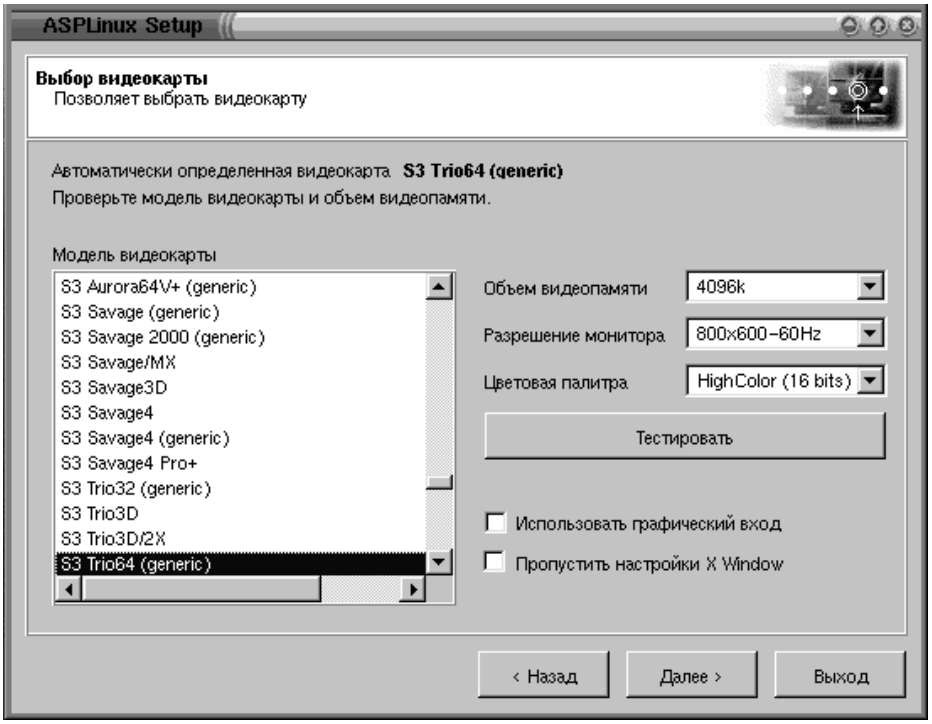



**Рис. 8.20.** Выбор видеокарты

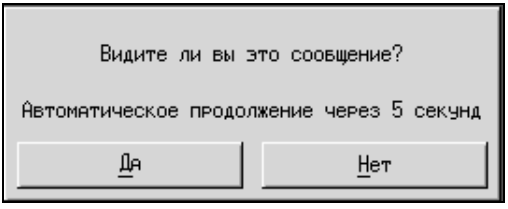



**Рис. 8.21.** Тестирование видеокарты

22. В окне на рис. 8.23 мы определяем, прежде всего, пароль пользователя root (суперпользователя). Пароль вводится два раза. Будьте аккуратны при вводе и не забудьте его. Пароль не следует выбирать коротким и слишком простым. При необходимости здесь можно добавить и новых пользователей, но лучше это будет сделать потом.
23. Следующее окно говорит об успешном завершении установки (рис. 8.24). Осталось нажать кнопку **Перезагрузить** — и можно будет познакомиться с миром Linux.

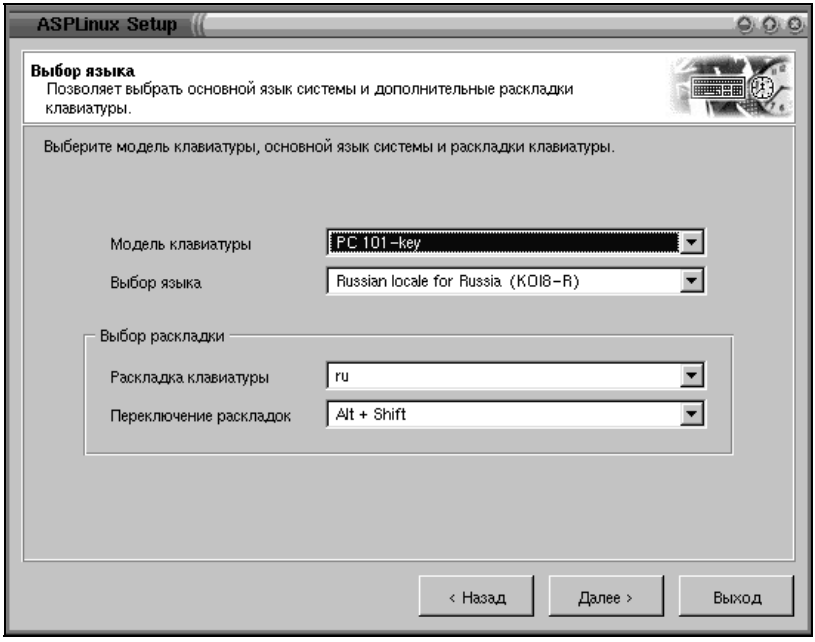



**Рис. 8.22.** Выбор языка и раскладки клавиатуры

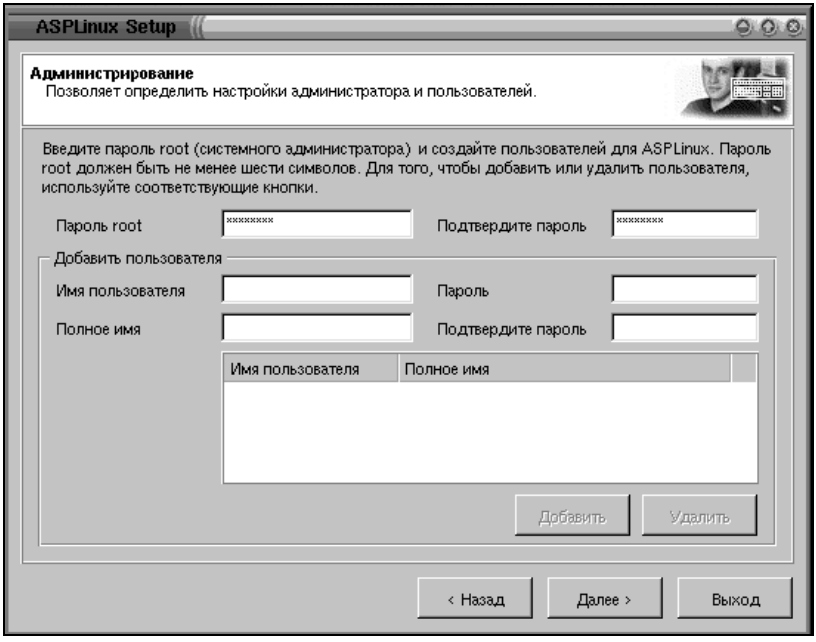



**Рис. 8.23.** Настройка пользователей

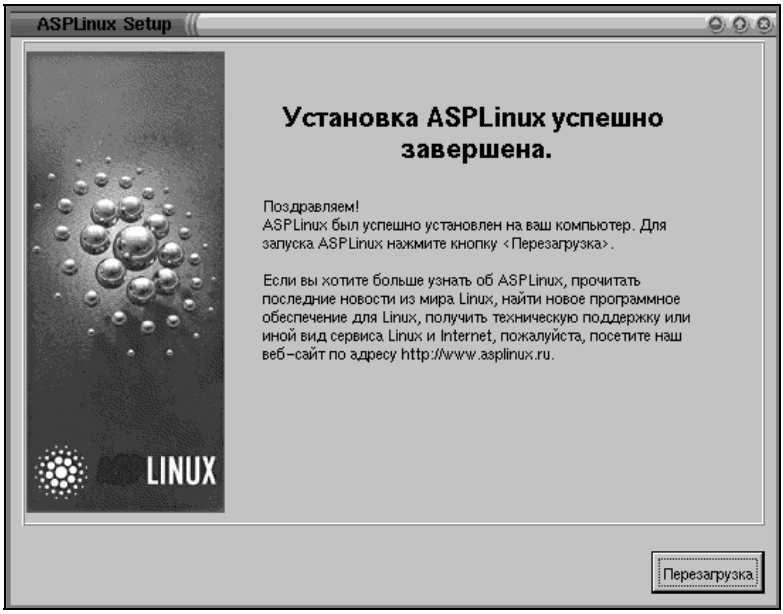



**Рис. 8.24.** Завершение установки

# 8.3. Первый вход в систему и настройка графического режима при помощи утилиты xvidtune

Окно загрузчика ASPLoader предложит один из вариантов загрузки: либо с дискеты, либо ОС Linux (рис. 8.25).

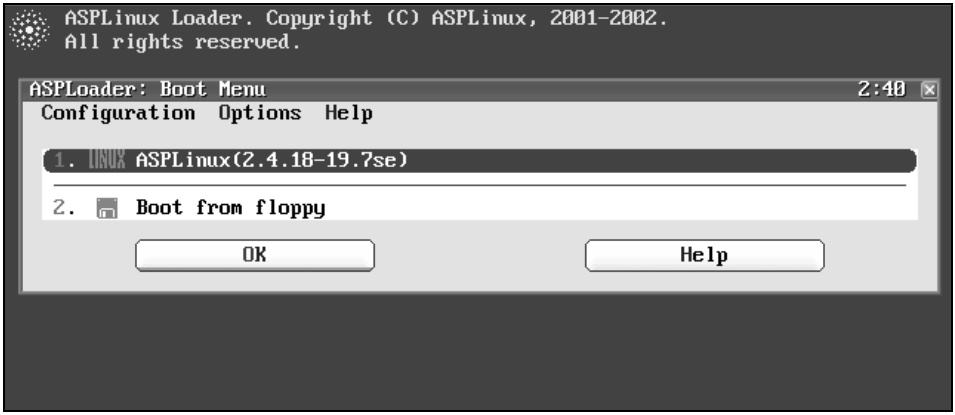



**Рис. 8.25.** Выбор типа загрузки

1. После того как вы выбрали тип загрузки (или по истечении 30 секунд), начнется загрузка операционной системы (рис. 8.26).

```
                    Welcome to ASPLinux
            Press 'I' to enter interactive startup.
Mounting proc filesystem:                                  [  OK  ]
Configuring kernel parameters:                             [  OK  ]
Setting default font (UniCyr_8x16):                        [  OK  ]
Setting clock  (localtime): Вск Мар  6 02:40:35 MSK 2005   [  OK  ]
Activating swap partitions:                                [  OK  ]
Setting hostname localhost:                                [  OK  ]
Checking root filesystem
/dev/hda5: clean, 69124/513024 files, 223773/1024135 blocks
                                                           [  OK  ]
Remounting root filesystem in read-write mode:             [  OK  ]
Finding module dependencies:                               [  OK  ]
Starting up RAID devices:
Checking filesystems
/dev/hda6: clean, 11/256512 files, 8065/512064 blocks
/dev/hda7: clean, 11/641280 files, 20141/1281175 blocks
                                                           [  OK  ]
```

**Рис. 8.26.** Процесс загрузки ОС Linux

2. Процесс загрузки операционной системы заканчивается приглашением, а поскольку мы сознательно не выбирали графический режим, то авторизоваться будем в текстовом режиме, для этого вводим имя **root** и пароль, который вы выбрали. Пароль при воде не отображается (рис. 8.27). Сделано это для того, чтобы кто-нибудь не подсмотрел ваш пароль из-за спины.

```
ASPLinux release 7.3 (Vostok)
Kernel 2.4.18-19.7se on an i686

localhost login: root
Password: _
```

**Рис. 8.27.** Авторизация в системе

После авторизации в системе командная строка на экране заканчивается символом #, что говорит о том, что вы root, у остальных пользователей командная строка заканчивается символом $.

3. Первой нашей задачей стоит настройка графического режима при помощи утилиты xvidtune.

Утилита xvidtune призвана облегчить настройку графического режима. К сожалению, xvidtune непосредственно не вносит изменения в файлы конфигурации, а выводит строку, которая определяет параметры системы.

Далее описаны действия по настройке графического режима.

**Рис. 8.28.** Вид утилиты xvidtune

4. В командной строке набираем **xinit**. Запускается Xserver.
5. Теперь набираем **xvidtune** (рис. 8.28). Появляется предупреждение, которое говорит о том, что вы используете программу на свой страх и риск. Жмем **OK**. Нажмем также кнопку **Auto**, она позволит видеть изменения сразу же. Далее, используя кнопки **Left**, **Right**, **Up**, **Down**, позиционируем изображение в центре экрана. А затем, используя **Wider**, **Narrower**, **Shorter**, **Taller**, развертываем (сжимаем) изображение. Ну вот, полдела сделано.
6. Теперь жмем кнопку **Show**, которая выведет необходимую строку настройки, запомните настройку, а лучше запишите. Теперь выходим из программы, нажав **Quit**.
7. Выходим из графического режима, нажав <Ctrl>+<Alt>+<BackSpace>.
8. Запускаем файловый менеджер Midnight Commander, для этого в командной строке набираем **mc** (*о его работе подробнее см. в главе 9*).
9. Выделяем курсором файл /etc/x11/XF86Config-4 (XF86Config) и жмем клавишу <F4>. В редактируемом файле необходимо найти секцию `monitor` и добавить туда строку `modeline`, а далее те параметры которые получили в п. 6. Записываем файл.
10. Покидаем графический сервер, нажав <Ctrl>+<Alt>+<BackSpace>.
11. Перезагружаем Xserver, набрав **startx** (в этом случае, помимо графического режима, запустится сразу и оболочка Gnome). Должно получиться изображение без искажений.

Теперь система установлена, и можно углубиться в изучение основ Linux.

## Если все сложнее, чем вы думали

Вполне возможно, что на этапе установки системы вы пропустили настройку графического режима либо настроили его неправильно. В этом случае после набора команд **xinit** или **startx**, скорее всего, вы увидите пустой (черный) экран.

Пустой экран может появиться и сразу после загрузки системы. Это будет в случае, если выбрали графический вход в систему. Тогда делаем следующее.

- Выходим из графического режима, нажав <Ctrl>+<Alt>+<BackSpace>.
- Входим в систему под администратором.
- В командной строке набираем **Xconfigurator**.

Следуя указаниям программы, вы можете настроить с нуля или перенастроить существовавший графический режим (рис. 8.30). Все сделанные вами изменения автоматически заносятся в систему. После окончания работы программы Xconfigurator можно будет запустить графический режим и довести его до кондиции программой xvidtune.

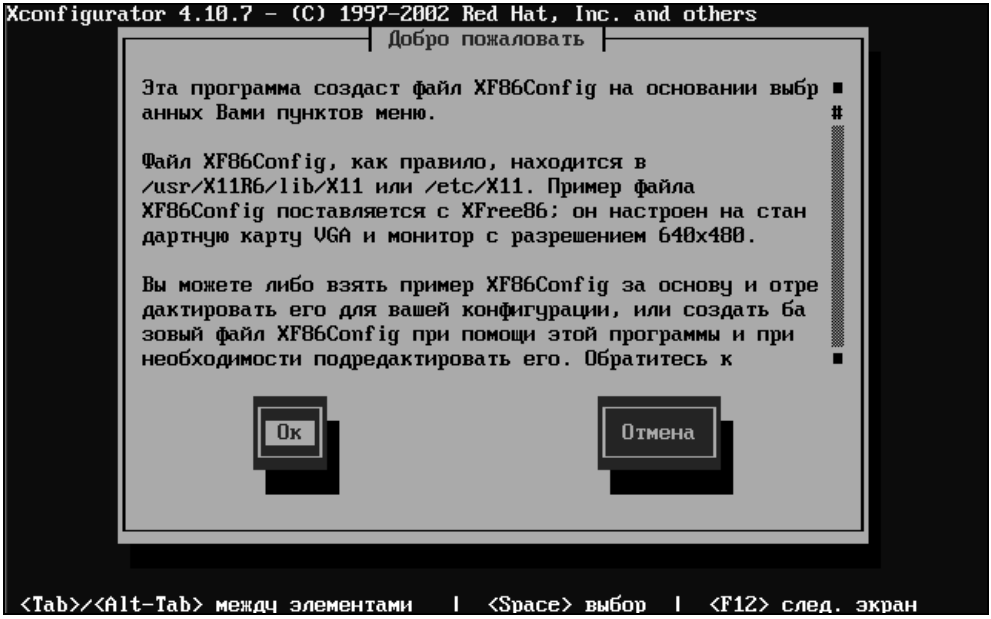



**Рис. 8.30.** Окно программы Xconfigurator

# Глава 9

# Основы администрирования Linux

В этой главе будут рассмотрены основы администрирования и некоторые моменты, касающиеся работы Linux. Будет предложено два метода: ручного редактирования конфигурационных файлов и посредством утилиты Webmin. Более подробно вопросы запуска Webmin будут рассмотрены в *главе 12*. Пока примем как данность, что у нас уже все работает. Тем более, что в ASPLinux 7.3 Server Edition, взятой нами за основу, так и будет.

Linux сама по себе достаточно дружественная система и может подсказать вам ответы на большинство вопросов. Вашим первейшим помощником должны стать страницы интерактивного справочного руководства. Просмотр справки осуществляется вызовом команды

`man` имя_команды (устройства)

либо

`man` ключевое_слово (выводит список страниц, в которые ключевое слово входит).

Существует большое количество сайтов, посвященных Linux (*см. Приложение 3*). На сайтах огромное количество авторских статей, руководств. Есть форумы, где вам окажут помощь. Ведь с той проблемой, с которой вы столкнулись, возможно, уже кто-то сталкивался, и он готов вам помочь. Возможно, кому-то поможете и вы. В любом случае знания можно получить либо из электронных и бумажных источников, либо путем общения. Главное — не опускать руки.

## 9.1. Загрузка системы

Перед тем, как вы увидите приглашение на вход в систему, успевает произойти достаточно много событий:

- включение питания и проверка аппаратного обеспечения средствами BIOS;

- запуск загрузчика ОС и передача ему управления;
- загрузка и запуск ядра и программы init.

Программа init считывает файл конфигураций /etc/inittab файла конфигурации системы (листинг 9.1).

**Листинг 9.1. Файл /etc/inittab**

```
#
# inittab       This file describes how the INIT process should set up
#               the system in a certain run-level.
#
# Author:       Miquel van Smoorenburg, <miquels@drinkel.nl.mugnet.org>
#               Modified for RHS Linux by Marc Ewing and Donnie Barnes
#

# Default runlevel. The runlevels used by RHS are:
#   0 - halt (Do NOT set initdefault to this)
#   1 - Single user mode
#   2 - Multiuser, without NFS (3, if you do not have networking)
#   3 - Full multiuser mode
#   4 - unused
#   5 - X11
#   6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:
#Уровень запуска по умолчанию
# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

# What to do in single-user mode.
~~:S:wait:/sbin/sulogin
#ссылка на каталог
l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
l2:2:wait:/etc/rc.d/rc 2
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6
```

```
# Things to run in every runlevel.
ud::once:/sbin/update

# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now

# When our UPS tells us power has failed, assume we have a few minutes
# of power left.  Schedule a shutdown for 2 minutes from now.
# This does, of course, assume you have powerd installed and your
# UPS connected and working correctly.
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting"

# If power was restored before the shutdown kicked in, cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Cancelled"



# Run gettys in standard runlevels
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6

# Run xdm in runlevel 5
# xdm is now a separate service
x:5:respawn:/etc/X11/prefdm -nodaemon
```

Строки, начинающиеся символом #, являются комментарием. Остальные строки состоят из 4-х полей, разделенных знаком двоеточия (табл. 9.1).

***Таблица 9.1.*** *Структура учетной записи в файле /etc/inittab*

| Номер поля в строке | Назначение поля |
|---|---|
| 1 | Идентификатор строки в файле длиной в два символа |
| 2 | Уровень работы для данной строки. Пустая ячейка значит, что работает на всех уровнях |
| 3 | Действие, которое стоит выполнить с процессом |
| 4 | Процесс или программа, которые будут выполнены |

Уровни работы для строки описаны в табл. 9.2.

***Таблица 9.2.*** *Уровни работы*

| Номер уровня | Комментарий |
|---|---|
| 0 | Останов системы |
| 1 | Однопользовательский режим |
| 2 | Многопользовательский режим за исключением работы в сети |
| 3 | Многопользовательский режим без ограничений |
| 4 | Не используется |
| 5 | Графический режим |
| 6 | Перезагрузка системы |

В каталогах находятся символические ссылки, которые останавливают или запускают процесс. Имена этих ссылок подчинены определенным правилам (табл. 9.3).

***Таблица 9.3.*** *Правила наименования ссылок в каталогах rc*

| Номер позиции | Допустимое значение | Комментарий |
|---|---|---|
| 1 | S или K | S — запустить процесс (Start), K — остановить (убить) процесс (Kill) |
| 2–3 | 00–99 | Порядок выполнения действия. Запуск производится в порядке возрастания номеров. Останов процессов в порядке их убывания |
| 4 и далее | – | Имя процесса, с которым происходит операция |

Рассмотрим пример в листинге 9.2.

**Листинг 9.2. Пример сценария запуска останова**

```
#! /bin/bash
# DHCPD запускает или останавливает DHCP Server
# chkconfig: 345 79 21
# description: dhcpd (DHCP)
# autor: free
```

```
#Проверим наличие необходимых библиотек и запуск сети
/etc/rc.d/init.d/functions
/etc/sysconfig/network
[ "${NETWORKING}" = "no" ] && exit 0RETVAL=0
# Определим переменную
prog="dhcpd"
# Собственно тело скрипта
start() {
# процедура запуска
echo -n $"Запуск $prog: "
# daemon dhcpd -u dhcpd
daemon /usr/sbin/dhcpd
RETVAL=$?
[ $RETVAL -eq 0 ] && touch /var/lock/subsys/dhcpd
echo
return $RETVAL
}
stop() {
# Процедура остановки
echo -n $"Останавливается $prog: "
# killproc dhcpd
killproc /usr/sbin/dhcpdRETVAL=$? [ $RETVAL -eq 0 ] && rm -f
/var/lock/subsys/dhcpd
echo
return $RETVAL
}
#Процедура перезапуска
restart() {
stop
start
}
# Выбор необходимой процедуры из start-stop-restart-reload-status
case "$1" in
start)
start
;;
stop)
stop
```

```
;;
restart|reload)
restart
;;
status)
status /usr/sbin/dhcpd
;;
*)
echo $"Usage: $0 {start|stop|restart|reload|status}"
exit 1
esac
exit $?
```

## 9.2. Пользователи, группы, учетные записи

Пользователей, а точнее их учетные записи, в Linux можно разделить на три категории:

- Учетная запись суперпользователя (root) — обладает неограниченными правами в системе и может удалить любой файл, остановить или запустить любой процесс, останавливать систему и др. К работе с этой учетной записью надо подходить очень осторожно и ответственно. Пользоваться учетной записью суперпользователя нужно только, если это действительно необходимо. Вообще лучше для решения отдельных задач, таких как создание учетных записей, управление теми или иными файлами, каталогами и другими задачами администрирования создавать отдельные учетные записи.
- Учетные записи обычных пользователей — например, для менеджера из отдела продаж или секретаря. Основное отличие их от root — в правах. Как правило, это ограниченный набор прав на доступ к тому или иному каталогу, процессу. Предоставление обычному пользователю лишних привилегий может выйти боком по понятным причинам.
- Системные учетные записи — не принадлежат реальным людям, а относятся к системным процессам. Характерной особенностью является отсутствие паролей, поскольку они не предназначены для регистрации в системе.

**Внимание!**

*Правильное управление учетными записями и разграничение прав доступа является залогом безопасности системы и вашего спокойного сна.*

Для удобства администрирования пользователи объединяются в группы. Группа — это объединение нескольких пользователей для решения определенной задачи.

Ключевым при рассмотрении пользователей и их учетных записей является файл /etc/passwd. Это обычный текстовый файл. Он доступен для чтения любому пользователю. Примерный вид этого файла приведен в листинге 9.3.

***Листинг 9.3 Файл /etc/passwd (сокращенный)***

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/var/spool/news:
gdm:x:42:42::/var/gdm:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
nscd:x:28:28:NSCD Daemon:/:/bin/false
ident:x:98:98:pident user:/:/sbin/nologin
postgres:x:26:26:PostgreSQL Server:/var/lib/pgsql:/bin/bash
radvd:x:75:75:radvd user:/:/bin/false
squid:x:23:23::/var/spool/squid:/dev/null
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
firebird:x:184:184:Firebird Database Administra-
tor:/opt/interbase:/dev/null
mysql:x:27:27:MySQL Server:/var/lib/mysql:/bin/bash
sapdb:x:500:500:SAPDB demo user:/home/sapdb:/bin/bash
postfix:x:89:89::/var/spool/postfix:/bin/true
amanda:x:33:6:Amanda user:/var/lib/amanda:/bin/bash
aspseek:x:93:93::/var/aspseek:/bin/bash
jabber:x:52:2:Jabber Server:/var/spool/jabber:/bin/bash
mailman:x:41:41:GNU Mailing List Manager:/var/mailman:/bin/false
```

```
netdump:x:34:34:Network Crash Dump user:/var/crash:/bin/bash
ldap:x:55:55:LDAP User:/var/lib/ldap:/bin/false
popa3d:x:84:84::/dev/null:/dev/null
pvm:x:24:24::/usr/share/pvm3:/bin/bash
user:x:501:501::/home/user:/bin/bash
```

Каждая строка этого файла соответствует определенному пользователю. Строка состоит из полей, разделенных символом двоеточия. Назначение каждого поля зафиксировано (табл. 9.4).

***Таблица 9.4.*** *Структура учетной записи в файле /etc/fstab*

| Номер поля в строке | Назначение поля |
|---|---|
| 1 | Имя учетной записи |
| 2 | Раньше здесь хранился пароль, но сейчас в большинстве систем используется система теневых паролей. Теневые пароли хранятся в файле /etc/shadow. Поэтому во втором поле скорее всего будет `x`. Это, собственно, и говорит о том, что используется система теневых паролей |
| 3 | Здесь находится идентификатор пользователя (UID, User ID) |
| 4 | В этом поле находится идентификатор группы пользователя по умолчанию GID (Group ID). Пользователь может принадлежать более, чем к одной группе. Информация о том, в каких группах числится пользователь, содержится в файле /etc/group |
| 5 | Поле для комментария, здесь хранится полное имя пользователя, контактные телефоны и пр. |
| 6 | Каталог, где окажется пользователь после регистрации |
| 7 | Оболочка по умолчанию |

В большинстве систем используется система теневых паролей, в этом случае пароли хранятся в файле /etc/shadow. Причина в том, что файл /etc/passwd доступен для чтения любому пользователю. То есть получается, что можно считать пароль в зашифрованном виде. Конечно, дешифровать его не получится, так как используются специальные алгоритмы. Но можно подобрать пароль методом прямого перебора. Для исключения этого зашифрованный пароль был перенесен из файла /etc/passwd в файл /etc/shadow (листинг 9.4).

**Листинг 9.4. Содержимое файла /etc/shadow**

```
root:$1$uTZ5$989pmoMOjzMcYTJhs/T1u1:12302:0:99999:7:::
bin:*:12302:0:99999:7:::
daemon:*:12302:0:99999:7:::
adm:*:12302:0:99999:7:::
lp:*:12302:0:99999:7:::
sync:*:12302:0:99999:7:::
shutdown:*:12302:0:99999:7:::
halt:*:12302:0:99999:7:::
mail:*:12302:0:99999:7:::
news:*:12302:0:99999:7:::
gdm:!!:12302:0:99999:7:::
rpcuser:!!:12302:0:99999:7:::
nfsnobody:!!:12302:0:99999:7:::
nscd:!!:12302:0:99999:7:::
ident:!!:12302:0:99999:7:::
postgres:!!:12302:0:99999:7:::
radvd:!!:12302:0:99999:7:::
squid:!!:12302:0:99999:7:::
vcsa:!!:12766:0:99999:7:::
firebird:!!:12766:0:99999:7:::
mysql:!!:12766:0:99999:7:::
sapdb:SARW90n3cnWxc:12766:0:99999:7:::
postfix:!!:12766:0:99999:7:::
amanda:!!:12766:0:99999:7:::
aspseek:!!:12766:0:99999:7:::
jabber:!!:12766:0:99999:7:::
mailman:!!:12766:0:99999:7:::
netdump:!!:12766:0:99999:7:::
ldap:!!:12766:0:99999:7:::
popa3d:!!:12766:0:99999:7:::
pvm:!!:12766:0:99999:7:::
user:$1$32cz$vUCsWQWyVFWdt6XjZwwL99:12766:0:99999:7:::
```

Как и в случае с файлом /etc/passwd, в файле /etc/shadow каждой учетной записи соответствует отдельная строка. Строка состоит из полей разделенных знаком двоеточия, назначение каждого поля зафиксировано (табл. 9.5).

***Таблица 9.5.*** *Назначение полей в файле /etc/shadow*

| Номер поля в строке | Назначение поля |
|---|---|
| 1 | Имя учетной записи соответствует имени учетной записи из файлов /etc/passwd. В этом случае имя связывает эти две строчки в разных файлах |
| 2 | В этом поле содержится пароль в зашифрованном виде |
| 3 | Дата последнего изменения пароля |
| 4 | Минимальный период между изменениями пароля |
| 5 | Максимальный период до изменения пароля (если –1, пароль может не изменяться) |
| 6 | Период до окончания срока действия пароля, в течение которого будет выдаваться предупреждение пользователю |
| 7 | Период, по истечении которого учетная запись, если не изменить пароль, будет заблокирована |
| 8 | Срок действия учетной записи, когда запись будет заблокирована. Если поле пустое, то учетная запись всегда активна |
| 9 | Зарезервированное поле |

Поскольку пользователь может принадлежать к нескольким группам, а в файле /etc/passwd значится только группа по умолчанию, то возникает вопрос, где хранится информация о группах. Ответ прост — в файле /etc/group (листинг 9.5).

**Листинг 9.5. Структура файла /etc/group**

```
root:x:0:root
bin:x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin,adm
adm:x:4:root,adm,daemon
tty:x:5:
disk:x:6:root
lp:x:7:daemon,lp
mem:x:8:
```

```
kmem:x:9:
wheel:x:10:root
mail:x:12:mail,postfix
news:x:13:news
ntp:x:38:
rpc:x:32:
gdm:x:42:
rpcuser:x:29:
nfsnobody:x:65534:
nscd:x:28:
ident:x:98:
postgres:x:26:
radvd:x:75:
squid:x:23:
wine:x:101:
vcsa:x:69:
firebird:x:184:
mysql:x:27:
sapdb:x:500:
postdrop:x:90:
postfix:x:89:
aspseek:x:93:
ccache:x:11:
haclient:x:60:
pppusers:x:44:
popusers:x:45:
slipusers:x:46:
mailman:x:41:
netdump:x:34:
ldap:x:55:
popa3d:x:84:
pvm:x:24:
user:x:501:
```

Каждая строка файла /etc/group соответствует отдельной группе. Назначение полей описано в табл. 9.6.

***Таблица 9.6.*** *Назначение полей в файле /etc/group*

| Номер поля в строке | Назначение поля |
|---|---|
| 1 | Имя группы |
| 2 | Поле пароля группы, как правило, не используется |
| 3 | Идентификатор группы. Именно этот идентификатор хранится в поле группы по умолчанию. Если в /etc/passwd пользователь является членом группы, а в /etc/group нет, то группе он принадлежит все равно |
| 4 | Список пользователей, входящих в группу. Пользователи перечисляются через запятую, без пробелов |

# 9.3. Управление учетными записями

## Добавление нового пользователя

Добавление нового пользователя может быть осуществлено либо из командной строки, либо с использованием утилиты администрирования Webmin, либо путем непосредственного редактирования файлов (последним лучше не заниматься без особой нужды).

Рассмотрим добавление пользователя командой `useradd`.

Наиболее употребительные ключи команды представлены в табл. 9.7.

***Таблица 9.7.*** *Ключи команды useradd*

| Ключ | Назначение |
|---|---|
| `-d` | Для нового пользователя будет создан домашний каталог. По умолчанию система добавляет имя пользователя к домашнему каталогу по умолчанию |
| `-e` | Дата, когда учетная запись пользователя будет отключена. Указывается в формате ГГГГ-ММ-ДД |
| `-f` | Через столько дней после истечения срока действия пароля учетная запись пользователя будет навсегда отключена. Значение 0 отключает использование учетной записи сразу после окончания действия пароля, а значение –1 отключает все описанные тут возможности. По умолчанию устанавливается в –1 |
| `-g` | Имя существующей группы или номер группы пользователя. Имя группы должно существовать. Номер группы должен соответствовать уже существующей группе. По умолчанию номер группы устанавливается в 1 |

***Таблица 9.7*** *(окончание)*

| Ключ | Назначение |
|---|---|
| -G | Список дополнительных групп, членом которых является пользователь. Группы разделяются запятыми, пробелы и пропуски запрещены. Ограничения для групп тут такие же, как и для групп в параметре -g. По умолчанию пользователь принадлежит только к начальной группе |
| -n | По умолчанию вместе с созданием пользователя создается группа с таким же именем. Данный ключ отключает эту специфическую для Red Hat особенность |
| -r | Этот параметр используется для создания системных учетных записей, то есть пользователя с идентификатором меньшим, чем минимальный идентификатор UID_MIN, определенный в файле /etc/login.defs. Заметим, что useradd не создаст домашний каталог для такого пользователя, даже если это определено в файле настроек /etc/login.defs. Вам необходимо указать -m, если хотите создать домашний каталог для системной учетной записи |
| -p | Шифрованный пароль пользователя, возвращаемый crypt или генератором паролей MD5. По умолчанию учетная запись отключена |
| -s | Имя оболочки входа пользователя. По умолчанию это поле остается пустым, что указывает системе устанавливать оболочку по умолчанию |

Допустим, нам необходимо ввести пользователя с именем user_01, входящим в группу group_01 (основная). Для добавления нового пользователя необходимо ввести команду

```
#useradd —c "My first User number 01"—g group_01 user_01.
```

Здесь мы создаем пользователя user_01. Группа group_01 должна уже существовать. В поле комментария записываем "My first User number 01".

Если вы хотите при этом создать группу, то строка должна быть

```
#useradd—c "My first User number 01"—g `groupadd group_01`  gser_01.
```

Посмотрим теперь, как те же действия можно выполнить при помощи утилиты Webmin. Утилита Webmin работает только в графическом режиме. Для входа в режим наберите с клавиатуры:

```
startx.
```

Webmin по умолчанию работает через порт 10 000. Выяснить, запущена ли утилита Webmin, можно командой

```
netstat -ln.
```

Если в списке процессов появятся строки, связанные с портом 10 000, то значит утилита работает. Запомните адрес, с которым связана эта утилита (рис. 9.1). Строка

```
0.0.0.0
```

означает любой адрес. Однако на деле надо вводить либо IP-адрес сервера, либо (если вы работаете с сервера) можно ввести:

```
127.0.0.1.
```

```
[root@localhost root]# netstat -ln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:6000            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:10000           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
udp        0      0 0.0.0.0:10000           0.0.0.0:*
udp        0      0 0.0.0.0:475             0.0.0.0:*
udp        0      0 0.0.0.0:111             0.0.0.0:*
Active UNIX domain sockets (only servers)
Proto RefCnt Flags       Type       State         I-Node Path
unix  2      [ ACC ]     STREAM     LISTENING     1766   /dev/gpmctl
unix  2      [ ACC ]     STREAM     LISTENING     1843   /tmp/.font-unix/fs7100
unix  2      [ ACC ]     STREAM     LISTENING     2302   /tmp/.X11-unix/X0
unix  2      [ ACC ]     STREAM     LISTENING     2348   /tmp/.ICE-unix/1480
unix  2      [ ACC ]     STREAM     LISTENING     2404   /tmp/.sawfish-root/localhost.localdomain:0.0
unix  2      [ ACC ]     STREAM     LISTENING     2498   /tmp/orbit-root/orb-2554822351232316616
unix  2      [ ACC ]     STREAM     LISTENING     2479   /tmp/orbit-root/orb-4382324014662482 50
unix  2      [ ACC ]     STREAM     LISTENING     2545   /tmp/orbit-root/orb-5927286401100424084
unix  2      [ ACC ]     STREAM     LISTENING     2550   /tmp/orbit-root/orb-14299163011858021831
unix  2      [ ACC ]     STREAM     LISTENING     2586   /tmp/orbit-root/orb-1975692173940929909
[root@localhost root]#
```

**Рис. 9.1.** Запуск netstat

**Рис. 9.2.** Запуск браузера

Теперь запускаем браузер и, если вы следовали указанным инструкциям, это будет Mozilla. На рис. 9.2 приведен пример запуска.

В адресной строке набираем IP-адрес вашего сервера. Теперь нажимаем <Enter>.

Появится окно, как на рис. 9.3. В нем необходимо ввести имя пользователя (в данном случае root) и пароль.

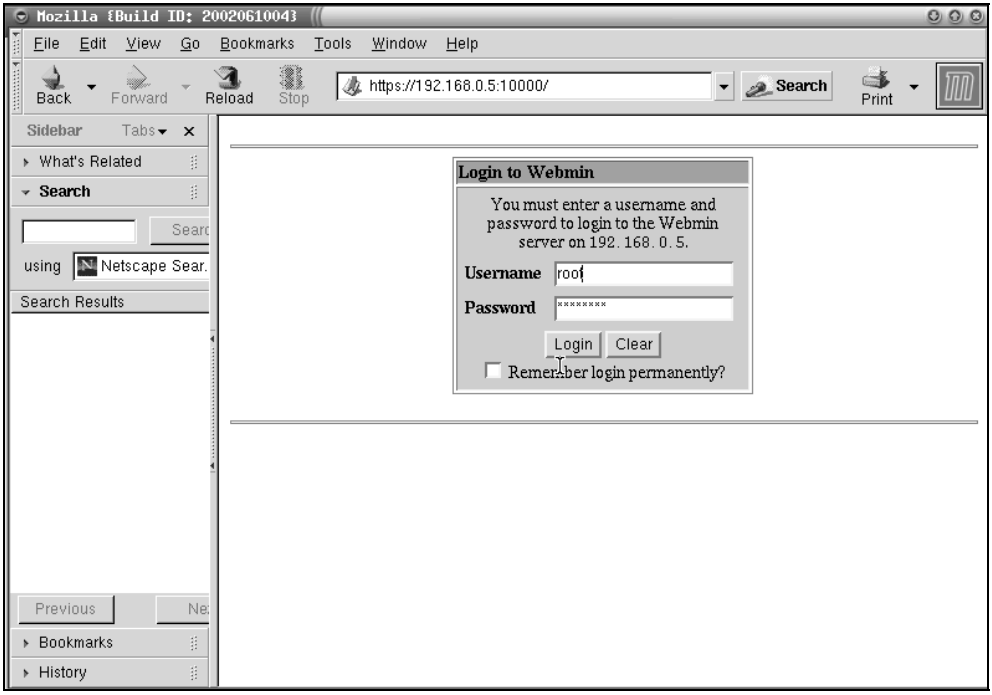



**Рис. 9.3.** Авторизация в Webmin

После авторизации перед вами появится главное окно Webmin. Для удобства правую панель лучше закрыть, тогда перед вами появится следующее окно, представленное на рис. 9.4.

Последовательно выбираем вкладку **System**, пункт **Users and Groups**.

Мы оказались в окне **Users and Groups**. Здесь отображаются таблица пользователей и таблица групп. Для создания или удаления пользователей и групп предназначены специальные ссылки. Они находятся вверху и внизу таблицы. Выбираем ссылку **Create a new user**, в открывшемся окне заполняем необходимые поля (рис. 9.5). Их названия говорят сами за себя.

Осталось нажать кнопку **Create**.

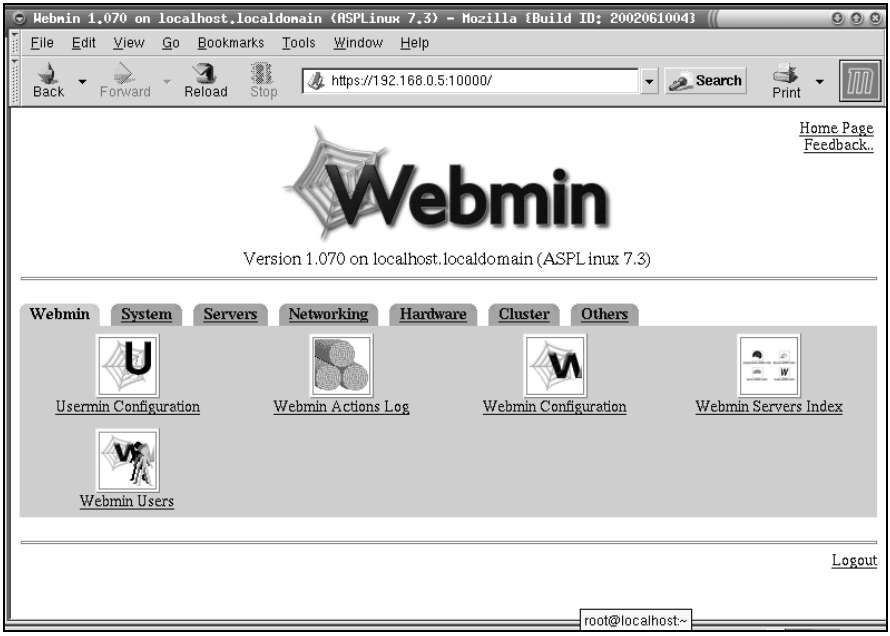



**Рис. 9.4.** Главное окно Webmin

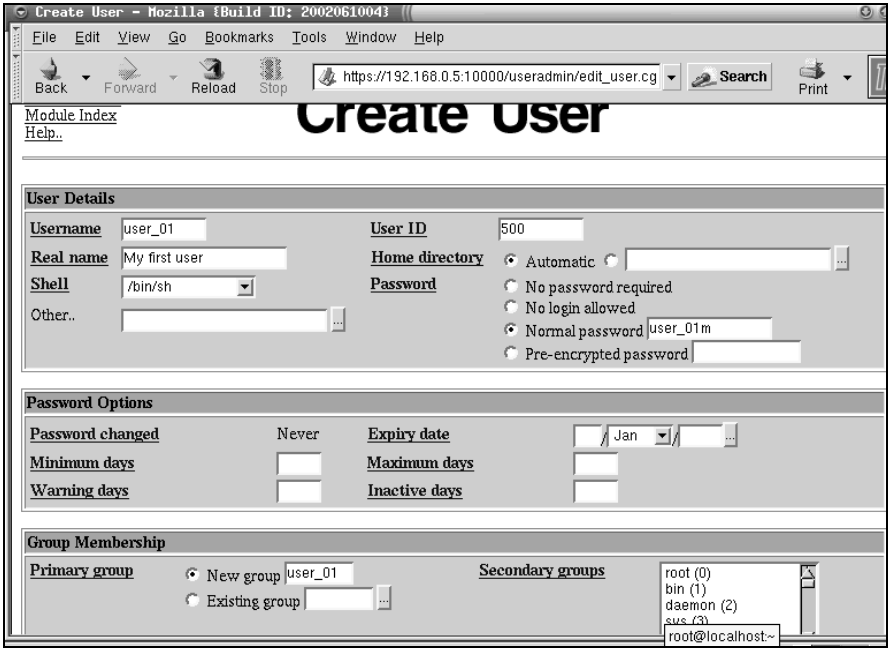



**Рис. 9.5.** Создание нового пользователя

Новый пользователь добавлен в систему. Теперь рассмотрим, как можно модифицировать нашего пользователя.

## Модификация существующего пользователя

Модифицировать существующего пользователя можно так же, как и в предыдущем случае тремя способами: либо из командной строки, либо с использованием утилиты администрирования Webmin, либо путем непосредственного редактирования файлов (опять же, последним лучше не заниматься без особой нужды).

Изменить существующего пользователя можно при помощи команды `usermod`. Наиболее употребительные ключи команды представлены в табл. 9.8.

***Таблица 9.8.*** *Ключи команды usermod*

| Ключ | Назначение |
|---|---|
| `-d` | Новый домашний каталог пользователя. Если указан ключ `-m`, то все содержимое текущего каталога пользователя перемещается в новый каталог (если его не существует, то он будет создан) |
| `-e` | Дата, когда учетная запись пользователя будет отключена. Указывается в формате ГГГГ-ММ-ДД |
| `-f` | Интервал после истечения срока действия пароля. Значение 0 отключает использование учетной записи сразу после окончания действия пароля, а значение –1 отключает все описанные ранее возможности |
| `-g` | Имя существующей группы или номер новой группы пользователя. Имя группы должно существовать. Номер группы должен соответствовать уже существующей группе. По умолчанию номер группы устанавливается в 1 |
| `-G` | Список дополнительных групп, членом которых является пользователь. Группы разделяются запятыми, пробелы запрещены. Ограничения для групп тут такие же, как и для групп в параметре `-g`. Если пользователь являлся пользователем группы, не перечисленной тут, то он будет удален из этой группы |
| `-l` | Имя пользователя будет изменено на новое. Больше ничего не должно изменяться. Но, видимо, будет изменено имя домашнего каталога пользователя в соответствии с его новым именем |
| `-p` | Новый шифрованный пароль пользователя, возвращаемый от `crypt(3)` |

***Таблица 9.8*** *(окончание)*

| Ключ | Назначение |
|---|---|
| -s | Имя новой оболочки входа пользователя. Сброс этого поля в пустое значение установит оболочку, определенную по умолчанию в системе |
| -u | Числовое значение идентификатора пользователя. Это значение должно быть уникальным, если только не используется параметр -o. Значение должно быть неотрицательным. Значения между 0 и 99 обычно резервируются для системы. Все файлы, владельцами которых является пользователь, расположенные в домашнем каталоге пользователя, автоматически поменяют идентификатор владельца. У файлов вне домашнего каталога необходимо сменить владельца вручную |
| -L | Блокирует пароль пользователя. Перед зашифрованным паролем помещается символ !. Этот ключ нельзя использовать вместе с ключами -p или -U |
| -U | Снимает блокировку пароля пользователя. Убирает символ ! перед паролем. Нельзя использовать этот параметр вместе с параметрами -p или -L |

Рассмотрим пример:

```
# usermod—c "Modify string"—g group_02 user_01.
```

В этом случае мы заменили у пользователя user_01 комментарий на "Modify string" и группу по умолчанию на group_02.

Выполним те же действия при помощи Webmin.

1. Запуск утилиты осуществляем, как и в предыдущем примере.
2. После открытия главного окна программы выбираем на вкладке **System** пункт **Users and Groups**.
3. В окне **Users and Groups** отображаются таблица пользователей и таблица групп. Находим там строку **user_01** и щелкаем по ней мышью — открывается окно редактирования пользователя (рис. 9.6).
4. В нем исправляем необходимые поля.

Данные будут модифицированы, а вы снова попадете в список пользователей и групп.

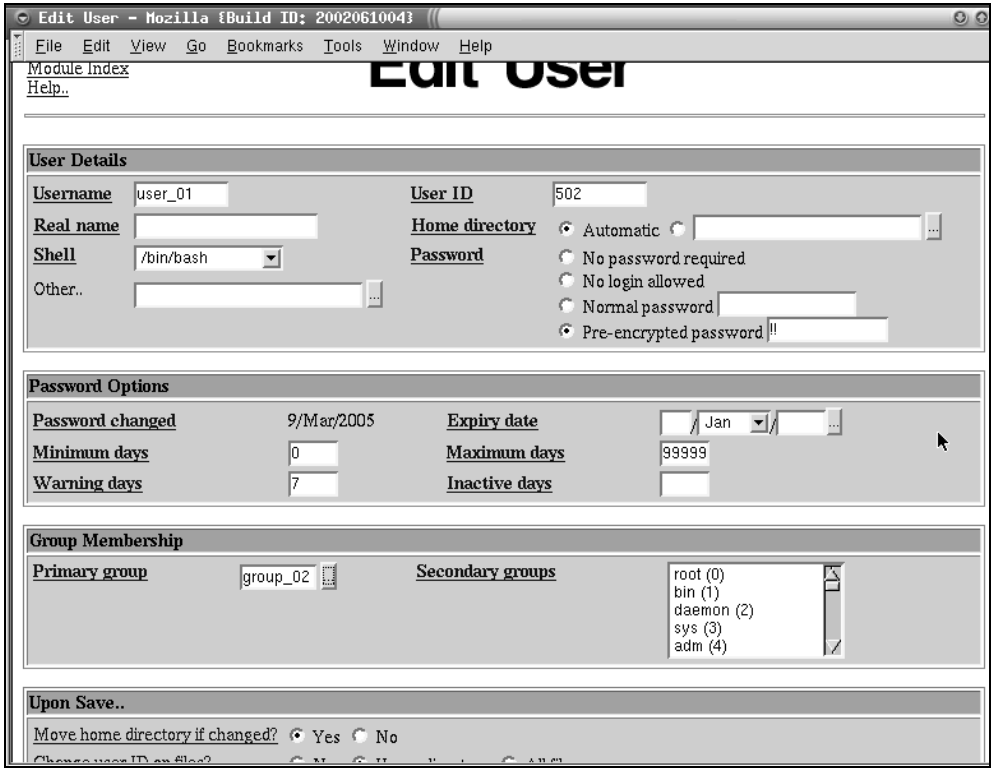



**Рис. 9.6.** Модификация существующего пользователя

## Удаление пользователя

Для удаления пользователя существует команда `userdel`. Команда `userdel` удаляет все ссылки на учетную запись из файлов /etc/passwd, /etc/shadow, /etc/usergroup. Каталог пользователя по умолчанию не уничтожается.

Синтаксис команды:

```
userdel [-r] имя_пользователя.
```

Команда `userdel` изменяет системные файлы учетных записей, удаляя все элементы, связанные с именем пользователя. Указанный пользователь должен существовать. Ключ `-r` означает, что файлы в домашнем каталоге пользователя будут удалены вместе с самим каталогом и почтовым спулером. Файлы других файловых систем должны быть обнаружены и удалены вручную.

Допустим, нам необходимо удалить пользователя `user_01`, тогда выполним

```
#userdel -r user_01.
```

Пользователь будет удален из системы.

Рассмотрим выполнение удаления в Webmin.

1. В главном окне на вкладке **System** выбираем **Users and Groups**. Появится уже знакомое нам окно со списком пользователей и групп.
2. Щелкнув мышью по имени пользователя (**user_01**), мы попадаем в окно редактирования пользователя.
3. Спускаемся при помощи полосы прокрутки в самый низ (рис. 9.7).

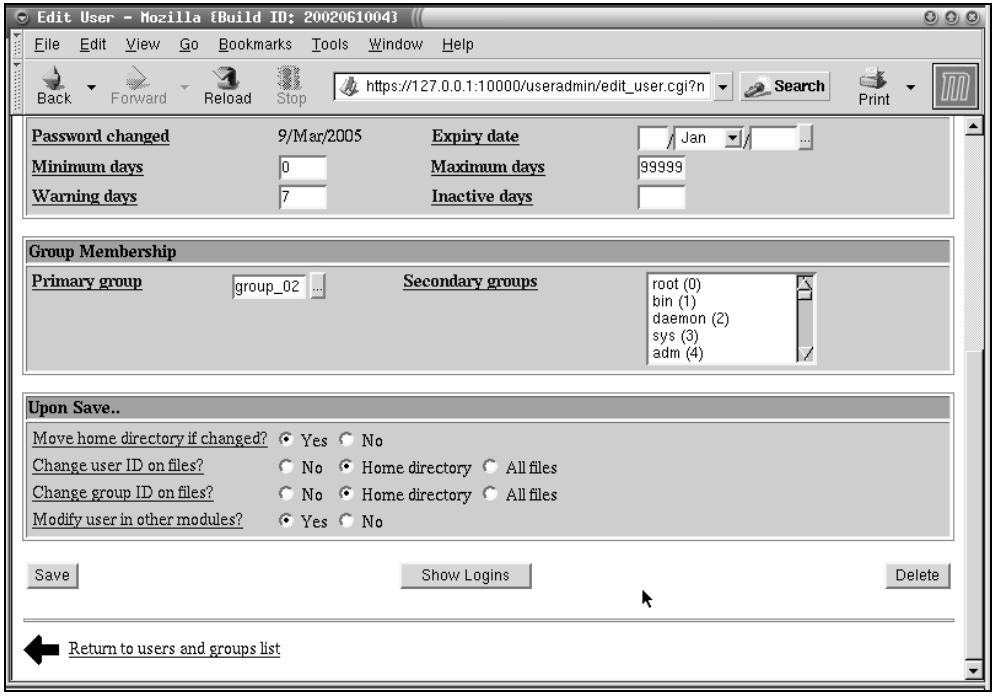



**Рис. 9.7.** Удаление пользователя

4. Теперь щелкнем по кнопке **Delete** на экране. Появится окно с двумя кнопками:
   - **Delete User** (Удалить пользователя);
   - **Delete User and Home Directory** (Удалить пользователя и домашнюю директорию).
5. Нажимаем **Delete User and Home Directory**. Пользователь удален. Осталось только вернуться на заглавную страницу, щелкнув в правом верхнем углу окна **Webmin Index**.

# 9.3. Процессы. Управление процессами. Останов системы

## Общая характеристика процессов

Процессом называется выполняющаяся программа. Все процессы: и системные, и прикладные — управляются одинаковыми командами.

В текущий момент времени системой может выполняться только один процесс. Однако операционная система переключается между процессами, предоставляя им по очереди свои ресурсы, в результате получается многозадачная операционная система.

В Linux существует три вида процессов:

- системные процессы (процесс init);
- процессы-демоны, выполняющиеся в фоновом режиме. В отлаженной системе демоны обычно запускаются при загрузке операционной системы и обеспечивают работу печати, http-сервис, samba-сервис;
- прикладные процессы.

Рассмотрим ряд характеристик процесса (табл. 9.9).

***Таблица 9.9.*** *Некоторые характеристики процессов*

| Характеристика | Комментарий |
|---|---|
| PID (Process ID) | Идентификатор процесса, назначенный ему ядром операционной системы по порядку. После завершения процесса его идентификатор высвобождается и может быть присвоен другому процессу |
| PPID (Parent Process ID) | Идентификатор процесса, породившего данный процесс. Исходный процесс называется родителем, а порожденный процесс дочерним процессом |
| UID (User ID) | Идентификатор пользователя, запустившего данный процесс |
| EUID (Effective UID) | Эффективный идентификатор пользователя служит для определения прав доступа данного процесса к системным ресурсам. У большинства процессов UID и EUID совпадают |
| GID (Group ID) | Идентификатор группы пользователя, породившего процесс |
| EGID (Effective GID) | Эффективный идентификатор группы пользователя служит для определения прав доступа данного процесса к системным ресурсам |

Важным является и понятие контекста процесса — это вся информация, необходимая для описания процесса:

- адресное пространство процесса (код, данные, стек процесса и другие области памяти, имеющие отношение к процессу);
- управляющая информация (тип управляющей структуры процесса);
- среда процесса (переменные среды, значения которых задаются непосредственно или наследуются от родителя);
- аппаратный контекст (значение регистров процесса).

Переключение между процессами, по сути, сводится к переключению между контекстами.

В ходе своего жизненного цикла процесс может находиться в одном из четырех состояний:

1. Выполнение — процесс выполняется и периодически получает доступ к системным ресурсам.
2. Ожидание выделения ресурсов системы — процесс переходит в это состояние, если он не может завершиться немедленно. Тогда процесс переходит в состояние ожидания и ждет наступления определенного события. Ресурсы данному процессу выделяются только после его пробуждения.
3. Зомби — в это состояние процесс переходит после выполнения команды `exit`. По сути, этого процесса уже не существует, но остаются записи, содержащие сведения о нем, и информация о завершении еще не поступила к породившему его процессу.
4. Останов — состояние аналогично ожиданию, однако выход из состояния останова возможен только с помощью другого процесса. Процесс входит в режим останова по команде `STOP`. Для того чтобы процесс восстановил работу, необходимо послать команду `CONT`.

## Получение информации о процессах

Если говорить о командной строке, то команда `ps` является основным инструментом для получения информации о процессе. Она выдает информацию об идентификаторах, приоритете, объеме памяти, затратах времени центрального процессора.

Команда `ps` может выполняться с ключами (табл. 9.10).

***Таблица 9.10.*** *Ключи команды ps*

| Ключ | Значение |
|---|---|
| `a` | Информация обо всех процессах в терминале |
| `x` | Информация о процессах за текущим терминалом |
| `u` | Выдавать информацию в формате, ориентированном на пользователя |

***Таблица 9.10*** *(окончание)*

| Ключ | Значение |
|---|---|
| -u | Выдавать информацию только по выбранному пользователю |
| -t | Выдавать информацию по процессам, имеющим отношение к указанному терминалу |
| -g | Выдавать информацию только по определенной группе |
| -e | Предоставлять информацию обо всех процессах |
| -l | Выдает длинный листинг |
| -f | Выдает полный листинг |

Для просмотра всех процессов в системе можно воспользоваться командой: `ps aux`.

```
[root@localhost root]# ps aux
USER       PID %CPU %MEM   VSZ  RSS TTY      STAT START   TIME COMMAND
root         1  0.1  0.4  1376  416 ?        S    00:08   0:02 init
root         2  0.0  0.0     0    0 ?        SW   00:08   0:00 [keventd]
root         3  0.0  0.0     0    0 ?        SW   00:08   0:00 [kapmd]
root         4  0.0  0.0     0    0 ?        SWN  00:08   0:00 [ksoftirqd_CPU0]
root         5  0.0  0.0     0    0 ?        SW   00:08   0:01 [kswapd]
root         6  0.0  0.0     0    0 ?        SW   00:08   0:00 [bdflush]
root         7  0.0  0.0     0    0 ?        SW   00:08   0:00 [kupdated]
root         8  0.0  0.0     0    0 ?        SW   00:08   0:00 [pagebufd]
root         9  0.0  0.0     0    0 ?        SW   00:08   0:00 [pagebuf_io_CPU0]
root        10  0.0  0.0     0    0 ?        SW   00:08   0:00 [mdrecoveryd]
root       561  0.0  0.5  1436  516 ?        S    00:11   0:00 syslogd -m 0
root       567  0.0  0.3  1372  360 ?        S    00:11   0:00 klogd -x
rpc        587  0.0  0.4  1512  412 ?        S    00:11   0:00 portmap
root       724  0.0  0.4  1368  400 ?        S    00:11   0:00 /usr/sbin/apmd -p 10 -w 5 -W -P /etc/sysconfig
root       798  0.0  0.3  1432  364 ?        S    00:11   0:00 /usr/sbin/hasplm
root       816  0.0  0.7  2632  728 ?        S    00:11   0:00 /usr/sbin/sshd
root       849  0.0  0.6  2256  588 ?        S    00:11   0:00 xinetd -stayalive -reuse -pidfile /var/run/xin
root       890  0.0  1.1  4748 1128 ?        S    00:11   0:00 sendmail: accepting connections
root       909  0.0  0.3  1412  376 ?        S    00:11   0:00 gpm -t ps/2 -m /dev/mouse
root       927  0.0  0.6  1584  600 ?        S    00:11   0:00 crond
xfs       1001  0.0  0.9  5544  900 ?        S    00:11   0:01 xfs -droppriv -daemon
daemon    1037  0.0  0.4  1412  444 ?        S    00:11   0:00 /usr/sbin/atd
root      1053  0.0  2.5  6348 2480 ?        S    00:11   0:01 /usr/bin/perl /usr/libexec/webmin/miniserv.pl
root      1057  0.0  0.6  2384  644 ?        S    00:11   0:00 login -- root
root      1058  0.0  0.3  1352  308 tty2     S    00:11   0:00 /sbin/mingetty tty2
root      1059  0.0  0.3  1352  308 tty3     S    00:11   0:00 /sbin/mingetty tty3
root      1060  0.0  0.3  1352  308 tty4     S    00:11   0:00 /sbin/mingetty tty4
root      1061  0.0  0.3  1352  308 tty5     S    00:11   0:00 /sbin/mingetty tty5
root      1062  0.0  0.3  1352  308 tty6     S    00:11   0:00 /sbin/mingetty tty6
root      1065  0.0  1.0  2492  976 tty1     S    00:12   0:00 -bash
root      1109  0.0  0.8  2248  868 tty1     S    00:12   0:00 /bin/sh /usr/X11R6/bin/startx
root      1120  0.0  0.4  2284  484 tty1     S    00:12   0:00 xinit /etc/X11/xinit/xinitrc --
root      1121  2.9  3.8 16656 3756 ?        S<   00:12   1:12 X :0
```

**Рис. 9.8.** Получение информации о процессах

Как видим, команда `ps` выдает свой отчет в виде таблицы (рис. 9.8). Каждому процессу соответствует отдельная строка. В самой верхней строке расписаны значения полей, которые выдает эта команда. Раскроем некоторые из них (табл. 9.11).

***Таблица 9.11.*** *Основные характеристики полей*

| Поле | Значение |
|---|---|
| %CPU | Процент загрузки CPU |
| %MEM | Процент использования памяти |
| C | Доля выделенного планировщиком времени |
| PID | Идентификатор процесса |
| PPID | Идентификатор родительского процесса |
| PRI | Текущий приоритет процесса |
| STIME | Время запуска процесса |
| SZ | Размер образа процесса в памяти в блоках (1 блок = 512 байтов) |
| TIME | Время выполнения процесса |
| TTY | Терминал, в котором выполняется процесс |
| UID | Идентификатор владельца процесса |
| WCHAN | Адрес события, которого ожидает процесс |

Для получения информации о процессах при помощи утилиты Webmin необходимо открыть вкладку **System** и выбрать **Running Processes**. Результатом будет окно, представленное на рис. 9.9.

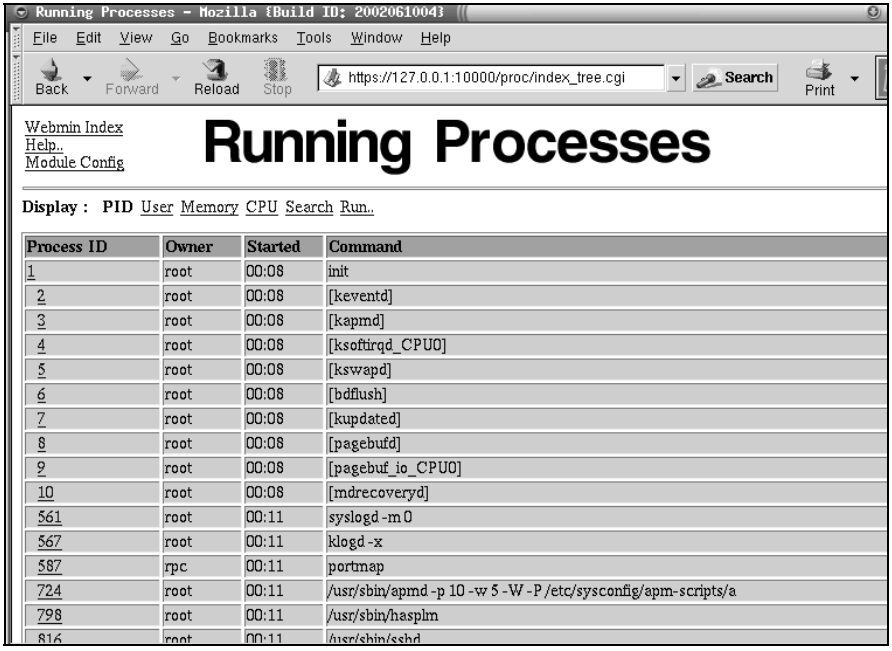



**Рис. 9.9.** Информация о процессах в утилите Webmin

Если щелкнуть мышью по имени процесса, можно узнать о нем подробную информацию, а также изменить его параметры (рис. 9.10).

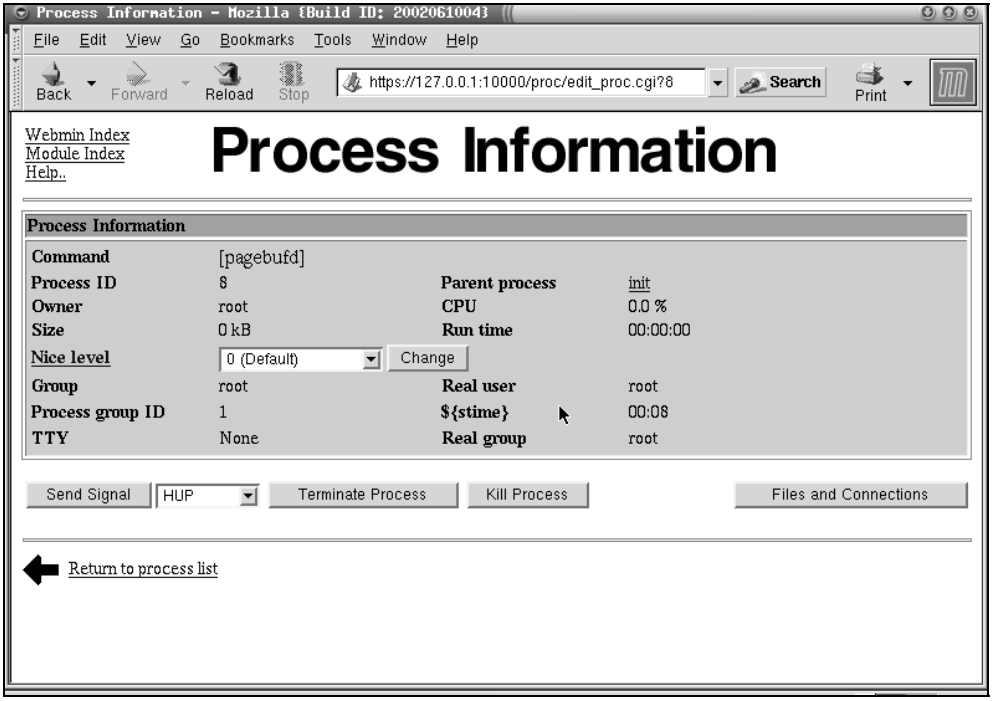



**Рис. 9.10.** Информация о процессе и управление им

## Управление приоритетом процессов

Существует такое понятие, как фактор уступчивости процесса. Это число, по которому ядро определяет, как себя вести в отношении этого процесса. По сути, фактор уступчивости определяет приоритет процесса. Чем выше фактор уступчивости, тем ниже приоритет процесса, и наоборот.

Фактор уступчивости может находиться в пределах −20 до + 20. По умолчанию дочерний процесс наследует приоритет своего родителя. Владелец процесса может только увеличивать фактор уступчивости, то есть снижать приоритет процесса. Сделано это для того, чтобы избежать порождений процессов с высоким приоритетом выполнения. Только root может в полной мере управлять уступчивостью того или иного процесса. Фактор уступчивости может быть установлен при создании процесса, это делается при помощи команды `nice`, команда `renice` изменяет приоритет текущего процесса.

Рассмотрим синтаксис команд:

`nice` [параметр] [ команда [аргумент]].

Например:

`#nice [-][+]n -10 mc`.

В этом примере мы повышаем приоритет процесса, так как уменьшаем его уступчивость.

Без указания параметров команда выдает текущий приоритет работы.

Для `renice` примером может служить

`renice +1 -u daemon root`.

Здесь изменяем приоритет для всех процессов, владельцы которых root и daemon.

Для изменения приоритета процесса в утилите Webmin выбираем вкладку **System** и пункт **Running Processes** (рис. 9.9). В окне щелкнем по интересующему нас процессу и, помимо необходимой информации, можем изменить уступчивость процесса (рис. 9.10).

## Уничтожение процессов

Для уничтожения процессов используется команда `kill`. Рядовые пользователи могут уничтожать только порожденные ими процессы, суперпользователь (root) может убить любой процесс.

Синтаксис команды `kill`:

`kill` [сигнал] pid,

где сигналы представлены в табл. 9.12.

***Таблица 9.12.*** *Таблица сигналов*

| Имя | номер | Описание |
|---|---|---|
| HUP | 1 | Отбой |
| INT | 2 | Прерывание |
| QUIT | 3 | Выход |
| KILL | 9 | Уничтожение |
| TERM | 15 | Запрос на завершение |
| STOP | – | Останов |
| CONT | – | Продолжение после останова |

Большинство сигналов приводят к завершению процесса. Однако только KILL и STOP нельзя ни заблокировать, ни перехватить.

Послать процессу необходимый сигнал или просто уничтожить его можно и при помощи утилиты Webmin. Для этого во вкладке **System** щелкнем по **Running Processes**, а в открывшемся окне — по интересующему нас процессу. Результатом будет уже знакомое нам окно (рис. 9.11). Далее наши действия зависят от того, что мы хотим сделать. Если послать сигнал, то выбираем необходимый сигнал и нажимаем **Send Signal**, если убить процесс, то **Kill Process**.

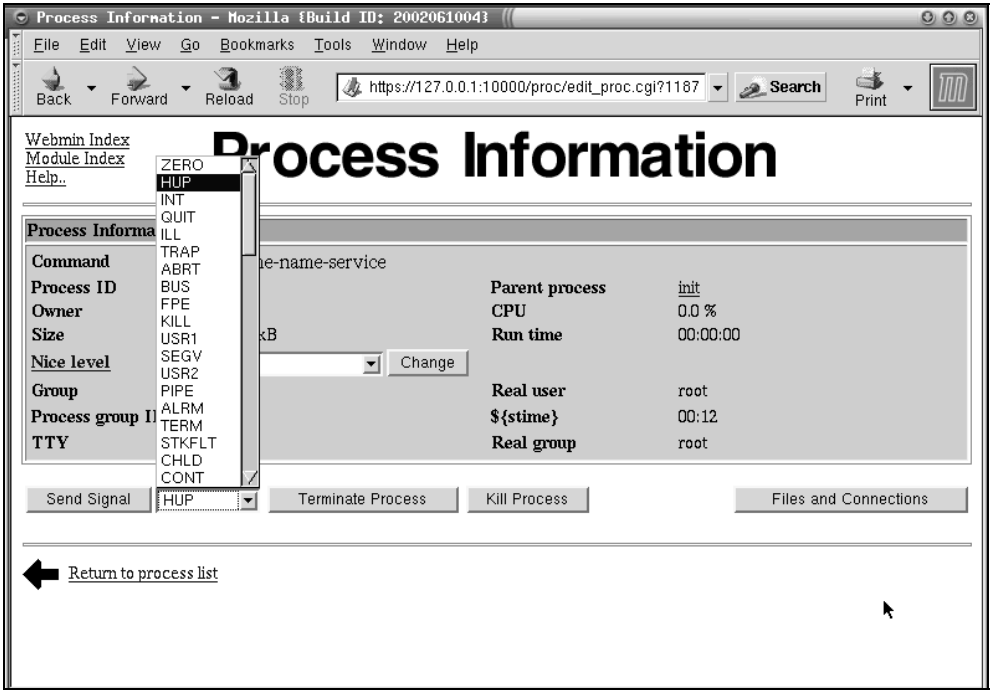



**Рис. 9.11.** Уничтожение процесса в Webmin

## Останов системы

Как и в других операционных системах, непосредственное выключение питания, путем выдергивания шнура из розетки — не лучший вариант останова системы. Это и понятно, однако в компьютерах с форм-фактором AT нажатие кнопки на корпусе аналогично выдергиванию шнура из розетки. Конечно, в современных компьютерах с ATX-блоками такого не произойдет.

Причин, по которым не следует выключать компьютер непосредственно, множество: от чисто технических до особенностей реализации операционной системы. В Linux реализовано кэширование работы с жесткими дисками. Это значит, что перед непосредственной записью на диск данные буферизуются в памяти компьютера и сохраняются только в определенные моменты времени. Это позволяет сгладить неравномерность обращений к жесткому диску и повысить производительность его работы. Однако в случае пропадания питания кэшируемые данные потеряются.

Даже если данные принудительно сохранены, все равно такой способ — не лучшее решение проблемы.

Существует несколько способов остановки системы:

- Команда `shutdown` — самый корректный и безопасный способ выключения питания. Команде `shutdown` при помощи опций можно задать ряд параметров, определяющих ее поведение. Например, опция `-r` приведет к перезагрузке системы, а опция `-h` к ее остановке. Можно также задать задержку выключения и предусмотреть посылку пользователям уведомления о том, что система выключится через определенное время.
- Команда `halt` это полный аналог `shutdown -h`. Команда корректно заканчивает все процессы и выключает компьютер.
- Нажатие трех волшебных кнопок <Ctrl>+<Alt>+<Del> также приведет к корректному выключению системы, правда, пользователи уже не будут уведомлены.

# 9.4. Файловая система. Структура каталогов. Работа с файлами

## Понятие файловой системы

Термин "файловая система" используется в двух значениях:

- физический (способ хранения информации);
- иерархическая (структура файлов и каталогов).

Структура каталогов похожа на перевернутое дерево. Самым верхним каталогом, от которого начинается файловая система, является корневой каталог (root directory). Обозначается он так символом /.

В Linux есть несколько типов файловых систем, однако основными являются системы Ext2 и Ext3. В табл. 9.13 приведены некоторые параметры Ext2.

***Таблица 9.13.*** *Файловая система Ext2*

| Параметр | Значение |
|---|---|
| Максимальная длина имени файла (символов) | 255 |
| Максимальный размер файла (Гбайт) | 2 |
| Максимальный размер файловой системы (Тбайт) | 4 |
| Минимальный размер блока (байт) | 1024 |

Каждый диск, физический или логический, имеет свою файловую систему, которая создается при разбиении диска на разделы (рис. 9.12).

**Рис. 9.12.** Структура системы Ext2

Важным отличием Linux и Windows является форма представления дискового пространства. В Windows каждый логический или физический диск предстает пользователю в виде отдельного устройства. В Linux все дисковое пространство, включающее в себя набор физических и логических дисков, представляется единым деревом. В Linux вы не найдете диска D:\, физический или логический диск просто обозначен определенным каталогом.

## Монтирование и демонтирование файловой системы

Отдельная файловая система является либо логическим разделом диска, либо физическим диском (отдельным винчестером). Изначально файловая

система недоступна для пользователей. Для того чтобы файловая система стала доступна, ее необходимо монтировать командой `mount`:

Синтаксис команды `mount`:

Пример монтирования:

```
#mount /dev/hda2 /home.
```

Эта команда монтирует файловую систему диска hda2 (соглашение о наименовании устройств смотри далее) к каталогу /home корневой файловой системы.

При загрузке системы возможно автоматическое монтирование постоянно используемых файловых систем. Список автоматически монтируемых файловых систем находится в файле /etc/fstab (листинг 9.6).

**Листинг 9.6. Файл /etc/fstab**

```
# /etc/fstab: static file system information.
#
# <file system> <mount point> <type> <options>              <dump> <pass>
/dev/hda7      none    swap    sw      0       0
/dev/hda6      /       ext2    defaults,errors=remount-ro   0       1
proc    /proc   proc    defaults        0       0
none                    /dev/shm                tmpfs   defaults       0 0
none    /dev/pts/       devpts  gid=5,mode=620 0       0
/dev/cdrom /mnt/cdrom iso9660 noauto,owner,kudzu,ro,iocharset=koi8-r
/dev/fd0 /mnt/floppy auto noauto,owner,kudzu,iocharset=koi8-r
/dev/hda5 /mnt/Disk_D vfat iocharset=koi8-r,user,umask=0,auto 0 1
```

Путем редактирования этого файла можно добавлять файловые системы, которые будут монтироваться автоматически. Для того чтобы это правильно сделать, нужно разобраться с форматом файла /etc/fstab. Как видно, каждая файловая система представлена в нем отдельной строкой.

Содержимое каждой строки состоит из полей, разделенных пробелами:

- имя устройства, которое мы монтируем;
- точка монтирования;
- тип файловой системы в этом разделе;
- опции монтирования, которые могут определять права доступа, тип кодировки и многое другое.

Для того чтобы монтировать раздел с файловой системой Fat32 (используемой Windows) к файловой системе Linux, необходимо в файле /etc/fstab добавить строку вида

```
/dev/hda1 /mnt/disk_c vfat iocharset=koi8-r,codepage=866,auto,user 0 0.
```

Обратным монтированию является демонтирования. Выполняется демонтирование при помощи команды `unmount`. Если файловая система занята, то есть находится в работе, то демонтировать ее невозможно.

Синтаксис команды `unmont` следующий:

`umount` [`-O` опции] устройство точка_монтирования.

## Соглашение об именовании устройств

В Linux приняты следующие правила именования носителей (табл. 9.14):

***Таблица 9.14.*** *Соглашение о наименовании физических носителей в Linux*

| Наименование | Комментарий |
|---|---|
| hda | Первый диск на первом канале IDE (Master) |
| hdb | Второй диск на первом канале IDE (Slave) |
| hdc | Первый диск на втором канале IDE (Master) |
| hdd | Второй диск на втором канале IDE (Slave) |
| sda | Первый SCSI |

Как известно, физические диски могут быть разбиты на логические. То есть на жестком диске могут существовать основной и расширенный разделы. Логические диски в основном разделе нумеруются цифрами от 1 до 4. Логические диски в расширенном разделе нумеруются цифрами от 5 и больше. Например, диск D:\ операционной системы Windows в Linux будет наименоваться как hda5. Оговоримся, что такое справедливо только если диск D — логический диск).

## Структура каталогов файловой системы

Структура каталогов файловой системы Linux подчинена строгой иерархии, определено место каждого каталога, а также характер информации, находящейся в данном каталоге. Некоторые каталоги, исходя из целей безопасности, могут быть оформлены в виде самостоятельной файловой системы. В табл. 9.15 приведены некоторые наиболее важные каталоги и характер их содержимого.

***Таблица 9.15.*** *Структура каталогов и их содержимое*

| Каталог | Содержимое |
|---|---|
| / | Корневой каталог. Основа файловой системы |
| /bin | Находятся системные и пользовательские утилиты |
| /boot | Ядро операционной системы, могут находиться файлы загрузчика |
| /dev | В этом каталоге находятся файлы устройств, связанные с физическими устройствами. Через эти файлы ОС Linux получает доступ к физическим устройствам. Представление любого физического устройства в качестве файла облегчает администрирование системы |
| /etc | Файлы запуска и конфигурации системы |
| /home | В этом каталоге хранятся данные пользователей. Для каждого пользователя здесь создан свой подкаталог |
| /lib | В этом каталоге хранятся совместно используемые библиотеки |
| /lost+foud | Иногда в файловой системе могут возникать сбои. После восстановления ее работоспособности сюда попадают потерянные данные |
| /mnt | Этот каталог по умолчанию используется для монтирования файловых систем. Например, CD-ROM, скорее всего, будет монтирован в каталог /mnt/cdrom |
| /opt | Вспомогательные пакеты программ |
| /proc | По сути, это специальная файловая система. Здесь хранятся образы запущенных процессов |
| /root | Это рабочий каталог суперпользователя |
| /sbin | Утилиты загрузки |
| /tmp | Каталог временных файлов |
| /usr | Дополнительные совместно используемые файлы и программы |
| /var | Предназначен для размещения изменяемых данных |

# Просмотр информации о файле

Для получения информации о файле (его типе, размере, правах доступа, времени создания) используется команда `ls`.

Синтаксис команды `ls`:

`ls` [параметр]... [файл]...

Наиболее употребительные ключи команды представлены в табл. 9.16.

***Таблица 9.16.*** *Ключи команды ls*

| Ключ | Значение |
|---|---|
| `-a` | Не скрывать файлы, начинающиеся с точки (.) |
| `-A` | Не выдавать . и .. |
| `-F` | Добавлять индикатор (один из символов */=@\|), показывающий тип файла |
| `-l` | Использовать широкий формат вывода. Выдаются права доступа, владелец, группа |
| `-r` | Обратный порядок при сортировке |
| `-R` | Выводить подкаталоги рекурсивно |
| `-C` | Выдавать элементы по столбцам |
| `--full-time` | Выводить дату и время объекта |

Результатом работы команды `ls` является листинг файлов. Формат выдаваемой информации зависит от того, с какими ключами была запущена команда `ls`. На рис. 9.13 приведен пример выполнения команды `ls`.

```
root@localhost:~
Файл  Правка  Настройка  Справка
[root@localhost root]# ls -a --full-time
итого 96
drwxr-x---   10 root     root         4096 Срд Мар 16 00:12:24 2005 .
drwxr-xr-x   22 root     root         4096 Срд Мар 16 00:10:58 2005 ..
-rw-------    1 root     root          741 Срд Мар 16 01:18:30 2005 .bash_history
-rw-r--r--    1 root     root           24 Вск Июн 11 01:00:15 2000 .bash_logout
-rw-r--r--    1 root     root          234 Чтв Июл 05 22:23:26 2001 .bash_profile
-rw-r--r--    1 root     root          176 Срд Авг 23 23:04:30 1995 .bashrc
drwx------    2 root     root         4096 Птн Мар 11 20:41:19 2005 .cedit
-rw-r--r--    1 root     root          210 Вск Июн 11 01:09:02 2000 .cshrc
drwxr-xr-x    5 root     root         4096 Срд Мар 16 00:12:32 2005 .gnome
drwxr-xr-x    2 root     root         4096 Птн Мар 11 18:57:57 2005 .gnome-help-browser
drwx------    3 root     root         4096 Птн Мар 11 19:03:04 2005 .gnome_private
-rw-------    1 root     root         1320 Срд Мар 16 00:12:24 2005 .ICEauthority
drwxr-xr-x    3 root     root         4096 Вск Мар 06 15:58:14 2005 .kde
drwxr-xr-x    3 root     root         4096 Срд Мар 16 01:35:05 2005 .mc
drwxr-xr-x    3 root     root         4096 Вск Мар 06 13:25:26 2005 .mozilla
drwxr-xr-x    3 root     root         4096 Вск Мар 06 13:24:29 2005 .sawfish
-rw-r--r--    1 root     root          196 Вто Июл 11 19:53:11 2000 .tcshrc
-rw-------    1 root     root          103 Чтв Мар 10 00:59:35 2005 .Xauthority
-rw-r--r--    1 root     root        17878 Вск Мар 06 15:38:51 2005 .xftcache
-rw-r--r--    1 root     root         1126 Срд Авг 23 23:02:38 1995 .Xresources
[root@localhost root]#
```

**Рис. 9.13.** Выполнение команды `ls`

# Типы файлов

В Linux выделяют несколько типов файлов.

- Обычный файл — наиболее общий тип файлов. Он представляет собой последовательность байтов. Это может быть текстовый документ, программа, файл базы данных. Признаком обычного файла является символ дефиса в первой позиции первого столбца. Обычные файлы создаются либо методом копирования, либо при помощи текстовых редакторов.
- Каталог — это файл, содержащий ссылки на хранящиеся в нем файлы. Читать содержимое каталога может любой процесс, имеющий на это право. Записывать содержимое каталога может только ядро системы. Создается каталог командой `mkdir`. Удаляется командой `rmdir`. Признаком каталога является символ d в позиции первого столбца.
- Файлы устройств (символьные и блочные) обеспечивают доступ к физическим устройствам. Доступ к устройству осуществляется путем открытия файла, чтения из него или записи в него:
  - символьные (character special device) файлы устройств используются для небуферизованного обмена данными с устройством. Признаком символьного файла является символ c в первой позиции первого столбца;
  - блочные (block special device) файлы позволяют вести обмен данными с устройством блоками, то есть пакетами фиксированной длины. Признаком блочного файла является символ b в первой позиции первого столбца. Файлы устройств может создавать только root. Создаются командой `mknod`, удаляются командой `rm`.
- Символическая ссылка. Вместо имени файла символическая ссылка указывает его псевдоним. Это аналог ярлыка в Windows. При обращении к ссылке обмен данными будет вестись не с ней, а тем файлом, на который эта ссылка указывает. Признаком символической ссылки является символ l в первой позиции первого столбца. Создается ссылка командой `ln`, удаляется командой `rm`
- Именованные каналы осуществляют взаимодействие между двумя процессами по принципу очереди FIFO (First Input First Output). Признаком именованного канала является символ p в первой позиции первого столбца. Создаются именованные каналы командой `mknod`, удаляются командой `rm`..
- Сокеты позволяют представить в виде файла сетевое соединение. Признаком сокета является символ s в первой позиции первого столбца. Создается сокет командой `socket`. Удаляется командой `rm`.

# Права доступа

Файловая система хранит для каждого файла несколько десятков информационных полей. Большинство из этих полей используются самой операционной системой. Администраторов, как правило, интересует тип файла (файл, каталог или ссылка), владелец файла, группа, права доступа, размер файла и время последнего обращения.

В Linux для файлов существуют три уровня доступа:

- владелец (owner);
- группа (group);
- все остальные (other).

В рамках каждой категории права разделяются еще на три уровня доступа:

- чтение (read);
- запись (write);
- выполнение (execute).

Пояснение правила чтения строк приводится на рис. 9.14.

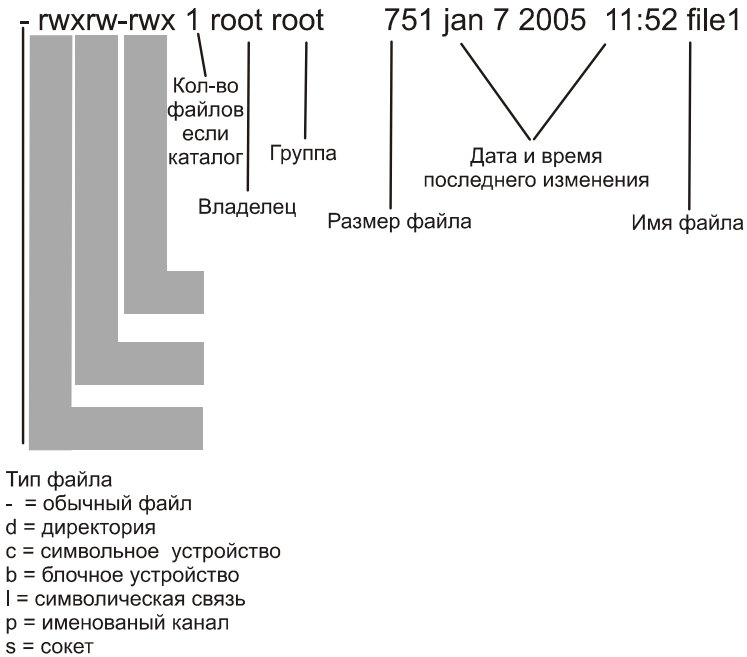



**Рис. 9.14.** Правила чтения информации о файле

# Изменение прав доступа

Изменение прав доступа к файлу осуществляется при помощи команды `chmod`. Изменить права доступа может только владелец файла и root.

Синтаксис:

`chmod` [`-fR`] абсолютные_права файл

`chmod` [`-fR`] символьное_изменение_прав файл.

Ключ `-f` говорит о том, что команда не будет сообщать о невозможности изменения прав доступа. `R` говорит о том, что права будут изменяться рекурсивно, то есть на все вложенные каталоги и файлы. Абсолютные права задаются восьмеричным числом и приведены в табл. 9.17.

***Таблица 9.17.*** *Коды прав доступа*

| Режим доступа | Двоичное число | Восьмеричное число |
|---|---|---|
| `- - -` | 000 | 0 |
| `- - x` | 001 | 1 |
| `- w -` | 010 | 2 |
| `- w x` | 011 | 3 |
| `r - -` | 100 | 4 |
| `r - x` | 101 | 5 |
| `r w -` | 110 | 6 |
| `r w x` | 111 | 7 |

Это число указывает право для каждой из трех категорий. Например:

```
chmod 711 file_1
```

назначает для file_1 следующие права: владелец — чтение, запись, выполнение; группа и остальные — только исполнение.

Для символьного изменения существуют определенные правила синтаксиса (табл. 9.18).

***Таблица 9.18.*** *Правила синтаксиса для chmod*

| Опция | Значение |
|---|---|
| `a` | Изменения касаются всех |
| `u` | Изменения касаются владельца |
| `g` | Изменения касаются группы |

***Таблица 9.18*** *(окончание)*

| Опция | Значение |
|---|---|
| o | Изменения касаются остальных пользователей |
| + | Назначение права |
| - | Снятие права |
| = | Установка |
| r | Назначить право чтения |
| w | Назначить право записи |
| x | Назначить право выполнения |

Соответственно, примером выполнения может служить

```
chmod u+rwx,go=x.
```

Эта команда выполняет те же действия, что и команда в предыдущем примере.

## Изменение владельца и группы

Владелец файла, а также root, помимо изменения прав доступа к файлу, могут менять также владельца и группу файла. При этом первоначальный владелец теряет права на файл. Теперь они переходят к новому владельцу.

Для изменения владельца и группы используется команда `chwon`:

`chown` [`-h`] [`-R`] владелец [:группа] файл.

Опция `-h` требует изменения владельца, на которого указывает символическая связь. Опция `-R` требует рекурсивного изменения.

Пример выполнения:

```
chown user_01:group_01 file_1
```

При этом файлу file_1 задается владелец user_01 и группа group_01.

Если требуется изменить только группу, не изменяя владельца, то можно воспользоваться командой `chgrp`, которая сменяет группу файла.

## Установка режима создания файла

Новый файл создается с правами, определяемыми пользовательской маской режима создания файла. Встроенная команда `umask` позволяет менять режим создания файла по умолчанию. Права доступа в команде `umask` задаются в виде трехзначного восьмеричного числа для каждой ка-

тегории (владелец, группа, все). Это число соответствует аннулируемым правам (табл. 9.19).

***Таблица 9.19.*** *Коды для umask*

| Режим доступа | Двоичное число | Восьмеричное число |
|---|---|---|
| r w x | 000 | 0 |
| r w - | 001 | 1 |
| r - x | 010 | 2 |
| r - - | 011 | 3 |
| - w x | 100 | 4 |
| - w - | 101 | 5 |
| - - x | 110 | 6 |
| - - - | 111 | 7 |

Например,

```
umask 067
```

разрешает все владельцу, группе разрешен только запуск, всем остальным ничего нельзя.

# 9.5. Текстовый редактор vi

Текстовый редактор vi существует уже очень давно (рис. 9.15). Он отличается от привычных графических редакторов, так как не обладает графическим интерфейсом, а общается с пользователем через систему команд. Его вызов содержится в большинстве общесистемных настроек, поэтому важно владеть основами работы с ним.

Вызов текстового редактора осуществляется следующим образом:

`vi` имя файла

Редактор находится в режиме просмотра. Вход в режим редактирования осуществляется при выборе одной из команд. Возвращение в командный режим по клавише <Esc>.

Для работы с редактором используется система команд. Основные команды приведены в табл. 9.20.

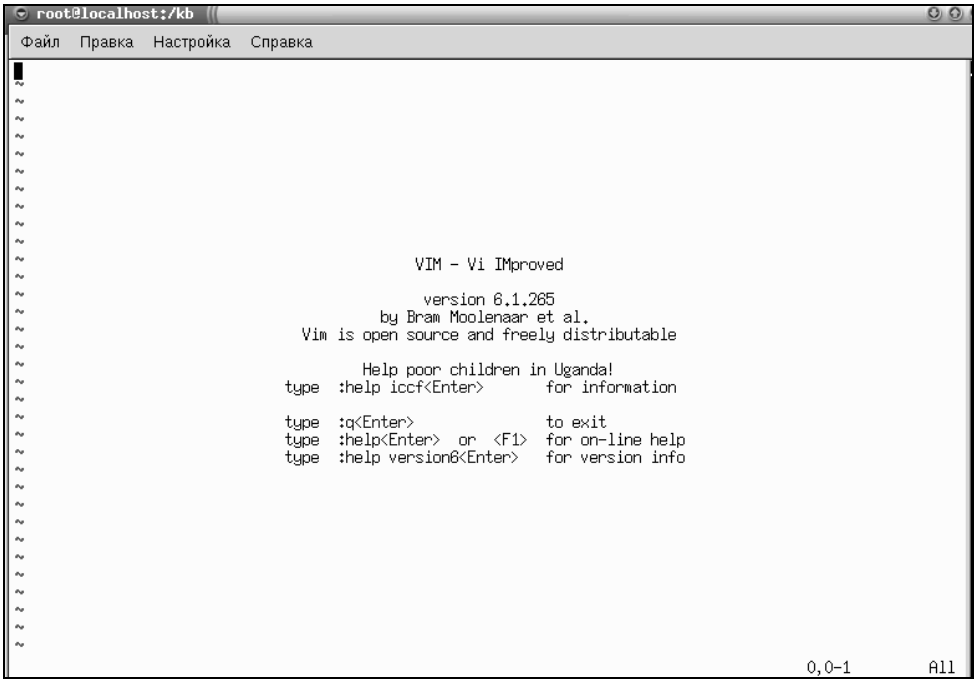



**Рис. 9.15.** Окно текстового редактора vi

***Таблица 9.20.*** *Команды редактора vi*

| Действия | Назначение |
|---|---|
| | **Перемещение** |
| <Ctrl>+<h> | Курсор влево |
| <Ctrl>+<n> | Курсор вниз |
| <Ctrl>+<p> | Курсор вверх |
| Пробел | Курсор вправо |
| | **Добавление текста** |
| a | Вставить текст после курсора |
| i | Вставить текст перед курсором |
| o | Создать новую строку ниже текущей строки |
| O | Создать новую строку выше текущей |

***Таблица 9.20*** *(окончание)*

| Действия | Назначение |
|---|---|
| | **Редактирование текста** |
| r | Замена одного символа |
| R | Замена нескольких символов |
| | **Удаление текста** |
| x | Удаление символа |
| dd | Удаление строки |
| | **Поиск и замена** |
| /строка | Поиск строки вперед |
| ?/строка | Поиск строки назад |
| n | Продолжить далее |
| N | Продолжить назад |
| :[range]s/old/new/[g] | Заменить old на new в указанном диапазоне строк range. Суффикс g означает замену во всем файле |
| | **Копирование текста** |
| yy | Копирование строки в целом |
| p | Вставить из буфера после (курсора, текущей строки) |
| P | Вставить из буфера перед (курсором, текущей строкой) |
| | **Выход из редактора** |
| :wq <Enter> | Запись и выход. Записать текст из буфера в файл и выйти из редактора |
| :q! <Enter> | Закончить редактирование без записи изменений |

**Замечание**

Более предпочтительным для работы является текстовый редактор, встроенный в файловый менеджер mc (Midnight Commander).

# 9.6. Файловый менеджер Midnight Commander

Midnight Commander во многом похож на Norton Commander, FAR и Total Commander. Как и большинство файловых менеджеров, он имеет классиче-

ский двухоконный интерфейс. Одна из панелей активная. Переход между панелями осуществляется при помощи клавиши <Tab>.

В верхней части экрана расположено меню программы. Вход в меню осуществляется по нажатию клавиши <F9>.

В нижней части расположена командная строка. Она служит для ввода команд.

Еще ниже командной строки расположен список функциональных клавиш <F1>–<F10>. Например, нажатие <F1> вызовет появление справки (рис. 9.16).

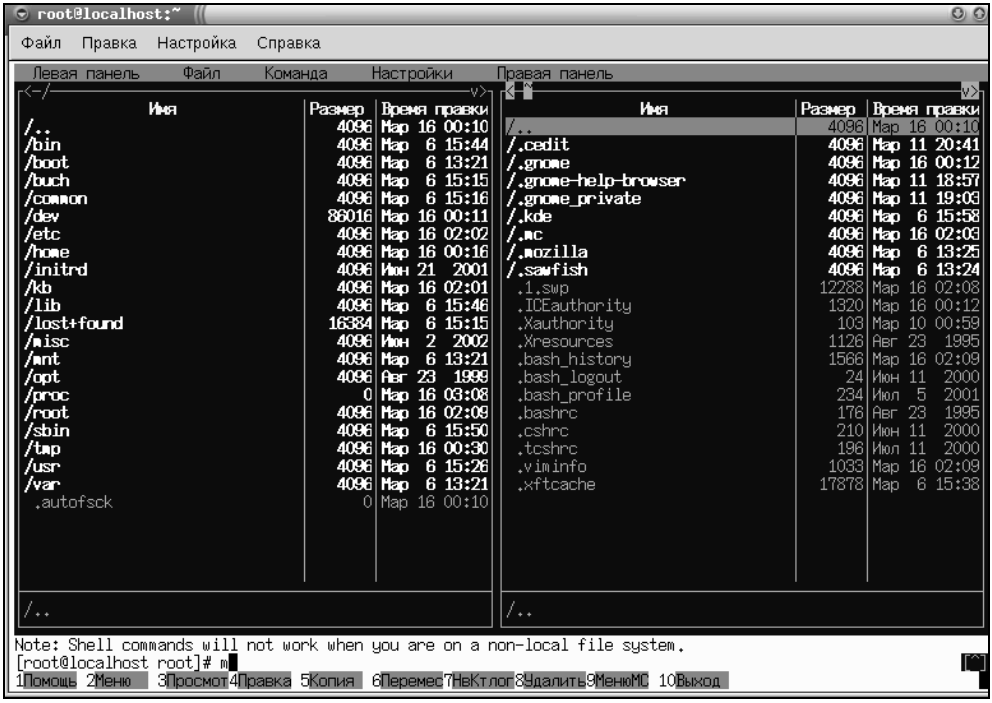



**Рис. 9.16.** Окно Midnight Commander

Поскольку работа с Midnight Commander аналогична работе с большинством файловых менеджеров, остановимся только на некоторых специфических особенностях использования.

## Настройка прав доступа

Для изменения прав доступа к файлу необходимо позиционировать курсор, нажать <F9>, затем войти в меню **Файл | Права доступа** (рис. 9.17).

После выбора откроется окно, как на рис. 9.18.

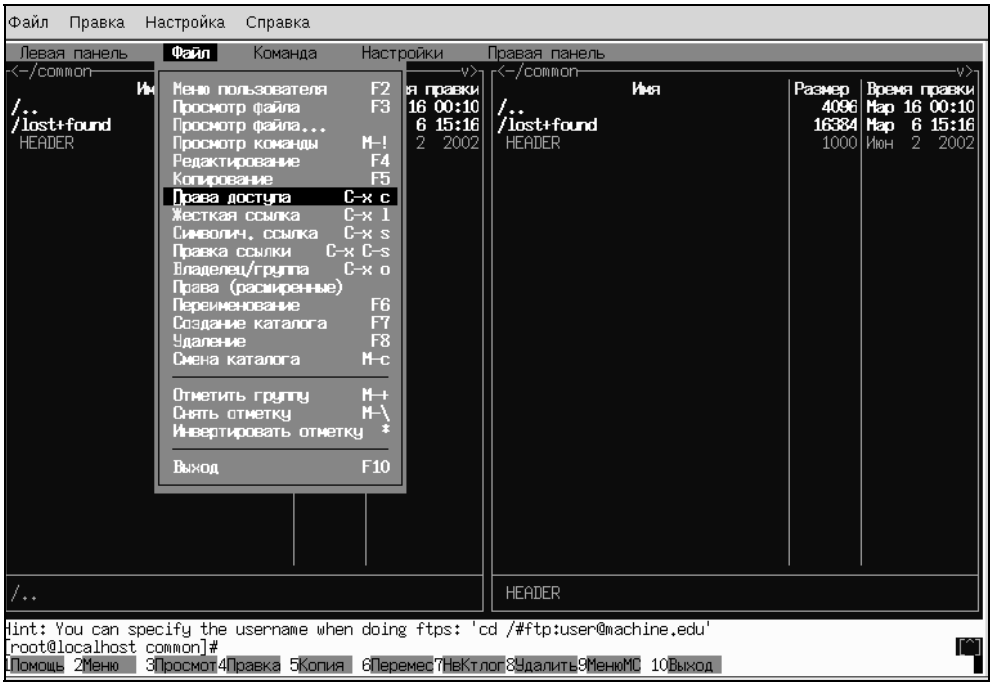



**Рис. 9.17.** Пункт меню файл

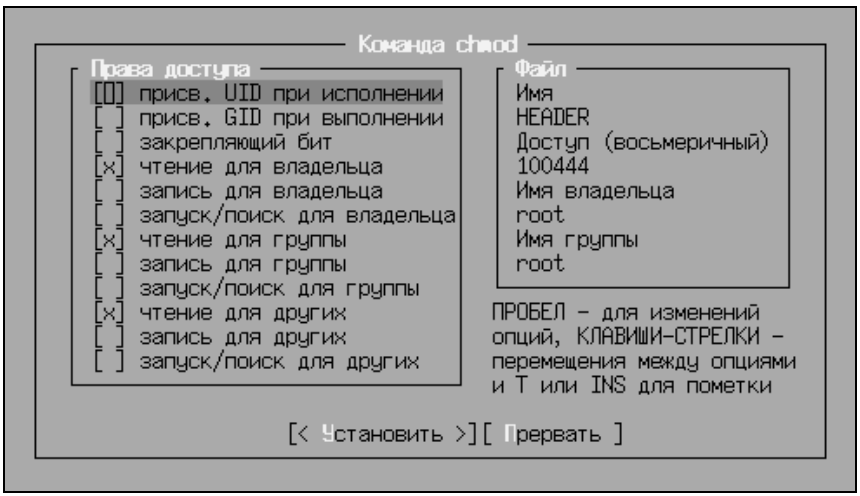



**Рис. 9.18.** Окно изменения прав доступа к файлу

Устанавливаем необходимые права доступа. Для этого устанавливаем соответствующий признак. В нашем случае всем разрешено только чтение.

## Настройка владельца и группы

Для настройки владельца и группы по аналогии с предыдущим выбираем **Файл | Владелец / Группа**. Откроется окно изменения владельца, в котором можно сделать необходимые изменения (рис. 9.19).

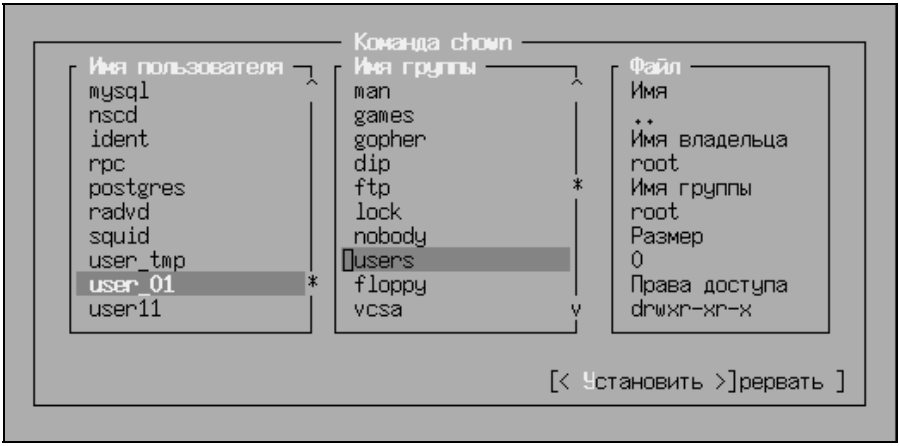



**Рис. 9.19.** Смена владельца / группы

# 9.7. Средства аудита

Система регистрации событий в Linux построена на демоне syslog. Он принимает от программ сообщения и в зависимости от выбранной политики аудита регистрирует его в журнале или отбрасывает. Прелесть демона syslog в том, что он собирает всю информацию воедино (до syslog каждая программа вела свой аудит) и обладает большой гибкостью настройки.

Демон syslog запускается одним из первых во время инициализации системы. После запуска начинает принимать от программ сообщения. То, как себя будет вести syslog, определяет его файл конфигурации /etc/syslog.conf (листинг 9.7).

**Листинг 9.7. Фрагмент файла /etc/syslog.conf**

```
# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.*                                                 /dev/console


# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
```

```
*.info;mail.none;authpriv.none;cron.none          /var/log/messages

# The authpriv file has restricted access.
authpriv.*                                        /var/log/secure

# Log all the mail messages in one place.
mail.*                                            /var/log/maillog



# Log cron stuff
cron.*                                            /var/log/cron

# Everybody gets emergency messages
*.emerg                                           *

# Save news errors of level crit and higher in a special file.
uucp,news.crit                                    /var/log/spooler

# Save boot messages also to boot.log
local7.*                                          /var/log/boo
```

Формат файла состоит из двух столбцов. Левый столбец определяет параметры регистрируемого события, правый — местоположение журнала, в который это событие будет записано. Строки, начинающиеся с символа #, являются комментарием и игнорируются. Запись, регистрирующая в файлах то или иное событие, состоит из двух частей, разделенных точкой. Левая часть представляет собой средство (табл. 9.21).

***Таблица 9.21.*** *Средства syslog*

| Средство | Комментарий |
|---|---|
| * | Все средства |
| auth, authpriv | Команды, связанные с авторизацией |
| cron | Демон cron |
| daemon | Все демоны |
| kern | Ядро |

***Таблица 9.21*** *(окончание)*

| Средство | Комментарий |
|---|---|
| lpr | Система спулинга печати |
| mail | Система sendmail |
| mark | Метки времени |
| news | Новости |
| syslog | Сообщения демона syslog |
| user | Пользовательские процессы |
| uucp | Система uucp |
| local0-local7 | Локальные сообщения |

Значение, расположенное справа от точки, представляет собой параметр уровня. Если не указано противное, то будут регистрироваться все сообщения с указанным уровнем, а также сообщения с более высоким уровнем. В табл. 9.22 приведены основные уровни сообщений в порядке возрастания их важности: в первой строке наименее важные сообщения, в последней — экстренные сообщения.

***Таблица 9.22.*** *Уровни важности сообщений*

| Уровень сообщения | Комментарий |
|---|---|
| debug | Сообщения, относящиеся к отладке |
| info | Информационные сообщения |
| notice | Замечания, заслуживающие внимания |
| warning (warn) | Предупреждения |
| err (error) | Ошибки |
| crit | Критическое состояние |
| alert | Срочные сообщения |
| emerg | Экстренные сообщения |

При описании уровней возможно применение спецсимволов: знак равенства означает, что правило справедливо только для этого уровня. Восклицатель-

ный знак (инверсия) означает, что события будут регистрироваться для всех уровней ниже текущего.

Параметр none используется для предотвращения регистрации. В правой колонке — имя файла журнала. Помимо того, что сообщения записываются в файл, они могут еще отправляться на компьютеры с указанными IP-адресами и выводиться на экран зарегистрированных пользователей (табл. 9.23).

***Таблица 9.23.*** *Некоторые дополнительные возможности syslog*

| Действие | Комментарий |
|---|---|
| @@IP_adr | Переслать сообщение на IP-адрес |
| * | Вывести сообщение на экраны всех зарегистрированных пользователей |
| user_01, user_02 | Вывести сообщения на экраны перечисленных пользователей |

Изменять файл конфигурации syslog можно любым текстовым редактором. Однако после сделанных изменений систему целесообразно проверить. Для проверки изменений в конфигурационном файле /etc/syslog.conf можно использовать команду `logger`.

```
logger -p kern.emerg "Выпей чашечку кофе"
```

Здесь `kern.emerg` — это средство и параметр уровня, для которого проводится проверка. Сообщение будет занесено в журнал регистрации.

Просмотр журнала регистрации может осуществляться в любом редакторе.

Возможен просмотр и при помощи утилиты Webmin. Для этого запускаем Webmin уже знакомым нам способом: в окне **System Logs** вкладки **System** можно просмотреть существующие log, добавить новые, отредактировать или удалить старые (рис. 9.20):

- Для того чтобы просмотреть журнал, достаточно нажать ссылку **View**, в открывшемся окне можно просмотреть все записи текущего лога.
- Для того чтобы создать новый лог, щелкнем по ссылке **Add New Sysytem log**. Заполняем форму создания нового лога и нажимаем кнопку **Save**. Создан новый лог.
- Для редактирования необходимо щелкнуть по ссылке, указывающей расположение лога, — **Log destination**. При этом появится окно, где будут уже внесены некоторые данные (рис. 9.21). Необходимо их отредактировать и сохранить изменения.
- Если в окне на рис. 9.21 нажать кнопку **Delete**, то лог благополучно удалится.

System Logs - Mozilla {Build ID: 2002061004}

File Edit View Go Bookmarks Tools Window Help

Back Forward Reload Stop https://127.0.0.1:10000/syslog/ Search Print

Webmin Index
Module Config

# System Logs

Search Docs..

Add a new system log

| Log destination | Active? | Messages selected | |
|---|---|---|---|
| File /dev/console | No | kern.* | |
| File /var/log/messages | Yes | *.info ; mail.none ; authpriv.none ; cron.none | View.. |
| File /var/log/secure | Yes | authpriv.* | View.. |
| File /var/log/maillog | Yes | mail.* | View.. |
| File /var/log/cron | Yes | cron.* | View.. |
| All users | Yes | *.emerg | |
| File /var/log/spooler | Yes | uucp,news.crit | View.. |
| File /var/log/boot.log | Yes | local7.* | View.. |

Add a new system log

Apply Changes Click this button to make the current configuration active by killing the running syslog process and restarting it.

Return to index

**Рис. 9.20.** Управление log-файлами в Webmin

Edit System Log - Mozilla {Build ID: 2002061004}

File Edit View Go Bookmarks Tools Window Help

Back Forward Reload Stop https://127.0.0.1:10000/syslog/edit_log.cgi?idx=2 Search Print

Webmin Index
Module Index

# Edit System Log

**Log destination**

Log to: File /var/log/messages ☑ Sync after each message?
Named pipe
Syslog server on
Local users
All logged-in users

Logging active? Yes No

**Message types to log**

| Facilities | | | Priorities | | | | |
|---|---|---|---|---|---|---|---|
| All | Many | | None | All | At or above.. | info |
| mail | Many | | None | All | At or above.. | |
| authpriv | Many | | None | All | At or above.. | |
| cron | Many | | None | All | At or above.. | |
| | Many | | None | All | At or above.. | |

Save View logfile Delete

**Рис. 9.21.** Редактирование log

# 9.8. Действия в случае аварии

Никто не застрахован от возникновения аварийных ситуаций. Оборудование имеет свойство иногда выходить из строя, причем случается это в самый неподходящий момент. Конечно, если оборудование на гарантии, его отремонтируют, но вот сохранность данных — это ваша забота.

Для того чтобы обеспечить сохранность данных и оперативное восстановление работоспособности оборудования, у вас есть все полномочия и возможности. Первым делом, необходимо составить подробный план действий на случай аварии. В него вы должны включить необходимые действия и процедуры в том порядке, в котором их необходимо выполнить. В список можно внести телефоны и координаты тех лиц, с которыми необходимо связаться для получения помощи при необходимости.

**Внимание!**

*Залогом быстрого восстановления работоспособности сети и данных всегда было и остается регулярное резервное копирование данных и подробное документирование состояния сети.*

Излишне будет рассказывать о том, как в результате выхода из строя жесткого диска терялись данные за весь прошедший период, причем в канун сдачи отчетов в фискальные органы. А между тем подобное происходит достаточно часто.

В следующем разделе описаны рекомендации, следуя которым, вы во многом облегчите себе жизнь при возникновении нештатных ситуаций.

## Составление плана действий на случай аварии

Содержание плана вы должны определить для себя сами, исходя из уровня вашей подготовки и особенностей построения сети. Однако общие рекомендации все же можно дать.

1. Составьте список тех людей, с которыми необходимо связаться в случае возникновения аварийной ситуации. Сюда включается весь персонал, который имеет отношение к обслуживанию сети (если вы отвечаете за сеть не один), представители службы охраны, более опытные товарищи (которые могут оказать вам помощь советом или делом). В список вносите подробные координаты: рабочие и домашние телефоны, пейджеры, сотовые телефоны.
2. Определите порядок восстановления системы: подразделения, которые должны получить доступ к сети и серверу в первую очередь. В торговой организации, скорее всего, это будут точки продаж и склад.
3. В план необходимо включить документы о структуре сети и структуре сервера: схему сети с расположением розеток, компьютеров, сервера,

карту патч-панелей и коммутаторов. Включите в план место хранения резервных копий, это может быть как шкаф с копиями на CD-носителях, так и расположение компьютеров, на которых хранятся резервные копии.

4. Составьте список информации, подлежащей резервному копированию, и график и строго придерживайтесь этого графика. Если резервное копирование производится автоматически, то проверяйте наличие резервных копий и проводите их тестовое восстановление. Бывает очень обидно, когда из-за проблем с правами доступа вместо копии необходимых данных вы получаете пустые файлы.
5. Составьте график анализа log-файлов и список лиц, которым можно поручить данное дело. Постоянно выполнять анализ log-файлов нет необходимости. Кроме того, монотонная работа притупляет внимание, и вы можете не увидеть важного. Привлечение других лиц к анализу log позволит подготовить других специалистов, которые при необходимости могут вас подменить на время отпуска.

## Создание аварийной загрузочной дискеты

Иногда бывает достаточно загрузки с дискеты и небольшой правки системы для восстановления работоспособности сервера. Поэтому немаловажным этапом является создание загрузочной дискеты.

Как правило, при установке Linux система предлагает создать загрузочную дискету. Если вы в свое время от этого отказались, то создайте дискету сейчас. Воспользуйтесь для этого командой `mkbootdisk`.

При создании дискеты, необходимо определить ядро, которое вы используете, и место его расположения. Для этого необходимо просмотреть файл конфигурации загрузчика: по умолчанию это ASP Loader, но может быть и Lilo или другой загрузчик (листинг 9.8).

**Листинг 9.8. Файл конфигурации загрузчика /etc/aspldr.conf**

```
[asplinux1@ASPLinux(2.4.18-19.7se)]
icon linux
kernel /boot/vmlinuz-2.4.18-19.7se root=/dev/hda5 ro

[SEPARATOR]

[floppy@Boot from floppy]
icon floppy
sysboot a:
```

```
 [BOOTMGR]
video graphics
default asplinux1
timeout 15
clock 24

[ACTIVATOR]
writembr on
writeboot off
biosnum 1
mbrdev /dev/hda
language en
```

Как видно из файла, образ ядра находится в каталоге /boot/vmlinuz-2.4.9-13smp. Теперь можно приступить к созданию загрузочной дискеты:

1. Входим в систему под именем root.
2. Вставляем в дисковод дискету.
3. В командной строке вводим:

```
mkbootdisk --device /dev/fd0 2.4.18-19.7se
```

Появится надпись:

```
Insert a disk in /dev/fd0. Any information on the disk will be lost.
Press <Enter> to continue or ^C to abort:
```

1. Подтверждаем создание загрузочной дискеты, нажав <Enter>.
2. Для проверки загрузитесь с только что созданной дискеты. Для этого в загрузчике необходимо выбрать загрузку с диска a:.
3. Если все прошло успешно, то вы только что стали счастливым обладателем загрузочной дискеты. Будем надеяться, что вам не придется ею воспользоваться.

## Резервное копирование и восстановление в Linux

Несмотря ни на что, резервное копирование остается единственной гарантией целостности ваших данных и быстроты восстановления системы. Резервное копирование может осуществляться на любом независимом от сервера носителе. Выбор носителя определяется в основном объемом подлежащей резервированию информации. Если это небольшой объем, то это может быть CD-ROM, для совсем больших объемов можно выделить дисковое пространство на других компьютерах в сети, при этом не забывайте

об обеспечении конфиденциальности информации (если в этом есть необходимость). Методов, которыми можно осуществить резервное копирование, много.

- Если речь идет о клиентах, то можно сделать его через планировщика заданий в Windows, либо приобретя специализированное программное обеспечение, например программу "Хранитель" фирмы "Гендалф". Это достаточно удобная программа с гибкой системой настроек.
- При резервировании на стороне сервера наиболее типичным является организация резервного копирования при помощи программы tar. Существуют также и другие пакеты, в том числе и коммерческие.

Ранее мы уже говорили о том, что нужно составить список резервного копирования. Сюда должны входить в основном те данные, которые не могут быть восстановлены другими способами. Например, нет никакого смысла в резервном копировании программ, поскольку их легко восстановить с исходных носителей. В то же время малейшее упущение при резервировании данных может стать причиной их повторного ввода вручную, а значит затрат времени, денег и нервов. Наиболее типичной является схема копирования с недельным циклом (например в пятницу в конце рабочего дня копируется весь объем данных, до наступления следующей пятницы копируются только те файлы, которые изменялись). Об изменении файла будет говорить его дата.

Ротация добавочных копий осуществляется в недельном цикле. Ротация полных копий осуществляется каждые 4 недели (должны быть диски полных копий Пятница_1, Пятница_2, Пятница_3, Пятница_4). Таким образом, понадобится 8 носителей резервной информации (компакт-дисков).

Синтаксис команды `tar`:

`tar` [- ключи] архив  место. Ключи описаны в табл. 9.24.

***Таблица 9.24.*** *Некоторые опции tar*

| Опция | Комментарий |
|---|---|
| c | Говорит о том, что мы создаем архив |
| x | Извлечение файлов из архива |
| p | Будут сохраняться права доступа |
| N | Сохраняет файлы с учетом даты |
| f | Указывает на то, что поле ключей закончено. Дальше будет имя устройства или файла |

## Создание резервной копии

Особенностью `tar` является то, что, когда он создает архивные файлы, он удаляет начальный символ / из пути к файлу. Чтобы избежать нарушения пути к файлу при его восстановлении, перейдите в корневой каталог.

Полные архивы по пятницам создаются следующей командой

```
# tar cpf /dev/rw1/a.tar --label=" Полный архив создан `date '+%d-%B-
%Y'`." --directory /buh
```

В остальные дни недели команда следующая:

```
# tar cpNf /dev/rw1/a.tar --label=" Дополнительный архив создан `date
'+%d-%B-%Y'`." --directory /buh
```

Опция `label` позволяет записать в архивный файл служебную информацию.

## Восстановление данных из копии

Для восстановления полной копии каталога /buh используйте команду `# tar xpf /dev/rw1/a.tar`

Возможно извлечение не всей заархивированной информации, а ее части:

```
# tar xpf /dev/rw1/a.tar \ buh/other/client.doc
```

В этом случае вы извлечете только файл /buh/other/client.doc.

# Глава 10

# Настройка Samba

Материал этой главы достаточно сложный. Причина в том, что разделы переплетаются и связаны друг с другом. Поэтому если вам что-то непонятно, сначала прочитайте главу целиком, а затем вернитесь к интересующей вас теме.

## 10.1. Общие сведения о Samba

На большинстве офисных компьютеров установлена операционная система Windows. Для объединения серверов с Linux и клиентов под Windows используется пакет программ Samba. Автором Samba является Andrew Tridgell (Эндрю Триджелл), первая версия вышла в 1992 г.

Пакет Samba основан на инженерном анализе протокола Lan Manager компании Microsoft. Основу этого протокола составляют Server Message Block (SMB) — блоки сообщений сервера, что и дало название программному продукту. Поскольку на анализ требуется время, то реализация некоторых специфичных функций в Samba отстает от Windows.

Несмотря на сложность анализа, пакет Samba постоянно обновляется, обеспечивая надежный механизм совместного использования ресурсов в сети. По сути, пакет Samba представляет Linux-машинам доступ в сеть, используя протоколы Microsoft. Прелесть такого решения состоит в том, что программное обеспечение устанавливается только на сервере. На рабочих местах с Windows ничего устанавливать не надо. В ASPLinux 7.3 Server Edition используется версия Samba 2.27 как наиболее стабильная и отлаженная.

Сервер Samba состоит из нескольких программ. Состав программ и выполняемые ими роли приведены в табл. 10.1.

Конфигурирование Samba может осуществляться вручную либо при помощи графических утилит администрирования. В любом случае все сводится к изменению конфигурационных файлов. У Samba их несколько (табл. 10.2).

***Таблица 10.1.*** *Состав пакета Samba*

| Пакет | Описание |
|---|---|
| smbd | Демон Samba, обеспечивающий сервис SMB |
| nmbd | Демон, обеспечивающий работу сервиса имен NetBIOS. Это позволяет узлам в сети получать репликацию сетевых имен на IP-адреса |
| smbclient | Утилита, позволяющая узлам на Linux получить доступ в сеть Microsoft |
| testparam | Само название этой утилиты говорит за себя. Она используется для проверки правильности написания файла конфигурации smb.conf |
| Swat | Средство администрирования при помощи web-интерфейса. Утилита включена в пакет Webmin |

***Таблица 10.2.*** *Конфигурационные файлы Samba*

| Файл | Описание |
|---|---|
| /etc/samba/smb.conf | Основной конфигурационный файл. Большинство настроек касается общих файловых ресурсов, общих принтеров, прав пользователей |
| /etc/samba/smbpasswd | Здесь содержится информация для идентификации пользователей |
| /etc/samba/lmhosts | Здесь содержится информация для обеспечения отражения имен на IP-адреса |
| /etc/init.d/smb | Файл, определяющий порядок запуска, остановки и перезапуска сервера |
| /etc/sysconfig/samba | Системный конфигурационный файл |
| /etc/pam.d/samba | Обеспечивает идентификацию пользователей при помощи PAM |

Предпочтительнее на первых парах использовать Swat с последующим анализом файла /etc/samba/smb.conf.

Необходимо помнить, что демон smbd работает с полномочиями root. Так что при неправильной настройке конфигурационного файла пользователи могут получить не предназначавшиеся им права суперпользователя. Даже если это было без злого умысла, система может пострадать.

Всю самую свежую информацию о проекте Samba можно получить на домашней странице проекта **http://www.samba.org**. Здесь же можно скачать готовые к установке новые версии пакета.

# 10.2. Одноранговая сеть. Основной файл конфигурации Samba /etc/smb.conf

Как уже упоминалось, основным файлом конфигурации является файл /etc/samba/smb.conf. В нем хранится описание основных сетевых ресурсов и правил доступа к ним. Большинство комментариев будем приводить прямо по тексту файла (листинг 10.1).

**Листинг 10.1. Файл /etc/samba/smb.conf**

```
[global]
     ; Эту строку раскомментируйте, если хотите дать доступ гостю
     ;guest account = nobody
     ; Строка привязки к определенной сетевой
     ; карте, если их больше одной
     ;     ;interfaces=192.168.0.5/24
     ;Задаем расположение log
     logfile=/var/log/sambalo-log.%m
     lock directory = /var/lock/samba
     share modes = yes
     ; Строки приказывают Linux использовать шифрованные пароли,
     ; по умолчанию это не делается
     encrypt passwords = Yes
     smb passwd file = /etc/samba/smbpasswd
     ; Строка синхронизирует Linux и Samba-пароли
     unix password sync = Yes
    ; последующие две строки избавят вас от проблем с русскими
    ; именами в файловой системе
     client code page = 866
    character set = KOI8-R

[homes]
    comment = Home Directoryes
    broweable = no
    read only = no
    creat mode = 0750

[printers]
```

```
comment = Printers
patch = /var/spool/samba
browseable = no
printable = yes
public = yes
writable = no
create mode = 0700

; пример публичного каталога
[public]
    comment = Public Stuff
    path = /home/public
    public = yes
    printtable = no
    ;write list = @stuff - на запись имеют право только
    ;пользователи группы stuff, остальным чтение

[buch]
    ; подготавливаем ресурс для бухгалтерии
    comment = Buchgalter files
    path = /buch
    public = yes
    printtable = no
    write list = @stuff
    ;на запись имеют право только
    ;пользователи группы stuff, остальным чтение

[kb]
    ; подготавливаем ресурс для конструкторского бюро
    comment = konstuktor
    path = /kb
    public = yes
    printtable = no
    write list = @stuff
    ; на запись имеют право только
    ; пользователи группы stuff, остальным чтение

;По такой же схеме создается и ресурс для временных файлов
[tmp]
```

```
comment =  Temporary file space
path = /tmp
read only = no
public = yes
```

Файл состоит из разделов (табл. 10.3). Каждый раздел начинается заголовком в квадратных скобках.

***Таблица 10.3.*** *Разделы файла smb.conf*

| Раздел | Описание |
|---|---|
| [global] | Здесь содержатся директивы, применяемые ко всем ресурсам |
| [homes] | Директивы, определяющие доступ к начальным каталогам пользователей на Linux-машине |
| [printers] | Здесь содержатся директивы, применяемые к общим принтерам |

Приведенный в листинге 10.1 файл конфигурации позволит пользователям Samba иметь доступ к своим домашним директориям и писать во временную директорию.

Необходимо также немного сказать о шифровании паролей. По умолчанию Samba не шифрует пароли, а Windows шифрует. В целях безопасности лучше, чтобы пароли шифровались. Однако в некоторых случаях, если возникают проблемы на стадии отладки, шифрование паролей можно отключить. Для этого необходимо подправить системный реестр Windows следующим образом.

- При использовании Windows 98 в разделе HKEY_Local_Machine\system\CurrentControlSet\Services\VxD\VNetsup добавить
  - тип DWORD;
  - имя EnablePlainTextPassword;
  - данные 0 × 01.
- При использовании Windows NT в разделе HKEY_Local_Machine\system\CurrentControlSet\Services\ Rdr\Parametrs добавить
  - тип DWORD;
  - имя EnablePlainTextPassword;
  - данные 0 × 01.
- При использовании Windows 2000 в разделе HKEY_Local_Machine\system\CurrentControlSet\Services\ LanmanWorkStation\Parametrs добавить
  - тип DWORD;
  - имя EnablePlainTextPassword;
  - данные 0 × 01.

Теперь немного по поводу ресурсов дискового пространства, выделяемых пользователю. Если доступ к каталогу для пользователя закрыт в Linux, то даже если вы разрешите его в Samba, пользователь все равно доступ не получит. Каталог-то увидит, но зайти не сможет.

Это проблема особенно касается монтируемых к Samba дисков с Fat32. Поскольку разграничение прав не предусмотрено параметрами файловой системы, то по умолчанию диски монтируются только для чтения (это в файле fstab). Для того чтобы открыть доступ к монтируемым DOS-дискам, в файле fstab один из параметров должен иметь `umask=0` (листинг 9.6).

## 10.3. Samba в качестве PDC

Вначале немного теории. Все узлы имеют кодовое значение уровня операционной системы. Роль компьютера в сети Microsoft определяется уровнем его операционной системы (по числу, определяющему уровень, определяется ведущий узел в сети). В частности, для первичного контроллера домена это число равно 64. Этот момент необходимо учесть при настройке Samba в качестве PDC (Primari Domain Controller — первичный контроллер домена).

Естественно, превращение Samba в PDC осуществляется редактированием файла smb.conf (листинг 10.2).

**Замечание**

Основной режим безопасности — это user. Единственный режим безопасности, который не будет работать по техническим причинам — share. Два других режима (domain и server) — это просто разновидности режима user.

Если вы хотите повысить степень защиты, то имеет смысл пароли для Samba и Linux сделать разными:

```
unix password sync = No
```

**Листинг 10.2. Файл /etc/samba/smb.conf для Samba в качестве PDC**

```
 [global]
; Здесь заданы имя домена и сетевое имя сервера
    workgroup = SAMBA
    netbios name = SAMBASERVER
; Эти параметры определяют первичный контроллер домена, для вас
; они меняться не будут
    os level = 64
    preferred master = yes
    domain master = yes
```

```
    local master = yes
;Уровень доступа тоже должен быть только таким
    security = user
; Устанавливаем шифрованные пароли
    encrypt passwords = yes
; Обработка входа в домен
    domain logons = yes
; Здесь будут сохраняться профили пользователей
; Такая комбинация позволит хранить профили в домашних каталогах
     logon path = \\%N\%U\profiles
     logon home = \\%L\%U\.profiles
 ; Домашние каталоги пользователей монтируются на диск H:
    logon drive = H:
;
    logon script = logon.cmd

    character set = KOI8-R
    client code page = 866
; Информационная строка, выводимая в сетевом окружении
    server string = Windows 3000 Server
; Здесь расположены файлы log для каждой машины,  если
; что-то не получается, анализируйте эти файлы
    log file = /var/log/samba/%m.log
; Если хотите, чтобы был сервером времени, раскомментируйте
;    time server = Yes
socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
; Эту строку раскомментируйте, если ваша программа добавления
пользователя
; позволяет добавлять пользователей, содержащих символ $
;   add user script = /usr/sbin/useradd -d /dev/null -g 100 -s /bin/false
-M %u

; Этот ресурс необходим контроллеру домена
[netlogon]
    path = /usr/local/samba/lib/netlogon
    read only = yes
    write list = root
; Ресурс для профилей пользователей
   [profiles]
    path = /export/smb/ntprofile
```

```
    read only = no
    create mask = 0600
    directory mask = 0700
[homes]
    comment = Home Directories
    path = /home/%U
    ; write list = user11,user12,user14,root
    ; force user = user11,user12,user14,root
    writeable = Yes
    guest ok = Yes
    browseable = No
    available = No
[printers]
    comment = All Printers
    path = /var/spool/samba
    printable = Yes
    browseable = No
; А это уже наши ресурсы, которые  мы планировали для отделов
[Kb]
       path = /kb
       comment = Konstruktor
       write list = user11,user12,user14,root

[buch]
       path = /buch
       write list = user21,user22,user24,root
```

Сценарий входа для пользователей представлен в листинге 10.3.

**Листинг 10.3. Содержание файла logon.cmd**

```
; Установка времени
net time \\SambaServer /set /yes
;Подключение дисков
net use Y: \\ SambaServer\kb
net use Z: \\ SambaServer\buch
```

**Примечание**

Кстати, команда net use правильно работает тоже только в версии 2.2.

Если имя задать как %u.cmd?, то можно установить отдельный сценарий для каждого пользователя.

После того как вы настроили Samba, необходимо создать пользователей, для которых будут доступны ресурсы Samba. Делается это в два этапа:

1. Создаете пользователей в Linux *(см. главу 9)*.

2. Создаете пользователей в Samba. Для этого с клавиатуры вводим:

```
#smbpasswd -a Имя_пользователя Пароль_пользователя
```

Также это можно сделать в Swat или в Webmin (об этом чуть позже). Обратите внимание на то, что вначале пользователь создается в системе, а затем только в Samba. При этом имена в Samba и Linux должны совпадать. По соображениям безопасности лучше, чтобы пароли не совпадали. Если вы все-таки установили синхронизацию паролей

```
unix password sync = yes,
```

то, соответственно, пароли должны совпадать.

# 10.4. Настройка входа в домен для Windows 98

## Настройка входа

Здесь предполагается, что сетевая карта уже установлена и IP-адрес введен.

Откройте **Пуск | Панель Управления | Пароли** и выберите вкладку **Профили пользователей** (рис. 10.1). Выберите нужный уровень установок профилей (рис. 10.2).

1. После сделанных изменений потребуется перезагрузка системы. Выполните ее.

2. После перезагрузки вас попросят указать имя пользователя и пароль (рис. 10.3). Пользователя можно указать реального, а пароль оставьте пустым, поскольку эти данные попадают в pwl-файлы, которые не обладают высокой степенью защиты и позволяют легко вскрыть пароли.

3. На вопрос, создавать ли на локальном компьютере учетную запись пользователя, отвечайте **Да**.

   Теперь вы создали локальную учетную запись (рис. 10.4). У вас свой собственный вид рабочего стола и элементов меню. Однако необходимо помнить, что это локальная копия, на сервере пока ее нет. В домен Samba вы еще не вошли, хотя и создали профиль пользователя. Только что созданный вами профиль будет впоследствии перенесен на сервер.

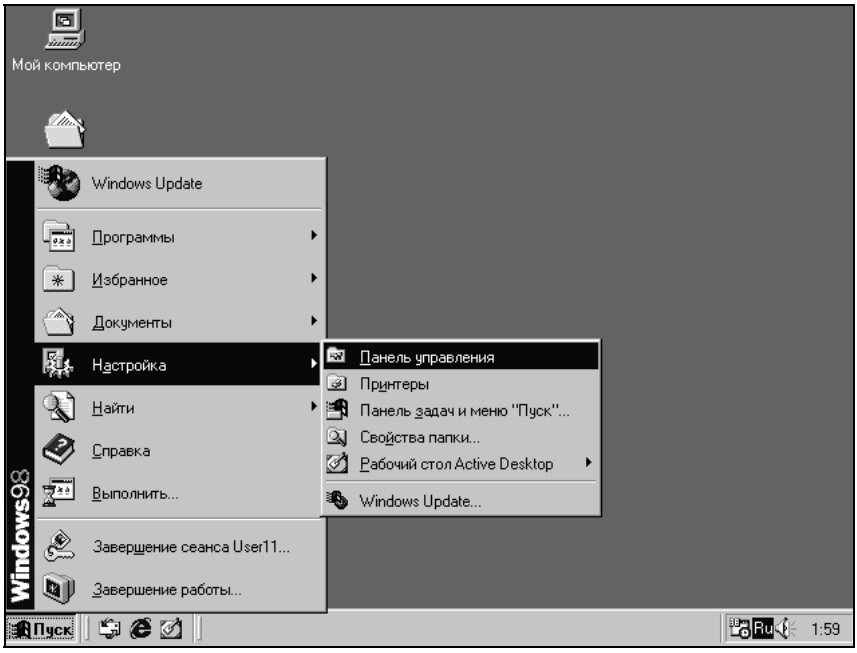



**Рис. 10.1.** Входим в панель управления

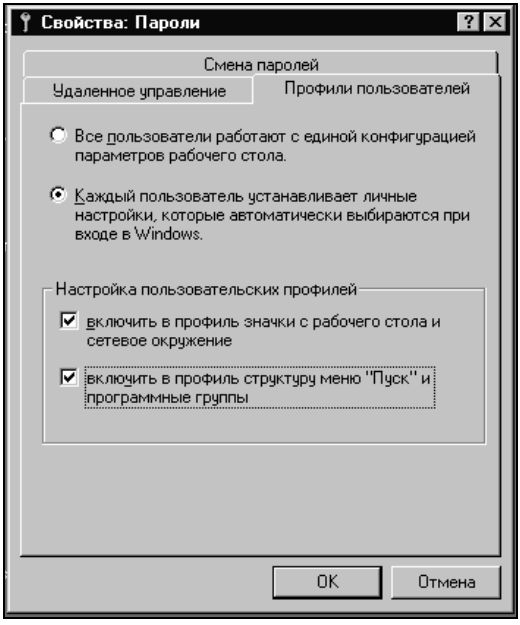



**Рис. 10.2.** Каждый пользователь со своим профилем

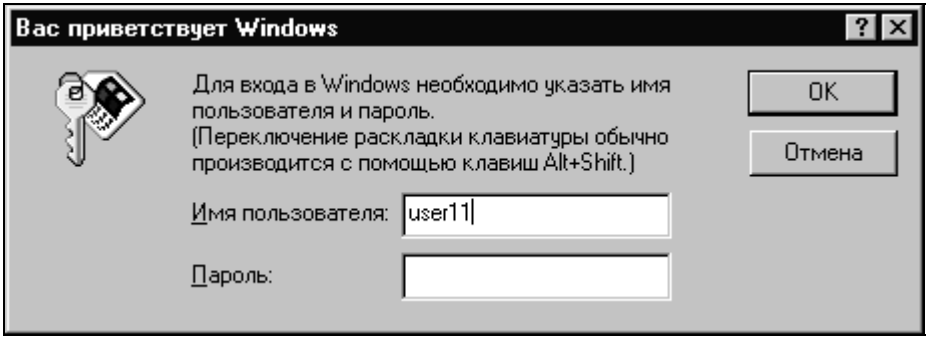



**Рис. 10.3.** Создание нового пользователя в Windows 98

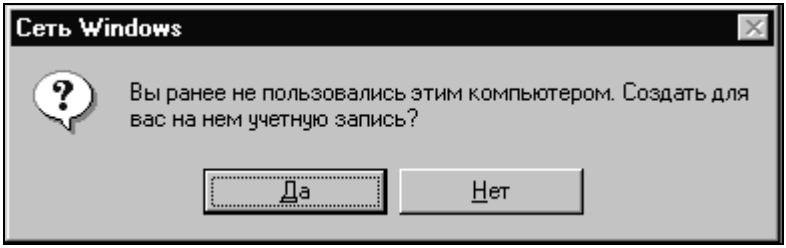



**Рис. 10.4.** Создание локальной учетной записи

4. Открываем **Пуск | Панель Управления**. Выбираем элемент **Сеть**. В открывшемся окне назначаем способ входа в сеть **Клиент для сетей Microsoft**, в противном случае компьютер не будет входить в сеть, и будут загружены локальные профили (рис. 10.5).
5. Здесь же выберите в верхнем окне **Клиент для сетей Microsoft**. Нажмите кнопку **Свойства** и выберите пункт **Входить в домен Windows NT** (рис. 10.6).
6. Сохраняем сделанные изменения, нажав кнопку **OK**.
7. Перезагрузите компьютер, появится окно входа в сеть, содержащее поля **Имя**, **Пароль**, **Домен** (рис. 10.7).
8. Введите данные и попытайтесь зарегистрироваться. Если имя и пароль не распознаны, причиной, кроме очевидных ошибок, могут быть:
   - неправильный формат файла, ссылка на который задана в файле smb.conf;
   - пользователю, под именем которого вы входите, закрыт доступ;
   - пользователь не существует;
   - неправильно выставлены IP-адрес и маска сети.

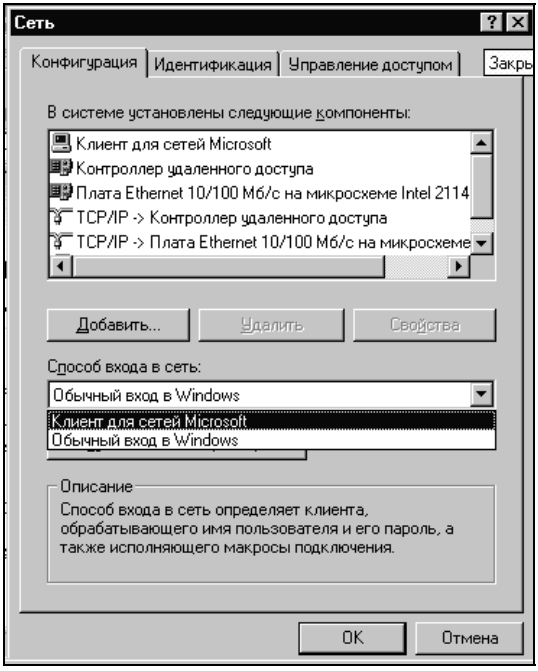



**Рис. 10.5.** Выбор способа входа в сеть

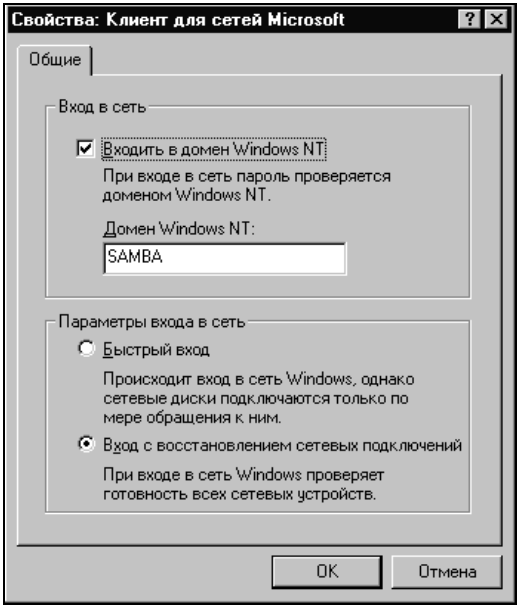



**Рис. 10.6.** Настройка имени домена

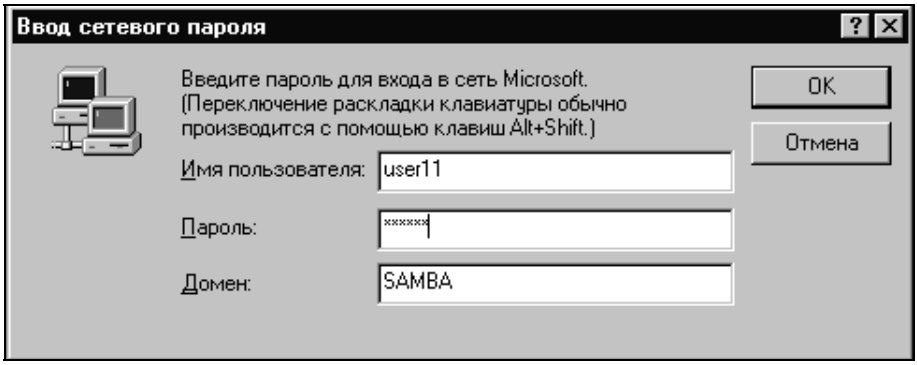



**Рис. 10.7.** Вход в домен

9. Теперь осталось настроить доступ к этому компьютеру со стороны членов сети (рис. 10.8). Открываем **Пуск | Панель Управления**. Выбираем элемент **Сеть**. В открывшемся окне назначаем **Управление доступом**. Доступ к ресурсам установите на уровне пользователей.

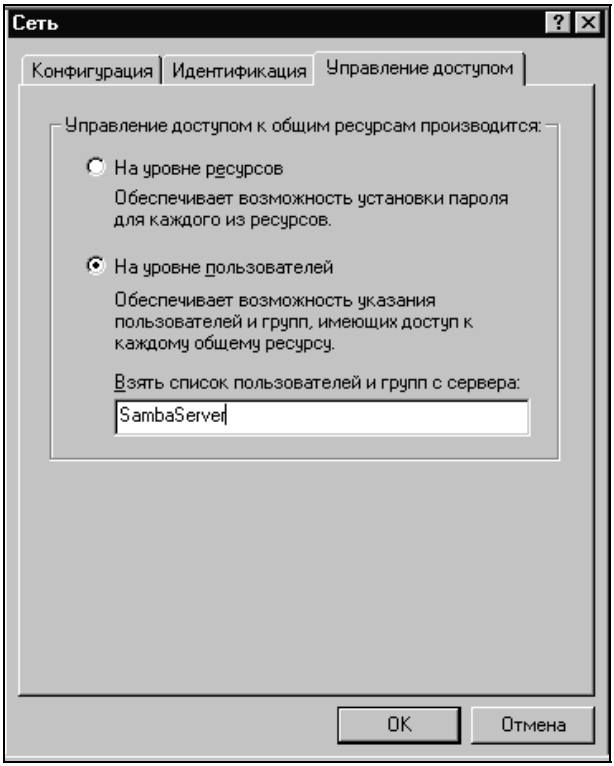



**Рис. 10.8.** Выбор разграничения ресурсов

10. Жмем **OK**. На вопрос об изменении режима доступа отвечаем **Да** (рис. 10.9).

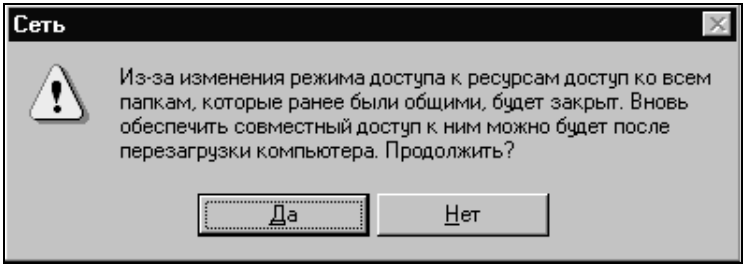



**Рис. 10.9.** Вопрос об изменении режима доступа

11. Затем перезагружаем компьютер. Все готово. В домен вы попали, можно назначать права доступа.

## Устранение проблем в Windows 98

1. Откажитесь от входа в сеть, нажав кнопку **Отмена**.
2. Запустите **Пуск | Выполнить**, наберите **regedit.exe**. В параметре HKEY_LOCAL_MACHINE\Windows\CurrentVersion\ProfileList удалите ключ ProfilePath для требуемого пользователя. В некоторых версиях положение ключа ProfilePath может отличаться. В этом случае воспользуйтесь поиском.
3. Удалите файл с расширением pwl пользователя в каталоге c:\windows (как правило, он похож по имени на пользователя, имя pwl-файла для конкретного пользователя можно уточнить в sistem.ini).
4. Завершите сеанс работы Windows.
5. Удалите файлы user.dat или user.man из Linux, они расположены в каталоге, указанном в logon_path.
6. Теперь попробуйте повторить настройку входа сначала

# 10.5. Настройка входа в домен для Windows NT/2000

Рабочие станции под управлением Windows NT/2000 используют машинные учетные записи. Пароль этой учетной записи служит ключом для защищенного соединения с контроллером домена, в целях предотвращения несанкционированного присоединения машин с именем NetBIOS, уже зарегистрированным в домене, и получения доступа к учетным записям

групп/пользователей. Рабочие станции под управлением Windows 9x/Me таких учетных записей не используют, поэтому не могут считаться полноправными членами домена.

Существуют следующие методы создания учетной записи:

1. Заранее. Для этого, обозначив machine_name имя NetBIOS, последовательно вводим команды:

```
root# /usr/sbin/useradd -g 100 -d /dev/null -c "комментарий" -s
/bin/false machine_name$
root# passwd -l machine_name$
root# smbpasswd -a -m machine_name
```

2. С помощью скриптов. В глобальную секцию добавим строку

```
add user script = /usr/bin/useraddscript %u.
```

   В каталоге /usr/bin должен быть скрипт useraddscript следующего содержания:

```
#!/bin/sh
/usr/bin/useradd -d /dev/null -g 100 -s /bin/false -M $1
```

3. Можно это сделать и одной строкой в секции [global]:

```
add user script = /usr/bin/useradd -d /dev/null -g 100 -s
/bin/false -M %u
```

К сожалению, программа useradd, используемая в ASPLinux Server 7.3, не позволяет создавать имена, содержащие спецсимволы. Это значит, что на лету создание учетной записи не получится, система не сможет самостоятельно добавить пользователя, имя которого заканчивается символом $.

Выход — в поиске другой программы, позволяющей добавлять такие имена, или в ручном создании имени без знака $ и последующем редактировании файлов /etc/passwd и /etc/shadow и добавлении в конец имени символа $.

Поскольку подключать машины к домену может только root, то в Samba необходимо добавить пользователя root. Причем пароли Linux и Samba для root должны различаться по соображениям безопасности.

Проще всего задать нового пользователя для Samba из командной строки (рис. 10.10).

```
[root@localhost root]# /usr/bin/smbpasswd -a root SambaRootPass
Added user root.
[root@localhost root]#
```

**Рис. 10.10.** Создание пользователя root для Samba

В реальности пароль, конечно, необходимо задавать более сложный, чем в приведенном примере.

## Создание учетных записей вручную

Пусть нам необходимо создать учетную запись для машины с сетевым именем kb_01. Из названия понятно, что это первая машина в конструкторском бюро. В этом случае последовательность команд будет, как на рис. 10.11.

```
[root@localhost root]# /usr/sbin/useradd -g 100 -d /dev/null -c "КБ первая машина" -s /bin/false kb01
[root@localhost root]# passwd -l kb01
Locking password for user kb01.
passwd: Success
[root@localhost root]#
```

**Рис. 10.11.** Создание новой учетной записи

Но это еще на все. Надо теперь отредактировать файлы /etc/passwd и /etc/shadow и добавить к имени kb_01 символ $. Отредактировать можно, например, воспользовавшись файловым менеджером mc (рис. 10.12).

```
user_tmp:x:501:505:Temp user:/home/user_tmp:/bin/bash
user_01:x:502:501::/home/user_01:/bin/bash
user11:x:500:510::/home/user11:/bin/sh
kb01$:x:503:100:КБ первая машина:/dev/null:/bin/false
```

**Рис. 10.12.** Последние строки файла /etc/passwd

Теперь настала пора добавить пользователя kb_01 в файл паролей Samba (рис. 10.13).

```
[root@localhost root]# smbpasswd -a -m kb01
Added user kb01$.
[root@localhost root]#
```

**Рис. 10.13.** Добавление пользователя для Samba

Подготовительный этап закончен, настало время пойти на компьютер kb_01 и подключить его к домену.

## Настройка входа

1. Входим с правами администратора на данном компьютере.
2. На значке **Мой компьютер** правой кнопкой мыши выбираем пункт **Свойства** (рис. 10.14).
3. В появившемся окне выбираем вкладку **Сетевая идентификация**. Доступны две активные кнопки: **Идентификация** и **Свойства**. При помощи любой из них возможно подключение компьютера к домену. Поскольку идентификация позволяет сразу же создать нового локального пользователя, поэтому воспользуемся ею (рис. 10.15).

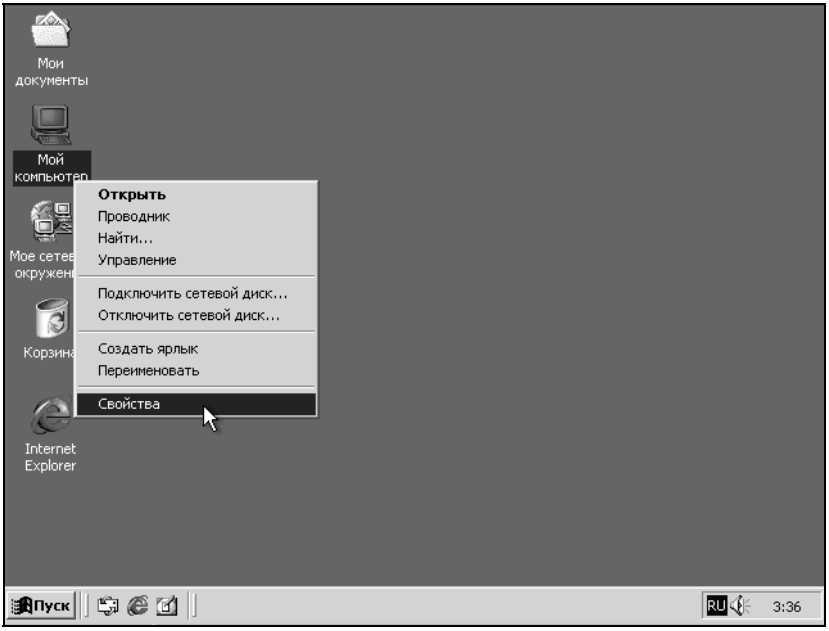



**Рис. 10.14.** Начинаем подключение компьютера к домену

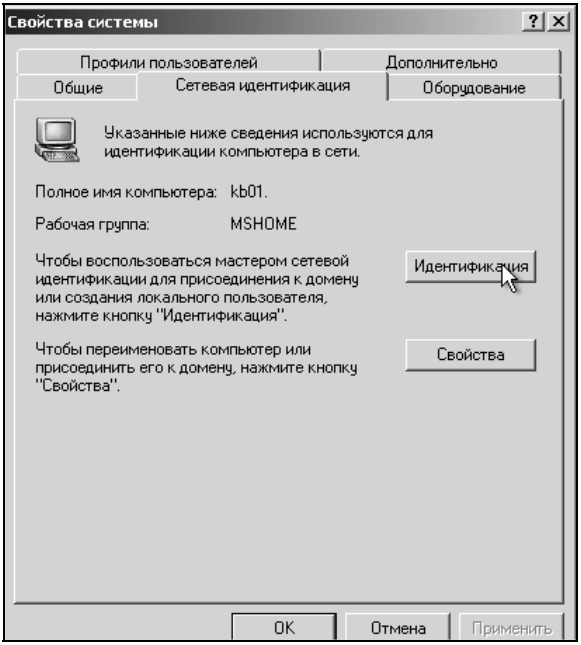



**Рис. 10.15.** Выбор способа подключения к домену при помощи мастера

4. При нажатии кнопки **Идентификация** начинает свою работу мастер подключения компьютера. Мы покажем работу мастера по шагам, сопровождая, где это необходимо, комментариями (рис. 10.16–10.20).

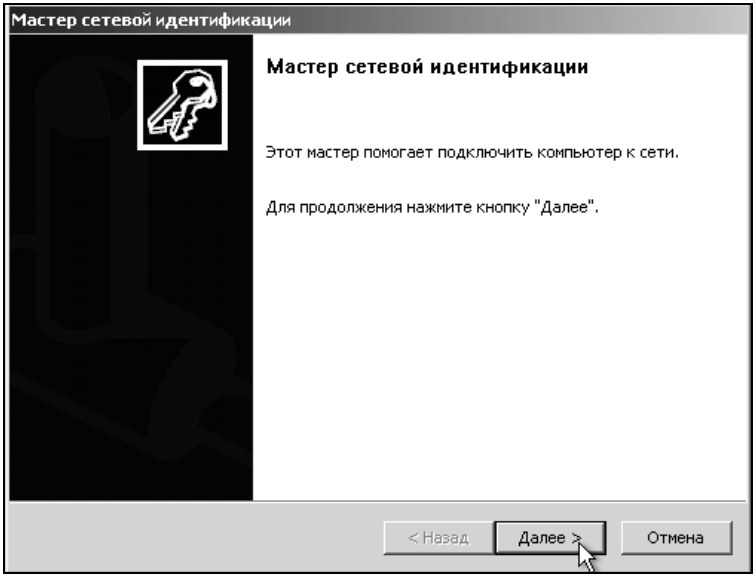



**Рис. 10.16.** Шаг 1

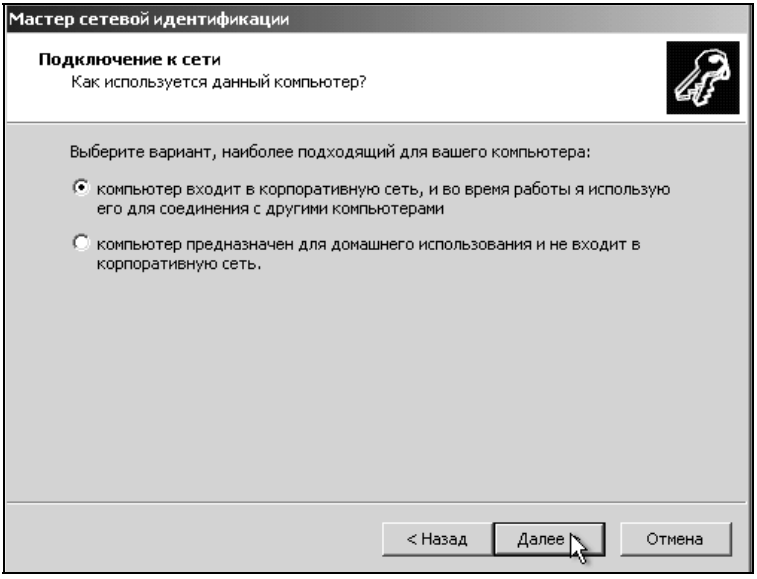



**Рис. 10.17.** Шаг 2

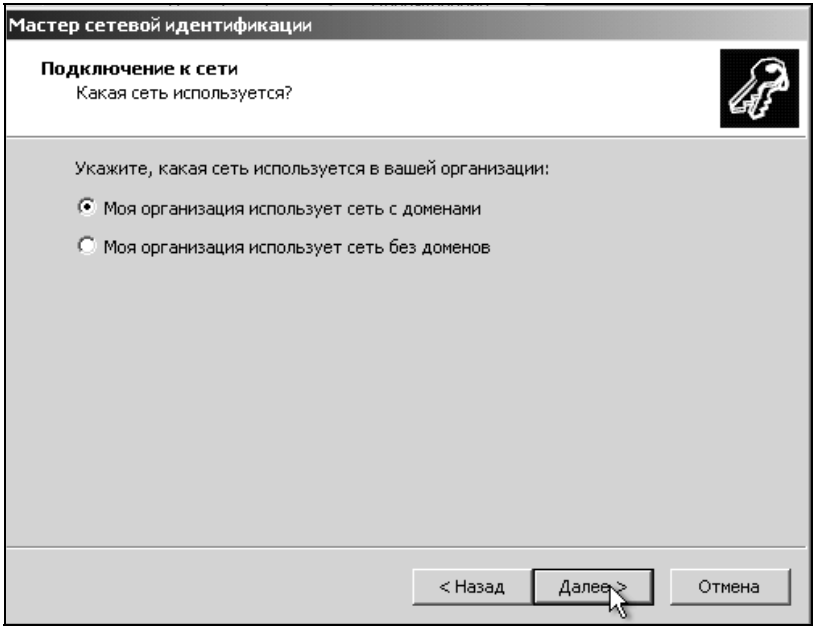



**Рис. 10.18.** Шаг 3. Сеть будет использовать домены

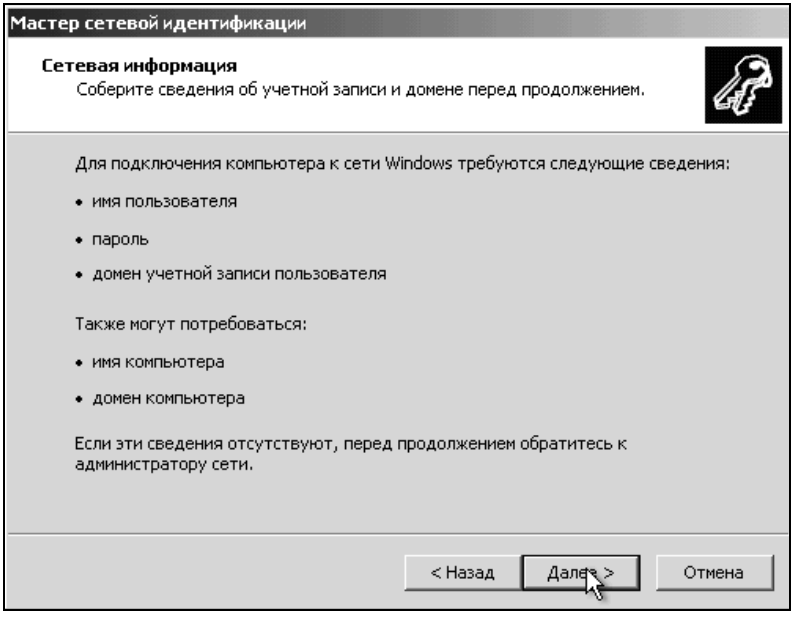



**Рис. 10.19.** Шаг 4. Информация о сведениях, которые пригодятся в дальнейшем

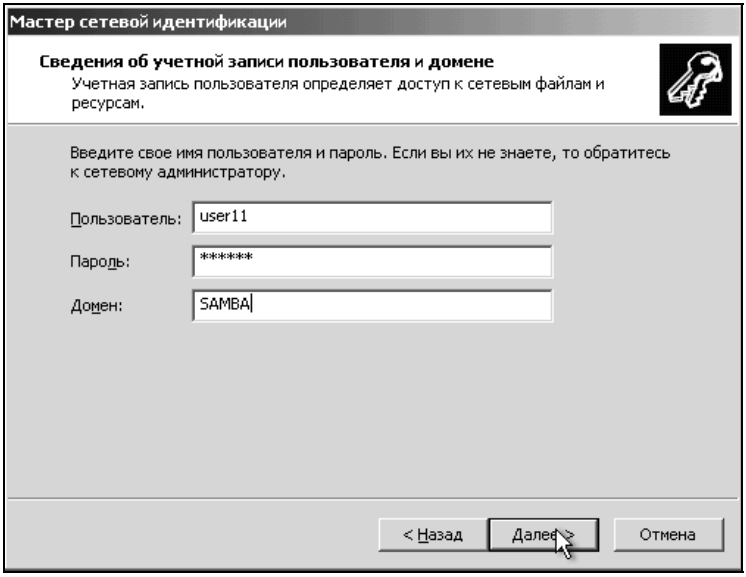



**Рис. 10.20.** Шаг 5

5. С шага 6 собственно и начинается регистрация компьютера в домене. Здесь вводим имя компьютера и домен, в который он входит. На данный момент эти учетные записи уже должны быть в Samba (рис. 10.21).

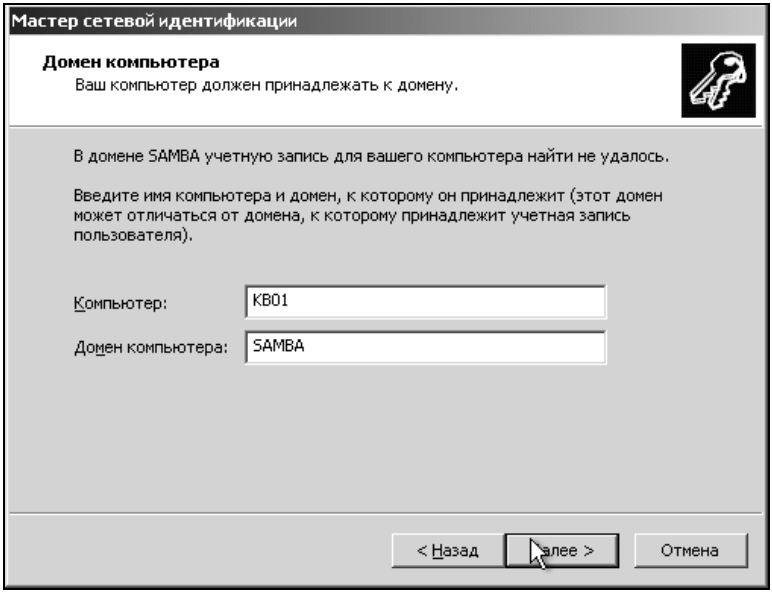



**Рис. 10.21.** Шаг 6

6. На следующем шаге необходимо ввести имя пользователя и пароль, имеющего право подключения к домену. Это пользователь root. Обратите внимание, что имеется в виду пароль для доступа к Samba (рис. 10.22).

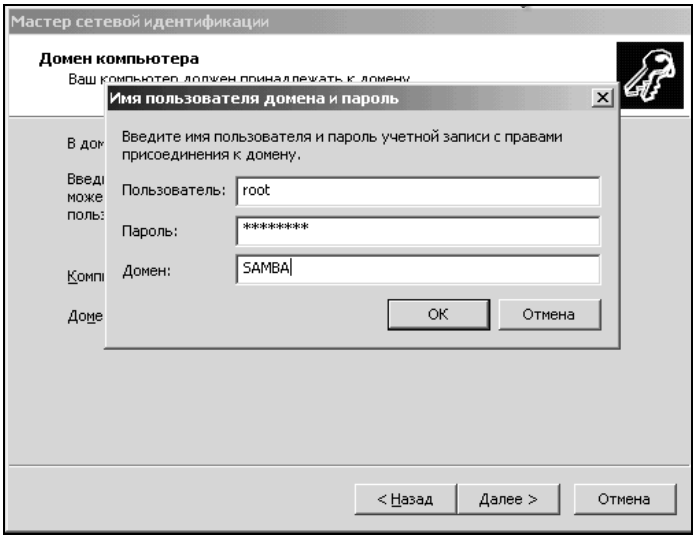



**Рис. 10.22.** Шаг 7

7. Компьютер подключен к домену, теперь создадим для пользователя учетную запись на этом компьютере. Здесь имеется в виду пользователь Samba, уже существующий на данный момент. У нас это user11 (рис. 10.23).

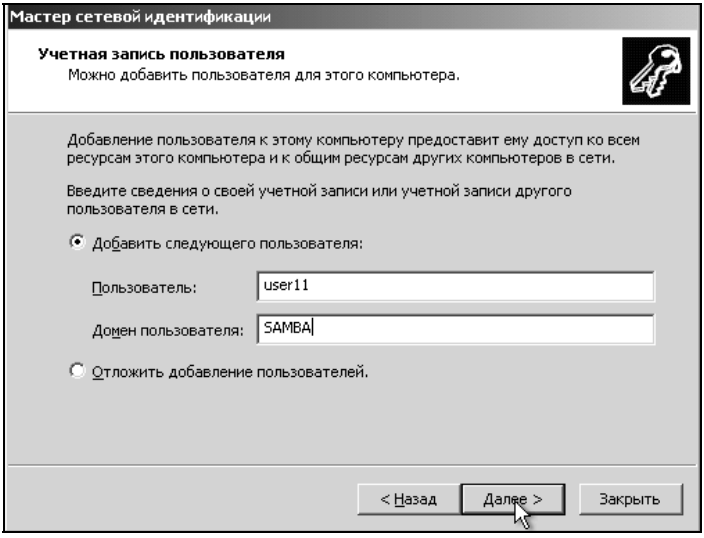



**Рис. 10.23.** Шаг 8

8. Следующим шагом будет назначение пользователю прав на данном компьютере (рис. 10.24).

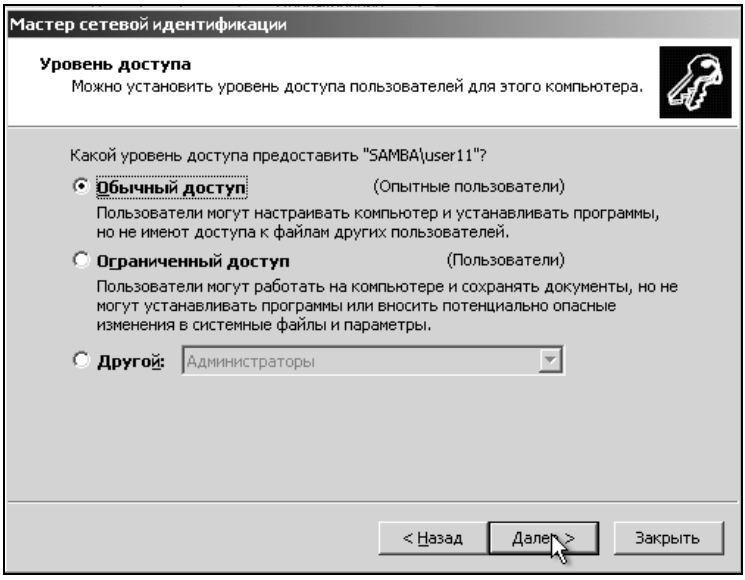



**Рис. 10.24.** Шаг 9. Выбор прав пользователя на этом компьютере

9. Ну, все, мы в домене. Осталось только перезагрузить компьютер (рис. 10.25).

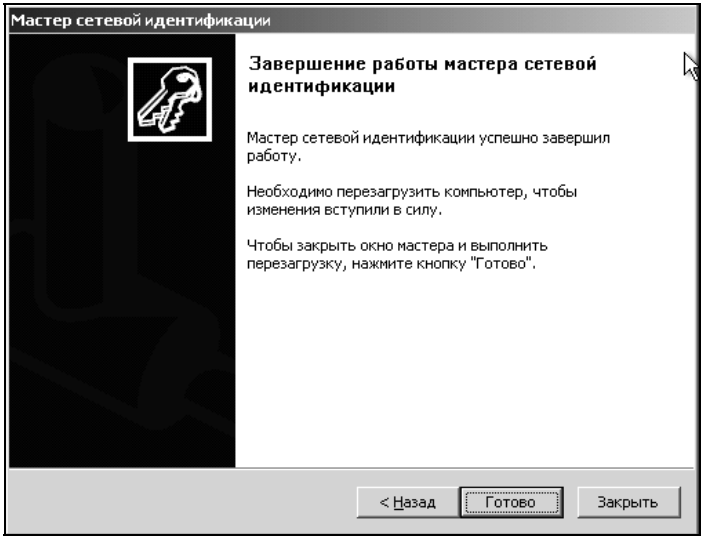



**Рис. 10.25.** Завершение работы мастера

10. На последнем шаге нам осталось войти в домен. Если вы обратили внимание, то приглашение на вход в систему изменилось. Выберите вход в домен Samba, введите имя пользователя и пароль из списка зарегистрированных в Samba. И войдите в систему. Удачной работы в домене (рис. 10.26).

**Рис. 10.26.** Вход в домен

# 10.6. Первый запуск Swat

Начиная со второй версии, в комплект Samba входит утилита графической настройки Swat (Samba web Administation Tool — средство администрирования с использованием Web-интерфейса). На первых порах при конфигурировании Samba лучше пользоваться именно этой утилитой. Она оградит вас от тонкостей синтаксиса smb.conf и предоставит возможность задавать необходимую конфигурацию.

По умолчанию в ASPLinux 7.3 Server Edition сервис Swat не устанавливается его надо выбирать либо во время установки, либо позднее.

Во время установки мы этого не сделали. Так что давайте исправим ситуацию и установим утилиту Swat. Заодно и потренируемся устанавливать пакеты. Для этого необходимо установить пакет: samba-swat-2.2.7a-3.7.3asp.i386.rpm, он располагается на втором диске.

1. Вставляем диск в привод.
2. Монтируем файловую систему на компакт-диске, для этого вводим команду

   ```
   # mount /dev/cdrom.
   ```

3. Запускаем mc (Midnight Commander) и копируем файл samba-swat-2.2.7a-3.7.3asp.i386.rpm во временный каталог.

4. Во временном каталоге вводим команду

```
# rpm -i samba-swat-2.2.7a-3.7.3asp.i386.rpm.
```

5. Пакет установлен.
6. Настраиваем автоматический запуск, для этого файл /etc/xinetd.d/swat приводим в соответствие:

```
# default: on
# description: SWAT is the Samba Web Admin Tool. Use swat \
#              to configure your Samba server. To use SWAT, \
#              connect to port 901 with your favorite web browser.
service swat
{
        port           = 901
        socket_type    = stream
        wait           = no
        only_from      = 127.0.0.1
        user           = root
        server         = /usr/sbin/swat
        log_on_failure += USERID
        disable        = no
}
```

После сделанных изменений необходимо проверить файлы /etc/hosts.allow /etc/hosts.deny и убедиться, что доступ не запрещен.

Проверку запуска можно осуществить, набрав

```
netstat -ln
```

и проверив строки с портом 901 (рис. 10.27).

В нашем случае это верхняя строка. Все нормально, запускаем браузер и в адресной строке набираем:

**http://192.168.0.5:901**.

Если вы работаете не на сервере, а на локальной машине, то получить доступ к этой утилите можно только через Webmin: на вкладке **Servers** следует выбрать пункт **Samba Windows File String** и щелкнуть кнопку **Swat**.

При запуске Swat появляется окно, как на рис. 10.28. Активные кнопки служат для управления. Их смысл ясен из названий.

- **Home** — начальная страница, можно получить доступ к документации;
- **Globals** — настройка глобальных параметров;
- **Shares** — настройка общих ресурсов;
- **Printers** — настройка принтеров;

- **Wizard** — утилита автоматического конфигурирования;
- **Status** — просмотр статистики Samba;
- **View** — просмотр файла конфигураций;
- **Password** — добавление пользователей, смена паролей.

```
> netstat -ln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:901             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:139             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:10000           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
udp        0      0 127.0.0.1:1025          0.0.0.0:*
udp        0      0 192.168.0.5:137         0.0.0.0:*
udp        0      0 0.0.0.0:137             0.0.0.0:*
udp        0      0 192.168.0.5:138         0.0.0.0:*
udp        0      0 0.0.0.0:138             0.0.0.0:*
udp        0      0 0.0.0.0:10000           0.0.0.0:*
udp        0      0 0.0.0.0:475             0.0.0.0:*
udp        0      0 0.0.0.0:111             0.0.0.0:*
Active UNIX domain sockets (only servers)
Proto RefCnt Flags       Type       State         I-Node Path
unix  2      [ ACC ]     STREAM     LISTENING     1459   /tmp/.font-unix/fs7100
unix  2      [ ACC ]     STREAM     LISTENING     1382   /dev/gpmctl
```

**Рис. 10.27.** Список запущенных процессов

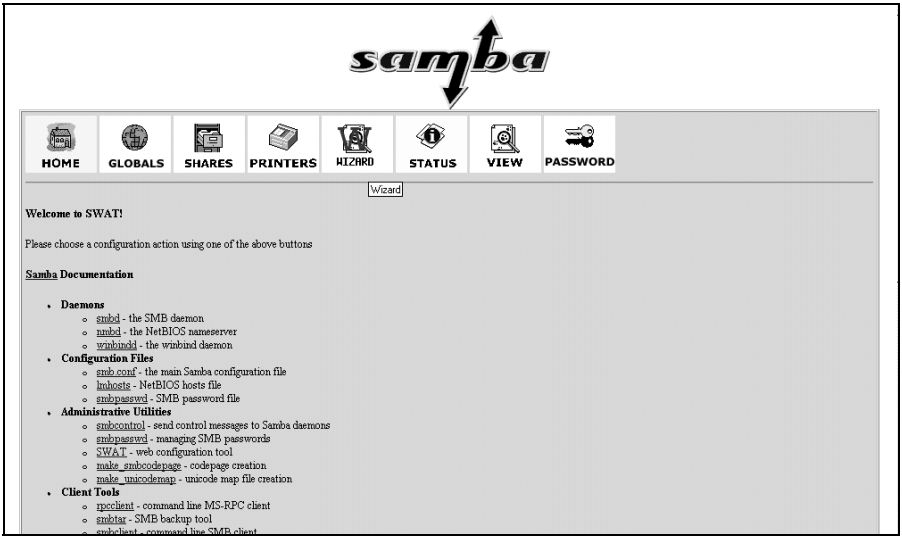



**Рис. 10.28.** Окно Swat

# 10.7. Некоторые переменные Samba

Список переменных Samba приводится в табл. 10.4.

***Таблица 10.4.*** *Переменные Samba*

| Переменная | Описание |
|---|---|
| %S | Название текущего сервиса |
| %P | Корневой каталог текущего сервиса |
| %u | Имя пользователя |
| %g | Имя основной группы пользователя |
| %U | Имя пользователя для сеанса |
| %G | Имя группы для сеанса |
| %H | Рабочий каталог пользователя |
| %v | Версия Samba |
| %m | Netbios-имя машины пользователя |
| %L | Netbios-имя сервера |
| %M | DNS-имя машины пользователя |
| %p | Путь к рабочему каталогу |
| %I | IP-адрес компьютера клиента |
| %T | Текущее время |

# 10.8. Автоматический запуск Samba

По умолчанию Samba не запускается автоматически. Ее можно запустить вручную. Для этого надо файл /etc/rc.d/rc3.d/k35smb переименовать в файл /etc/rc.d/rc3.d/S65smb. Кроме того, заголовок файла /etc/rc.d/init.d/smb необходимо изменить следующим образом:

```
#!/bin/sh
#
# chkconfig: 3 65 35
# description: Starts and stops the Samba smbd and nmbd daemons \
#              used to provide SMB network services.
```

Это обеспечит автоматический запуск сервиса. В варианте работы с Webmin на вкладке **System** выбрать **Boot and Shutdown** и щелкнуть по надписи **smb**.

Следует указать, что при следующем запуске сервис должен стартовать. Затем нажать **Save** и сохранить изменения (рис. 10.29).

**Примечание**

Конечно, такая конструкция работоспособна, вот только можно запутаться, поскольку можно заметить K35smb, не заметив S99smb, и думать, что процесс запускаться не будет. Мелочь, но обратите на это внимание, если используете утилиту.

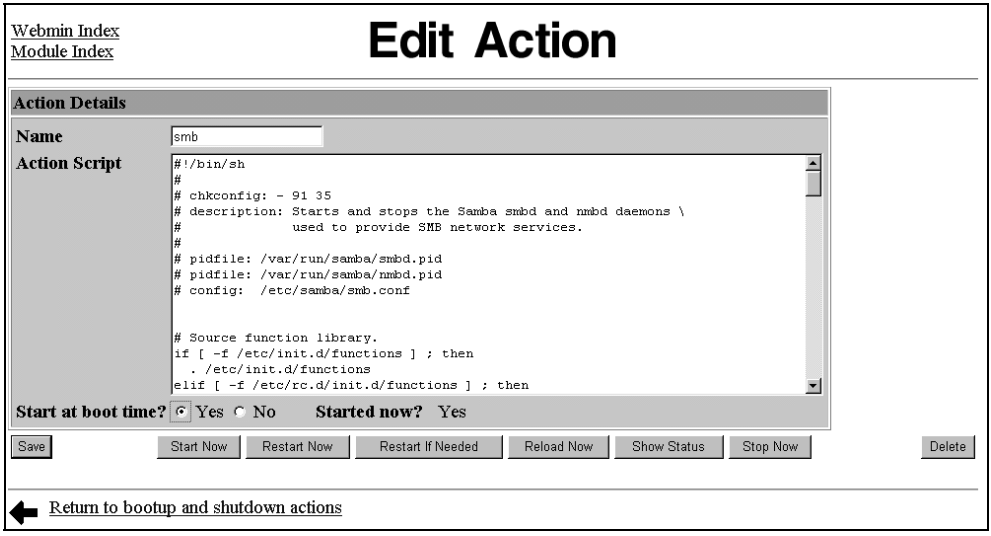
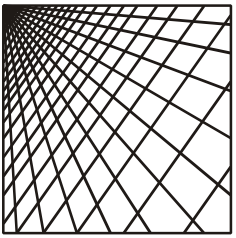



**Рис. 10.29.** Настройка автоматического запуска Samba



# Глава 11

# Службы DNS и DHCP

При построении локальной сети можно обойтись и без служб DNS и DHCP. Конечно, сеть будет не такой гибкой, однако если вы испытываете сложности с настройкой, то можете отказаться, в особенности, от службы DHCP, которая используется достаточно редко.

## 11.1. Краткая характеристика DNS

Компьютеры в сетях TCP/IP распознают друг друга при помощи локальных IP-адресов. Запоминать IP-адрес не совсем удобно, кроме того, числа не несут информации о содержании ресурса. Поэтому компьютерам присваивают символьные имена. В самом деле, проще запомнить **www.webmin.com**, чем IP-адрес 66.35.250.210.

Ясно, что если пользователь оперирует символьными именами, а в сети используются IP-адреса, то кто-то должен заниматься постановкой соответствия "IP-адрес — имя хоста". Этим ведает специальная служба, называется она DNS (Domain Name Service — служба доменных имен).

Работа службы DNS описана в документах RFC882, RFC883, RFC1034, RFC1035. Документы RFC1034, RFC1035 были созданы в 1987 г., но до сих пор считаются основными спецификациями службы DNS, несмотря на то, что существует ряд документов, дополнивших их (например RFC2671 – RFC2673, RFC2535 и др.).

DNS-служба определяет следующие моменты:

- иерархию построения имен компьютеров в сети;
- протокол обмена информацией о соответствии "имя — адрес" между серверами DNS;
- механизм поиска служб в сети;
- правила построения распределенной базы данных "имя — компьютер".

Пространство имен компьютеров в сети имеет вид дерева, ветвями вниз. Началом всему служит корень дерева. Обозначается он точкой. Поскольку

точка располагается в конце имени, то она, как правило, не пишется. Под корнем располагаются домены первого уровня. Далее идут домены второго уровня, третьего и т. д. Например, Linux.ru — это доменное имя второго уровня. Точка в конце пропущена (как мы и говорили). Читать имя надо с конца: ru — домен первого уровня, Linux — доменное имя второго уровня.

Имена первого уровня достаточно стабильны. Помимо страны (например ru, us, de), доменные имена первого уровня могут иметь и другой смысл. Например, com —коммерческие организации, gov — правительственные учреждения, mil — военные ведомства и др.

## 11.2. Как работает служба DNS

Наличие определенной структуры в дереве имен предполагает наличие определенных принципов работы службы DNS.

Каждый домен является главным на своем уровне и знает обо всех компьютерах этого уровня. В то же время он знает о существовании домена более высокого уровня. Поэтому, когда производится поиск имени за пределами своего домена, то в начале запрос идет к корневому серверу, а затем к серверам более низкого уровня, вплоть до разрешения имени (рис. 11.1).

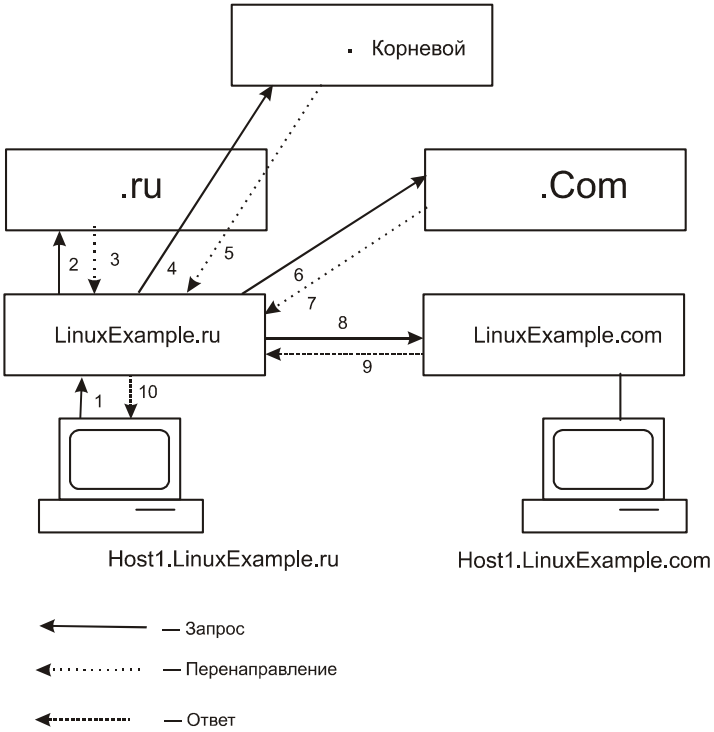



**Рис. 11.1.** Порядок работы службы DNS

Рисунок сделан в предположении, что в серверах имен не использовалось кэширование. Запросы такого рода предполагают значительную загрузку сети, что может привести к сбоям в работе. Поэтому применяется кэширование записей на серверах DNS. Кэширование предполагает сохранение на серверах DNS соответствий "имя — адрес" на наиболее частые запросы. Запись из кэша может удаляться по прошествии определенного времени. Система стремится использовать в кэше наиболее часто встречаемые запросы. Поэтому обычно запросы на серверы первого уровня и популярные серверы всегда хранятся в кэше. Разрешать запросы на верхнем уровне приходится редко.

## 11.3. Пакет BIND и его компоненты

Существует несколько программных реализаций сервера DNS. Наиболее распространенной реализацией является BIND (Berkley Internet Name Domain — система доменных интернет-имен университета Беркли). Существуют три основных разновидности пакетов: BIND 4, BIND 8, BIND 9. Поскольку программный продукт свободно распространяемый, то во все современные дистрибутивы Linux входит, как правило, пакет BIND 9.

В пакет BIND входят следующие компоненты:

- named — демон сервера имен;
- утилиты nslookup, dig, host, позволяющие проверить работу сервисов из командной строки.

Демон named собственно реализует функцию разрешения имен на IP-адреса. Если ему неизвестно какое-либо имя, то он осуществляет разрешение этого имени при помощи других серверов. Серверы имен могут быть нескольких типов (табл. 11.1) и обычно один сервер совмещает сразу несколько режимов работы.

***Таблица 11.1.*** *Типы серверов DNS*

| Тип сервера | Описание |
|---|---|
| Авторитарный | Официальный представитель зоны, ответы этого сервера самые точные и не устаревшие |
| Первичный | Основное хранилище данных зоны. Вся информация хранится здесь |
| Вторичный | Копирует данные с главного сервера и страхует главный сервер |
| Неавторитарный | Отвечает на запросы из своего кэша. В принципе, ответ может быть некорректным, хотя такое встречается редко |
| Рекурсивный | Осуществляет запросы от имени клиента до полного разрешения адреса. Клиенту выдается соответствие "имя —адрес" |
| Нерекурсивный | Перенаправляет клиента к другому серверу. Как правило, все серверы верхнего уровня нерекурсивные, так как они обрабатывают большое число запросов |

Все серверы, кроме Linux.ru, на рис. 11.1 были нерекурсивными. Сервер Linux.ru рекурсивный.

Для отладки работы службы DNS используются программы nslookup, dig, host. Здесь они приводятся обзорно. Если вас интересует их работа, то обращайтесь к документации по этим утилитам. Тем не менее даже столь малые сведения позволят вам проверить свой сервер имен.

## Утилита nslookup

Утилита nslookup предназначена для передачи запросов серверу имен. Если программу запустить без параметров, то она обратится к первому серверу имен, взятому из файла /etc/resolv.conf. Конкретный сервер имен задается принудительно, для этого в строке вызова после nslookup указывается дефис и DNS-сервер (рис. 11.2).

```
[root@localhost root]# nslookup -sil
> set q=any
> samba.
Server:         127.0.0.1
Address:        127.0.0.1#53

samba
        origin = dns.samba
        mail addr = admni.mail.samba
        serial = 2005031501
        refresh = 10800
        retry = 3600
        expire = 60448
        minimum = 7200
samba   nameserver = dns.samba.
samba   mail exchanger = 10 mail.samba.
>
```

**Рис. 11.2.** Пример выполнения nslookup

## Утилита dig

Утилита dig, так же как и nslookup, предназначена для отладки и выдает данные о формате, который может быть использован непосредственно сервером DNS. В ответе с двумя символами точки с запятой идут строки комментария (рис. 11.3).

```
[root@localhost root]# dig -?
Invalid option: -?
Usage:  dig [@global-server] [domain] [q-type] [q-class] {q-opt}
        {global-d-opt} host [@local-server] {local-d-opt}
        [ host [@local-server] {local-d-opt} [...]]

Use "dig -h" (or "dig -h | more") for complete list of options
[root@localhost root]# dig kb1.samba.

; <<>> DiG 9.2.1 <<>> kb1.samba.
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64319
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;kb1.samba.                     IN      A

;; ANSWER SECTION:
kb1.samba.              7200    IN      A       192.168.0.6

;; AUTHORITY SECTION:
samba.                  7200    IN      NS      dns.samba.

;; ADDITIONAL SECTION:
dns.samba.              7200    IN      A       192.168.0.5

;; Query time: 26 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sat Mar 26 14:00:13 2005
;; MSG SIZE  rcvd: 77

[root@localhost root]#
```

**Рис. 11.3.** Пример запуска программы dig

## Утилита host

Утилита host возвращает IP-адрес хостов. Программа выдает только IP-адрес, поэтому ее применение для отладки ограничено.

# 11.4. Как читать файл настроек BIND

## Файл named.conf

Демон сервера имен имеет несколько конфигурационных файлов. Основным файлом настроек является файл /etc/named.conf (листинг 11.1).

**Листинг 11.1. Файл настроек /etc/named.conf (пример)**

```
options {
        directory "/var/named";
        /*
         * If there is a firewall between you and nameservers you want
```

```
         * to talk to, you might need to uncomment the query-source
         * directive below.  Previous versions of BIND always asked
         * questions using port 53, but BIND 8.1 uses an unprivileged
         * port by default.
         */
        // query-source address * port 53;
};

//
// a caching only nameservers config
//
controls {
        inet 127.0.0.1 allow { localhost; } keys { rndckey; };
};
zone "." IN {
        type hint;
        file "named.ca";
};

zone "localhost" IN {
        type master;
        file "localhost.zone";
        allow-update { none; };
};

zone "0.0.127.in-addr.arpa" IN {
        type master;
        file "named.local";
        allow-update { none; };
};

include "/etc/rndc.key";
```

Разделы файла заключены в фигурные скобки. Строки, начинающиеся символом //, являются комментарием. Инструкция

```
options {
        directory
```

задает расположение файлов зоны. Следующие операторы описывают файлы зон.

## Файлы баз данных зон

Пути к файлам баз данных зон определены файлом named.conf. Файлы баз данных зон состоят из ресурсных записей. Перечень ресурсных записей приведен в табл. 11.2.

***Таблица 11.2.*** *Ресурсные записи файла баз данных зон*

| Запись | Комментарий |
|---|---|
| SOA | Начало зоны. Содержит административную информацию |
| NS | Идентифицирует серверы имен для данного домена |
| A | Отображение имени в адрес, составляют основную часть базы данных |
| PTR | Обратный указатель обеспечивает обратный перевод IP-адресов в имена |
| MX | Используется системой электронной почты для более эффективной маршрутизации |
| CNAME | Каноническое имя, позволяет назначать узлу дополнительные имена |
| SRV | Определяют местонахождение служб в пределах домена |
| TXT | Позволяют добавлять произвольный текст в файл |

Кроме того, в файлах зон также могут использоваться специальные символы (табл. 11.3).

***Таблица 11.3.*** *Спецсимволы, используемые в файлах зон*

| Символ | Значение |
|---|---|
| @ | Обозначает имя текущего домена |
| ( ) | Объединяет несколько строк в одну строку |
| * | Метасимвол. Может применяться только имени. Обозначает все возможные комбинации |
| ; | Строки комментария |

Рассмотрим более подробно формат основных записей.

## Запись SOA

SOA (Start of authority — начало полномочий) обозначает начало зоны. Зона продолжается до тех пор, пока не встретится другая запись SOA. Наличие

SOA указывает на то, что данный сервер DNS является первичным источником информации для данного домена.

Разберем подробнее возможную форму записи для ресурса. Имена узлов приведены для примера и являются вымышленными.

```
stuff.org. IN SOA first.stuff.org. admni.first.stuff.org. (
                  2004123101        ; Порядковый номер
                  10800             ; Период обновления в секундах
                  3600              ; Интервал между попытками обновления
                                    ; в секундах
                  60448             ; Период устаревания в  секундах
                  7200)             ; Время жизни в секундах
```

Данный файл описывает зону stuff.org. Класс зоны — Internet (IN). Первичным сервером зоны является сервер first.stuff.org. Адрес электронной почты администратора домена admin.first.stuff.org, читать этот адрес надо как admin@first.stuff.org. Следует также обратить внимание на точку в конце строки.

Порядковый номер определяет момент, когда надо загружать новую конфигурацию. При каждом изменении конфигурации порядковый номер должен монотонно возрастать. Первая часть — это дата изменения, а последняя — это номер изменения в рамках данной даты. В нашем случае это 31 декабря 2004 г.

Следующие четыре периода времени по умолчанию имеют формат в секундах, однако допускается и другая запись, для этого используются буквы:

- w — неделя;
- d — дни;
- h — часы;
- m — минуты.

Период обновления обозначает, как часто вторичные сервера обновляют данные. Обновление производится в том случае, если в конфигурации произошли изменения. О наличии изменений говорит разница в порядковых номерах. Обновление производится в том случае, если порядковый номер на первичном сервере больше. Интервал между попытками обновления означает период времени, через который будет произведена повторная попытка обновления, в случае если предыдущая попытка не удалась.

Интервал — период устаревания, в течение которого вторичные сервера будут обслуживать домен, если первичный сервер не отвечает (невозможно провести обновление файла зоны).

Время нахождения в кэше отрицательных ответов описано последним параметром.

## Запись NS

Эта запись задает сервер имен для данного домена. Обратите внимание на точку в конце записи.

Пример:

```
stuff.org. IN NS first.stuff.org.
stuff.org. IN NS second.stuff.org.
```

Здесь для домена stuff.org. задано два сервера имен: first.stuff.org. и second.stuff.org.

## Запись A

Эта запись является основной записью базы данных DNS и отображает имя в адрес. Для каждого сетевого интерфейса существует одна запись A. Для компьютера может существовать несколько записей, по каждой на один сетевой интерфейс.

Пример:

```
first.stuff.org. IN A 128.100.241.212
first.stuff.org. IN A 128.100.241.213
second.stuff.org.IN A 128.100.241.214
```

Здесь первые две записи относятся к одному компьютеру с именем first.stuff.org. и двумя сетевыми картами, имеющими IP-адреса 128.100.241.212 и 128.100.241.213.

Третья запись относится ко второму компьютеру second.stuff.org., сетевая карта которого имеет адрес 128.100.241.214.

## Запись PTR

Эта запись обеспечивает обратный перевод, то есть перевод IP-адресов в символьные имена. Точно так же, как и в предыдущем случае, для каждого сетевого интерфейса существует только одна запись PTR. В простейшем случае это записанный наоборот IP-адрес *(см. пример записи A)* с суффиксом in-addr.arpa.

Пример:

```
212.241.100.128. in-addr.arpa. IN PTR first.stuff.org.
213. 241.100.128. in-addr.arpa. IN PTR first.stuff.org.
214. 241.100.128. in-addr.arpa. IN PTR second.stuff.org.
```

## Запись MX

Эта запись используется в электронной почте для более эффективной маршрутизации. Запись MX подменяет адресатов сообщений.

Пример:

```
stuff.org. IN MX 10 first.stuff.org.
stuff.org. IN MX 20 second.stuff.org.
```

Эта запись означает, что вместо сервера почты stuff.org. будут использоваться серверы first.stuff.org. и second.stuff.org., 10 и 20 означают приоритет: чем меньше число, тем больше приоритет. Допустимым является диапазон приоритета 0 – 65535. На практике приоритет используется с шагом 10, чтобы можно было поставить что-то между 10, 20 и 30, не переписывая весь файл.

## Запись CNAME

Каноническое имя позволяет назначать узлу дополнительные имена. Эта запись используется для закрепления за компьютером определенных функций.

Пример:

```
www. stuff.org. IN CNAME second.stuff.org.
```

Здесь узлу second.stuff.org. присвоено каноническое имя www.stuff.org. Соответственно, при вводе в адресной строке

**www. stuff.org.**

вы попадете на узел second.stuff.org.

## Запись SRV

Эта запись определяет местонахождение служб в пределах домена. С помощью этой записи легко менять адреса служб.

Формат записи:

```
Служба.протокол.имя IN SRV приоритет вес порт сервер
```

реализован в примере:

```
http.tcp.www IN SRV 10 0 80 second.stuff.org..
```

## Запись TXT

Эта запись задает произвольный текст. Например:

```
IN TXT "Это произвольный текст".
```

# 11.5. Простой пример собственного домена

При выборе имени домена надо быть осторожным и помнить о том, чтобы никого не беспокоить во внешнем мире. Например, домен можно назвать localdomain, но в нашем случае это будет samba.

При чтении листингов обязательно обращайте внимание на точки в конце имен. Отсутствие точки означает, что отсчет будет идти не от корневого домена, а от текущего.

Также обратите внимание, что в файле ресурсов named.conf и зон точка не ставится (листинги 11.2—11.5).

**Листинг 11.2. Файл /etc/named.conf**

```
; Вначале опишем локальную зону
       zone "0.0.127.in-addr.arpa" {
                  type master;
                  file "/etc/domain/ld127";};
     ;создаем основной  сервер для  0.0.127.in-addr.arpa файл зоны в
     ; файле "domain/ld127"
; создадим нашу основную зону
; она будет описана в файле domain/ld
zone "samba"{
           notify no
           type master
           file "domain/ld";};
; Обратная зона поиска
zone "0.168.192.in-addr.arpa" {
          notify no;
          type master;
          file "ld168";};
```

**Листинг 11.3. Файл зоны ld127**

```
@ IN SOA dns. samba. admin.mail. samba. (
                          20050315  ; Порядковый номер
                           10800    ; Период обновления, в секундах
                           3600    ; Интервал между попытками обновления,
                                   ; в секундах
                          60448     ; Период устаревания, в  секундах
                          7200)     ; Время жизни, в секундах
```

```
 ; Зададим сервер имен,  первую часть  - @ IN можно опустить
                NS       dns. samba.
; Это значит, что машина с адресом 127.0.1 называется localhost
     1          PTR  localhost.
```

**Листинг 11.4. Файл зоны ld**

```
; Файл зоны samba

@ IN SOA dns. samba. admni.mail. samba. (
                          2005031501       ; Порядковый номер
                          10800     ; Период обновления, в секундах
                           3600     ; Интервал между попытками
                                    ; обновления, в секундах
                          60448     ; Период устаревания, в секундах
                          7200)     ; Время жизни, в секундах
     NS   dns. samba.;
     MX   10 mail. samba. ; Почтовый сервер
; Записи о ресурсах
; Сервер
localhost      A    127.0.0.1
dns            A    192.168.0.5
mail           A    192.168.0.5
smb            A    192.168.0.5 ; зарезервируем  имя

; Рабочие станции
kb1            A    192.168.0.6
               TXT  "Рабочая станция в конструкторском бюро!"
buch1          A    192.168.0.7
               TXT  "Рабочая станция в бухгалтерии"
```

**Листинг 11.5. Файл зоны ld168**

```
@ IN SOA dns. samba. admni.mail. samba. (
                          2005031501       ; Порядковый номер
                           10800    ; Период обновления, в секундах
                           3600     ; Интервал между попытками
                                    ; обновления, в секундах
                          60448     ; Период устаревания, в секундах
                          7200)     ; Время жизни, в секундах
```

```
   NS   dns. samba.;
; Обратные записи о ресурсах
5  PTR  dns. samba.
6  PTR  kb1. samba.
7  PTR  buch1. samba.
```

## Проверка

Теперь запускаем службу DNS. Для этого вводим

```
# ndc start.
```

То же самое можно сделать и через Webmin. Запускаем Webmin, переходим на вкладку **Servers** и щелкнем по пункту **BIND DNS Server**. Открывается окно управления сервером DNS (рис. 11.4). В самом низу этого окна (на рисунке не поместилась) находится кнопка **Start Name Server** (или **Apply Change**, если сервер запущен), эту кнопку необходимо нажать. Сервер будет запущен, во втором случае будут приняты сделанные изменения.

Отправляем проверочный запрос:

```
# nslookup
>set q=any
> kb1.samba.
```

Результаты запроса представлены на рис. 11.5.

Как видим, сервер DNS работает. Теперь проверим, работает ли обратный поиск. Для этого снова воспользуемся программой nslookup (рис. 11.6).

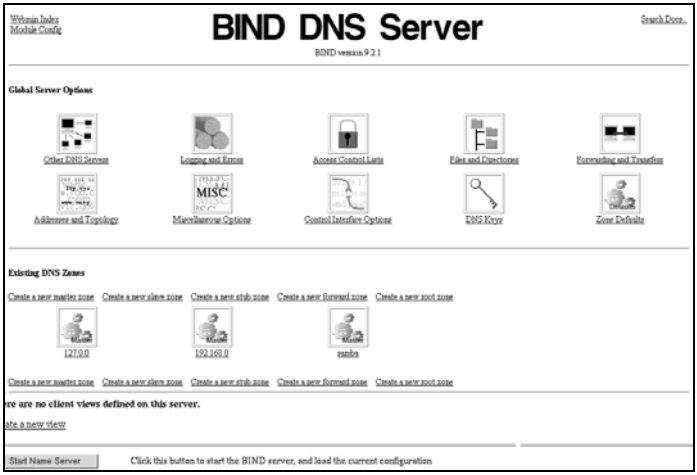



**Рис. 11.4.** Окно управления DNS-сервером

```
[root@localhost root]# nslookup -sil
> set q=any
> kb1.samba
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:   kb1.samba
Address: 192.168.0.6
>
```

**Рис. 11.5.** Работа с утилитой nslookup

```
[root@localhost root]# nslookup -sil
> set q=any
> 192.168.0.6
Server:         127.0.0.1
Address:        127.0.0.1#53

6.0.168.192.in-addr.arpa        name = kb1.samba.
>
```

**Рис. 11.6.** Проверка работоспособности обратного поиска

Из запроса видно, что сервер работает. Итак, DNS мы настроили, теперь настроим DHCP.

# 11.6. Краткая характеристика DHCP

Даже в крупных локальных сетях (100 и более машин) нет недостатка в IP-адресах. Тем не менее для полноты картины мы рассмотрим, как организовать несложный DHCP-сервер.

Протокол DHCP (Dynamic Hosts Configuratin Protocol — протокол динамического конфигурирования узлов) призван облегчить конфигурирование хостов при включении их в сеть. Протокол определен в стандартах RFC2131 и RFC2132.

Протокол DHCP относится к классу клиент-серверных протоколов. На стороне сервера за его работу отвечает демон DHCP (dhcpd).

Через протокол DHCP централизовано выделяются IP-адреса в локальной сети, а также автоматически настраиваются сетевые и административные параметры хостов в сети. Основными параметрами, выдаваемыми DHCP-сервером, являются:

- IP-адрес, выделяемый в аренду для текущего хоста;
- сетевая маска;

- адреса DNS-серверов;
- адреса WINS-серверов;
- адреса шлюзов.

Существует множество других параметров, которые настраиваются при помощи DNS, однако они используются крайне редко, так что о них говорить не имеет смысла.

Работа протокола DHCP выглядит следующим образом (рис. 11.7).

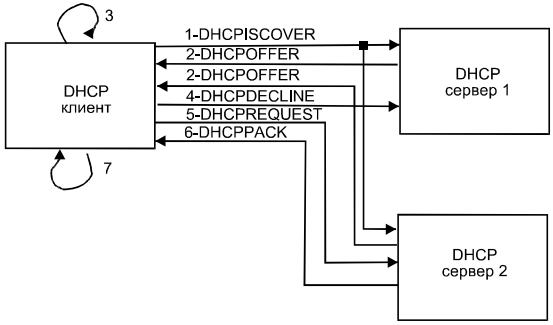



**Рис. 11.7.** Стадии процесса выделения IP-адреса

DHCP-клиент посылает широковещательный запрос DCHPISCOVER, который принимают все DHCP-серверы.

Все DHCP-серверы посылают ответ DHCPOFFER клиенту, запросившему сервис DHCP.

Клиент рассматривает DHCPOFFER в порядке их поступления.

Ненужные клиент отклоняет посылкой DHCPDECLINE.

Принятие клиент подтверждает посылкой DHCPREQUEST (одному из серверов).

Тот сервер, услуги которого приняты, посылает DHCPPACK.

Информация принимается и обрабатывается клиентом.

## 11.7. Файл настроек dhcpd и их параметры

Файл конфигураций DHCP /etc/dhcpd.conf состоит из нескольких разделов. Каждый из его разделов отвечает за свою область. Параметры в глобальной секции справедливы для всех разделов.

Если вы изменяли файл конфигурации, то для того, чтобы система считала его, и изменения вступили в силу, необходимо перезапустить сервер.

После начала работы DHCP-сервер начинает выделять IP-адреса клиентам. Выделенные адреса записываются в файл /var/dhcpd/dhcp.leases (в принципе, он может располагаться и в другом месте).

Запись в файл dhcp.leases идет до тех пор, пока его размер не достигнет указанной величины. Тогда делается копия файла dhcp.leases и создается новый файл dhcp.leases, но уже без устаревшей информации.

В принципе, DHCP-сервер достаточно прост в настройке. Здесь мы приводим только основные параметры, позволяющие настроить несложную конфигурацию. Если у вас возникла более сложная ситуация, и данных этой книги не достаточно, обращайтесь к страницам справочного руководства. Используйте для этого команду man.

## Глобальные параметры

Эти параметры относятся ко всем секциям файла dhcp.conf (листинг 11.6). Большинство параметров начинается со слова `option`.

**Листинг 11.6. Файл /etc/dhcpd.conf (фрагмент)**

```
; Идентификация сервера
server-identifier dhcp.samba;
; Имя домена, обратите внимание,  точки в конце нет
option domain-name samba;
; имя сервиса
option domain-name-servers dhcp.samba;
; Маска подсети
option sub-netmask 255.255.255.0;
; Широковещательный адрес
option boardcast-address 192.168.0.255;
; Шлюз по умолчанию
option routers 192.168.0.12;
; Адрес сервера netBios указывает на то, что обслуживаются клиенты,
; имеющие ОС windows
option netbios-name-servers 192.168.0.5
; Время лизинга адреса по умолчанию, в секундах (10 lytq)
default-lease-time 864000;
; Максимальное время лизинга адреса, в секундах (30 дней — 1 месяц)
max-lease-time 2592000;
```

Если опции содержат несколько параметров, то они перечисляются в порядке важности. Строка всегда заканчивается знаком точки с запятой. Пробелы и символы табуляции игнорируются демоном при чтении файла. Сделано это для красоты оформления и удобства последующего чтения файла конфигурации.

## Опция subnet

Опция subnet присутствует в любом конфигурационном файле dhcp.conf и представляет собой описание подсети (листинг 11.7). Эта опция указывает демону, какая подсеть обслуживается, также здесь задается пул адресов, из которого сервер может выдавать IP-адреса в аренду. Если задать опции, описанные в глобальных параметрах, то локальные параметры их перекроют. Это очень удобно, поскольку можно задавать параметры, отличные от глобальных для какой-то отдельной секции.

**Листинг 11.7 Файл /etc/dhcpd.conf (фрагмент)**

```
Subnet 192.168.0.0 netmask 255.255.255.0 {
                        range 192.168.0.6   192.168.0.145;
                        range 192.168.0.150 192.168.0.199;
}
```

## Опция shared network

Опция shared network позволяет сгруппировать две или несколько подсетей с целью совместного использования (листинг 11.8) и используется для разбиения больших сетей на подсети. В нашем случае эту опцию вряд ли придется использовать. Для каждой из подсетей можно задавать отдельно параметры, например: маршрутизатор по умолчанию, время лизинга или любой другой.

**Листинг 11.8. Файл /etc/dhcpd.conf (фрагмент)**

```
shared-network OOO-fortuna-sm {
         Subnet 192.168.0.0 netmask 255.255.255.0 {
                        range 192.168.0.6   192.168.0.145;
                        range 192.168.0.150 192.168.0.199;
         }
}
```

# 11.8. Установка DHCP-сервера. Связь DNS и DHCP

Перед установкой DHCP-сервера важно понять следующее. Пока адреса статичны (задаются администратором при настройке сети, каждой машине свой IP-адрес), то все нормально, важно только поддерживать файлы зон DNS в актуальном состоянии. Но как только распределение IP-адресов перекладывается на плечи DHCP-сервера, то, скорее всего, IP-адрес компьютера будет меняться, поскольку время лизинга IP-адреса ограничено. Это приведет к несоответствию реальных адресов и тех, что закреплены за узлами в базе данных DNS.

Выхода из ситуации два. Первый состоит в том, чтобы конкретной машине выделялся конкретный IP-адрес. Делается это на основе MAC-адреса сетевой карты. Второй состоит в том, чтобы DHCP сервер вносил изменения в базу данных DNS.

Недостаток первого способа в том, что для каждого подключения необходимо в конфигурационном файле DCHP-сервера вести отдельную запись, причем эта запись будет меняться при замене сетевой карты.

Недостаток второго состоит в относительной сложности (для начинающего пользователя) установки и настройки DCHP-сервера.

Возможно, именно эти обстоятельства, а также то, что объективных причин для использования DHCP-сервера в небольших сетях мало, привели к тому, что DHCP-сервер по умолчанию не устанавливается.

Мы рассмотрим оба способа настройки DHCP-сервера. Начнем с того способа, по которому компьютеру выделятся жестко заданные IP-адреса. В состав ASPLinux Server входит DHCP-сервер второй версии, который по умолчанию не поддерживает обмена данных с DNS. На этапе установки мы не установили DHCP-сервер. Поэтому нам его предстоит установить. Возьмите файл dhcp-2.0pl5-8.i386.rpm со второго диска ASPLinux, скопируйте его во временный каталог и дайте команду:

```
rpm -i dhcp-2.0pl5-8.i386.rpm
```

Все, пакет установлен. Теперь создаем простенький файл конфигурации. Выделение жестко заданных IP-адресов достигается использованием опции host в разделе subnet. В этом случае изменение IP-адресов, выделяемых узлу, осуществляется уже в файлах настроек DNS. Соответственно, для каждого узла необходимо иметь отдельную запись в файлах конфигурации DNS (листинг 11.9).

**Листинг 11.9. Файл /etc/dhcpd.conf для домена samba**

```
option domain-name-servers 192.168.0.5; # Адрес DNS-сервера
option domain-name "samba"; #Имя домена
```

```
shared-network OOO-fortuna-sm {
            Subnet 192.168.0.0 netmask 255.255.255.0
                      { range 192.168.0.6   192.168.0.145;
                        range 192.168.0.150 192.168.0.199;
                       host kb1 {
                               hardware ethernet 04:ee:17:98:f5:87;
                               fixed-address kb1.samba;
                               }
                       # и так далее, для каждой машины в сети
                       # MAC-адрес сетевой карты можно взять из
                       # документации на карту или проанализировав log
                       # DHCP-сервера.
                       }
}
```

Начиная с версии 3.0, в пакете DHCP реализовано обновление для DNS-сервера. Итак, вам потребуется скачать свежую версию DHCP-сервера по адресу **http://www.isc.org/products/DHCP/**. Объем дистрибутива составляет порядка 1 Мбайт. После того как скачали, ее нужно установить. Для этого выполняются следующие действия:

1. Создадим специальный каталог, куда скопируем архив.
2. Распакуем архив, для чего, войдя в каталог, выполним:

   `tar xvfz dhcp-3.0pll.tar.gz.`

3. Перейдем во вновь созданный программой распаковки каталог.
4. Выполним ./configure.
5. Далее собираем пакет командой make.
6. Инсталлируем пакет командой make install.

Если команду make система не распознала, значит, вам необходимо установить пакет gcc-2.96-113asp.i386.rpm. Устанавливаем уже знакомым нам методом:

```
rpm -i gcc-2.96-113asp.i386.rpm.
```

Обновлять DNS-сервер нет необходимости, так как он достаточно свежей версии. Теперь собственно к настройке связи DNS и DHCP-сервера. Первым делом создаем секретный ключ для того, чтобы серверы доверяли друг другу. Для DNS-сервера 9-й версии делаем это командой:

```
dnssec-keygen -a HMAC-MD5 -b 128 -n # USER DHCP_UPDATER.
```

После того как вы выполните эту команду, в текущем каталоге появится файл, он и будет содержать этот секретный ключ.

Затем в глобальные параметры dhcpd.conf добавляем:

```
ddns-update-style interim;
```

```
key DHCP_UPDATER {
Algorithm: 157 (HMAC_MD5);
Key: s4e6lkdZx70vKpAUPHs1Zg== }.
```

В файл named.conf добавляем:

```
key DHCP_UPDATER {
Algorithm: 157 (HMAC_MD5);
Key: s4e6lkdZx70vKpAUPHs1Zg==};.
```

Возможен и другой вариант, просто добавляем в dhcpd.conf:

```
ddns-update-style ad-hoc;.
```

Выбор одного из трех вариантов за вами. Но на первых порах лучше отказаться от использования DHCP-сервера. Когда придет опыт, тогда и настроите.

## 11.9. Запуск служб DNS и DHCP

В *главе 10* мы обсуждали, как настроить автоматический запуск сервисов на примере Samba. Здесь ситуация в точности такая же. Мы можем настроить автоматический запуск сервисов путем непосредственного редактирования файлов в каталоге /etc/rc.d/rc3.d либо через Webmin (вкладка **Bootup and Shotdown**).

Начнем с DNS-сервера. Переименовываем /etc/rc.d/rc3.d/K45named в /etc/rc.d/rc3.d/S55named. Заголовок файла /etc/rc.d/init.d/named заменяем:

```
#!/bin/bash
#
# named           This shell script takes care of starting and stopping
#                 named (BIND DNS server).
#
# chkconfig: 3 55 45
```

Теперь DHCP-сервер: переименовываем /etc/rc.d/rc3.d/K35dhcpd в /etc/rc.d/rc3.d/S65dhcpd. Заголовок файла /etc/rc.d/init.d/dhcpd заменяем:

```
#!/bin/sh
#
# dhcpd         This shell script takes care of starting and stopping
#               dhcpd.
#
# chkconfig: 3 65 35
# description: dhcpd provide access to Dynamic Host Control Protocol
```

На вкладке **System** щелкнем по **Bootup and Shootdown** и по имени соответствующего процесса: для DNS-сервера это — named; для DHCP-сервера это — dhcpd. Отвечаем **Start at boot time? — Yes** и сохраняем **Save**.

На рис. 11.8 показана операция для DHCP, для DNS она аналогична.

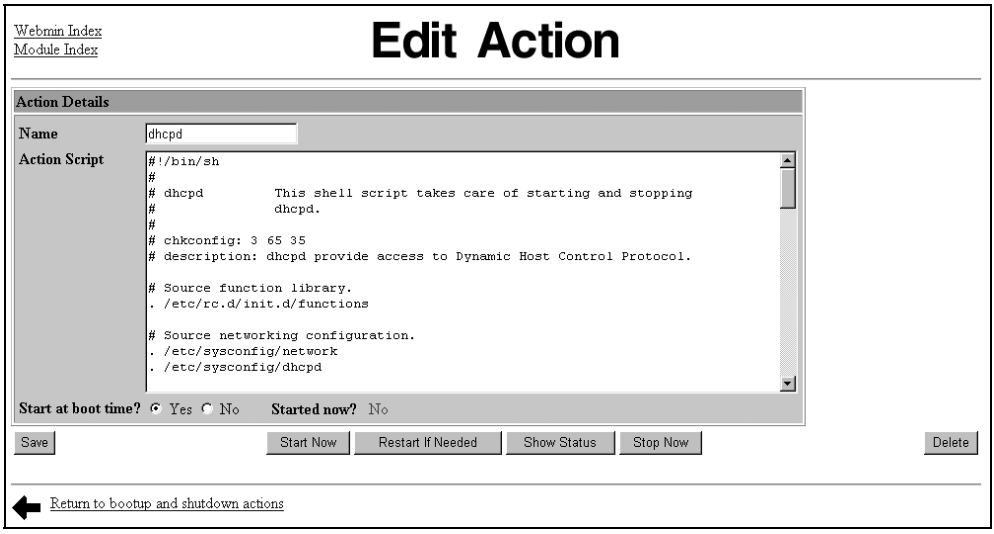



**Рис. 11.8.** Настройка запуска dhcpd

## 11.10. Настройка клиентской части Windows 98

Итак, в предыдущих разделах мы рассмотрели настройку серверов DHCP и DNS, теперь настало время настроить клиентскую часть операционных систем.

1. Начнем с настройки службы DNS (рис. 11.9).
2. На рабочем столе правой кнопкой мыши щелкнем по значку сетевого окружения.
3. Выбираем пункт **Свойства**.
4. В открывшемся окне вбираем связку **TCP/IP — Сетевая карта**.
5. Щелкаем по кнопке **Свойства**.
6. В открывшемся окне выбираем включить DNS.
7. Если хотите настроить работу компьютера с DHCP-сервером, то на вкладке **IP-адрес** выберите **Получить IP-адрес автоматически**.
8. Вводим имя локального компьютера, его домен и адрес DNS. Жмем **OK**.
9. Перезагружаем машину, чтобы изменения вошли в силу.

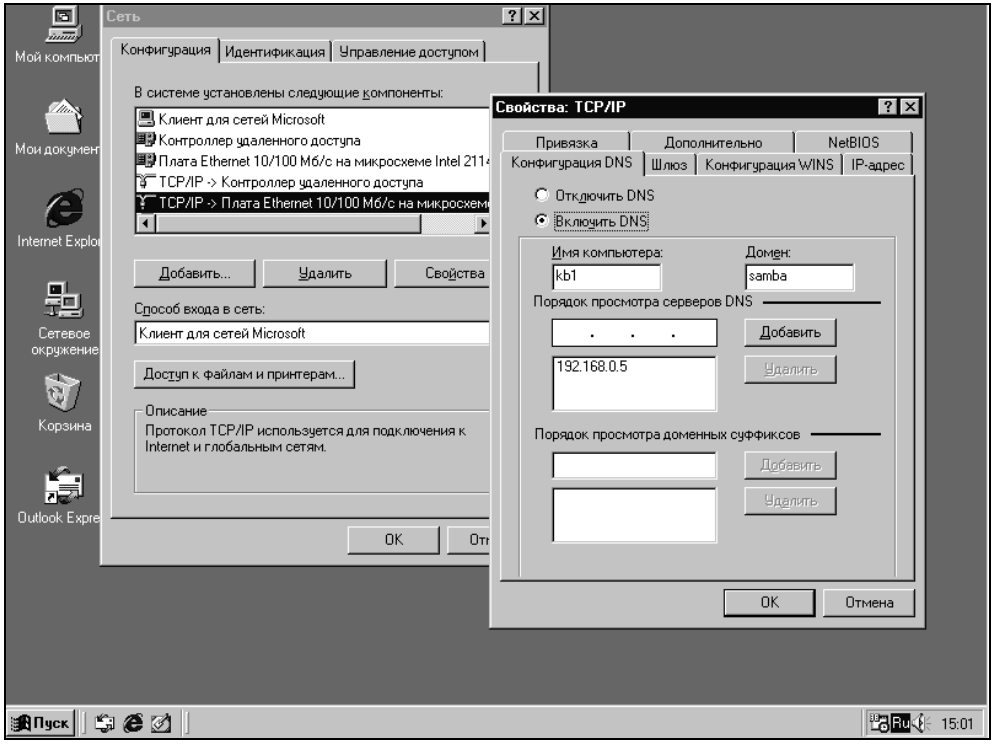



**Рис. 11.9.** Настройка DNS для Windows 98

# 11.11. Настройка клиентской части Windows 2000

Настройка в Windows 2000 ничуть не сложнее.

1. Последовательно выбираем: **Пуск | Настройка | Сеть** и удаленный доступ к сети.
2. На значке **Подключение по локальной сети** нажимаем правую кнопку мыши и выбираем пункт **Свойства**.

В окне **Подключение по локальной сети — Свойства** выбираем строку **Протокол TCP/IP**.

Нажимаем кнопу **Свойства**.

В открывшемся окне свойств протокола TCP/IP выставляем адрес предпочитаемого DNS-сервера (рис. 11.10).

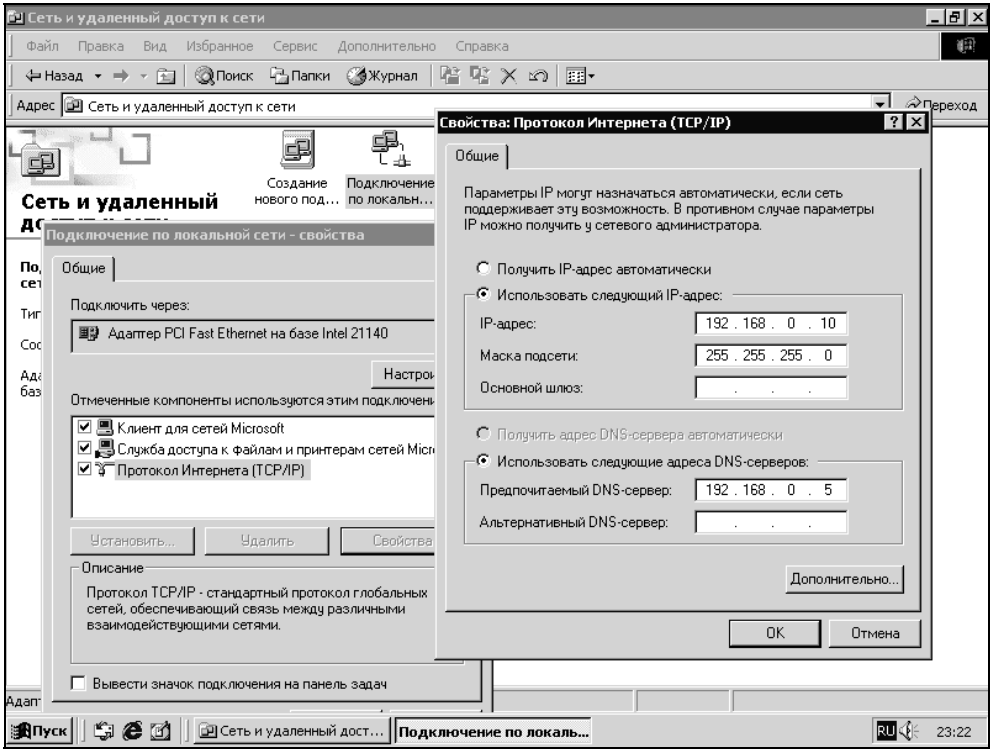



**Рис. 11.10.** Настройка DNS в Windows 2000

# Глава 12

# Запуск Apache и Webmin

## 12.1. Почему Apache

Стремительное развитие Интернета привело к тому, что сегодня деятельность любого предприятия немыслима без использования технологий глобальной сети. Одна из самых известных и широко используемых технологий — это служба WWW (World Wide Web — всемирная паутина). В основе ее работы лежит протокол HTTP (Hyper Text Transfer Protocol — протокол передачи гипертекста). Это протокол семейства TCP/IP. Протокол передачи гипертекста позволяет передавать связанные между собой документы сложной структуры. В состав документа могут входить изображения, анимация, звуковые фрагменты и др.

HTTP — это клиент-серверный протокол. Нажав кнопкой мыши в том или ином месте экрана, вы посылаете запросы удаленным серверам, которые отвечают, пересылая информацию. Согласно стандарту, протокол HTTP "слушает" порт 80. На стороне пользователя запрос формирует браузер, специальная программа, предназначенная для взаимодействия с web-серверами и преобразования информации к должному виду. Наиболее популярным HTTP-сервером для пользователей Linux является, конечно, сервер Apache (около 50 % от общего числа HTTP-серверов). Он входит в состав большинства дистрибутивов Linux. Характерной особенностью сервера является простота его установки и настройки, а также хорошие показатели производительности и устойчивости (последнее во многом зависит от администратора). Полезным является и то, что при определенной настройке Apache способен создавать динамические страницы, то есть страницы, которые формируются на лету, в зависимости от запросов пользователя.

Apache относится к свободно распространяемому программному обеспечению, и при необходимости свежую версию можно скачать по адресу

**www.apache.org** (сайт некоммерческой организации Apache Software Foundation, целью которой является поддержка и обновление Apache).

Возвращаясь к тематике нашей книги, Apache используется в качестве Web-сервера, который можно организовать в локальной сети, чтобы своевременно выкладывать на общее обозрение документы, печатные формы, приказы и т. д. При помощи Apache также возможна организация корпоративной базы данных со стандартизованным интерфейсом доступа, причем этот подход становится достаточно популярным.

## 12.2. Основы конфигурирования Apache

Конфигурирование для Apache представляет собой редактирование файлов в каталоге /etc/httpd/conf/ или /usr/local/apache/conf. Файлов конфигурации несколько. В табл. 12.1 мы приведем их в последовательности обработки Web-сервером.

***Таблица 12.1.*** *Назначение конфигурационных файлов Apache*

| Наименование | Комментарий |
|---|---|
| httpd.conf | Основной файл конфигурации Web-сервера |
| srm.conf | Сохранен для совместимости с предыдущими версиями, использовать не рекомендуется. Ранее здесь размещались директивы обработки запросов и дерево документов Apache |
| access.conf | Ранее здесь хранились сведения о правах пользователей на доступ к каталогу, использовать не рекомендуется |
| mime.types | Определяет mime для различных типов файлов |

Изменение конфигурационных файлов может быть осуществлено либо путем непосредственного редактирования файлов, либо при помощи утилиты упрощенного администрирования Webmin.

## 12.3. Базовые параметры, используемые при настройке Apache

Директивы Apache представлены в табл. 12.2.

***Таблица 12.2.*** *Назначение директив Apache*

| Наименование | Комментарий |
|---|---|
| ServerRoot | Начальный каталог сервера, здесь расположен файл демона httpd |
| BindAddress | IP-адрес сервера, принимает следующие значения:<br>* — отвечать на все адреса;<br>123.123.123.123 — отвечать только на этот адрес;<br>name.domen — отвечать на IP-адрес имени |
| Port | По умолчанию 80. Поскольку это стандартное значение, то его лучше не изменять. Не стоит думать, что нестандартный номер может служить защитой от взлома. Сканеры портов достаточно быстро находят все открытые порты |
| ServerName | Задает имя сервера, должно совпадать с доменным именем компьютера, поскольку используется в обмене данными между клиентом и сервером |
| ServerAdmin | Электронный адрес администратора сервера, выдаваемый пользователям |
| User, Group | Определяют пользователя и группу, от имени которой будет работать сервер. При настройке или отладке сервера целесообразнее запускать его с привилегиями root, после отладки серверу назначается пользователь и группа с ограниченными правами. Лучше для этих целей создавать отдельного пользователя или группу |
| Maxclients | Определяет максимальное число клиентов, подключенных к серверу. Необходимо помнить, что один сеанс, открытый пользователем на HTTP-сервере, отнимает у него порядка 20 Мбайт оперативной памяти. Для серверов использование параметра обязательно |
| KeepActive | Поддержка постоянного соединения с клиентом. Позволяет сократить трафик за счет экономии на открытии и закрытии соединения, однако снижает безопасность сервера |
| ServerAdmin | Адрес электронной почты администратора сервера. Например, apache@mail.samba |
| DocumentRoot | Путь к главному дереву каталогов сервера, открытому для пользователей сервера. При открытии пользователям доступа к файлам надо быть внимательным и проверить, чтобы в открытых каталогах не было конфиденциальной информации и ссылок на другие закрытые каталоги |

***Таблица 12.2*** *(окончание)*

| Наименование | Комментарий |
|---|---|
| UserDir | Путь к каталогам отдельных пользователей |
| DirectoryIndex | Указание серверу, какой файл использовать в качестве индексного (с этого файла начинается просмотр директории) файла. Как правило, это index.html или index.htm |

# 12.4. Коды ошибок, выдаваемых сервером

Несомненную помощь окажет анализ кодов ошибок, приведенных в табл. 12.3.

***Таблица 12.3.*** *Коды ошибок выдаваемых сервером*

| Код ошибки | Содержание ошибки |
|---|---|
| 401 | Отказ в доступе к указанному документу. Появляется форма для ввода имени пользователя и пароля для доступа к документу |
| 403 | Доступ к документу или каталогу запрещен |
| 404 | Пожалуй, с этой ошибкой сталкивались все. Код ошибки возникает тогда, когда документ не найден |
| 405 | Эта ошибка возникает, когда запрос отправлен на форму, не являющуюся сценарием |
| 500 | Ошибка выполнения скрипта на сервере. Причина может быть в том, что не установлены библиотеки, которые используются в скрипте |
| 502 | Отказ обработки запроса клиента из-за большой нагрузки сервера |

# 12.5. Регистрация ошибок сервера

Через журналы регистрации можно получать самую разную информацию. Наиболее интересен журнал регистрации ошибок. Согласно железной логике разработчиков, в этом журнале регистрируются ошибки, возникшие при работе сервера.

В ASPLinux 7.3 Server ошибки прописываются в файл /var/log/httpd/error_log. Если вы хотите перенаправить функцию обработки ошибок программе syslog, необходимо задать

```
ErrorLog syslog
```

По умолчанию директива loglevel управляет уровнем регистрации ошибок, для ASPLinux 7.3 Server установлено значение warn. Однако возможны и другие значения (табл. 12.4).

***Таблица 12.4.*** *Возможные уровни регистрации ошибок*

| Директива | Описание |
|---|---|
| emerg | Авария, система прекращает работу |
| alert | Критическая ситуация. Необходимо срочное вмешательство в работу системы |
| crit | Критическая ситуация |
| error | Стандартная ошибка |
| warn | Предупреждение |
| notice | Стандартное событие, заслуживающее внимания |
| info | Информационное сообщение |

При стандартной строке для ASPLinux 7.3 сообщение об ошибках имеют следующий формат (рис. 12.1):

```
[Mon Mar 28 05:22:28 2005] [error] [client 192.168.0.1] File does not exist: /var/www/html/43
[Mon Mar 28 05:22:34 2005] [error] [client 192.168.0.1] File does not exist: /var/www/html/manual/index.html
[Mon Mar 28 05:22:51 2005] [error] [client 192.168.0.1] File does not exist: /var/www/html/manual/mod/core.html
```

**Рис. 12.1.** Примерное содержание файла ошибок

Структура файла ошибок следующая: первый столбец содержит дату и время возникновения ошибки, второй — характер записи, третий столбец указывает источник возникновения ошибки, а четвертый дает развернутую информацию по ошибке.

В конфигурационном файле также определены форматы для регистрации различных событий:

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U"  eferrer
LogFormat "%{User-agent}i" agent
```

Если вы хотите использовать предложенный формат, то необходимо в конце строки указать, какой именно:

```
CustomLog  /var/log/httpd/custom_log combined
```

Понять структуру журнала регистрации поможет табл. 12.5.

***Таблица 12.5.*** *Типовые структуры, используемые в строке Logformat*

| Символ | Комментарий |
|---|---|
| %h | IP-адрес узла от которого поступил запрос. |
| %u | Имя пользователя. Имеет смысл, если пользователь входил в систему. |
| %t | Время происхождения события. Формат времени определяется системой |
| %f | Имя файла |
| %p | Порт сервера, к которому шло обращение |
| %U | Запрашиваемый клиентом путь (URL) |
| %r | Строка запроса клиента |
| %s | Код состояния запроса |

Это лишь некоторые идентификаторы, но и они должны сильно облегчить вам жизнь.

## 12.6. Настройка автоматического запуска Apache

Запуск Apache можно настроить либо вручную, либо при помощи утилиты Webmin. Начнем с ручного метода.

В каталоге /etc/rc.d/rc3.d переименовываем ссылку K15httpd в S85httpd.

В каталоге /etc/rc.d/init.d исправляем заголовок файла httpd на следующий:

```
#!/bin/bash
# Startup script for the Apache Web Server
#
# chkconfig: 3 85 15
```

В данном случае строка # chkconfig: 3 85 15 говорит о третьем уровне, очередь запуска 85, удаления — 15.

**Внимание!**

Второй и третий параметры # chkconfig должны давать в сумме 100.

Для варианта с Webmin производятся следующие действия (рис. 12.2).

1. Запускаем браузер mozilla (с консоли сервера) либо Internet Explorer (с рабочей станции).

2. Запускаем уже знакомым образом Webmin.
3. Выбираем вкладку **System**, пункт **Bootup and Shutdown** (Загрузка и завершение).
4. Находим строчку **httpd** и щелкаем по ней.
5. В открывшемся окне выбираем **Start at boot time** (Запускать во время загрузки).

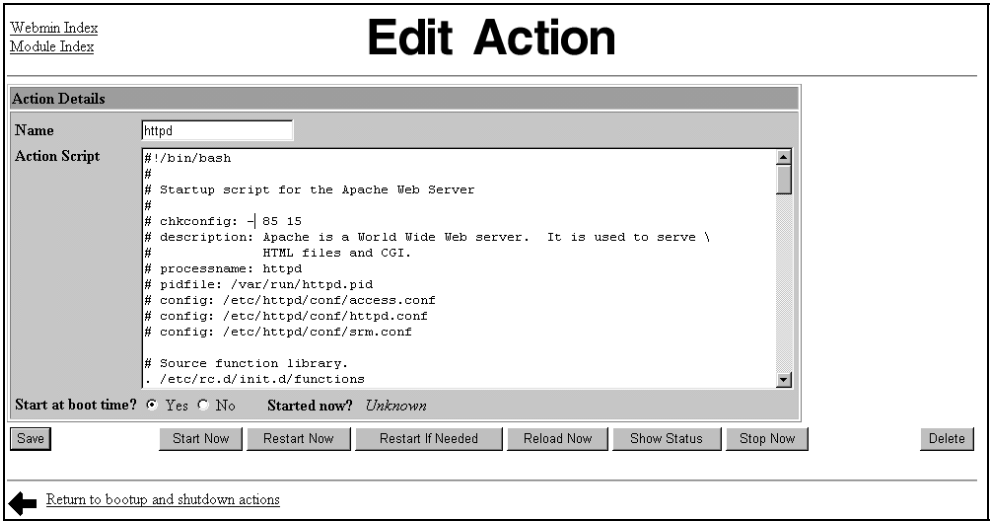



**Рис. 12.2.** Настройка автоматического запуска Apache при помощи Webmin

Нажимаем **Save** и сохраняем сделанные изменения.

Как и ранее, автоматическая настройка не удалит файл, убивающий сервис, а просто добавит новый, который будет запускать его девяносто девятым по очереди. Так что лучше используйте ручное редактирование файлов запуска.

## 12.7. Самый простой способ организовать Web-сервер в организации

При помощи Web-сервера можно автоматизировать делопроизводство в организации. Возможно, что со временем вы сможете организовать корпоративную базу данных с web-интерфейсом. Apache будет выступать в качестве программы, формирующей интерфейс доступа к услугам СУБД. Стоит заметить, что такой подход становится достаточно популярным.

Организация информационного сервера на базе Apache состоит из четырех пунктов:

- настройка Apache;
- настройка клиентов (папка Избранное);
- создание наполнения информационного сервера;
- обновление сервера.

## Настройка Apache

Подойдем к этому вопросу с позиций максимальной простоты, будем делать минимум изменений.

- `ServerName` — имя сервера, должно совпадать с доменным именем компьютера:

  ```
  ServerName smb.samba
  ```

- Maxclients указывает максимальное число клиентов при одновременном обращении. Расчет этого числа ведется, исходя из объема оперативной памяти и того факта, что каждый клиент Apache в среднем требует 20 Мбайт оперативной памяти. Для оперативной памяти 512 Мбайт целесообразно выставить ограничение до 10 пользователей. Для 1024 Мбайт — на уровне 25 пользователей. Подходить к настройке этого параметра надо гибко. В зависимости от целесообразности использования Apache и загрузки сервера решайте, увеличивать ли число пользователей или уменьшать. Если сервис стал необходимым в работе организации, можно подумать о выделении отдельного физического сервера под Apache.

- `DocumentRoot` указывает путь к информационному порталу. Файлы здесь открыты для большинства пользователей, проверьте лишь, чтобы в открытых каталогах не было конфиденциальной информации, а также символических ссылок на другие закрытые каталоги. Оставляем его без изменений:

  ```
  DocumentRoot "/var/www/html"
  ```

- `DirectoryIndex` указывает, какой файл использовать в качестве индексного при входе в каталог. Для повышения безопасности сервера этот файл должен быть в каждом каталоге. Если этого не сделать, Apache покажет содержимое каталога, в который вы обратились. Приведем варианты описания индексного файла:

  ```
  DirectoryIndex index.html index.htm index.shtml index.php index.php4
  index.php3 index.phtml index.cgi
  ```

- `BindAddress` — IP-адрес сервера, на который он будет откликаться. В нашем случае это 192.168.0.5, а вы задаете свой адрес:

  ```
  BindAddress 192.168.0.0.5
  ```

- Port — порт, оставляем по умолчанию:

```
Port 80
```

## Настройка клиентов

Настройка клиентов не столько необходимая операция, сколько способ облегчить жизнь пользователям. По сути, она сводится к добавлению ссылок в папку Избранное клиента. Для того чтобы добавить ссылку, делаем следующее:

1. Запускаем Internet Explorer.
2. Набираем в адресной строке нужный адрес и нажимаем <Enter> (рис. 12.3).
3. Нажимаем **Избранное / Добавить в избранное**.
4. В открывшемся окне вводим имя ссылки и **OK**.

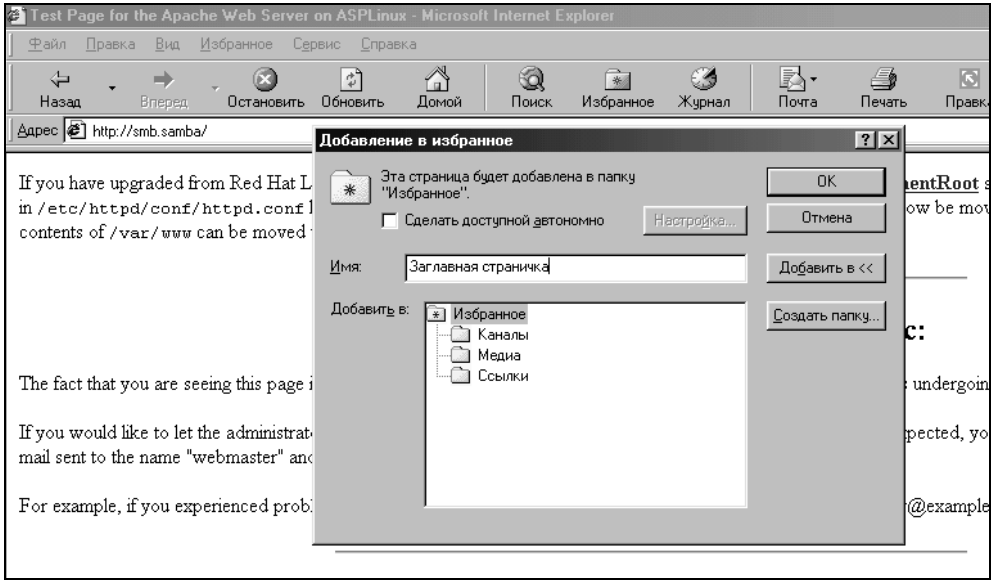



**Рис. 12.3.** Добавление в избранное (фрагмент экрана)

Уже введенную закладку можно отредактировать. Для этого:

1. Выберите ее и нажмите правую кнопку мыши.
2. Выберите пункт **Свойства**.
3. И на вкладке **Документ Интернета** отредактируйте адрес URL, например: **http://smb.samba:8000**. Вы должны поставить нужный адрес и порт, если изменили настройки по умолчанию. В примере на рис. 12.4 предполагается, что порт изменен на 8000.

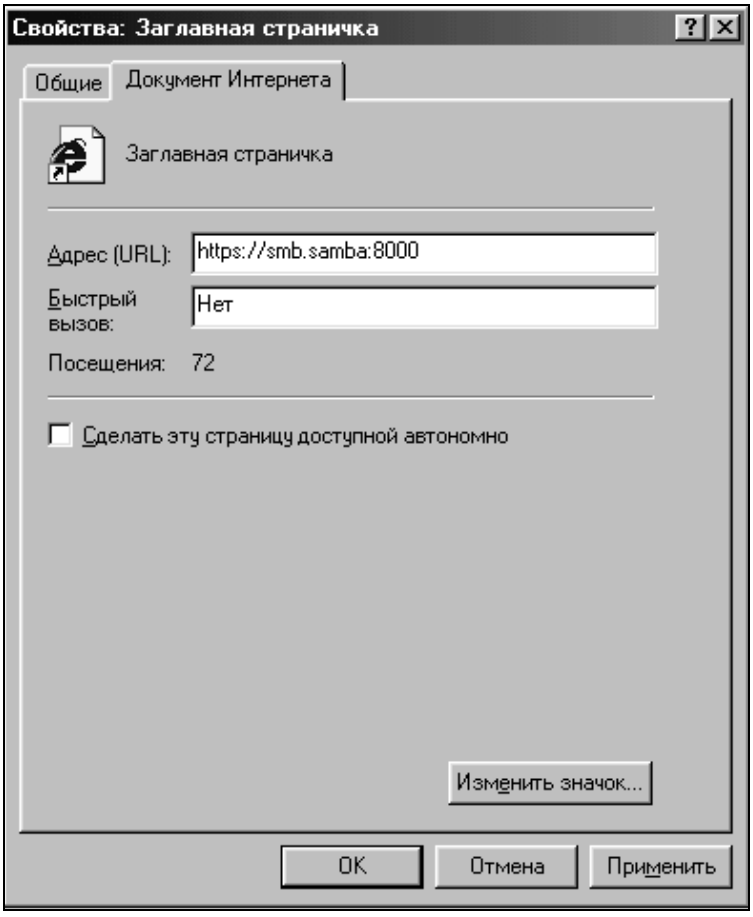



**Рис. 12.4.** Редактирование папки Избранное

## Создание информационного наполнения сервера

Основная информация, которая будет размещена на сервере, связана при помощи гипертекста. Необходимо сформировать Html-страницы и ссылки на ресурсы (документы, файлы архивов, мультимедийные файлы) при помощи Html-редакторов. Это могут быть FrontPage от Microsoft, Dream Weaver от Macromedia или любой другой. На рис. 12.5 представлено окно редактора Dream Weaver.

Обновление информации на сервере состоит из добавления новых Html-файлов или к замене старых. Производиться оно может в любом файловом менеджере, с управлением доступа для записи только ограниченному кругу лиц, непосредственно связанных с заполнением.

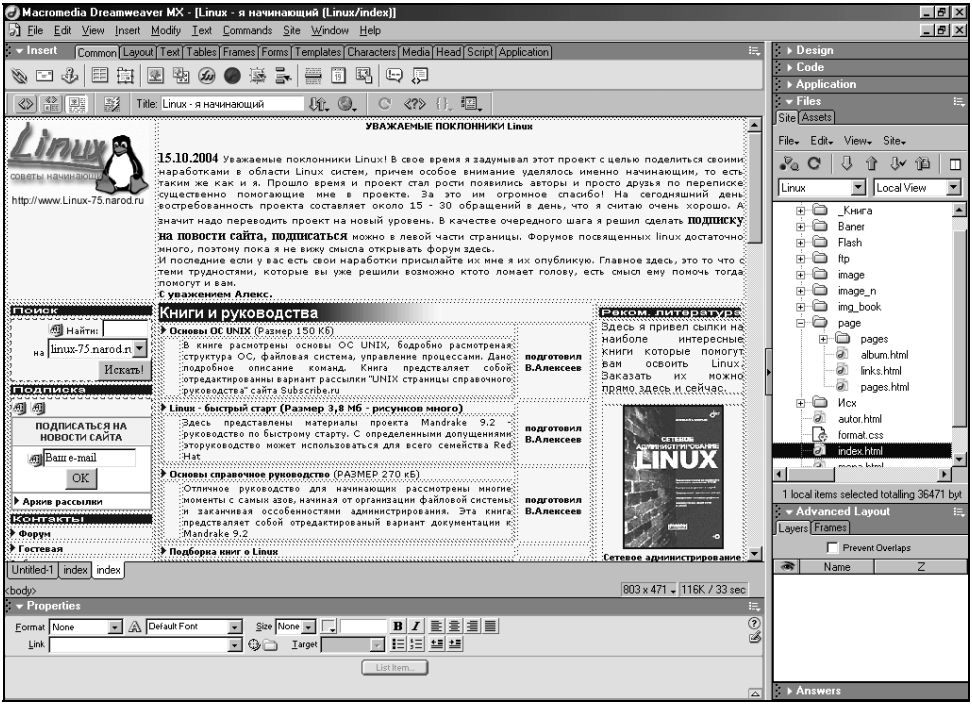



**Рис. 12.5.** Окно редактора Dream Weaver

# 12.8. Краткие сведения о Webmin и запуск

Помимо Webmin, существует еще целый ряд программ графического администрирования, встроенных в оболочки (например KDE) и как самостоятельные продукты (например Linuxconf). Однако, самым популярным и удобным инструментом администрирования остается Webmin.

Утилита Webmin уже входит в пакет ASPLinux 7.3 Server Edition и при выборе типа установки **Сервер рабочей группы** запускается автоматически. Проверить, запустилась ли утилита, можно при помощи команды

```
netstat -ln.
```

Webmin по умолчанию занимает порт 10 000. В примере на рис. 12.6 видно, что строка с портом 10 000 присутствует — Webmin запущен. Если такая строка отсутствует, то сделать запуск можно командой:

```
# /etc/rc.d/init.d/webmin start
```

Теперь можно обращаться к Webmin:

```
https://localhost:10000
```

Префикс `https` показывает, что это защищенное SSL-соединение.

```
[root@localhost root]# netstat -ln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:901             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:139             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:6000            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:10000           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:3000          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:443             0.0.0.0:*               LISTEN
udp        0      0 192.168.0.5:137         0.0.0.0:*
udp        0      0 0.0.0.0:137             0.0.0.0:*
udp        0      0 192.168.0.5:138         0.0.0.0:*
udp        0      0 0.0.0.0:138             0.0.0.0:*
udp        0      0 0.0.0.0:10000           0.0.0.0:*
udp        0      0 0.0.0.0:475             0.0.0.0:*
udp        0      0 0.0.0.0:111             0.0.0.0:*
Active UNIX domain sockets (only servers)
Proto RefCnt Flags       Type       State         I-Node Path
unix  2      [ ACC ]     STREAM     LISTENING     1649   /tmp/.font-unix/fs7100
unix  2      [ ACC ]     STREAM     LISTENING     1571   /dev/gpmctl
unix  2      [ ACC ]     STREAM     LISTENING     17402  /tmp/.X11-unix/X0
unix  2      [ ACC ]     STREAM     LISTENING     17445  /tmp/.ICE-unix/4612
unix  2      [ ACC ]     STREAM     LISTENING     17595  /tmp/orbit-root/orb-19410392841961743142
unix  2      [ ACC ]     STREAM     LISTENING     17640  /tmp/orbit-root/orb-1198106839745869374
unix  2      [ ACC ]     STREAM     LISTENING     17576  /tmp/orbit-root/orb-631328104497381612
unix  2      [ ACC ]     STREAM     LISTENING     17647  /tmp/orbit-root/orb-8350971821593104740
unix  2      [ ACC ]     STREAM     LISTENING     17683  /tmp/orbit-root/orb-20802142961513464974
unix  2      [ ACC ]     STREAM     LISTENING     17498  /tmp/.sawfish-root/localhost.localdomain:0.0
[root@localhost root]#
```

**Рис. 12.6.** Проверка состояния Webmin

Если в поставке отсутствует Webmin, (например ASPLinux 7.1), то скачать дистрибутив можно по адресу **http://www.webmin.com**. Для rpm-пакета установка осуществляется следующим образом:

```
# rpm -Uvh webmin-X.XXX.noarch.rpm
```

Команда `U` позволит провести корректную установку, если уже был установлен Webmin более ранней версии.

При установке, помимо всего прочего, будет создан скрипт запуска /etc/rc.d/init.d/webmin (листинг 12.1).

**Листинг 12.1. Скрипт запуска /etc/rc.d/init.d/webmin**

```
#!/bin/sh
# description: Start/stop Webmin
# chkconfig: 2345 99 01
case "$1" in
'start')
        /etc/webmin/start >/dev/null 2>&1 </dev/null
        RETVAL=$?
        if [ "$RETVAL" = "0" ]; then
```

```
                touch /var/lock/subsys/webmin
        fi
        ;;
'stop')
        /etc/webmin/stop
        RETVAL=$?
        if [ "$RETVAL" = "0" ]; then
                rm -f /var/lock/subsys/webmin
        fi
        ;;
'status')
        pidfile=`grep "^pidfile=" /etc/webmin/miniserv.conf | sed -e
's/pidfile=//g'`
        if [ -s $pidfile ]; then
                pid=`cat $pidfile`
                kill -0 $pid >/dev/null 2>&1
                if [ "$?" = "0" ]; then
                        echo "webmin (pid $pid) is running"
                        RETVAL=0
                else
                        echo "webmin is stopped"
                        RETVAL=1
                fi
        else
                echo "webmin is stopped"
                RETVAL=1
        fi
        ;;
'restart')
        /etc/webmin/stop && /etc/webmin/start
        RETVAL=$?
        ;;
*)
        echo "Usage: $0 { start | stop }"
        RETVAL=1
        ;;
esac
exit $RETVAL
```

Для настройки автоматического запуск утилиты Webmin необходимо для соответствующего уровня работы создать символическую ссылку на сценарий запуска Webmin:

```
# ln -s /etc/rc.d/init.d/webmin /etc/rc.d/rc3.d/S99webmin
```

можно настроить запуск Webmin и в графическом режиме. Для этого необходимо запустить Webmin из командной строки и точно так же, как мы делали с запуском Apache, настроить автоматический запуск Webmin.

## 12.9. Настройка Webmin

Используя дистрибутив ASPLinux 7.3 Server Edition, можно использовать русский язык, внешний вид, отличный от классического, и другие вкусности.

Все настройки и управление Webmin осуществляется на вкладке Webmin (рис. 12.7).

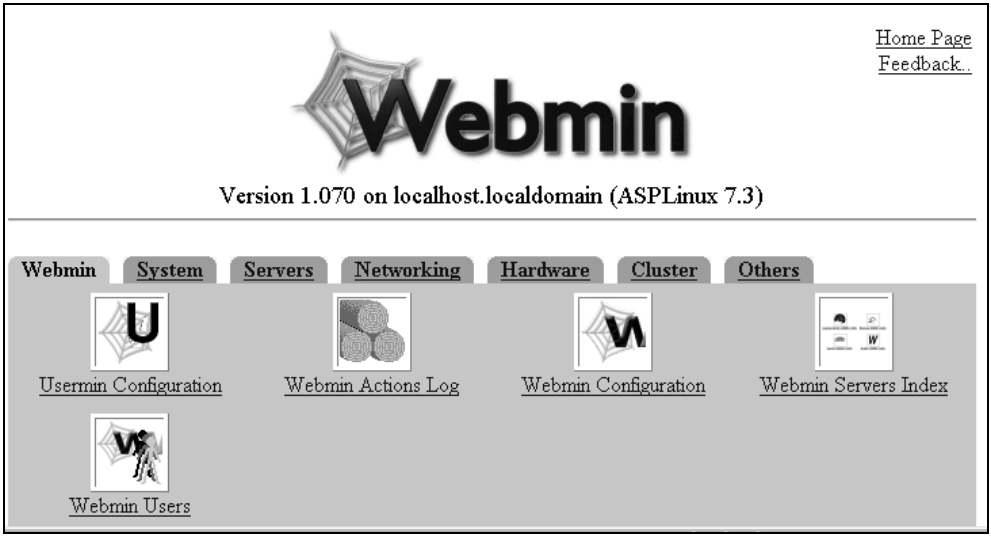



**Рис. 12.7.** Вкладка Webmin

Для настройки языка последовательно выбираем **Webmin configuration**, далее **Language** и в выпадающем списке **Russian**. Нажимаем **Change language** (рис. 12.8).

Ну вот, теперь общаться будет немного попроще.

Еще одной приятной возможностью является возможность настройки внешнего вида. Для этого в **Webmin Configuration** выберем пункт **Webmin Theme**. Появится окно, представленное на рис. 12.9.

Webmin Index
Module Index

# Language

This page allows you to choose which language Webmin will use for displaying titles, prompts and messages

**Webmin Language**

**Display in language** English (EN)

**Use language specified by browser?**

Italian (IT)
Japanese (JA_JP.EUC)
Korean (KO_KR.EUC)
Norwegian (NO)
Polish (PL)
Portuguese (PT)
Portuguese (Brazilian) (PT_BR)
Russian CP1251 (RU_RU)
Russian KOI8 (RU_SU)
Simplified Chinese (ZH_CN)
Slovak (SK)

Change Language

Return to webmin configuration

**Рис. 12.8.** Смена языка

Webmin Themes - Microsoft Internet Explorer

Файл Правка Вид Избранное Сервис Справка

Назад Вперед Остановить Обновить Домой Поиск Избранное Журнал Почта Печать Правка

Адрес https://192.168.0.5:10000/webmin/edit_themes.cgi Переход Ссылки

Webmin Index
Module Index

# Webmin Themes

Themes control the appearance of the Webmin user interface, including icons, colours, backgrounds and possibly the layout of pages. The selection box below can be used to choose one of the themes installed on your system.

**Current theme :** Old Webmin theme Change

Old Webmin theme
ASPLinux Management Console
Caldera Theme
MSC.Linux Theme

Use the form below to install new themes on your system. Themes are typically distributed in .wbt files, but can also be installed from RPM files if supported by your operating system.

From local file ...
From uploaded file Обзор...
From ftp or http URL
Install Theme

This form can be used to delete one of the themes installed on your system that is not currently in use.

**Theme to delete :** ASPLinux Management Console Delete

root logged into Webmin 1.070 on localhost.localdomain (ASPLinux 7.3) Интернет

**Рис. 12.9.** Смена темы в Webmin

По умолчанию доступны четыре темы:

- Old Webmin theme;
- ASPLinux Management Console;
- Caldra Theme;
- MSC Linux Theme.

Темы отличаются в цветовом и оформительском решениях. Выбор любой из них за вами. Автору более приглянулась классическая тема.

В открывшемся окне можно подключить новые темы (с расширением wbt) или удалить лишние.

Если соединение SSL еще не используется Webmin, то в адресной строке надо набирать

`http://localhost:10000`,

без буквы s, указывающей, что соединение SSL.

По умолчанию в ASPLinux 7.3 Server Edition утилита Webmin уже использует защищенное SSL-соединение, однако в некоторых дистрибутивах этого может не быть. Для настройки необходимо проверить, какой порт оно использует. Если используется порт ниже 1024 (такое может быть в некоторых старых дистрибутивах), то необходимо изменить его на 10 000 или любой другой, с номером выше 1024. Делается это через **Webmin configuration** редактированием **Port and Adress**. После этого выбираем **SSL Encryption**, далее выбираем **Enable SSL in aviable** и сохраняем изменения (рис. 12.10).

Для того чтобы была возможность использовать SSL-соединение, модули SSL должны быть уже установлены (рис. 12.11).

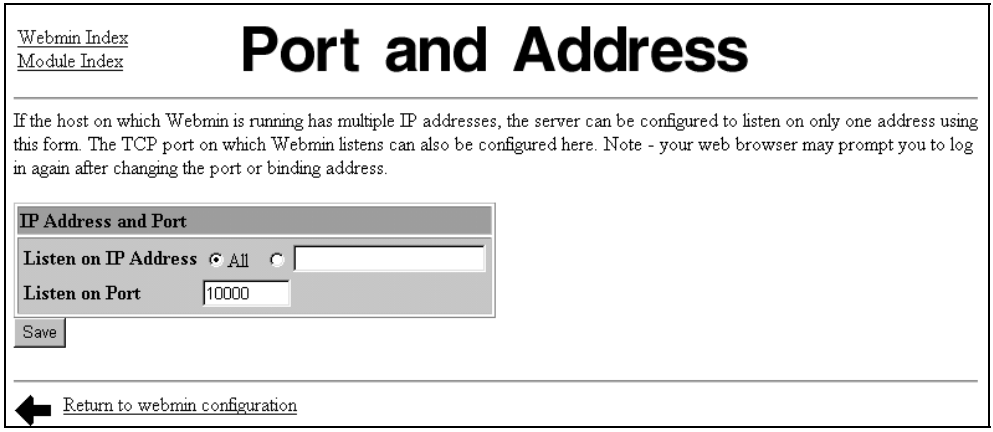



**Рис. 12.10.** Установка номера порта

Webmin Index
Module Index

# SSL Encryption

The host on which Webmin is running appears to have the SSLeay Perl module installed. Using this, Webmin supports SSL encrypted communication between your browser and the server. If you are accessing your Webmin server over the Internet, then you should definately consider using SSL to prevent an attacker capturing your Webmin password.

Warning - only turn on SSL support if you have a browser that supports SSL (such as Netscape or IE), and there is no firewall blocking **https** requests between your browser and the Webmin host

**SSL Support**

| | |
|---|---|
| **Enable SSL if available?** | Yes No |
| **Private key file** | /etc/webmin/miniserv.pem |
| **Certificate file** | Same file as private key |

Save

This form can be used to create a new SSL key for your Webmin server.

**Create SSL key**

| | |
|---|---|
| **Server name in URL** | Any hostname 192.168.0.5 |
| **Email address** | webmin@localhost.localdomain |
| **Department** | |
| **Organisation** | Webmin Webserver |
| **State** | |
| **Country code** | |
| **Write key to file** | /etc/webmin/miniserv.pem |
| **Use new key immediately?** | Yes No |

Create Now

**Рис. 12.11.** Настройка SSL-соединения

При необходимости здесь можно создать ключ для сервера SSL (рис. 12.11).

# Глава 13

# Антивирусная защита

Несмотря на то, что эта глава располагается в конце книги и занимает относительно небольшой объем, вопрос антивирусной защиты очень важен. В настоящее время количество известных вирусов и их модификаций насчитывает несколько сотен тысяч. Некоторые совершают серьезные деструктивные действия, вплоть до выхода из строя электронных компонентов компьютера, некоторые относительно безвредны.

Вирусы пишутся с самыми различными целями, начиная от хулиганских побуждений и заканчивая целенаправленной деятельностью по воровству информации.

Пути проникновения вирусов самые различные: Интернет, дискеты, модули Flash-памяти. В общем, путь проникновения может быть любой. Все, что содержит информацию, может содержать вирус. В общем, полностью изолироваться от внешнего мира нельзя, а значит, нельзя избавиться от вирусов.

Раз нельзя избавиться, с ними надо бороться. Выбор антивирусного пакета полностью определяется пользователем. В данной книге рассмотрена установка антивируса Dr.Web. Здесь мы рассмотрим, как установить демон drwebd и как наладить его связь с Samba для проверки файлов, размещенных на Samba, в автоматическом режиме. Выкладки, представленные здесь, представляют собой адаптированные рекомендации, имеющиеся в технической документации.

Прежде всего, надо скачать самую свежую версию антивирусного обеспечения. Скачать можно по адресу **http://www.dialognauka.ru** в разделе загрузки. Здесь же на сайте можно скачать полную техническую документацию по установке и настройке.

Демон drwebd представляет собой усовершенствованный антивирусный сканер. Как и любой другой демон, он постоянно находится в памяти и готов к выполнению работы.

Существует два варианта демона:

- В виде rpm-пакета — ориентирован он на системы Linux, использующие rpm в качестве системы управления пакетами (для нас этот вариант предпочтительнее).
- В виде tar.gz — не привязан ни к какому конкретному дистрибутиву, однако потребует ручной установки, хотя и в этом ничего сложного нет.

## 13.1. Установка drwebd и настройка скрипта запуска

Установка из пакетов rpm проще в том плане, что идет в автоматическом режиме. Однако могут возникнуть проблемы со связанностью пакетов. Чтобы избежать таких проблем, надо знать, какой у вас дистрибутив и выбирать rpm-пакеты только для него. В нашем случае rpm-пакеты собраны для Red Hat 7.X.

Исследуем установку из rpm-пакета.

Входим в систему с правами root.

Входим в каталог, где хранятся пакеты с демоном антивируса.

Набираем в командной строке:

```
# rpm -ihv drweb-4.32.2-rh10.i586.rpm
```

В результате пакет установится в каталог /opt/drweb, конфигурационный файл — в каталог /etc/drweb.

Стоит обратить внимание только на параметр **User**, который задает пользователя для работы демона. Лучше создать отдельного пользователя drweb и назначить ему права в контексте решаемых задач.

```
# useradd drweb
```

Крайне нежелательно запускать drweb с правами root. Хотя подобный запуск упростит настройки, но при этом снизится безопасность системы.

Также стоит обратить внимание на параметр **Key**. Он определяет путь к файлу ключа антивируса. Если путь указан неверно или ключ отсутствует, или истек срок лицензии, то запустить демона не получится (рис. 13.1).

```
[root@localhost root]# /etc/rc.d/init.d/drwebd start
Starting up drwebd: Dr.Web (R) daemon for Linux, version 4.32.2 (2004-11-01)
Copyright (c) Igor Daniloff, 1992-2004
Doctor Web Ltd., Moscow, Russia
Support service: http://support.drweb.com
To purchase: http://buy.drweb.com

Engine version: 4.32b
Loading /var/drweb/bases/drwebase.vdb - Ok, virus records: 51982
Loading /var/drweb/bases/drw43201.vdb - Ok, virus records: 364
Total virus records: 52346
Key file: /opt/drweb/drweb32.key
Key file number: 0010000000
Your license key file has expired!
                                                          [ СБОЙ ]
[root@localhost root]#
```

**Рис. 13.1.** Лицензия просрочена

Теперь надо каталогу, где будет расположен pid, назначить владельца, от имени которого будет работать Samba (в нашем случае /var/drweb/run). Проще это будет сделать в mc. Также необходимо разрешить запись и чтение пользователю drweb на все общие каталоги (рис. 13.2).

```
[root@localhost root]# /etc/rc.d/init.d/drwebd start
Starting up drwebd:                                       [  OK  ]
[root@localhost root]#
```

**Рис. 13.2.** А вот так все нормально

После проделанных процедур на компьютере будет установлен демон drwebd. Теперь необходимо настроить его запуск. Проверьте наличие файла /etc/rc.d/init.d/drwebd и его содержание. Этот файл служит для запуска в автоматическом режиме.

Затем настроим на уровне работы по умолчанию автоматический запуск демона drwebd. Для этого в каталоге /etc/rc.d/rc3.d необходимо создать символическую ссылку на скрипт запуска демона, делается это командой

```
# ln -s /etc/rc.d/init.d/drwebd /etc/rc.d/rc3.r/S34drwebd
```

## 13.2. Проверка работоспособности drwebd

Для коммуникации демон использует socket. Параметры его задаются в файле /etc/drweb/drweb32.ini в строке:

```
Socket= {…..},
```

по умолчанию используется порт 3000. Поэтому для проверки достаточно проверить открытые порты:

```
# netstat -ln
```

```
[root@localhost root]# netstat -ln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:901             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:139             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:6000            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:10000           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:3000          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:443             0.0.0.0:*               LISTEN
udp        0      0 127.0.0.1:1026          0.0.0.0:*
udp        0      0 192.168.0.5:137         0.0.0.0:*
udp        0      0 0.0.0.0:137             0.0.0.0:*
udp        0      0 192.168.0.5:138         0.0.0.0:*
udp        0      0 0.0.0.0:138             0.0.0.0:*
udp        0      0 0.0.0.0:10000           0.0.0.0:*
udp        0      0 0.0.0.0:475             0.0.0.0:*
udp        0      0 0.0.0.0:111             0.0.0.0:*
Active UNIX domain sockets (only servers)
Proto RefCnt Flags       Type       State         I-Node Path
unix  2      [ ACC ]     STREAM     LISTENING     18837  /tmp/.ICE-unix/4572
unix  2      [ ACC ]     STREAM     LISTENING     1467   /tmp/.font-unix/fs7100
unix  2      [ ACC ]     STREAM     LISTENING     1392   /dev/gpmctl
unix  2      [ ACC ]     STREAM     LISTENING     18797  /tmp/.X11-unix/X0
unix  2      [ ACC ]     STREAM     LISTENING     18973  /tmp/orbit-root/orb-16644854921598301628
unix  2      [ ACC ]     STREAM     LISTENING     18986  /tmp/orbit-root/orb-4030679271295225518
unix  2      [ ACC ]     STREAM     LISTENING     19028  /tmp/orbit-root/orb-13004997142074404271
unix  2      [ ACC ]     STREAM     LISTENING     19048  /tmp/orbit-root/orb-2433785917538544 94
unix  2      [ ACC ]     STREAM     LISTENING     19073  /tmp/orbit-root/orb-18076265711154235399
unix  2      [ ACC ]     STREAM     LISTENING     18886  /tmp/.sawfish-root/localhost.localdomain:0.0
[root@localhost root]#
```

**Рис. 13.3.** Проверка открытия портов

Как видим на рис. 13.3, порт 3000 открыт, значит, все нормально.

# 13.3. Установка Samba Spider

Наличие демона, проверяющего вирусы, само по себе мало что дает нам. Необходимо настроить его так, чтобы он проверял файлы на Samba. Для этого предназначен Samba Spider — монитор файловых операций, работающий совместно с Samba. Он является клиентом для Dr.Web Daemon.

Для этого потребуется:

- файл drweb-samba-4.32.2-rh7.i586.rpm;
- запущенный демон drwebd версии 4.30 или выше;
- Samba 2.2.1 и выше.

О версиях Samba стоит сказать особо. В версиях 2.2.1—2.2.3 интерфейс vfs не входил в стандартную поставку. Поэтому придется произвести исправления. В комплект ASPLinux Server Edition входит Samba 2.2.7, поэтому наложения исправления не требуется. Переходим к установке.

Устанавливаем из rpm-пакета:

1. Входим в систему с правами root.
2. Входим в каталог, где хранятся пакеты с демоном антивируса.
3. Набираем в командной строке

```
# rpm -ihv drweb-samba-4.32.2-rh7.i586.rpm
```

При установке будут внесены изменения в файл /etc/samba/sbm.conf. Теперь в общие ресурсы добавиться строка:

```
# vfs object /opt/drweb/smb_spider.so
```

Если вы хотите, чтобы демон проверял общий ресурс, то эту строку необходимо раскомментировать (листинг 13.2).

**Листинг 13.2. Измененный smb.conf (фрагмент)**

```
[Kb]
        vfs object = /opt/drweb/smb_spider.so
       comment = Konstruktor
       path = /kb
       writelist = drweb,user11
       guest ok = Yes

[buch]
       comment = Buchgalteria
        vfs object = /opt/drweb/smb_spider.so
       path = /buch
       writelist = drweb,user11
       guest ok = Yes
```

**Замечание**

*Пользователю drweb (демону drwebd) все разделяемые ресурсы должны быть доступны и для чтения (проверка), и для записи (лечение).*

## 13.4. Настройка действия антивируса на Samba

Настройка действий антивируса осуществляется в файле /etc/drweb/smb_spider.conf. Подробное описание каждого параметра можно найти в документации. Мы рассмотрим только некоторые наиболее интересные параметры.

- включить (или выключить) эвристический анализатор для поиска неизвестных вирусов

```
HeuristicAnalysis = BOOL;
```

- режимы сканирования

  `ScanMode =… ;`
- файлы будут проверяться всегда: при открытии и при закрытии

  `onAccess onAccess`;
- файлы будут проверяться только при открытии

  `onRead`;
- файлы будут проверяться только при закрытии

  `onWrite`;

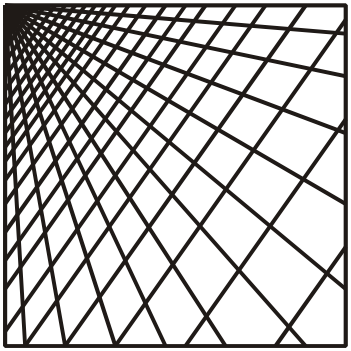
  действие, предпринимаемое при нахождении зараженного файла Infected:

  - reject — запретить операции с такими файлами;
  - discard — удалять такие файлы;
  - rename — запретить операции и переименовывать такие файлы;
  - quarantine — перемещать такие файлы в карантин;
  - cure — пытаться лечить такие файлы.

# Заключение

В этой книге я попытался отразить свой опыт в прокладке сетей и настройке сервера под управлением операционной системы Linux. Надеюсь, что материал, изложенный в книге, окажет вам помощь в проектировании и обслуживании вашей сети.

Успехов вам в освоении Linux.



# ПРИЛОЖЕНИЯ

# Приложение 1

# Полезные сочетания клавиш и некоторые команды

Сочетания клавиш и команды приведены в табл. П1.1 и П1.2.

***Таблица П1.1.*** *Полезные сочетания клавиш*

| Клавиши | Действие |
|---|---|
| <Ctrl>+<Alt>+<Fn> | Переключение на консоль n, n может принимать значения от 1 до 6. Седьмая консоль — графическая, доступна только в случае, если запущен графический режим. (Запуск графического режима в ASPLinux осуществляется командой `startx`) |
| <Shift> + <PgUp> | Просмотр содержимого консоли вверх по экрану. Используется, когда необходимо просмотреть содержимое ушедшее за границы экрана |
| <Shift> +<PgDn> | Просмотр содержимого консоли вниз по экрану. Используется совместно с <Shift> + <PgUp> |
| <Ctrl>+ <Alt> + <+> | Изменение разрешения экрана в графическом режиме верх по списку |
| <Ctrl>+<Alt>+<–> | Изменение разрешения экрана в графическом режиме вниз по списку |
| <Ctrl>+<Alt>+< > | Убивает графический режим. Если графический режим настроен как режим по умолчанию, то ничего не получится. Система вновь перейдет в состояние графического режима. В этом случае используйте <Ctrl>+<Alt>+<F1> |
| <↑> | Пролистывание списка введенных команд в обратном порядке, от последней команды к первой |
| <↓> | Пролистывание списка введенных команд в прямом порядке: от первой введенной команды до последней. Используется совместно с предыдущей |

***Таблица П1.1*** *(окончание)*

| Клавиши | Действие |
|---|---|
| <Tab> | Дополняет набор символов введенных в текстовой консоли до конца |
| <Ctrl>+<Alt>+<Del> | Перезагрузка компьютера. Процесс перезагрузки компьютера происходит корректно. Вначале закрываются все приложения, затем останавливается система |

***Таблица П1.2.*** *Некоторые полезные команды*

| Команда | Назначение |
|---|---|
| | **Работа с файлами и каталогами** |
| cd | Замена текущего каталога на указанный в строке |
| cd.. | Замена текущего каталога на родительский: подняться на один уровень вверх по дереву каталогов |
| cd~ | Замена текущего каталога на рабочий каталог пользователя |
| ls | Отображение списка файлов, находящихся в текущем каталоге |
| mkdir <имя> | Создать новый каталог с указанным именем |
| rmdir <имя> | Удалить указанный каталог |
| cat <имя> | Вывести на экран файл с указанным именем |
| cp <источник> <назначение> | Копировать файл |
| mv <источник> <назначение> | Переместить файл источника в назначение |
| rm <путь> | Удалить указанный файл |
| | **Работа с сетью** |
| netstat | Выводит на экран статус сетевых соединений |
| route | Выводит на экран текущую таблицу маршрутизации |
| ifconfig | Выводит на экран информацию о сконфигурированных интерфейсах |
| nslookup | Утилита для проверки работоспособности DNS-сервисов |
| | **Управление процессами** |
| ps | Показать список имеющихся процессов |
| kill <PID> | Убить процесс с соответствующим PID |
| killall | Убить все процессы |
| nice | Установить приоритет выполнения запускаемого процесса |
| renice <PID> | Изменить приоритет выполняемого процесса с соответствующим PID |

***Таблица П1.2*** *(окончание)*

| Команда | Назначение |
|---|---|
| | **Основные команды администрирования** |
| su | Смена пользователя. Без указания имени пользователь изменяется на root |
| useradd <имя> | Создать пользователя с заданным именем |
| passwd <имя> | Изменить пароль у соответствующего пользователя |
| userdel | Удалить соответствующего пользователя |
| grouppadd <имя> | Добавить группу с соответствующим именем |
| chmod <права> <путь> | Сменить права доступа к файлу с указанным именем |
| chown <пользователь> <путь> | Сменить владельца файла |
| **hgrp <пользователь> <путь>** | **Сменить группу-владельца** |
| | **Некоторые дополнительные команды** |
| reboot | Перезагрузка системы |
| mc | Запустить файловый менеджер |
| startx | Запустить графическую консоль |
| time | Получить информацию о системном времени |
| who | Получить информацию о пользователях, работающих в данный момент в системе |
| whoami | Получить информацию о себе |

# Приложение 2

# Аналоги Linux и Windows-программ

В табл. П2.1 приведены аналоги популярных программ.

***Таблица П2.1.*** *Аналоги Linux и Windows-программ*

| Программа для Windows | Программа для Linux |
|---|---|
| MS Office | Star Office |
| | Open Office |
| The Bat! | Kmail |
| Outlook | Mozilla |
| Internet Expoler | Mozilla |
| Opera | Opera |
| Norton Commander | Midnight Commander |
| Far | |
| Flachget | wget |
| Download master | |
| Delphi | Kylix |
| C++ Builder | |

Теперь об опыте перевода рабочего места секретаря с Windows на Linux.

# Постановка задачи

Исследуем используемое программное обеспечение и составляем список. В большинстве случаев он будет выглядеть примерно следующим образом:

- Windows 98 (2000) — в качестве рабочей среды;
- MS Office (Word, Excel) — для создания, редактирования и просмотра документов;
- Outlook для работы с почтой;
- Internet Explorer для работы в сети Интернет.

# Поиск аналогов используемых программ для Linux

В табл. 2.1 ищем программы аналоги для Linux. В частности, это могут быть:

- Star Office;
- Mozilla (почтовый клиент);
- Mozilla (браузер).

# Установка программного обеспечения и обучение работе

Устанавливаем выбранное программное обеспечение..

В качестве графической оболочки лучше выбрать KDE, так как он при настройках по умолчанию больше напоминает Windows.

Устанавливаем выбранное программное обеспечение.

Обучаем секретаря работе. Упор надо делать на аналог работы в предыдущем и вновь установленном программном обеспечении.

# Приложение 3

# Источники информации о Linux

## Русскоязычные источники

**http://www.ASPLinux.ru** — портал производителя операционной системы Linux.

**http://www.redhat.ru** — портал производителя операционной системы Red Hat.

**http://www.linux.ru** — крупный портал, посвященный Linux.

**http://www.linuxportal.ru** — портал документации по Linux-тематике.

**http://www.linuxbegin.ru** — сайт для начинающих пользователей Linux.

**http://www.samba.org.ru** — портал, посвященный Samba.

**http://www.lug.ru** — регистрация групп пользователей операционной системы Linux (Linux User Group).

**http://www.Linux-75.narod.ru** — портал, посвященный Linux и поддержке этой книги.

## Англоязычные ресурсы

**http://www.redhat.com** — портал производителя операционной системы Red Hat.

**http://www.kde.com** — портал производителя графической оболочки KDE.

**http://www.kernel.org** — портал разработчиков ядра Linux.

**http://www.linux.com** — портал информации, ссылок и новостей.

**http://www.samba.org** — портал информации о Samba.

**http://www.webmin.com** — портал производителя одноименной оболочки.

**http://www.openssl.org** — основная страничка проекта OpenSSL.

# Предметный указатель


## A

Apache 244
ARP-протокол 10
ASPLinux 119
AT 96
Auto negotuiation 22

## B

Backpressure 40
BIND 223
Bridge 34

## C

Caldera 119
Chmod 177
Collision 17
Collision domain 18
CSMA/CD 17

## D

DHCP 234
Dig 224
DNS 221
Dr. Web 261
DRAM 97

## E

EIA T568A 86, 87
EIA T568B 86, 87
Ethernet 16
Ext2 169
Ext3 169

## H

Halt 169
HDD 98
Host 225
Hub 32

## I

IP-адрес 9

## K

Kill 167

## L

LAN-tester 110
Ls 173

## M

MAC адрес 8
Mandrake 119
Midnight Commander 181
MMF кабель 14
Mother Board 94

## N

Nice 167
Nmbd 195
Nslookup 224

## P

PDC 199
PDV 17
Ps 163

## R

RAID 0 100
RAID 1 100
RARP-протокол 10
Red Hat 119
Renice 167
Router 34

## S

Samba 194
Samba Spider 264
SCSI 100
SDRAM 98
Shutdown 169
Smbclient 195
Smbd 195
SMF-кабель 14
SNMP 33
Socket 8
Spanning Tree Algorithm 50
STP-кабель 13
Structured Cabling System 55
Swat 195, 216
Switch 34
Syslog 184

## T

Testparam 195

## U

Umount 172
USB 95
Userdel 160
UTP кабель 12

## V

Vi 179
VLAN 51

## W

Webmin 254

## X

Xvidtune 138

## А

Агрессивный захват 40
Активное сопротивление 11
Алгоритм покрывающего дерева 50
Антивирусная защита 261
ATX 96

## В

Виртуальная локальнаях сеть 51
Витая пара 12
Волновое сопротивление 11
Волоконно-оптический кабель 14
Время двойного оборота 17
BTX 96

## Г

Гнездо 8
Горизонтальная система 57

## Д

Дейтограмма 7
Демонтирование файловой системы 170

Домен коллизий 18
Доменное имя 8

## Ж

Жесткий диск 98

## З

Задержка передачи 48
Затухание 11

## И

Инструмент для снятия изоляции 76

## К

Кабельный канал 80
Канальный уровень 6
Каталог
    /bin 173
    /boot 173
    /etc 173
    /etc/fstab 171
    /etc/group 151
    /home 173
    /lib 173
    /lost+foud 173
    /mnt 173
    /opt 173
    /proc 173
    /root 173
    /sbin 173
    /tmp 173
    /usr 173
    /var 173
Класс IP сетей 9
Коаксиальный кабель 12
Коллизия 17
Коммутатор 34
    на основе коммутационной матрицы 47
    с общей шиной 46
    с разделяемой памятью 45
Коммутации 43, 44
Концентратор 32

## Л

Локальный адрес 8

## М

Маршрутизатор 34
Маска сети 10
Материнская плата 94
Модель OSI 5
Модули памяти 97
Монтажные клещи 76
Монтажный шкаф 78
Монтирование файловой системы 170
Мост 34

## О

Обратное давление 40
Объем буфера порта 48
ОЗУ 97, 98
Оперативная память 97

## П

Пакет 7
Патч-корд 90
Патч-панель 88
Первичный контроллер домена 199
Перекрестные наводки 11
Погонная емкость 11
Полудуплексный режим 38
Порт 7
Поток 7
Права доступа 177
Прикладной уровень 5

Прозрачный мост 36
Пропускная способность 48
Протокол
IP 7
TCP 7
Процесс 162
Процессор 96

## Р

Размер адресной таблицы 48

## С

Сеансовый уровень 6
Сетевой
адаптер 31
уровень 6
Система здания 57
Скорость
продвижения 48
фильтрации 47
Стандарт
100Base-FX 22
100Base-T4 21
100Base-TX 21
10Base-F 19
10Base-T 19

Структурированная кабельная система 55

## Т

Топология
звезда 14
кольцо 15
общая шина 14
Транспортный уровень 6

## У

Ударный инструмент для разделки контактов 76
Универсальная последовательная шина 95
Уровень представления 5

## Ф

Файл
/etc/aspldr.conf 190
/etc/dhcpd.conf 236, 238
/etc/named.conf 225, 231
/etc/samba/smb.conf 195, 196, 199
/etc/samba/smbpasswd 195
/etc/shadow 149
/etc/sysconfig/samba 195
/etc/syslog.conf 184
/etc/init.d/smb 195
/etc/inittab 143
/etc/passwd 148, 149
/etc/rc.d/init.d/webmin 255
/etc/samba/lmhosts 195
Файловая система 169
Физический уровень 6
Форм-фактор 96

## Ш

Шина
AGP 95
PCI 95

## Э

Экранированная витая пара 13