Государственный Комитет по высшему образованию

**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ
ИНЖЕНЕРНО-ФИЗИЧЕСКИЙ
ИНСТИТУТ
(Технический университет)**

**Кафедра "КОМПЬЮТЕРНЫЕ СИСТЕМЫ
И ТЕХНОЛОГИИ"**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к дипломному проекту:

"Построение локальной компьютерной сети масштаба малого предприятия на основе сетевой ОС Linux"

Студент-дипломник: ***Селезнев Д.А.***

Руководитель: Начальник отдела АСУ
Подольского узла электросвязи,
руководитель региональной
провайдерской фирмы,
Моисеев А.В.

Рецензент: **аспирант**
Вавренюк А.Б.

Зав. кафедрой: **профессор,**
д.т.н. ***Забродин Л.Д.***

Москва
1999

Аннотация

Данный дипломный проект посвящен проблемам использования ОС Linux как основы сети офиса малого предприятия, решениям межплатформенного сетевого взаимодействия, Internet и Intranet-компонентам.

СОДЕРЖАНИЕ:

1. ВВОДНАЯ ЧАСТЬ

Обзор возможностей построения компьютерных сетей и сетевых операционных систем

1.1 Анализ информационных потребностей фирмы

Современный офис, пусть даже небольшой, сложно в наше время представить без сети в том или ином виде.

К началу работы над данным дипломным проектом в офисе фирмы существовала одноранговая сеть с топологией "шина", реализованной на "тонком" ethernet'е (стандарт 10Base2, волновое сопрпотивление кабеля RG-58 - 50 Ом). В ранний период развития офиса подобное построение сети было оправдано: низкая стоимость кабеля RG - 58, малая длинна сети и собственно простота ее организации. Рабочие станции работали под управлением Microsoft Windows for WorkGroups 3.11, Microsoft Windows 95 и Novell Personal Netware (Netware Lite).

По мере покупки новых рабочих станций и оборудования, роста информационных потребностей фирмы, появления новых информационных технологий и необходимостисти интеграции сети офиса в мировую сеть Интернет возникла необходимость качественного скачка - перехода к сети с выделенным сервером.

При анализе информационных потребностей фирмы автор определил для себя несколько ключевых моментов:

- наличие одноранговой сети с существующими рабочими станциями;
- возрастающая необходимость централизованного хранения и обработки даннных;
- надежность сети и всех ее компонент;
- удобство для конечных пользователей рабочих мест;
- существование различных платформ;
- гибкость администрирования и возможность масштабирования;
- информационная безопасность и защита информации;

При работе над проектом автор определил следующие функциональные требования:

1. Коллективная работа групп над едиными проектами;
2. Централизованный доступ к данным. Хранение данных на выделенном файл-сервере с разграничением прав доступа к информации;
3. Сетевые базы данных. Распределенные системы учета и автоматизация бухгалтерских расчетов;
4. Сетевая печать.
5. Доступ к электронной почте, группам новостей;
6. Использование Internet-технологий при ведении бизнеса. Intranet.
7. Общий доступ к глобальным сетям и мировым информационным ресурсам;
8. Организация удаленного доступа и удаленных рабочих мест;
9. Обеспечение информационной безопасности и сохранности данных;
10. Обеспечение межплатформенного взаимодействия;

1.2 Выбор сетевой ОС

1.2.1 Сравнение сетевых ОС

Ключевым звеном в сети является сетевая операционная система, своего рода"сердце сети". Поэтому, большая часть работы освещает различные аспекты настройки, администрирования сетевой ОС и ее взаимодействие с клиентскими станциями.

При стоящих перед автором задачах и имеющемся аппаратном обеспечении это могли быть лишь три системы: Microsoft Windows NT 4.0 Server, Novell Netware 4.11 или 5.0 (Intranetware) и UNIX.

Следует сразу отметить, что одним из важнейших критериев выбора являлись затраты, необходимые на приобретение как собственно ОС, так и программных продуктов для нее.

При всем уважении к продуктам фирмы Novell, следует отметить, что хотя в последних версиях ОС Netware и появилась поддерка протокола TCP/IP и традиционных Internet-сервисов, новшества эти носят характер этаких "присадок" и реализуются в виде дополнительных модулей. Вот фрагмент официального описания свойств Novell Netware 5:

NetWare 5 supports IP-only networks while retaining compatibility with IPX™. It enables you to implement TCP/IP, the standard protocol of the Internet, in your existing network infrastructure without additional routing overhead and without jeopardizing control, security, or performance.

Собственно говоря, Netware 5 является продуктом, позволяющим ранее постороенным на основе Netware и протокола IPX сетям адаптироваться к современным сетевым технологиям, прежде всего, к Интернет и TCP/IP. Поскольку сеть фирмы ранее использовала технологии Novell лишь в малой степени, автор пришел к выводу о неоптимальности выбора Netware 5 в плане функционального соответствия параметров ОС требованиям сети офиса.

Однако, помимо стоимости собственно ОС к возможным затратам фирмы следует прибавить и стоимость программных продуктов, обеспечивающий весь необходимый функциональный набор интернет-хоста, роутера, Веб-сервера, mail-сервера и прочих жизненно необходимых для сети сервисов.

Вторым возможным кандидатом на использование в качестве сетевой операционной системы была Windows NT 4.0 Server корпорации Microsoft, кажущаяся простота которой часто сбивает с толку начинающих системных администраторов. И хотя Microsoft позиционируют данную систему как серверную платформу для малого и среднего бизнеса, общеизвестно, что серьезные сетевые проекты в большинстве случаев по-прежнему базируются на платформе UNIX.

Сама по себе WindowsNT Server не является базой для законченных решений, здесь, в частности, следует упомянуть отсутствие в поставке таких необходимых вещей как полноценного сервера удаленного доступа, дисковых квот и серверных компонент Internet/Intranet.

Следует отметить явно завышенные по мнению автора требования к

аппаратному обеспечению, так, например, для полноценного функционирования сервера требуется не менее 96 мегабайт оперативной памяти.

Практика использования этой системы показала, что файловый сервис для Macintosh реализован далеко не в лучшем виде в плане скорости передачи файлов. Данный факт является отрицательным по отношению к тем сетям, в которых используются не только WindowsPC, но и Macintosh.

Эксплуатация данной ОС сопровождается установкой т.н. Service Pack, своеобразных “заплат” системы, устраняющих ошибки функционирования системы и бреши в защите.

И последний, весьма весомый фактор: WindowsNT Server - коммерческий программный продукт, стоимость которого превышает 1000 долларов, что при существующей экономической ситуации в стране является чрезвычайно болезненным аспектом использования этой системы в малом бизнесе.

Итак, Windows NT была отвергнута по следующим причинам:

- NT - коммерческий программный продукт, цена которого достаточного велика для фирмы;
- общее недоверие к программным продуктам Microsoft, их ненадежность, большое количество ошибок;
- высокие требования к аппаратному обеспечению при достаточно низкой производительности;
- недостаток свободных программных продуктов для этой системы;
- определенная функциональная неполноценность Windows NT как сетевой ОС;(здесь следует отметить, например, отсутствие в стандарной поставке дисковых квот и средств удаленного управления)
- соответвующие программные продукты для данной ОС являются коммерческими и требуют дополнительных затрат на их приобретение;

С другой стороны, операционная система Linux обладает следующими неосопоримыми достоинствами:

- незначительные требования, предъявляемые к аппаратному обеспечению;
- бесплатное распространение системы по лицензии GNU;
- гибкость настроек при одновременной мощности и традиционной высокой функциональности UNIX-систем;
- огромное количество свободно распространяемых продуктов (в том числе и в виде исходных текстов);
- отличная репутация ОС у коллег;
- полнота начальной дистрибъюции системы, позволяющая обеспечить функционирование большинства требуемых сервисов и служб; полная документированность;

Операционная система Linux стала привлекательной альтернативой коммерческих ОС для всех, кто работает на персональных компьютерах. Лавинообразный рост интереса к Linux во всем мире подтверждает это. В ней объединены мощь и гибкость рабочей UNIX-станции, возможность использования полного набора приложений Internet и полнофункциональный графический интерфейс (например, X Window). Все это свободно инсталлируется и прекрасно

функционирует на любом IBM PC-совместимом компьютере, оснащенном 386, 486, Pentium, Pentium II процессоре (существуют версии Linux для платформ PowerPC, Mac 68K, Alpha).

Linux разработал в начале 90-х годов Линус Торвальдс (Linus Torvalds), студент факультета вычислительной техники Хельсинского университета при участии программистов из разных стран мира. Эта ОС сочетает в себе скорость, эффективность и гибкость UNIX, используя при этом все преимущества современных персональных машин.

С финансовой точки зрения Linux обладает одним весьма существенным достоинством: это бесплатная система. В отличе от других операционных систем, Linux распространяется бесплатно по генеральной открытой лицензии GNU (см. соответсвующую главу) в рамках Фонда бесплатного программного обеспечения (Free Sofware Foundation), что делает эту ОС доступной для всех желающих. GNU лицензия сотавлена таким образом, что Linux остается бесплатной и в то же время стандартизованной системой.

Как уже говорилось, Linux является версией UNIX для персональных компьютеров. В отличие от большинства других операционных систем, UNIX разрабатывали в университетской, академической среде. Ее разработка шла параллельно с революцией в области вычислительной техники и коммуникаций, которая длится вот уже несколько десятилетий. Профессионалы по части компьютерной техники нередко разрабатывали на базе UNIX новые технологии.В частности, это касается развития Internet и средств для работы с Internet.

Как и все UNIX-системы Linux - многозадачная и многопользовательская ОС. Главным принципом Linux автор считает подход "все нужное собери себе сам". Собственно, Linux - есть ядро + подгружаемые модули+ приложения; и то и другое пользователь может пересобирать на основе предоставляемых исходных текстов согласно собственным конкретным потребностям и задачам. Этот принцип в сочетании с прозрачностью и гибкостью конфигурирования обеспечивает Linux чрезвычайную производительность, стабильность и эффективность при низким требованиям к аппаратуре.

1.2.2 Проект GNU

Фонд свободного программного обеспечения (FSF - Free Software Foundation) представляет собой очень интересное и во многих отношениях исключительное явление в современном мире программирования и программного обеспечения.

Идеология FSF и общие цели проекта GNU

FSF - это программистская организация, основанная и возглавляемая Ричардом Столлманом (Richard Stallman). В самой общей постановке задачей FSF является устранение ограничений по копированию, распространению, изучению и модификации программ для компьютеров. Для достижения этой общей задачи FSF стимулирует разработку и использование свободного программного обеспечения, ориентированного на широкий класс применений.

В своем "Манифесте GNU", написанном еще в 1985 г., Р. Столлман в качестве основной идеи, приведшей к возникновению FSF и проекта GNU, выдвигает свое неприятие права собственности на программы. Особенности взаимоотношений в сообществе программистов часто ставят людей перед выбором следования естественному чувству дружбы и взаимопомощи или подчинения, препятствующего этому закону о собственности. При использовании свободного программного обеспечения необходимость такого обременительного выбора исчезает.

Создание интегрированной свободной программной системы позволяет избежать дублирующей работы программистов (которая часто требуется только по причине наличия программ в чьей-либо собственности). Свободное распространение исходных текстов программ облегчает их сопровождение и приспособление к нуждам конкретного пользователя (не требуется прибегать к услугам только компаний - владельцев лицензий на исходные тексты). Появляется дополнительная и очень важная возможность использования хорошего программного обеспечения в учебных целях.

Как утверждает Р. Столлман, при переходе к свободному программному обеспечению программисты не вымрут от голода (хотя, видимо, будут зарабатывать несколько меньше). Ограничение на копирование программ - это не единственный способ зарабатывать деньги. Основная идея Столлмана состоит в том, что нужно продавать не программы, а труд программиста. В частности, источником дохода может быть сопровождение программных систем или их настройка для использования на новых компьютерах и/или в новых условиях, преподавание и т.д.

"Манифест" Столлмана написан очень эмоционально и местами слишком утопичен. Тем не менее, как кажется, идеи свободного программного обеспечения исторически близки традиционным (за исключением самых последних лет) отношениям в среде российских программистов. Возможно, именно линия FSF - наиболее естественный путь к глубокой интеграции отечественного и мирового сообществ программистов.

Более конкретно, FSF ведет разработку программ в рамках проекта GNU (аббревиатура GNU раскрывается рекурсивно - GNU's Not Unix). Целью проекта GNU является создание полной интегрированной программной системы, средства которой совместимы с возможностями среды ОС Unix (как правило, возможности программ GNU шире возможностей аналогов среды Unix).

Программное обеспечение FSF является "свободным" в двух смыслах. Во-первых, любую программу можно свободно копировать и передавать кому угодно. Во-вторых, наличие исходных текстов программ обеспечивает возможность свободного изучения программ, их улучшения и распространения доработанных вариантов.

Подобно тому, как права обычных компаний, производящих программное обеспечение, охраняются их знаком авторских прав (copyright), "свобода" программных систем FSF защищается "copyleft" - комбинацией copyright и присутствующим во всех текстах FSF документом с заголовком "GNU General Public License". В этом документе говорится о правах, которыми располагает любой текущий владелец данного текста, и о невозможности лишения этих прав у любого другого субъекта.

Основная деятельность FSF состоит в разработке новых составляющих свободного программного обеспечения в рамках проекта GNU. Большей частью проект GNU развивается плановым образом, но FSF принимает для свободного распространения и программы, разработанные фирмами и частными лицами по собственной инициативе. Кроме того, FSF занимается производством и продажей лент со свободным программным обеспечением, подготовкой, публикацией и распространением руководств по различным компонентам программного обеспечения GNU, а также поддерживает и распространяет справочник услуг - список фирм и частных лиц, которые оказывают платные услуги пользователям программ и систем GNU.

Финансовой основой FSF является продажа лент и документации, а также спонсорство коммерческих фирм и частных лиц.

В настоящее время существуют и развиваются тысячи GNU-проектов, над которыми трудятся программисты разбросанные по всему Миру. Значительная часть данного проекта реализована (как будет показно далее) с использованием GNU-приложений. Вот лишь некоторые из них:

Apache - HTTP сервер, используемый примерно на 50% Web-сайтов в Интернете. Он содержит обширный API для расширения с помощью модулей, множество способностей и большое количество подключаемых модулей; очень гибок, работает на большом количестве популярных операционных систем, имеет активную группу разработки и сообщество пользователей.
Bash, Bourne Again SHell, один из расширенных UNIX shell;
GNU Finger - утилита, позволяющая пользователям UNIX-хостов в сети Интернет получать информацию о других хостах;
GCC - свободный компилятор C, C++ и Objective C;
Ghostscript - интерпретатор языков Postscript и Adobe PDF;
gzip - GNU-вариант утилиты сжатия и разжатия zip;
Midnight Commander - UNIX файл менеджер, подобный Norton Commander;
Shell-утилиты в составе: `basename', `chroot', `date', `dirname', `echo', `env', `expr', `factor', `false', `groups', `hostname', `id', `logname', `nice', `nohup', `pathchk', `printenv', `printf', `pwd',`seq', `sleep', `stty', `su', `tee', `test', `true', `tty', `uname', `uptime', `users', `who', `whoami', and `yes';
GNOME - GNU desktop, обеспечивающий графический интерфейс пользователя огромного числа программ от таблиц до почтовых клиентов;

Emacs В 1975 году Ричард Столмен разработал первую версию Emacs, расширяемого, настраиваемого экранного редактора реального времени, а также среды для работы с машиной. GNU Emacs -- это вторая его реализация. Он предоставляет настоящий Lisp -- хорошо интегрированный с редактором -- для написания расширений и обеспечивает интерфейс с системой X Window. Emacs работает на Unix, MS-DOS и Windows NT или 95. В дополнение к своему собственному мощному набору команд, Emacs может эмулировать редакторы vi и EDT (редактор из операционной системы VMS фирмы DEC). У Emacs есть еще множество свойств и способностей, делающих его полноценной средой для работы с машиной. Руководство по GNU Emacs и справочная карточка поставляются в комплекте. Исходные тексты Справочника по языку Lisp редактора GNU Emacs и Руководства по програмированию на Emacs Lisp и введение поставляются в отдельных пакетах.

1.2.3 Различные дистрибутивы Linux

Существует несколько дистрибутивов Linux: Caldera, Slackware, Debian, RedHat, SuSE, KSI и другие. Эти дистрибьюции разлчаются наборами прикладных программ, средств разработки, основополагающих библиотек и некоторыми индивидуальными особенностями.

Вот краткие характеристики раличных дистрибьюций:

RedHat Linux

URL: www.redhat.com
Последняя версия: 6.0
Ядро: 2.2
libc: glibc 2.1
Менеджер пакетов: rpm
Си компилятор: egcs 1.1
Init: SysV-style

Дистрибутив RedHat Linux в данный момент является наиболее популярным. Входящие в дистрибутив удобные средства администрирования системы делают его простым для начинающих пользователей. Компания RedHat регулярно выпускает обновления для своих дистрибутивов (в том числе и для нескольких предыдущих его версий). RedHat так же спонсирует некоторые перспективные программные разработки для ОС Linux. RedHat Linux выпускается для платформ i386, Alpha, Sparc.

Slackware

URL: www.slackware.com
Последняя версия: 3.6 (4.0beta3)
Ядро: 2.0
libc: libc5
Менеджер пакетов: pkgtool (tgz)
Си компилятор: egcs
Init: BSD-style

Простота и логичность организации этого дистрибутива позволят вам до конца разобраться с устройством Linux. Большинство настроек производятся "напрямую", без дополнительных конфигураторов и других "прослоек". Это делает дистрибутив немного сложноватым для начинающих, но он пользуется заслуженной популярностью у большого количества пользователей. Плюсом является отсутствие long file names в дистрибутиве - слакварь можно ставить с досовского раздела винта, переносить на дискетах и винтах с fat16. Использование стандартного для юникса формата tar.gz в пакетах инсталляции - тоже достаточно удобная вешь. Обновления пакетов появляются достаточно регулярно.

Debian GNU/Linux

URL: www.debian.org
Последняя версия: 2.1
Ядро: 2.0
libc: glibc 2.0
Менеджер пакетов: dpkg (deb)
Си компилятор: egcs 1.0
Init: SysV-style

На текущий момент Debian является самым большим дистрибутивом. Создатели Debian'а очень щепетильно относятся к лицензированию, поэтому Debian является самым "чистым" дистрибутивом. Большое внимание уделяется тестированию готового продукта. Из недостатков можно отметить неудобный frontend deselect к менеджеру пакетов dpkg, который в будущем будет заменен на deity.

Debian выпускается для платформ i386, m86k (amiga, atari, macs), Alpha и Sparc.

KSI-Linux

URL: www.ksi-linux.com
Последняя версия: 2.0
Ядро: 2.2
libc: glibc 2.0
Менеджер пакетов: rpm
Си компилятор: egcs 1.0
Init: SysV-style

Данный дистрибутив построен на основе и с использованием идеологии Red Hat Linux, так что те, кто имел дело с Red Hat, найдут в нем много знакомого. Поддержка русского языка сделана правильным способом, т.е. с использованием правильной locale в кодировке koi8-r. KSI Linux был создан на Украине Сергеем Кубушином и по этому содержит большое количество русифицированного ПО. Процедура инсталяции проходит на русском языке.В состав входит K Desktop Environment.

SUSE

URL: www.suse.com, русский перевод на сервере iplabs
Последняя версия: 6.1
Ядро: 2.0
libc: glibc 2.0
Менеджер пакетов: rpm
Си компилятор: egcs 1.0
Init: BSD-style

S.u.S.E. Linux - один из самых популярных в Европе. Родной язык -

немецкий, переведен (вместе с подробным руководством) на английский, французский, итальянский и испанский. Компание SuSe является одним из основных разработчиков X-серверов для XFree86 - графической системы Linux. Поддержка новых видеокарт часто появляется сперва в дистрибутиве S.u.S.E. и только спустя некоторое время - в составе XFree86 и других дистрибутивов. Дистрибутив имеет очень хорошую программу установки и администрирования YaST, включает в себя более 800 пакетов.

S.u.S.E. Linux может устанавливаться на FAT16 с использованием live file system, а входящее в комплект 450- страничное руководство - лучшее в своем жанре. S.u.S.E. содержит 10 оконных менеджеров, KDE, Gnome. S.u.S.E Linux удовлетворит и новичков и старых поклонников Linux. Полный (коммерческий) дистрибутив SuSe состоит из 5-ти дисков.

Black Cat Linux

URL: linux.geon.donetsk.ua
Последняя версия: 5.3 Ядро: 2.2
libc: glibc2
Менеджер пакетов: rpm
Си компилятор: egcs 1.0.2
Init: SysV-style

Дистрибутив ОС Linux, созданный на основе популярного дистрибутива RedHat и под влиянием Mandrake , а также с учетом некоторого опыта в инсталляции и настройке Linux-серверов. Кроме обновленного GPL RedHat 5.1, Black Cat Linux 5.1 Spitfire включает в себя: все необходимые средства русификации в кодировке KOI8-R, KDE, дополнительные средства для работы в сетях Relcom и Fidonet и другие приятные и полезные мелочи.

Русский Linux "Красная Шапочка"

URL: www.magister.msk.ru/tech/linux/rh-rus.htm
Последняя версия: 6.0
Ядро: 2.0
libc: glibc 2.0
Менеджер пакетов: rpm
Си компилятор: egcs 1.0
Init: SysV-style

"Красная Шапочка" - русская дистрибуция Linux на базе дистрибуции RedHat, русифицированная и с набором русифицированных программ.

Stampede Linux

URL: www.stampede.org
Последняя версия: Europe 0.89
Ядро: 2.2

libc: glibc2.1
Менеджер пакетов: slp (not slackware!)
Си компилятор: pgcc
Init: BSD-style

Дистрибутив Stampede Linux предназначен только и исключительно для Pentium процессоров. Хотя это не означает невозможность работы на 386/486. Все пакеты скомпированы pgcc (PentiumGCC), что дает прирост производительности.Версии пакетов самые последние, даже свежее ;-) Включено достаточное количество библиотек для разработки, вследствие чего компилируются практически любые исходные тексты. Пакеты сжаты bzip2 - это уменьшило размер дистрибутива раза в полтора по сравнению с rpm. Недостатки: дистрибутив “сыроват”. Инсталлятор недоделанный. Пока не для новичков.

Открытое Ядро

URL: www.usoft.spb.ru
Последняя версия: 5.2
Ядро: 2.0
libc: libc6 (glibc)
Менеджер пакетов: rpm
Си компилятор: egcs
Init: SysV-style

Является достаточно полной копией текущего дистрибутива RedHat с добавлением пакетов русификации и большого количества документации (в том числе и на русском языке)

Mandrake Linux

URL: www.linux-mandrake.com
Последняя версия: 5.3
Ядро: 2.0
libc: libc6 (glibc)
Менеджер пакетов: rpm
Си компилятор: egcs 1.0
Init: SysV-style

Mandrake Linux - дистрибутив Linux, основанный на RH5.2. Он сделан по схеме "RedHat + KDE", т.е. он содержит KDE и некоторые дополнительные наработки для интеграции RedHat Linux и KDE, а также некоторые незначительные изменения.

Автор остановил свой выбор на RedHat 5.2 с ядром 2.0.36 из-за ее наибольшей распространенности, большого количества существующих RPM (RedHat Package Manager) версий прикладных программ, легкости инсталляции и наличии развитых средств администрирования.

2. РАСЧЕТНО-КОНСТРУКТОРСКАЯ ЧАСТЬ

2.1 Топология сети. Схема сети.

Общая схема сети офиса ПФГ “ПРОМЭКСПОРТ” изображена в Приложении 1.

Топология сети представляет из себя сочетание шины, выполненной на коаксиальном кабеле и сегменте, управляемом хабом. Компьютеры Учебного центра (alpha, beta, gamma, delta), главного бухгалтера (buh), коммерческого директора (nalim), компьютер медицинского центра (med) используют ethernet-соединения стандарта 10Base2 (“тонкий ethernet”). Географически коаксиальный кабель соединяет между собой достаточно удаленные друг от друга помещения, и с этой точки зрения его использование оправдано. При всей дешевизне соединения коаксиальным кабелем, следует отметить его главный недостаток: разрыв соединения в любой точке кабеля делает полностью нерабочим целый сегмент сети.

По этой причине особо критичные устройства сети (UNIX-сервер, рабочие станции Macintosh, сетевой принтер HP LaserJet 5M, менеджерская машина Компьютерного центра и машины Интернет-кафе) соединены с основной сетью при помощи 8-портового концентратора NetGear EN108 известной фирмы BayNetwork. Соединения при помощи витой пары 10BaseT (UTP) и концентратора более надежны и работоспособны: выход из строя одной линии не влияет на остальных участников сети, хаб имеет специальную световую индикацию установления соединения, загруженности и числа коллизий, что сильно упрощает визуальную диагностику и контроль аварий. К недостаткам витой пары следует отнести большое число кабельных отрезков и их суммарную длинну.

На схеме отражены соединение сервера по выделенной асинхронной линии с маршрутизатором провайдера при помощи модема для выделенных линий (скорость до 115200 бит/сек), а также доступ удаленных клиентов к сети фирмы.

2.2 Отбор необходимых протоколов

2.2.1 Протокол TCP/IP

Протокол TCP/IP (Transmission Control Protocol/Internet Protocol) - основное средство современного сетевого и межсетевого взаимодействия. Не секрет, что большинство современных систем поддерживают данный протокол. Распространению данного протокола способствовало , в частности, развитие сети Internet и использование TCP/IP в качестве универсального транспорта. TCP/IP предоставляет пользователям однородный интерфейс, обеспечивающий взаимодействие с сетевыми аппаратными средствами различных типов. Этот протокол гарантирует возможность обмена данными между системами, невзирая на многочисленные различия, существующие между ними. TCP/IP, кроме того, позволяет соединять на программном уровне отдельные физические сети в более крупную и более гибкую логическую сеть.

В состав комплекта TCP/IP входит несколько компонентов:

- межсетевой протокол (Internet Protocol, IP), который обеспечивает транспортировку без дополнительной обработки данных с одной машины на другую;

- межсетевой протокол управления сообщениями (Inernet Control Message Protocol, ICMP), который отвечает за различные виды низкоуровневой поддержки протокола IP, включая сообщения об ошибках , содействие в маршрутизации, подтверждение получения сообщения;

- протокол преобразования адресов (Adsress Resolution Protocol, ARP), выполняющий трансляцию логических сетевых адресов в аппаратные;

- протокол пользовательских дейтаграмм (User Datagram Protocol, UDP) и протокол управления передачей (Transmission Control Protocol, TCP), которые обеспечивают пересылку данных из одной программы в другую с помощью протокола IP. Протокол UDP обеспечивает транспортировку отдельных сообщений без проверки , тогда как TCP более надежен и предполагает проверку установления соединения.

Сетевые пакеты могут достичь пункта назначения только при наличии правильного адреса. Протокол TCP/IP использует сочетание нескольких схем адресации. Самый нижний уровень адресации задается сетевыми аппаратными средствами. Так, например, ethernet-устройствам при изготовлении присваиваются шестибайтовые аппаратные адреса. В некоторых сетях с двухточечным соединением (SLIP, PPP, используемых в сети “ПРОМЭКСПОРТа”) аппаратные адреса вообще не нужны: адрес пункта назначения указывается непосредственно при установлении соединения.

На следующем более высоком уровне используется Internet-адресация (которую чаще называют IP - адресацией). Каждому включенному в сеть устройству

присваивается один четырехбайтовый IP-адрес (например: 195.133.132.17). IP-адреса глобально-уникальны и не зависят от аппаратных средств. Их назначение - содействовать процессу маршрутизации пакетов из одной сети в другую с тем, чтобы машины, находящиеся в разных физических сетях могли взаимодействовать друг с другом. Если первым байтом адреса является число 127, оно обозначает закольцованный интерфейс - фиктивную сеть, не имеющую реального аппаратного интерфейса и состоящую только из локальной хост-машины. Закольцовывающий адрес 127.0.0.1 всегда обозначает текущую машину, ее символическое имя - local-host.

Соответствие между IP-адресами и аппаратными адресами сетевых устройств реализуется на канальном уровне модели TCP/IP. В современных сетях, допускающих широковещательный режим (broadcasting), протокол ARP обеспечивает автоматический поиск соответствий без участия администратора. В качестве широковещательного адреса используют последний адрес машинной части подсети. Например, в сети ПФГ “ПРОМЭКСПОРТ” с адресом 195.133.132.16 и маской подсети 255.255.255.240, определяющей 16 адресов сети последний адрес, т.е. 195.133.132.31 является широковещательным (broadcast) адресом.

IP-адреса недостаточно конкретны для адресации отдельных процессов и служб, они идентифицируют лишь машины (вернее, сетевые интерфейсы, которых может быть несколько на одной машине). Протоколы TCP и UDP IP-адреса концепцией портов. Порт в данном случае представляет собой двухбайтовое число, добавляемое к IP-адресу и указывающее конкретного адресата той или иной сетевой службы. Вот пример такого соответствия номера порта сетевым службам (фрагмент файла /etc/services головной UNIX-машины фирмы):

```
tcpmux          1/tcp       # TCP port service multiplexer
echo            7/tcp
echo            7/udp
discard         9/tcp       sink null
discard         9/udp       sink null
systat          11/tcp      users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd            17/tcp      quote
msp             18/tcp      # message send protocol
msp             18/udp      # message send protocol
chargen         19/tcp      ttytst source
chargen         19/udp      ttytst source
ftp-data        20/tcp
ftp             21/tcp
# 22 - unassigned
telnet          23/tcp
# 24 - private
smtp            25/tcp      mail
```

Доменная система имен (Domain Name System, DNS) - это распределенная база данных, которая содержит информацию о компьютерах, включенных в сеть Internet. Доменная система имен выполняет несколько задач, но основная ее работа - преобразование имен компьютеров в IP-адреса и наоборот. DNS определяет:

- иерархически организованное пространство имен машин;
- таблицу машин, реализованную в виде распределенной базы данных;
- библиотечные подпрограммы запросов этой базы данных;
- усовершенствованные средства маршрутизации электронной почты;
- протокол обмена информацией об именах;

Узлам, подключенным к Internet, доменная система имен нужна для полноценного участия в работе сети. Так домен, делегированный сети фирмы ПРОМЭКСПОРТ региональным провайдером Интернет - px.podolsk.ru. Соответственно, полные буквенные имена машин офиса сформированы из названия машины+название домена, например, головная машина unix.px.podolsk.ru имеет численный адрес 195.133.132.17.

Вообще, функционирование DNS является достаточно сложным и объемным вопросом. Отмечу лишь что DNS фирмы реализована на UNIX-системе BIND (Berkley Internet Name Domain) и логически делится на внутреннее пространство имен домена px.podolsk.ru (прямая и рекурсивная части), обращающейся к авторитетному DNS-серверу провайдера (forwarding). Подробно настройка DNS описана в соответствующей главе.

Одним из важнейших понятий и компонент сетей TCP/IP является маршрутизация. В двух словах это система обеспечения известности маршрута следования пакета от отправителя к получателю. В качестве маршрутизатора офисной сети выступает все та же UNIX-машина со статической маршрутизацией. Подробно маршрутизация, ее концепции и реализация на примере сети ПФГ “ПРОМЭКСПОРТ” будет описана ниже.

2.2.2 Протокол AppleTalk

Протокол AppleTalk был разработан фирмой Apple для обеспечения совместного использования принтеров первыми компьютерами Macintosh. Позже этот комплект был расширен и стал включать полный набор средств поддержки сетей. Этот протокол встроен во все системы Macintosh, а также во многие принтеры.

AppleTalk почти не требует конфигурирования на машинах конечных пользователей, и одноранговые сети машин Macintosh организуются довольно просто. Тем не менее, маршрутизаторы AppleTalk должны быть высокоинтеллектуальными и требуют тщательного выполнения сложных работ по конфигурированию.

AppleTalk работает с целым рядом аппаратных средств, реализующих обмен информацией через последовательные порты (LocalTalk) и Ethernet (EtherTalk). Фирма Apple реализовала протоколы TCP/IP для Макинтош в расширении системы MacOS (точнее, это расширение входит в состав стандартной поставки OpenTransport, полноценного сетевого набора MacOS).

При работе в среде Ethernet протоколы AppleTalk называются EtherTalk.EtherTalk имеет две разновидности: Phase 1 и Phase 2. Phase 1 была первой попыткой фирмы Apple использовать сеть Ethernet; эта попытка выявила целый ряд проблем. Наиболее серьезной из них было пристрастие Phase 1 к широковещательному режиму, что ухудшало показатели работы всей сети. Такие недостатки создали протоколу AppleTalk очень плохую репутацию среди администраторов сетей, и это предубеждение сохранилось до сегодняшнего дня.

В Phase 2 фирма Apple устранило большинство этих проблем, в основном путем замены широковещательной адресации на групповую. В Phase 1 были собственные типы пакетов, тогда как Phase 2 оформляет пакеты заголовком, соответствующим стандарту 802.3 (SNAP). Групповые пакеты AppleTalk представляют собой SNAP-пакеты, направляемые Ethernet-адресам начиная с 9:0:7.

Как IP-адрес, адрес AppleTalk состоит из номера сети и номера узла. Эта комбинация однозначно определяет каждый объект сети. Номер узла динамически присваивается устройству согласно протоколу преобразования адресов AppleTalk (AARP), когда устройство подключается к сети. Поскольку номер узла присваевается “на лету”, то при перезапуске машины он часто меняется. Наряду с этим, существует и возможность конкретного назначения адреса тому или иному устройству.

Устройства на том или ином узле AppleTalk группируется по зонам, чтобы их легче было найти. В каждой сети имеется перечень зон. Объекты сети могут относится к любой из зон, связанных с сетью, а зона может включать устройства, принадлежащие нескольким сетям. Как и сетевые номера, имена зон статически присваиваются сети маршрутизатором. Поскольку общее число компьютеров Macintosh в сети фирмы не велико, протокол настраивается на зону, оговоренную как зона по умолчанию: * .

Помимо имени зоны, каждый объект AppleTalk имеет тип объекта и тип. Имена формируются без учета регистра и могут содержать пробелы, знаки препинания и восьмибитовые символы. Имя объекта обозначает конкретное устройство, например, имя сетевого принтера в нашей сети выглядит так: “HP

LaserJet 5M". Поле типа содержит описание устройства. Для получения сетевых и узловых номеров из триплетов объект-тип-зона клиенты пользуются протоколом привязки имен AppleTalk.

Помимо сетевого принтера и временных разделенных дисковых разделов рабочих станций в сети EtherTalk успешно функционирует GNU-версия эмулятора AppleShare Server - netatalk, работающая под управлением ОС Linux. Для компьютеров Macintosh специально выделенный для этой цели раздел файловой системы Linux выглядит как обычный разделяемый ресурс AppleShare, что особенно важно для хранения файлов Macintosh, состоящих из двух ветвей (data fork и resource fork). Данная область используется для резервного копирования данных и временного высвобождения дискового пространства рабочих станций (например, для записи CD ROM).

Следует отметить, что решение об использовании протокола AppleTalk естественно вытекает из удобства использования традиционных сетевых средств Macintosh для конечного пользователя, наличия необходимых драйверов сетевого принтера HP LaserJet 5M и, собственно, необходимости взаимодействия рабочих станций Macintosh в режиме одноранговой сети. (См. раздел " 2.3.2 Межстанционнное взаимодействие").

2.2.3 Протокол IPX

IPX (Inernetwork Packet Exchange) - это протокол обмена между сетями, разработанный фирмой Novell и входящий в состав ее сетевого программного продукта NetWare. Фирма Novell достигла ошеломляющего успеха в продаже программного обеспечения для локальных вычислительных сетей на базе персональных компьютеров. Достаточно отметить, что большинство производителей операционных систем включили клиентское программное обеспечение NetWare в состав своих ОС.

Последние версии NetWare (4.11, Inranetware) используют и TCP/IP, однако большинство систем все еще используют протокол IPX.

Протокол IPX использует в своем Ethernet-пакете заголовок, соответствующий стандарту 802.3. Данный протокол реализует механизм “гнезд” (sockets) с доставкой дейтаграмм.

На протокол IPX опираются множество других протоколов, в том числе:

- протокол данных маршрутизации (RIP);
- протокол обмена нумерованными пакетами (Sequenced Packet Exchange, SPX);
- протокол ECHO, посылающий пакеты обратно отправителю;
- протокол сообщений об ошибках (ERROR);
- протокол обмена пакетами (Packet Exchange Protocol, PEP), используемый большинством служб NetWare;
- протокол сервисных объявлений (Service Advertisement Protocol, SAP-брокер сервер-адресов);

Данный протокол используется в сети фирмы для соединение клиентских станций с эмулятором NetWare Server 3.12 -mars_nwe, работающим под управлением UNIX (подробно об этом рассказано в разделе “2.3.1 Централизованные файловые сервисы”,так же сетевая печать с Windows-станций на сетевой лазерный принтер осуществляется по протоколу IPX (См. “2.5.7 Сетевая печать”).

2.2.4 Протокол PPP

В качестве альтернативы аппаратным сетевым соединениям, таким как Ethernet, можно воспользоваться модемом и телефонными линиями. Существуют два протокола, которые позволяют передавать IP-пакеты по коммутируемым телефонным каналам. Это SLIP (Serial Line Internet Protocol - межсетевой протокол для последвоательного канала) и протокол PPP (Point-to-Point Protocol - протокол “точка-точка”). SLIP - старый протокол, а PPP - более современный и очень стабильный. Эти протоколы обычно используются для соединений машин пользователья с Internet-провайдерами, а так же для организация удаленного доступа к ресурсам корпоративных сетей.

PPP - это “универсальный” протокол оформления (инкапсуляции) пакетов. Он позволяет передавать мультипротокольные пакеты по одному каналу. Описание этого протокола приведено в RFC1331. Он отличается большей гибкостью, чем SLIP, который обрабатывает только IP-пакеты.

В состав протокола PPP входят три компонента:

- процедура инкапсуляции дейтаграмм для передачи их по последовательным каналам;
-“протокол управления каналом” (Link Control Protocol, LCP), предназначенный для установления, конфигурирования и тестирования соединения на канальном уровне;
- семейство “протоколов управления сетью” (Network Control Protocols, NCP), обеспечивающий конфигурирование и функционирование различных протоколов сетевого уровня.

Протокол PPP отличается рядом интересных особенностей, которыми не обладает протокол SLIP. В частности, PPP может инкапсулировать пакеты, соответствующие различным протоколам, для передачи их по одной последовательной линии. В нем есть встроенные средства коррекции ошибок и компрессии.

Протокол PPP связывает внутреннюю сеть с маршрутизатором CISCO, установленным у регионального провайдера, обеспечивая фирме постоянный выделенный канал в сеть Internet. На обоих концах соединения установлены асинхронные модемы для выделенных линий, обеспечивающие скорость соединения до 115200 бит в секунду.

Использование протокола PPP достигается его встроенными реализациями в ядре Linux и программными компонентами, основу которых составляет демон pppd.

2.3 Клиент-серверные компоненты

2.3.1 Централизованные файловые сервисы

Централизованные файловые сервисы работают на головной машине под управлением ОС Linux и призваны обеспечить централизованное хранение данных с высокой надежностью, четкое разделение прав доступа к ним и работу различных пользователей с одними и теми же документами .

Данная задача решена посредством трех компонент:

- Samba Server - сервер сетей Microsoft;
- MARS NW Server - эмулятор сервера Novell Netware;
- пакет NetAtalk 1.4 - эмулятор сервера AppleShareIP;

Остановимся более подробно на каждом из них:

2.3.1.1 Samba

Версия 1.9.18 пакета Samba, свободная версия SMB и CIFS -клиентов и серверов для платформы UNIX, разработана Andrew Tridgell в рамках проекта GNU (см. соответствующий раздел) и поддерживается командой разработки SAMBA Team.

Вот лишь сокращенный вариант свойств и возможности Samba 1.9.18:

- SMB сервер, обеспечивающий файл- и принт-сервисы в стиле Windows NT и LAN Manager-style для SMB клиентов таких как Windows 95, Warp Server, smbfs и других.
- Netbios (согласно rfc1001/1002) сервер имен, обеспечивающий наряду с прочими особенностями сетевой броузинг. Samba по желанию может являться ведущим броузером сети.
- ftp-образный SMB клиент обеспечивает доступ к разделенным ресурсам (дискам и принтерам) компьютеров, работающих под управлением Unix, NOVELL Netware и других операционных систем.
- Клиентское tar-расширение, позволяющее производить резервное копирование дисковых ресурсов сети.

Относящиеся к данному проекту пакеты включают в себя:

- smbfs, файловая система Linux, позволяющая монтировать разделы удаленных SMB-файловых систем.
- tcpdump-smb, расширение стандартного tcpdump, позволяющее диагностировать работу SMB по протоколам NetBeui и TCP/IP.
- smblib, библиотека сетевых smb-функций для разработки сетевых приложений, основанных на SMB.

В сети фирмы Samba служит для доступа к разделенным файловым ресурсам головного сервера и удаленного резервного копирования наиболее критичных разделов рабочих станций Windows (см. раздел "Резервное копирование").

2.3.1.2 Mars NWE

Пакет Mars_nwe, свободно распространяемый в рамках лицензии GNU (см. раздел “Проект GNU”) эмулятор NetWare(tm) для ОС Linux и UnixWare, написан Martin Stover, Германия.

Вот краткое изложение функциональных возможностей пакета:

- mars_nwe - чрезвычайно функциональная замена сервера NetWare, работающая на ОС Linux. Он прекрасно работает с обычным клиентским программным обеспечением DOS, поставляемым с Netware.
- mars_nwe осуществляет файл-, bindery- и принт-print сервисы для клиентского программного обеспечения Netware.
- отмечается, что пакет mars_nwe работает медленней NetWare 3.12, но БЫСТРЕЕ NetWare 4.1 при одинаковом аппаратном обеспечении.
- mars_nwe не содержит каких-либо лицензионных ограничений по его перекомпиляции или/и запуске нескольких mars_nwe в сети.
- mars_nwe содержит демон RIP/SAP, обеспечивающий функции IPX-роутера.

Относящиеся к данному проекту пакеты включают в себя:

- mars_dosutils: Набор утилит, позволяющих улучшающих работу с DOS-клиентами.
- ncpfs: файловая система ОС Linux, позволяющая монтировать разделы серверов NetWare на ОС Linux.

В рамках данного проекта пакет MARS используется как альтернативный файловый сервер рабочих станций Windows95 (следует отметить, что клиентская часть NetWare включается в поставку Windows95).

2.3.1.3 Netatalk

Последний пакет, представляет, пожалуй, особый интерес. Netatalk представляет собой свободно распространяемую версию средств и сетевых возможностей протокола AppleTalk для ОС UNIX. Пакет разработан Группой системных исследований UNIX Мичиганского университета (Research Systems Unix Group, The University of Michigan) и распространяется свободно. Данный пакет поддерживает такие протоколы как EtherTalk Phase I и II, DDP, RTMP, NBP, ZIP, AEP, ATP, PAP, ASP, и AFP. Общая структура поддерживаемого стека протоколов и сервисов показана на Рис 1.

Поддержка DDP осуществлена на уровне ядра (см. раздел "Инсталляция Linux") . Демон atalkd включает в себя RTMP, NBP, ZIP, AEP и является эквивалентом демона роутинга routed ОС UNIX в терминологии AppleTalk. Он также включает в себя клиент-ориентированную библиотеку NBP и инкапсулированные ATP и ASP - библиотеки. Демон papd позволяет станциям Macintosh посылать задания в очередь печати ОС UNIX lpd, а pap позволяет UNIX-машинам использовать для печати принтеры, подсоединенные по AppleTalk. psf является PostScript- фильтром для lpd при использовании с pap. Демон afpd является эмулятором традиционного для Macintosh-сетей Apple FileShare Server, обеспечивая клиентским станциям Macintosh возможность монтирования файловых ресурсов ОС UNIX традиционными средствами MacOS.

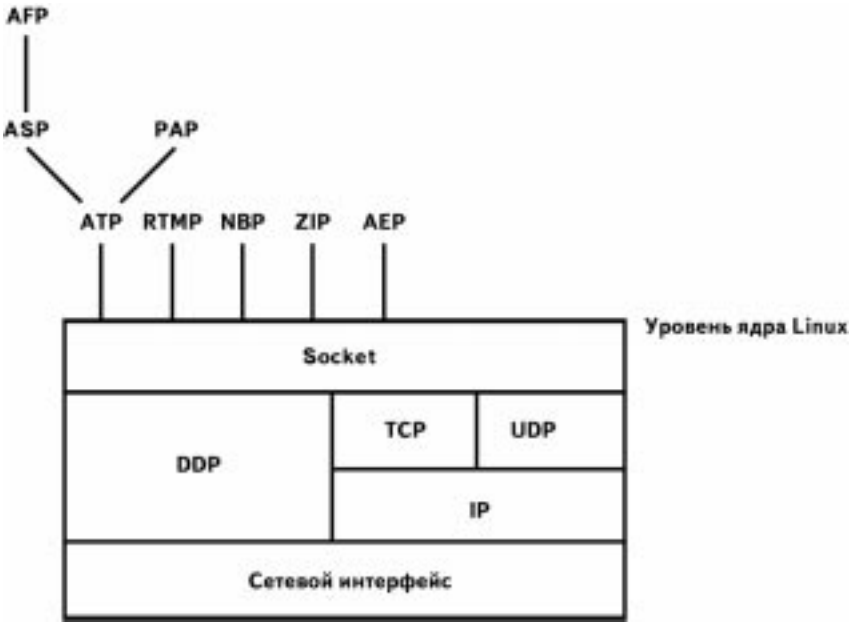


Рис. 1

Данный пакет используется в обновленной модификации netatalk-1.4b2+asun2.1.0 с исправлениями и добавлениями, сделанными Адрианом Сан (Adrian Sun). Коллеги автора по всему Миру, имеющие опыт совместного использования ОС UNIX и Macintosh в сетях отмечают чрезвычайную значимость новых свойств пакета.

Вот перечень усовершенствований, функциональных расширений и исправлений по сравнению со стандарным пакетом netatalk-1.4b2:

- Поддержка стандарта AFP 2.2 (AppleShare TCP/IP). Данное усовершенствование является главным, т.к. позволяет использовать поставляемое с MacOS клиентское программное обеспечение AppleShare IP и "вездесущий" TCP/IP в качестве протокола транспортного уровня. Эта возможность существенно увеличивает скорость передачи файлов между сервером Linux и рабочими станциями Macintosh, а также качественно упрощает доступ к ресурсам сервера фирмы для удаленных станций Macintosh (например, с использованием PPP).
- Поддержка дисковых разделов большого объема.
- Поддержка сообщений сервера для клиентов Macintosh.
- Поддержка tcp wrapper.
- Исправление ошибок информации о сервере, широковещательного просмотра, информации о файлах и директориях.
- Поддержка дисковых квот.
- Поддержка стандарта Apple II ProDOS.
- Поддержка методов аутентификации и криптования паролей Randnum и двунаправленного Randnum.
- Исправление ошибки файлового дескриптора.
- Исправление ошибки т.н. "плящущих иконок" - порчи ресурсов иконок при передае файлов.
- Возможность аутентификации реальных имен пользователей ОС UNIX.
- Отсутствие необходимости описания пользовательского раздела .AppleVolumes в конфигурационном файле.
- Поддержка byte locks при работе с NFS-разделами.

Исключительной особенностью всех трех пакетов является их свободное распространение в GNU-подобной манере при устойчивой эксплуатации в течении достачно продолжительного периода.

2.3.2 Межстанционное взаимодействие

Собственно под межстанционным взаимодействием администратор сети подразумевает сетевое взаимодействие рабочих станций на уровне одноранговой сети, т.е. ситуация, когда рабочие станции должны обладать как клиентским программным, так и серверной частью. Подобное взаимодействие необходимо, например, для рабочей группы в рамках работы над одним проектом.

2.3.2.1 Рабочие группы Windows

Взаимодействие рабочих станций Windows 95 осуществляется штатными средствами системы (“Клиент для сетей Microsoft”, “Служба доступа к файлам и принтерам сетей Microsoft”). В сети фирмы выделены две рабочие группы сети Microsoft: Office - группа внутреннего пользования сотрудников фирмы и группа Uch_Class для взаимодействия рабочих станций учебного класса. Так, в частности, в Учебном центре фирмы накопителем CD-ROM оснащена лишь одна машина Alpha, остальные машины Учебного центра при необходимости используют данный сетевой ресурс для инсталляции программного обеспечения и для совместного использования учебных программ центра на CD-ROM. На Рисунке 2 представлено окно “Проводника” Windows 95 рабочей станции Win:

Рис. 2

В левой части окна отображено дерево “Вся сеть”, в котором представлены рабочие группы Office и Uch_Class с активными на тот момент рабочими станциями.

2.3.2.2 DAVE Client-Sharing

Особый интерес представляет собой интеграция рабочих станций Macintosh в сеть Microsoft. Для этого использован программный продукт Dave 2.1 американской фирмы THURSBY SOFTWARE SYSTEMS, INC.. Красота подобного решения заключается в том, что Dave инсталлируется лишь на рабочие станции Macintosh, а установки особых клиент-серверных компонент для Windows-машин не требуется.

Вот характеристики этого программного продукта:

DAVE - программное решение, позволяющее пользователям Macintosh интегрироваться в сети Microsoft с файл-, принт- и сервисом сообщений. DAVE предлагает полный диапазон операций типа “станция-к-станции” между Mac’ами и Windows PCs, и обеспечивает полные клиентские свойства в сети NT.

DAVE устанавливается прямо на компьютеры Macintosh, обеспечивая пользователям Macintosh возможность монтирования разделенных директорий машин под управлением Windows NT (Server и Workstation), Windows 95, Windows 98 и Windows for Workgroups. В дополнение к этому, пользователи Windows PC могут монтировать разделенные директории на Mac’ах и производить печать на Macintosh-совместимых принтерах стандарта PostScript.

DAVE выполняет все коммуникационные операции, основываясь на промышленном стандарте-протоколе TCP/IP с поддержкой Domain Name Service (DNS), драйвере NetBIOS в полном соответствии с документами RFC 1001/1002.

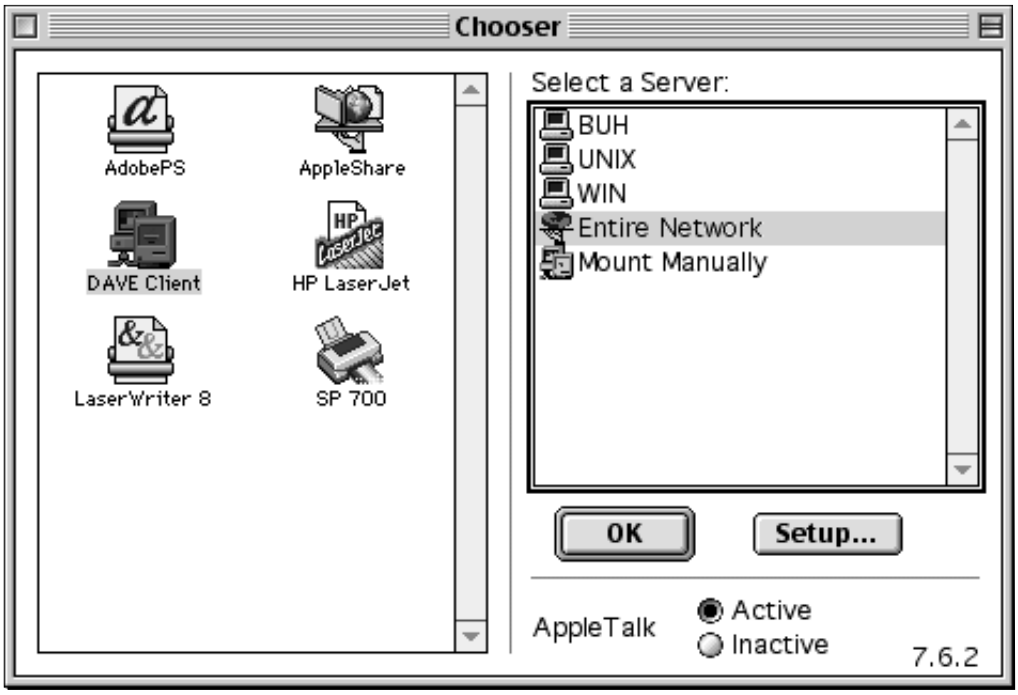


Рис. 3

Драйвер NetBIOS может быть отконфигурирован на поддержку DHCP для упрощения процедуры управления сетью. DAVE поддерживает Windows Internet Name Server (WINS) и все сервисы, ориентированные на использование протокола Common Internet File System (CIFS). CIFS является улучшенным расширением кросс-платформенного протокола представления разделяемых файловых ресурсов, называемого Server Message Block (SMB). DAVE оснащен необходимым набором драйверов, расширений (extentions) и приложений, которые отвечают возможным индивидуальным требованиям пользователей компьютеров Macintosh.

Как видно из Рис. 3, системное раширение MacOS DAVE Client позволяет пользователям Macintosh выбирать разделенные ресурсы сети Microsoft через традиционное окно Chooser, что делает процесс таким же простым и понятным как и при использовании традиционных для Macintosh сервиса AppleShare.

DAVE обеспечивает доступ Windows-машин к Macintosh-системам, позволяя видеть их через стандартное для Windows 95 окно “Сетевое окружение” посредством утилиты DAVE Sharing.

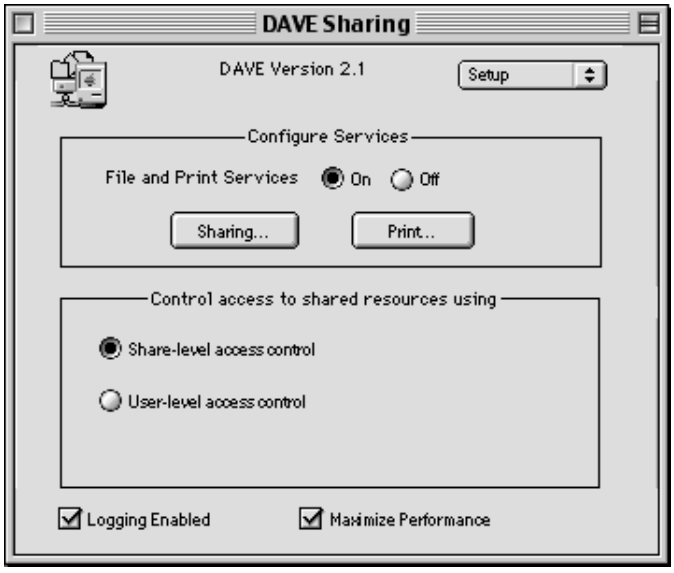


Рис. 4

DAVE поддерживает криптованные пароли форматов LanManager и Windows NT. Пользователи могут управлять разделенными ресурсами на уровнях Share-level и User-level (см. пример окна Dave Sharing на рис. 4).

Пользователи Macintosh могут отдавать по сети разделы и принтеры PostScript другим Macintosh-системам, используя DAVE.

Утилита Popup Message работает напрямую с сервисом Windows WinPopup. Она позволяет пользователям составлять и отправлять послания. При этом сервис приема может быть активизирован или выключен при загрузке системы по желанию пользователя.

DAVE полностью совместим и управляем языком AppleScript.Пользователи имеют возможность писать собственные сценарии монтировки и отмонтировки сетевых разделов, процессов аутентификации в сети и послания сообщений.

2.3.2.3 Взаимодействие станций Macintosh

Естественной третьей компонентой межстанционного сетевого взаимодействия являются традиционные для компьютеров Macintosh сервисы и клиенты Apple File Sharing с использованием протокола AppleTalk Phase 2. Данные средства встроены в стандартный набор сетевых возможностей операционной системы MacOS и по простоте использования подобны аналогичным средствам организации одноранговых сетей Microsoft. Собственно, средства эти представляют собой расширения (extentions) и конрольные панели (control panels), находящиеся в папке System загрузочного раздела компьютеров Macintosh. Так, например, контрольная панель AppleTalk отвечает за привязку протокола AppleTalk к конкретному сетевому интерфейсу (это может быть Ethernet-карта, принтерный и модемный порты, используемые для соединения двух Маков по нуль-модемному кабелю, и контроллер удаленного доступа Apple Remote Access). Общий вид панели

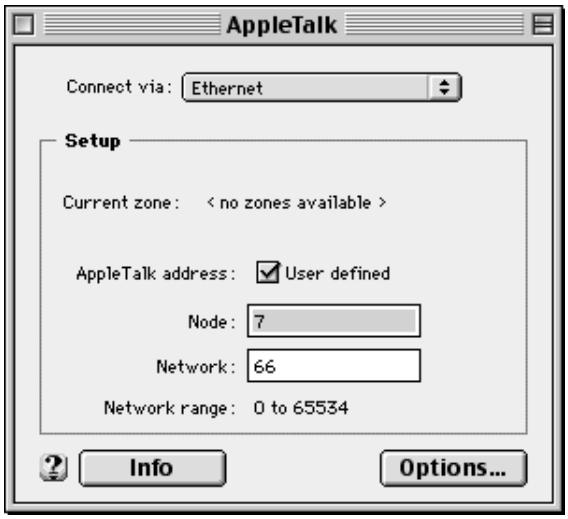


Рис. 5

Рис. 6

представлен на Рис. 5.

В нашем примере проткол AppleTalk привязан к Ethernet, сетевой адрес машины 66.7 (66 - номер сети, 7- номер узла в сети). В окне, приведенном на Рис. 6, отражена информация о сетевом адресе, MAC-адресе сетевой карты, аппаратном роутере и версиях реализации протокола AppleTalk и драйверов.

Ресурсы выделяются пользователем при помощи запуска контрольной панели File Sharing (см. Рис. 7).

Верхняя часть панели определяет имя владельца ресурса, пользовательский пароль и имя компьютера, под которым он регистрируется в сети. Средняя часть определяет включение и выключение разделения файловых ресурсов машины, нижняяя часть, соответственно, управляет включением и выключением разделения программных ресурсов системы.

Закладка Activity Monitor наглядно отображает текущее состояние подключений, имена пользователей, разделенные ресурсы, к которым пользователи получили доступ, и загруженность системы сетевым трафиком.

После общего включения разделения файловых ресурсов необходимо выбрать разделы (папки), которые подлежат выделению в качестве разделяемого сетевого ресурса. Это делается обыной навигацией в Finder и выделением желаемой папки

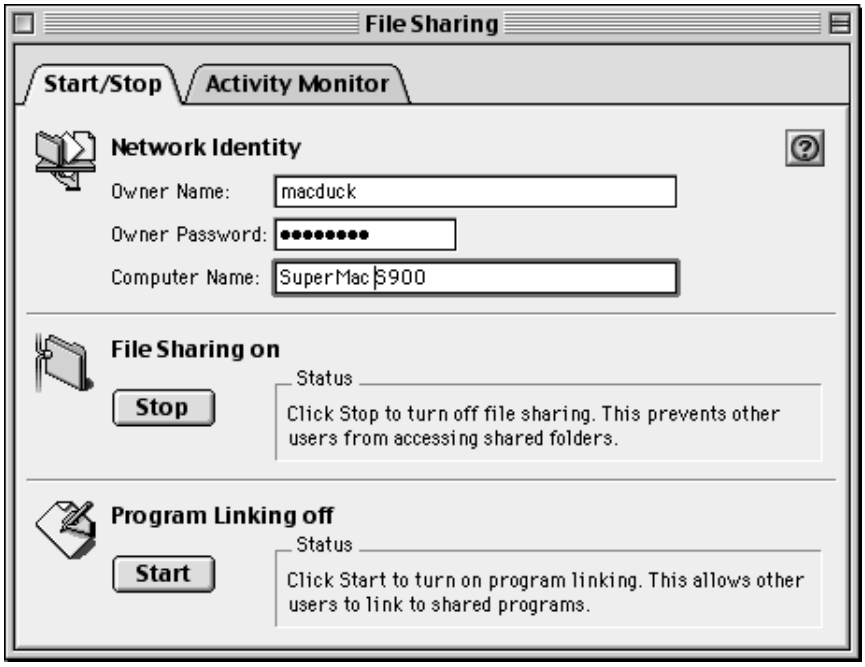


Рис. 7

щелчком мыши.

В данном примере (Рис. 8) отображена закладка Sharing, становящаяся доступной из окна информации о папке (UltraStar: Applications: Hotline: Total

Рис. 8

Downloads). Включенный чекбокс “Share this items and its contents” позволяет определить права доступа к данной папке по всей иерархии пользователей (собственно, владелец папки “Owner”, отдельные права по группам и пользовательским именам и права всех прочих).

При этом закладка Activity Monitor уже упомянутой контрольной панели FileSharing позволяет определить загруженность системы и сетевой карты работой сетевых клиентов с выделенными данной системой ресурсами, увидеть (а при

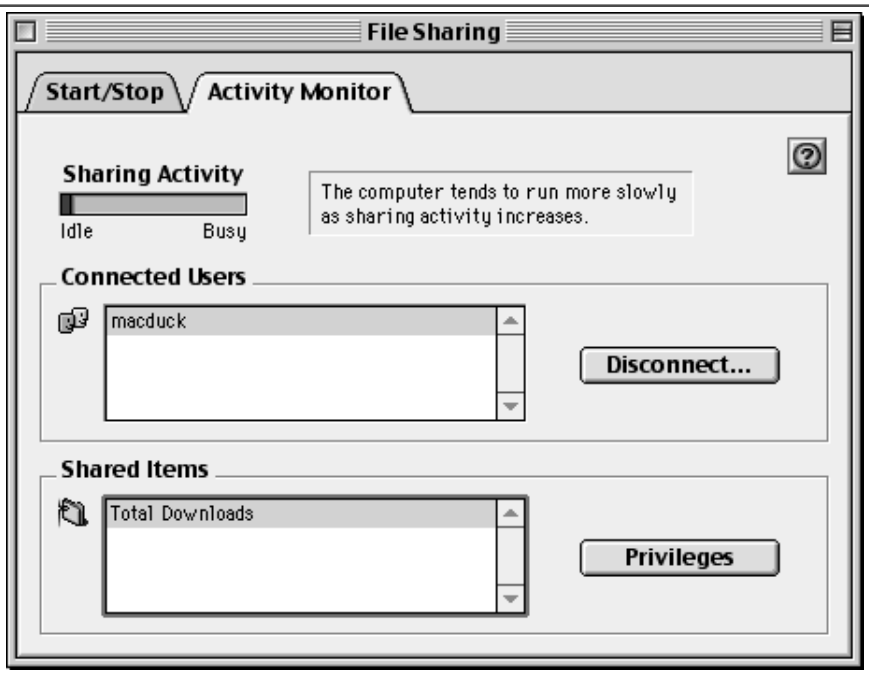


Рис. 9

желании и отсоединить) пользователей, видеть и управлять выделенными ресурсами (см. Рис. 9).

Подключение (монтирование) сетевых ресурсов станций Macintosh производится стандартной ситемной утилиты Chooser, вызываемой из “яблочного

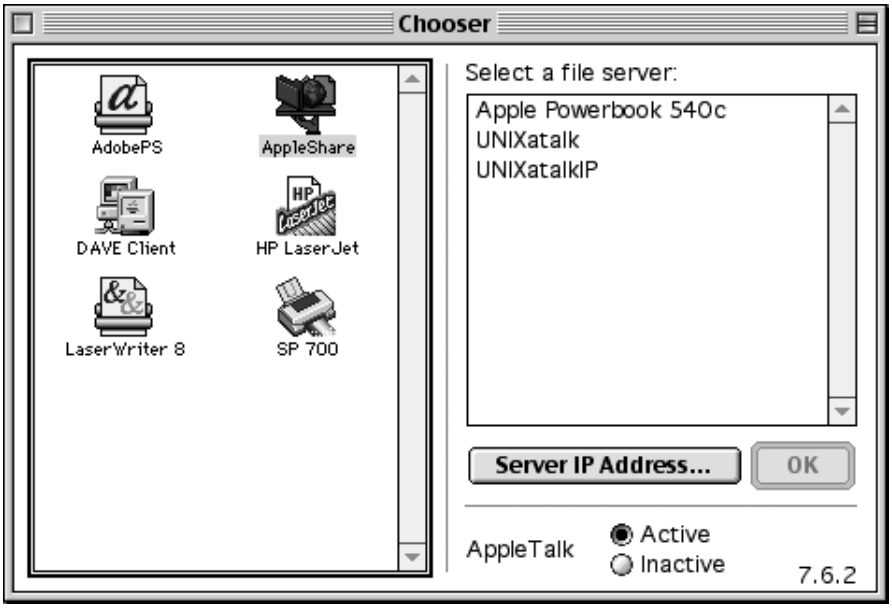


Рис. 10

меню” (Рис. 10).

После выбора в левой стороне окна ресурсов AppleShare, в правой стороне открывается список активных в данный момент серверов AppleShare. Выбрав сервер и введя имя и пароль, пользователь получает доступ к списку разделенных ресурсов данного файл-сервера:

Включив чекбокс, соответствующий определенному ресурсу, пользователь

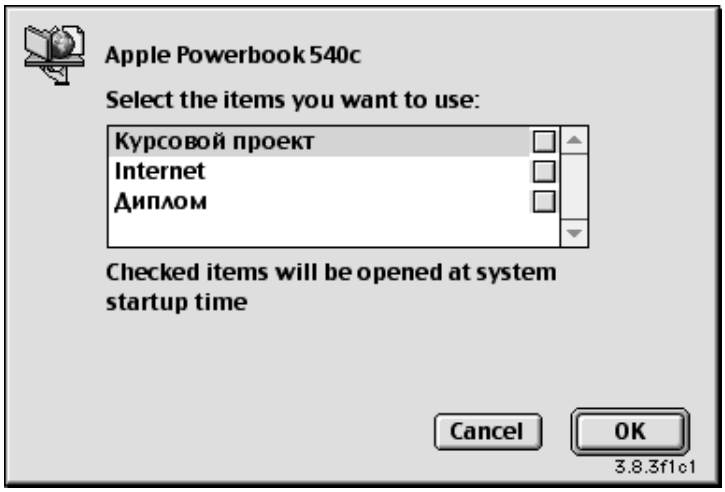


Рис. 11

получает возможность монтировать данный ресурс каждый раз при входе в свою систему автоматически (Рис. 11).

2.3.3 СУБД Oracle и клиенты

СУБД Oracle представляет собой чрезвычайно мощное средство управления базами данных. Oracle широко используется в мире, имеет развитый инструментарий и мощнейшие средства разработки приложений.

На момент принятия решения об установке данного продукта в качестве сервера баз данных существовала большая потребность в централизованной обработке данных и собственных приложениях баз данных, позволяющих автоматизировать рутинный учет первичной бухгалтерской документации (прежде всего, аптеки фирмы), ее последующую обработку и интеграцию в основной поток бухгалтерской отчетности фирмы. Централизация баз данных фирмы позволила бы осуществлять эффективный доступ к информации, который может обеспечить быстрое принятие решений, возможность быстрого реагирования на изменение условий ведения бизнеса и создание на его базе управляемых, производительных и экономически эффективных решений.

Ключевые моменты, предопределившие установку именно Oracle 8:

- Выпуск в недавнем времени Oracle 8.0.5 для платформы Linux;
- Большое количество средств разработки клиентских приложений, таких как Oracle Developer, Oracle Designer/2000;
- Безусловная перспективность и масштабируемость решений на базе Oracle;

Oracle8 обеспечивает единую систему управления базами данных, которая способна удовлетворить требованиям, предъявляемым новыми типами данных сейчас и в последующие годы. С помощью картриджа Oracle ConTex® Cartridge Oracle8 может управлять текстами с той же степенью интеллектуальности, масштабируемости, защищенности и целостности, что и для структурированных данных. А применение картриджа Oracle Video Cartridge™ позволяет записывать, управлять и получать по корпоративным сетям с сервера Oracle на клиентских компьютерах высококачественные аудиозаписи и полноэкранные видеоизображения с высоким разрешением. Разработчики могут легко расширить возможности Oracle8, используя картриджи собственной разработки.

Oracle8 предлагает наиболее передовые и масштабируемые платформы для баз данных с архитектурой “клиент/сервер” и “тонкий клиент”. Oracle Server оптимизирован с целью извлечения преимуществ из конкретных особенностей используемых операционных систем, например, модель потоков в Windows NT, или драйверы post-wait и асинхронный ввод/вывод в системах Unix.

Огромное число “мастеров” Oracle8 облегчает использование всех его возможностей и управление ими.

В состав стандарного пакета Oracle 8.05 for Linux входят продукты:

Distributed Database 8.0.5
Oracle8 Server 8.0.5.1
PL/SQL 8.0.5.1
Java Database Connect (JDBC) 8.0.5, включая драйверы JDBC OCI и JDBC Thin
Object Type Translator 8.0.5

Oracle Call Interface 8.0.5
Oracle Server Manager 3.0.4
Oracle LINUX Installer 4.0.3
Pro*C/C++ 8.0.5.1
SQL*Plus 8.0.5

Обязанностью автора работы являлась собственно установка Oracle 8 на головную машину фирмы, первичная настройка пакета и включение основных стартовых последовательностей сервера в списки соответствующих уровней запуска, предоставление соответствующих полномочий администратору баз данных, выделение дисковых квот для файловых структур и процессов сервера и общее поддержание UNIX-машины как платформы для функционирования сервера баз данных в рабочем состоянии.

2.3.4 Internet/Intranet компоненты

В данном разделе речь пойдет о традиционно сложившихся компонентах сетей, основанных на Internet-технологиях и протоколе TCP/IP. Уже никого не удивляет, что типичные для Internet сетевые службы нашли широкое применение как собственно в сети Internet, так и внутри локальных сетей. Гибкость, удобство использования, стандартизованность и широкая распространенность этих компонент позволяет строить на их базе корпоративные информационные решения (т.н. Intranet).

Большинство программных продуктов, использованных для реализации традиционных сетевых служб входят в состав дистрибутива RedHat 5.2

2.3.4.1 Mail-сервер

Электронная почта или, как ее называют, e-mail уже давно заняла достойное место в мировых средствах коммуникации. Это весьма доступный, быстрый, многофункциональный (многофункциональность заключается в возможности передавать не только текстовые послания, но и любые данные, представимые в цифровом формате) и достаточно надежный способ передачи сообщений. Сеть Internet сделала электронную почту популярной и доступной почти в каждой точке земного шара.

Системы электронной почты весьма сложны, электронную почту (несмотря на простоту самой концепции) реализовать очень трудно. В RFC822 описана форма электронного почтового сообщения для Internet и установлены стандарты на адреса и заголовки.

В частности, спецификации Internet требуют, чтобы каждая организация определяла псевдоним postmaster, который относится к лицам, сопровождающим систему электронной почты. Данное требование реализуется при помощи механизма псевдонимов.

Существует стандарт, определяющий включение в почтовые сообщения объектов мультимедиа. Он называется MIME (Milti-Purpose Internet Mail

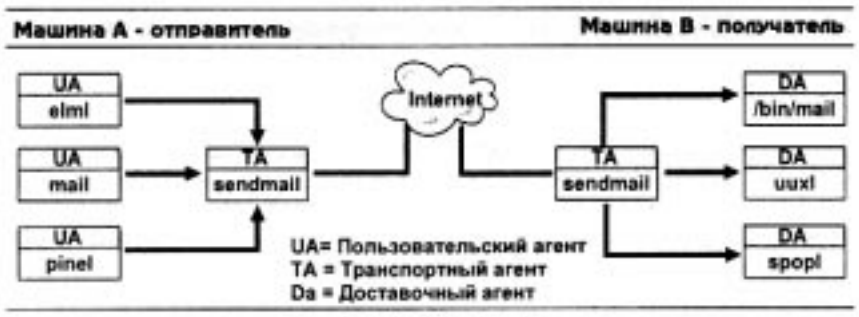


Рис. 12

Extensions - многоцелевое расширение электронной почты для Internet). Этот стандарт поддерживает большинство пользовательсяких агентов.

Система электронной почты сотоит из трех компонентов: пользовательского агента, который позволяет пользователям сотавлять и читать сообщения (в нашем случае это Microsoft Outlook Express, Netscape Communicator на рабочих станциях сети и программы mail, elm, pine, emacs на UNIX-машине), транспортного агента,

который пересылает сообщения с одной машины на другую, и доставочного агента, который помещает сообщения в почтовые ящики пользователей-получателей. Схема работы агентов электронной почты представлена на Рис. 12. Обычно ящики пользователя размещаются на почтовой хост-машине (в нашем случае это, разумеется, все та же Linux-машина.)

Самым первым пользовательским агентом была программа mail, разработанная AT&T. Все современные UNIX-системы включают в свой состав эту программу по умолчанию. Помимо этого, например, Linux RedHat 5.2 имеет в составе свое дистрибуции ряд более удобных GNU пользовательских агентов: elm, pine, emacs для консольного режима и ряд X Window программ.

Пользовательские агенты взаимодействуют с программой sendmail по протоколу SMTP (Simple Mail Transport Protocol - простой протокол транспортировки почты, определенный в RFC821). Демон Sendmail “прослушивает” 25 порт TCP на предмет входящих SMTP-обращений.

Задача транспортного агента - принимать почту от пользовательского агента, интерпретировать адреса получателей и переправлять почту на соответствующие машины для последующей доставки. В качестве транспортного агента выбрана уже упомянутая программа sendmail.

Доставочный агент отвечает за прием почты от транспортного агента и ее доставку соответствующим получателям. Почта может доставляться конкретному лицу, в список рассылки, файлы и даже программу. Для обслуживания получателя каждого типа может понадобиться отдельный агент. Программа mail - доставочный агент для локальных пользователей, демон же pop3d, обеспечивает доставку почты конечным рабочим станциям по протоколу pop3. Данный демон запускается супердемоном inetd каждый раз при попытке пользователя установить pop3 соединение с mail-хостом для доставки почты на локальную станцию.

Механизм адресации локальной почты прост, потому что регистрационное имя пользователя является уникальным идентификатором (имя пользователя@хост). Есть два вида адресов элетронной почты: маршрутно-зависимые (относительные) и маршрутно-независимые (абсолютные). При использовании первого способа требуется, чтобы отправитель знал промежуточные машины, через которые должно пройти сообщение, для того чтобы попасть в пункт назначения. В адресе второго вида просто указывается пункт назначения. UUCP-адреса являются маршрутно-зависимыми, а Internet-адреса от маршрута не зависят. Следует отметить что доставка почты по UUCP редко встречается в наши дни; почтовая система фирмы основана на Internet-адресации, которая в свою очередь предусматривает использование протокола TCP/IP в качестве универсального маршрутизируемого транспортного средства и системы имен (DNS) для преобразования имен хостов в численные значения IP-адресов.

Основу mail-хоста составляет программа sendmail - UNIX-программа, которая принимает почту на mail-хост, разбирает и маршрутизирует e-mail.

2.3.4.2 News-сервер

Второй и, пожалуй, наиболее сложной компонентой является собственный сервер новостей (usenet).

Usenet (или Internet News) - программная система, которая предназначена для рассылки сообщений (“статей”) в множество пунктов по всему миру. Это не сеть определенного типа, а скорее совокупность протоколов, форматов файлов и связей между системами. Фактическое транспортирование сообщения, как правило, обеспечивается средствами Internet или UUCP (последнее средство используется все реже).

Usenet напоминает электронную почту, но имеет одно отличие: все статьи доступны для чтения абсолютно всем. Права же написания в ту и ли иную группу новосте задаются в каждом отдельном случае по-разному. Система Usenet охватывает свыше 5 миллионов человек в 150000 пунктов, а многие тематические группы (“телеконференции”) получают сотни статей в день. Каждая телеконференция предназначена для обсуждения одной темы, которая может охватывать широкий круг проблем или, наоборот, может быть посвящена одной очень узкой теме. Существует более пяти тысяч телеконференций, и некоторые из них получают до тысячи сообщений в день. Программы чтения телеконференций позволяют пользователям подписываться только на интересующие их телеконференции, а некоторые программы еще и обеспечивают возможность создания в рамках телеконференций цепочек тем.

Большинство телеконференций организовано по принципу “без ведущего (модератора)”, т.е. опубликовать статью может каждый. Однако около пяти процентов телеконференций имеют ведущего (модератора). Пользователи, желающие поместить свои статьи в телеконференции с ведущим, должны посылать их ему для просмотра.

Подобно каталогам файловой системы, телеконференции имеют иерархическую структуру. Имя конференции является аналогом полного путевого имени файла, но в качестве разделителя используется не косая черта, а точка. Сходство здесь не случайно. Такая структура имени выбрана из соображения удобства: новости хранятся в дереве каталогов, имена которых формируются путем замены точек косыми чертами. На верхнем уровне находятся базовые классы новостей. Каждый класс включает произвольное число подразделов. Подраздел может заключать в себе другие подразделы.

Система новостей фирмы основана на собственном сервере новостей. Следует сразу оговориться, что данный сервер работает в режиме Intranet, т.е. он не соотносится с конференциями глобального уровня распространения, а наоборот служит в первую очередь для обеспечения функционирования локальных конференций фирмы. Локальные конференции фирмы предназначены для обсуждения оперативных вопросов деятельности и административных вопросов.

Необходимости в подписке сервера на конференции внешних источников не существует; здесь следует упомянуть ограниченные дисковые ресурсы сервера, которые определяют невозможность хранения огромных объемов мировых конференций. Однако пользователи рабочих станций имеют возможность подписаться на интересующие их внешние конференции на news-сервере провайдера (news.podolsk.ru), обладающем значительными дисковыми ресурсами и хорошей пропускной спообностью канала.

Основная конференция фирмы - promexport.local.

Сервер INN функционирует на базе пакета INN 1.7.2. Данный продукт производится и свобоно распространяется Internet Software Consortium. Основа пакета - innd. Эта программа работает как демон, опрашивает порт NNTP (119-й порт TCP) и устанавливает соединения с поставщиками и локальными клиентами для чтения новостей. Прграмма запускается из стартового скрипта /etc/rc.d/init.d/innd в соотвествующих уровнях исполнения. Одновременно с основным демоном в системе обычно функционирует и ряд дополнительных процессов:

```
15220 p0 S   0:00 /usr/lib/news/bin/actived
15222 ?  S   0:01 /usr/sbin/innd -p4 -r -i0 -L
15224 p0 S   0:00 sh /etc/rc.d/rc.news
15228 ?  S   0:00 /usr/lib/news/bin/crosspost -s -
15229 ?  S   0:00 /usr/lib/news/bin/overchan
15230 ?  S   0:00 /usr/lib/news/bin/innfeed -y
15234 ?  S   0:00 -bigmac.px.podolsk.ru ARTICLE
15235 p0 S   0:00 sh /usr/lib/news/bin/innwatch
```

Локальные клиенты обслуживаются демоном nnrpd. Программа expire удаляет из дерева статей устаревшие статьи. Со временем эти статьи удаляются и из базы данных. Процесс innwatch запускается скриптом rc.news, постоянно присутствует в системе вместе с демоном innd и является своеобразным монитором системы INN. Каждые 600 секунд innwatch анализирует суммарную загрузку UNIX-системы, количество свободных файловых ресурсов в каталоге спуллинга. Так, например, если на дисковом пространстве сервера нет свободного места, то данный процесс приостанавливает нормальную работу сервера, автоматически разблокируя его при нормализации ситуации с дисковым пространством. Innwatch автоматически генерирует почтовые сообщения администратору системы в ответ на различные критические события, связанные с функционированием INN.

Программа nntpsend и ее помощник innxmit передают статьи нижестоящим и вышестоящим узлам, причем nntpsend - это препроцессор, а фактическую работу выполняет innxmit.

Утилита ctlinnd позволяет управлять системой INN при помощи посылки определенных сигналов демону innd. При помощи ctlinnd можно создавать и изменять группы, управлять их параметрами, останавливать и запускать сервер и проч.

В файле active содержится перечень телеконференций, о которых известно системе:

```
control 0000000002 0000000001 y
junk 0000000001 0000000001 y
test 0000000003 0000000002 y
to 0000000001 0000000001 y
promexport.local 0000000009 0000000001 y
pxtest 0000000000 0000000001 y
```

Записи в этом файле расшифровываются следующим образом: первый элемент

строки - название ньюс-группы, второй - номер самой новой статьи, а третий - самой старой. Последий элемент означает текущеее состояние телеконференции. Так, буква “y” обозначает, что данная конференция принимается и возможна публикация в нее.

Главные конфигурационные файлы находятся в каталоге /etc/news:

actsync.cfg
actsync.ign
cleanfeed.conf
control.ctl
distrib.pats
expire.ctl
hosts.nntp
hosts.nntp.nolimit
inn.conf
innfeed.conf
innwatch.ctl
moderators
newsfeeds
nnrp.access
nntpsend.ctl
overview.fmt
passwd.nntp

Подробно о назначении этих файлов и индивидуальных настройках сервера новостей будет рассказано в разделе “3.2.7 Конфигурирование сервера новостей”

2.3.4.3 FTP-сервер и клиенты

FTP (File Transfer Protocol, данный протокол описан в RFC 959) - наиболее простая клиент-серверная система, которая позволяет премещать файлы по сети.

Программа ftp устанавливает соединение сервером и приглашает ввести имя пользователя и пароль, а затем предоставляет shell-подобный интерфейс с простейшим набором команд. В сети Internet широко практикуется анонимный доступ к ftp-серверам (сайты бесплатных программ, обновления, документация и проч.)

Надо сказать, что ftp существует практически с момента возникновения UNIX-систем. В настоящее время существует множество ftp-клиентов для различных платформ и операционных систем (Fetch, AnarchyPro, Vicom ftp-client на Macintosh, ftp, поставляемая с Windows95, ncftp, wget, ftp на UNIX-системах). Наряду с клиентами для большинства современных систем реализована и серверная часть.

Интерфейсы этих приложений совершенно разные: от консольных (ftp в UNIX) до графических (Anarchy для MacOS). На рисунке 13 представлен вид

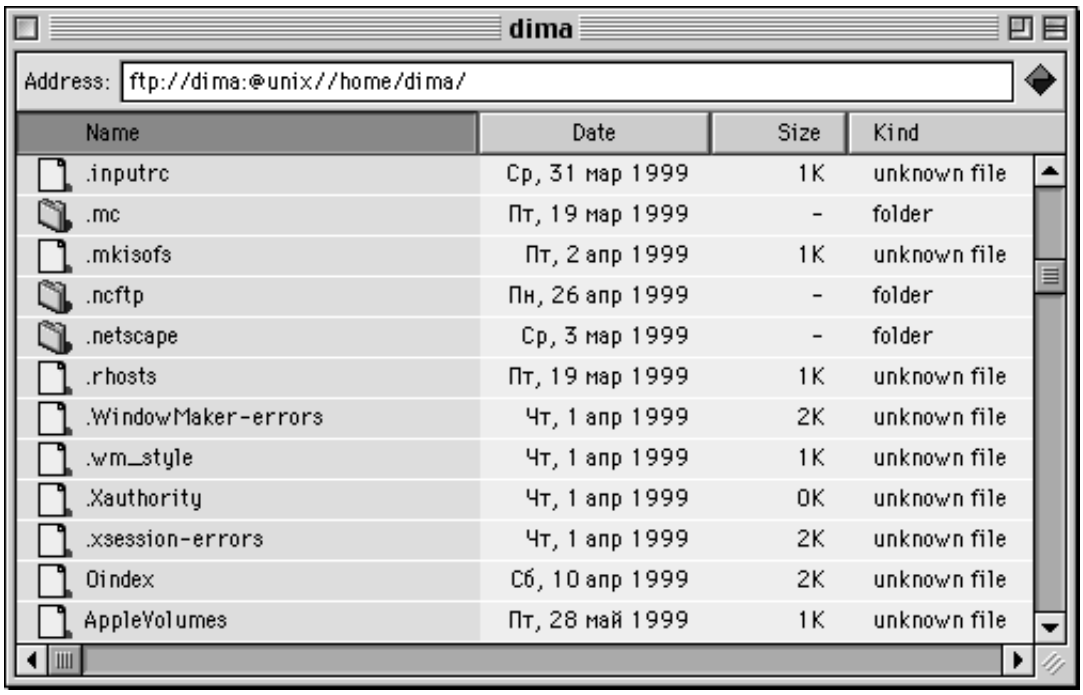


Рис. 13

графического окна Anarchy Pro 3.5. Файлы и директории можно перемещать так же просто, как будто мы имеем дело не с удаленной файловой системой, а директорией на локальном диске.

FTP-сервер фирмы основан на демоне ftpd пакета wu-ftpd 2.4.2-b-18, разработанного в Вашингтонском университе и распространяемом свободно. Данный сервер помимо стандартных свойств ftp-сервера поддерживает концепцию виртуальных ftp-серверов. В заключении добавлю, что демон ftpd запускается супердемоном inetd при обращениях ftp-клиентов.

2.3.4.4 Web-сервер

WWW (World-Wide Web) является, пожалуй, самой популярной технологией современного Internet. Собственно, пространство WWW состоит из огромного числа независимых, но взаимосвязанных серверов. Когда пользователи просматривают “информационное пространство” WWW, они плавно перемещаются от странице к странице в пределах одного сервера и между серверами. Технология WWW основана на концепции документов с гипертестовыми ссылками, обогащенной богатым языком форматирования документов и более удобной моделью доступа. Клиентская же часть представлена т.н. “броузерами” (“browsers”). Среди них надо отметить двух бесспорных лидеров Netscape Navigator и Microsoft Internet Explorer (они существуют для большинства современных систем с графическим интерфейсом). Помимо графических броузеров существуют и текстовые (например, lynx для UNIX, OS/2).

Броузеры позволяют просматривать содержимое гипертекстовых документов, перемещаться по ссылкам, сохранять различные документы и файлы.

Обмен между сервером и клиентом производится по протоколу http (hyper text transfer protocol) с использованием 80-го порта TCP.

Следует отметить, что язык описания гипертекстовых страниц называется html. Вот фрагмент такого файла:

```
<HTML>
<HEAD>
  <META HTTP-EQUIV="Content-Type" CONTENT="text/html">
  <META NAME="Author" CONTENT="Dmitry Seleznev">
      <META NAME="GENERATOR" CONTENT="Mozilla/4.03 (Macintosh; I; PPC) [Netscape]">
  <META NAME="Description" CONTENT="PROMEXPORT">
  <META NAME="KeyWords" CONTENT="PROMEXPORT">
  <TITLE>PROMEXPORT</TITLE>
</HEAD>
<BODY TEXT="#000000" BGCOLOR="#FFFFFF" LINK="#0000FF" VLINK="#551A8B" ALINK="#0000FF">
<CENTER>
<HR WIDTH="100%"></CENTER>
<BR>
<CENTER><TABLE COLS=1 WIDTH="70%">
<TR>
<TD>
<CENTER><A HREF="csc.html"><IMG SRC="csc.jpg" ALT="Компьютерный сервис-центр" BORDER=0></A></CENTER>
```

Последние две строки заключают в себе как раз гипертекстовую ссылку на другой html-документ, загрузить который можно либо нажатием мышкой на соответствующем графическом изображении, задаваемом в файле csc.jpg (вариант графического броузера), либо выбором текстовой строки “*Компьютерный сервис-центр*” в текстовом броузере.

Концепция HTML-документов расширяется такими компонентами как CGI,

Java, ActiveX, позволяющими организовывать активное взаимодействие клиента с компонентами сервера посредством гипертекстовых документов. Это получило широкое использование при организации доступа к базам данных, переменным структурам и проч. Компоненты взаимодействия открывают широкие перспективы для применения WWW-технологий в Intranet-сетях.

Собственный Web-сервер фирмы построен на системе Apache-1.3.4rus. Apache - один из самых популярных Web-серверов. Он разрабатывается и поддерживается Apache Group и распространяется в рамках лицензии GNU.Он содержит обширный API для расширения с помощью модулей, множество способностей и большое количество подключаемых модулей; очень гибок, работает на большом количестве популярных операционных систем, имеет активное сообщество пользователей.

В проекте используется русская версия Apache, поддерживаемая российскими учасниками Apache Group. Главным достоинством русской версии является возможность автоматического распознавания кодировки клиентской стороны с последующим перекодированием страниц в требуемую кодировку. Так, например если HTML-ресурсы сервера хранятся в кодировке KOI8, а к серверу обращается броузер Windows-машины, то Apache на лету перекодирует страницу в кодироку 1251 и “отдает” содержимое страницы клиентской стороне.

Данный продукт был сгружен с одного из анонимных российских ftp-серверов поддержки Apache в виде исходных текстов. Подробно о настройке Web-сервера будет рассказано в главе “3.2.2 Установка и настройка Apache”.

2.3.5 Традиционные сервисные компоненты UNIX

Следует отметить, что деление сервисов UNIX на сервисные и общепользовательские чисто условно. Однако в сети фирмы системы X Window и telnet занимают особое положение. Основное их предназначение - помощь в администрировании и управлении сервером (который, чаще всего, лишен монитора) стандартными и расширенными средствами UNIX и специальным программным обеспечением, установленным на клиентских станциях.

2.3.5.1 Система X Window

Система X Window - большая и сложная компонента UNIX-систем, обеспечивающая UNIX графическое окружение. X Window давно уже стала промышленным стандартом, и практически каждая UNIX-система так или иначе использует ее. Свободно распространяемый порт MIT X Window System версии 11, релиз 6 (т.н. X11R6) для UNIX-систем на основе 80386/80486/Pentium развивается командой программистов, изначально возглавляемой Дэвидом Вексельблатом (David Wexelblat). Данный продукт, известный как XFree86, доступен для System V/386, 386BSD, и других x86-вариаций UNIX, включая Linux. Продукт включает в себя исполняемые файлы, вспомогательные файлы, библиотеки и утилиты.

Система X Window является клиент-серверной. Сервер - собственно графическая оболочка, клиенты - приложения, написанные под X Window, взаимодействующие с X-сервером по протоколу X11R6. Так, например, клиенты - control panel, xterm. X-сервер может находиться как на локальной машине, так и на удаленной. Существует бесконечное количество т.н. Window Manager, здесь следует упомянуть KDE, Enlightment, AfterStep, xvwm95 (см. примеры различных менеджеров в Приложениях 5 и 8).

Помимо настройки X Window собственно на сервере Linux, есть необходимость использования удаленного X-сервера, например, на рабочих станциях Macintosh. Для этой цели на станциях Macintosh установлен eXodus 6.1.2 фирмы WhitePine Inc. Для удаленного запуска X-компонент используется специальное приложение Xdm, запускаемое в режиме демона. Xdm управлет наборами предоставляемых X-дисплеев согласно разработанному X-консорциумом стандарту XDMCP (X Display Manager Control Protocol). Этот демон существенно упрощает управление удаленным доступом, обеспечивая аутентификацию пользователей, устанавливая параметры и запуская сессии.

В Приложении 4 показан процесс аутентификации удаленного пользователя X Window, управляемый Xdm.

Система X Window и соответствующие графические прграммные средства активно используются автором для управления и администрирования системы, управления бюджетами пользователей и дисковым пространством сервера.

2.3.5.2 Telnet

Другим важным средством удаленного администрирования и управления является telnet. Telnet также является клиент-серверным приложением, позволяющим получать удаленный доступ к командному Shell системы UNIX по TCP/IP. С точки зрения UNIX каждое telnet-соединение обеспечивается индивидуальным процессом in.telnetd. Клиентское программное обеспечение (например MacBlueTelnet, BetterTelnet и проч.) устанавливается на рабочие станции, имеющие TCP/IP-доступ к серверу. При этом клиентская машина превращается в текстовый терминал, каждая консоль которой функционирует независимо от другой. Процесс удаленного соединения выглядит так: клиент telnet посылает свой запрос на порт 23/tcp IP-интерфейса сервера. Супердемон inetd “слышит” запрос, передает управление демону сервера telnet in.telnetd, который в свою очередь вызывает процесс login. После аутентификации пользователя предоставляется доступ к соответствующему shell, описанному в файле /etc/password.

Следует отметить, что telnet - универсальная компонента любой UNIX-системы, позволяющая используя даже слабый канал связи полноценно управлять системой и производить операции администрирования.

2.3.5.3 Сервер DNS

Для человека, поработавшего даже небольшое время в сети, становится совершенно естественным, что у каждого компьютера, подключенного к интернет, есть свое название, имя, которое легко запомнить. Система, которая позволяет нам использовать эти привычные для человека имена, избегая других неудобных способов "маркировки" компьютеров, называется DNS (Domain Name System, доменная система имен).

В интернете существует, вообще говоря, два основных способа адресации компьютеров. Первый - численный (или IP-адрес; например, 195.133.132.66), второй - символьный (ns.podolsk.ru). DNS создана для того, чтобы поставить в соответствие один способ другому.

DNS определяет:

- иерархически организованное пространство имен машин;
- таблицу машин, реализованную в виде распределенной базы данных;
- библиотечные подпрограммы запросов этой базы данных;
- усовершенствованные средства маршрутизации электронной почты;
- протокол обмена информацией об именах;

Чтобы облегчить упорядочивание наименований, вся структура компьютерных имен устроена таким образом: есть отдельные уровни (домены), которые могут включать в себя как другие поддомены, так и имена компьютеров. Все названия должны состоять только из латинских букв, цифр и, может быть, знака "минус". Отдельные уровни доменов разделяются точкой.

Типичное полное доменное имя компьютера может выглядеть так: computer3.otdel-5.firma.msk.ru В этом примере такой адрес мы присвоили компьютеру номер три, который стоит в отделе (номер сообразите сами) фирмы с изысканным английским названием "firma", которая находится в Москве ("msk"), в России ("ru"). Локальным именем компьютера (hostname) здесь является "computer3", а ".ru" обычно называется доменом верхнего уровня. Домен msk.ru, соответственно, является доменом второго уровня; firma.msk.ru - третьего...

В пределах домена каждого уровня есть группа людей, которые отвечают за этот домен. Они могут добавлять имена вновь появившихся компьютеров, менять их или удалять. И по сути дела, то, как будет называться та или иная машина зависит от того, что им подскажет фантазия написать в конфигурационном файле DNS.

Тот, кто имеет право администрировать домен, может делать изменения только в пределах этого домена. Например, системный администратор отдела N5 может, скажем, изменить имя "computer3" на "computer4" или на что-то более человеческое, например, назвать этот компьютер "julia" (Тогда полный его адрес станет julia.otdel-5.firma.msk.ru). Но для того, чтобы изменить имя домена 4-го уровня "otdel-5", администратору придется просить об этом у сисадмина фирмы (если, конечно, это не одно и тоже лицо). Процедура получения имени, например, в зоне .ru или .com называется регистрацией домена. Конечно, каждая компания, подключающаяся к интернет, стремится зарегистрировать как можно более естественное и легкое для запоминания имя. Так, для "Microsoft inc." логично

зарезервировать домен microsoft.com.

Доменов верхнего уровня очень немного - всего около 250. Большая часть из них - так называемые, географические домены. Например, .de (Deutschland, Германия), .ru (Russia, Россия), .iq (Iraq, Ирак). Оставшиеся негеографические домены верхнего уровня - .com (для коммерческих компаний), .net (для сетевых ресурсов), .edu (образовательные учреждения), .mil (военные организации), .org (некоммерческие организации), .gov (правительственные ведомства), .int (интернациональные корпорации).

Если кому-либо бы захотелось зарегистрировать еще один домен верхнего уровня, то потребовалось для этого предоставить такие серьезные обоснования, что гораздо проще было бы организовать свое маленькое государство и для него уже получитьгеографический домен.

К началу 1998 года во всем интернете зарегистрировано около 30 миллионов хостов. Распределение по доменам верхнего уровня приведено ниже:

домен	число хостов	описание
com	8201511	Commercial
net	5283568	Networks
edu	3944967	Educational
jp	1168956	Japan
mil	1099186	US Military
us	1076583	United States
de	994926	Germany
uk	987733	United Kingdom
ca	839141	Canada
au	665403	Australia
org	519862	Organizations
gov	497646	Government

Россия в этом списке находится на 28-м месте. Под доменом .ru зарегистрировано около 100 тысяч компьютеров. А в Антарктике (.aq), как оказывается, нет ни одного компьютера, подключенного к интернет.

При текущем уровне развития коммуникаций в России, все больше компаний встают перед необходимостью подключения своих локальных сетей к Интернет. Если опустить все организационные и коммерческие вопросы подключения, то в техническом отношении этот процесс сводится к следующей последовательности действий:

- Подключение к провайдеру. Физическое подключение может быть выполнено многими различными способами: от обычного модема до радиосетей и оптоволокна. Способ подключения и, соответственно, оплата оговариваются непосредственно с провайдером. (В нашем случае сеть фирмы подключена к каналу провайдера через выделенную линию)

- Получение численных (IP) адресов. Для того, чтобы собственные компьютеры, подключенные к Интернет, стали доступны, необходимо выделить для них уникальные численные адреса. Обычно в договоре на подключение к интернет указывается, сколько и каких адресов отдается в пользование компании.

Формат IP-адресов такой: четыре числа от 1 до 255, отделенных точками. Например, 193.124.134.101 - IP-адрес какого-то компьютера в сети. (Провайдером выделена 16 адресов с сети 195.133.132.)

- Настройка DNS. Это означает, что при выделении собственных IP-адресов следует правильно сконфигурировать систему имен и корректно настроить работу серверов с этими именами.

- Процедура регистрации доменного имени. Она сильно различается для различных доменов верхнего уровня, и может быть как бесплатной, так и по оплате. Фирма получила поддомен px.podolsk.ru в домене podolsk.ru, делегированного региональному провайдеру.

- Дальнейшая установка программного обеспечения на компьютеры, требующего явного указания доменного имени (например, web-сервера). Так для традиционного удобства внешних пользователей Web-сервера фирмы было выбрано имя www.px.podolsk.ru.

Традиционной системой, обеспечивающей реализацию всех компонент DNS на UNIX-машинах является BIND (Berkley Internet Name Domain), свободно поставлемой с большинством UNIX-систем. В состав дистрибутива Linux RedHat 5.2 входит BIND версии 8.1.2., которая и успешно используется. Система BIND состоит из трех компонент:

- демон named, который отвечает на запросы;
- библиотечные программы, которые отвечают на запросы машин, используя DNS;
- командные интерфейсы пользователей DNS: dip, nslookup, host

Прежде всего, для полноценной работы DNS необходимо два или больше компьютеров, так называемых, name-серверов, которые независимо друг от друга подключенны к интернет (лучше, если они будут находиться в разных сетях или даже разных странах). Такая структура обеспечит неизменную работу системы преобразования символьного адреса в числовой и обратно, даже если какое-то время некоторые из этих компьютеров будут недоступны по сети. На таких компьютерах запускается специальная программа-демон named, которая обрабатывает запросы на преобразование адресов и отвечает на них. Настроить DNS - означает корректно написать конфигурационные файлы named. Подробно конфигурация системы BIND, установленной на главной Linux-машине предприятия будет рассказано позже в соотвествующем разделе.

Name-сервера бывают primary и secondary. Иногда их называют первичными и вторичными, а также master и slave. Primary name-сервер может быть только один. На нем хранится вся информация о доменах, и если происходят изменения, то конфигурация правится только на нем. Secondary name-серверов может быть несколько, но обычная практика - один secondary nameserver. Дополнительные вторичные name-сервера служат для повышения скорости расшифровывания адреса и для повышения устойчивости такого преобразования. Для небольших сетей три и больше вторичных name-сервера - это уже излишество. Secondary name-

сервера с заданной периодичностью в автоматическом режиме считывают текущую конфигурацию с primary-сервера. Заметим, что один и тот же компьютер может одновременно являться primary-сервером для одних доменов и secondary nameserv-er'ом для нескольких других.

2.3.6 Сервер удаленного доступа

Сервер удаленного доступа выполняет две функции:

- доступ сотрудников фирмы к ресурсам сети, обеспечивающий работу над проектами на удаленных рабочих местах (например, из дома);

- удаленное администрирование сети и, в частности, управление центральной Linux-машиной;

В частности, сервер удаленного доступа позволяет приблизиться к концепции распределенного офиса, которая заключается в снижении нагрузки на центральный офис предприятия перенося часть рабочих в дома сотрудников. Данная концепция находит все большее распространение в Европе и Соединенных Штатах.

Помимо удаленных рабочих мест весьма актуальна проблема удаленного администрирования. В этом аспекте UNIX предоставляет, пожалуй, наиболее гибкие возможности: практически все действия по управлению системой с консоли платформа UNIX поддерживает по умолчанию. Управление системными и пользовательскими процессами, конфигурирование, настройки сетевых интерфейсов, пользовательские бюджеты, даже пересборка ядра машины и ее перезапуск доступны, если есть хотя бы низкоскоростное соединение по коммутируемой линии.

Сервер удаленного доступа фирмы использует одну из телефонных линий офиса и установленный на Linux-машине внутренний модем US Robotics Sportster 33600. Данная линия связи в штатных ситуациях используется лишь в ночное время.

Следует отметить, что дистрибутив RedHat 5.2 содержит все необходимые программные средства, необходимые для организации сервера удаленного доступа.

К эти средствам относятся: комплект mgetty-1.1.14, позволяющий работать с факс-модемами, системный демон initd, вызывающий демон mgetty, системный шедулер crond и комплект средств протокола PPP (Point-to-Point Protocol, см раздел “2.2.4 Протокол PPP”).

Работа mgetty совершается в традиционной для UNIX манере. Конфигурационный файл процесса initd содержит строчку запуска демона mgetty (это необходимо для постоянного поддержания mgetty в рабочем состоянии) :

```
S2:35:respawn:/sbin/mgetty -s 38400 -n 3 -i /etc/mgetty.issue -m '"" AT&F1M0 OK'
 /dev/ttyS2
```

S2 указывает, на то, что процесс mgetty работает на порту ttyS2, ресурсы которого принадлежат модему. Опция -s означает скорость захвата (“залочки”) порта. Опция -n указывает, что модему необходимо поднимать трубку после третьего входящего звонка. Дальнейшее содержимое строки - текстовый issue-файл, который предшествует запросу на аутентификацию в терминальном окне пользователя:

```
[root@unix /root]# less /etc/mgetty.issue
  PROMEXPORT Remote Access Service
       UNIX (Linux)
If problems occured contact Dmitry Seleznev
```

Это приветствие с указанием информации об удаленном хосте выводится удаленному пользователю каждый раз при установлении модемного соединения.

Последний параметр строки содержит инициализационные команды модему. О своем “праве” поднимать трубку и пытаться установить модемное соединение демон mgetty узнает из факта существования флагового файла /etc/nologin.ttyS2. Если файл существует, то демон mgetty находится в пассивном состоянии и никаких действий не предпринимает. Если же файлового флага в каталоге /etc нет, то после третьего звонка mgetty попытается установить соединение с удаленным модемом.

Данный флаговый файл порождается и удаляется системным процессом cron в определенное время. Таким образом организуется удаленный доступ в определенное время суток и различные дни недели.

Помимо собственно терминального доступа к консоли Linux-машины для организации полноценного сетевого взаимодействия удаленных пользователей с компьютерами сети используется протокол PPP (См. раздел 2.2.4)

2.4 Параметры сетевой ОС Linux

2.4.1 Требования к ядру

Ядро Linux (как и всех UNIX-систем) - важнейшая его компонента, своеобразное “сердце” системы. Ядро vmlinuz размещается обычно к корне основной файловой системы, хотя это совершенно необязательно, т.к. его местонахождение задается в конфигурационном файле программы начальной загрузки lilo (LInux LOader).

Поскольку в состав большинства дистрибуций Linux входят исходные тексты ядра и системных компонент, то возможна (и даже желательна) перекомпиляция ядра и модулей с целью получения оптимального по своим функциональным возможностям и минимального по размеру ядра, наиболее соответствующего конкретным аппаратным средствам и требованиям к Linux-системе.

Конфигурирование будущего ядра выполняется поредством команды /usr/src/linux/make menuconfig (а при отсутствии библиотек ncurces - при помощи /uusr/src/linux/make config; конфигурирование при этом выполняется в терминальном режиме вопросов и ответов).

Меню конфигурационной утилиты включает в себя несколько больших разделов. Многие из опций имеют три возможных состояния: включено, выключено и реализовать в виде модуля. Рассмотрим каждый из разделов и обоснуем соответствующие состояния опций ядра и модулей.

1. Code maturity level options. Данный раздел имеет лишь один подпункт Prompt for development and/or incomplete code/drivers (Разрешение на на использование драйверов и кода, не прошедшего полное тестироваие.) Включение этой опции означает согласие на использование альфа и бета-версий программ, драйверов, библиотек и экспериментальных фрагментов кода, поставляемого в рамках GNU-лицензии. Автор посчитал необходимым включить данную опцию.

2. Loadable module support. Развернувшееся подменю состоит из трех позиций:
Enable loadable module support (Разрешить поддержку загружаемых модулей) - выбрано разрешить.
Set version information on all symbols for modules - данная опция позволяет использовать старые модули после сборки нового ядра.
Kernel daemon support (e.g. autoload of modules) - Поддержка демона ядра (автозагрузка модулей). Включение данной опции разрешает автоматическую загрузку и выгрузку модулей демоном ядра kerneld. Включение опции необходимо.

3. General setup (Основные опции). Раздел включает следующие пункты:

Kernel math emulation (Эмуляция сопроцессора). Поскольку ОС Linux инсталлирована на компьютер с процессором Pentium, опция выключена.
Networking support (Сетевая поддержка). Разумеется, автор счел необходимым включить данную поддержку.
Limit memory to low 16MB (Ограничить использование памяти выше 16 Мб). Необходимость во включении данной опции существует лишь при работе с

устаревшими материнскими платами, у которых возможны проблемы с использованием областей памяти выше 16 мегабайт. Опция выключена.
PCI bios support (Поддержка материнских плат, использующих шину PCI). Опция включена.

System V IPC. Опция отвечает за поддержку межпроцессорных взаимодействий (Inter Process Communication), набора системных функций и вызовов, обеспечивающих процессам возможность синхронизации и обмена данными. Опция включена.

Kernel support for a.out binaries - Поддержка выходного формата ассемблера (a.out); поддержка включена.

Kernel support for ELF binaries - поддержка ELF (Executable and Linkable Format), формата библиотек и исполняемых файлов межплатформенного взаимодействия. Опция необходима.

Kernel support for JAVA binaries - поддержка скомпилированных Java-приложений. Необходимости в данной опции нет.

Compile kernel as ELF - if your GCC is ELF-GCC - производить компиляцию ядра в ELF-формате. Опция включена.

Processor type - Выбор типа процессора. Для использования преимуществ процессора Pentium указана опция Pentium.

Handle buggy SMP BIOSes with bad MTRR setup - опция используется в некоторых мультипроцессорных системах.

4. Floppy, IDE, and other block devices - опция поддержки дисководов, IDE-устройств и других блок-ориентированных устройств. Рассмотрим лишь наиболее важные входящие сюда опции:

Normal floppy disk support - поддержка обычных дисководов. Опция включена.

Enhanced IDE/MFM/RLL disk/cdrom/tape/floppy support - улучшенная поддержка жестких дисков форматов IDE, MFM, RLL, CD ROM, ленточных накопителей и дисководов. Опция включена для оптимального использования современных жестких дисков.

Initial RAM disk (initrd) support - возможность использования виртуального диска при начальной загрузке системы. Опция включена.

Intel 82371 PIIX (Triton I/II) DMA support - поддержка каналов прямого доступа к памяти набора микросхем Intel Triton. Опция включена.

Loopback device support - опция позволяет монтировать образы файловых систем как реальную файловую систему. Поддержка реализована в модуле.

Parallel port IDE device support - поддержка IDE-накопителей, подключенных к параллельному порту. Опция выключена.

5. Networking options (Сетевые опции). Данный раздел представлен следующим списком:

Network firewalls - включение поддержки файрволлинга, системы фильтрации пакетов как средства безопасности внутренней сети. Опция включена.

Network aliasing - включение опции позволяет иметь несколько виртуальных IP-адресов на одном сетевом устройстве нижнего уровня. Опция включена.

TCP/IP networking - разрешение протокола TCP/IP. Опция включена.

IP: forwarding/gatewaying - включение IP-форварда и функции шлюза. Опция включена.

IP: multicasting - разрешение IP-многопоточности. Опция включена.

IP: syn cookies - включение защиты от аттак SYN-переполнения. Опция включена.

IP: firewalling - включение поддержки IP-файрволлинга, системы фильтрации IP-пакетов как средства безопасности внутренней сети. Опция включена.

IP: firewall packet logging - поддержка сообщений файрволла. Опция включена.

IP: masquerading - IP-маскарадинг. Опция включена.

IP: ICMP masquerading - поддержка макарадинга ICMP-пакетов. Опция включена.

IP: accounting - опция, обеспечивающая учет IP-пакетов. Опция включена.

IP: optimize as router not host - оптимизация ядра для машины-роутера, а не конечного хоста. Функция выключена.

IP: tunneling - опция, разрешающая инкапсулирование различных протоколов в IP-пакеты. Опция реализована в модуле.

IP: multicast routing (EXPERIMENTAL) - экспериментальная поддержка возможности мультипоточной маршрутизации. Опция включена.

IP: aliasing support - возможность использования различных IP-адресов на одном адаптере.

IP: PC/TCP compatibility mode - опция обеспечивает возможность подключения некоторых нестандартных telnet-клиентов. Опция выключена.

IP: Reverse ARP - включение возможности реверсивных ARP-запросов. Опция

включена.

IP: Drop source routed frames - разрешение пакетов с маршрутизацией от источника. Опция включена.

IP: Allow large windows (not recommended if <16Mb of memory) - опция расширенной буферезации IP-пакетов.

The IPX protocol - поддержка протокола IPX. Опция включена.

Appletalk DDP - поддержка протокола AppleTalk. Опция включена.

6. SCSI support (Поддержка SCSI устройств). Выбранный для установки Linux компьютер не имеет SCSI-контроллеров, поддержка отключена.

7. Network device support (Поддержка сетевых устройств). Раздел имеет опции поддержки различных сетевых карт, протоколов PPP, SLIP. Включена поддержка NE2000-совместимой карты, 10 и 100-мегабитных Ethernet-стандартов.

8. ISDN subsystem (Поддержка ISDN). Сервер не оснащен ISDN-адаптером, использование каналов ISDN в обозримом будущем не предвидится. Поддержка выключена полностью.

9. CD-ROM drivers (not for SCSI or IDE/ATAPI drives) (Поддержка нестандартных накопителей CD ROM). В офисе используются только CD ROM'ы стандартов SCSI и ATAPI, использование накопителей иных стандартов не предвидится. Опции раздела полностью отключены.

10. Filesystems (Файловые системы). Раздел включает в себя поддержку дисковых квот (включено), различных файловых систем (включена модульная поддержка формата CD ROM ISO9660, файловой системы Linux ext2fs, файловой системы MS DOS FAT, файловой системы Windows95 VFAT, файловой системы OS/2 HPFS, файловых систем SMB, NFS, NCP, System V), поддержку виртуальной файловой системы /proc (включена), поддержку различных кодовых страниц (ISO8859, CPP, KOI8-r). При выборе кодовых страниц принято во внимание возможность монтирования разделов с русскими названиями файлов.

11. Character devices (Устройства ввода-вывода). В данном разделе существует возможность выбора опций поддержки различных портов, многопортовых карт, мыши, накопителей на магнитной ленте (стриммеров), устройств бесперебойного питания и системного таймера. Включена поддержка сериальных портов, параллельного порта принтера и раширенного управления системным таймером.

12. Sound (Поддержка звука). Сервер не оснащен звуковой картой, и установка ее не предусматривается. Опции раздела полностью отключены.

13. Kernel hacking (Расширенный доступ к параметрам ядра). Раздел представлен единственной опцией Kernel profiling support, позволяющей производить

расширенный анализ загруженности ядра различными процессами при помощи обращения с файлу виртуальной системы /proc/profile. Данная опция выключена.

Оптимизация ядра для конкретной системы позволила уменьшить его размер до 428 килобайт. Следует отметить, что процедура конфигурирования ядра требует достаточно хорошей осведомленности системного администратора LINUX относительно функциональных требований к системе, находящейся в его ведении, и определенных профессиональных навыков.

2.4.2 Необходимые компоненты

Огромное количество программных продуктов входит в состав дистрибутива Linux RedHad 5.2. Достаточно заметить, что полная инсталляция всех его компонент заняла бы порядка 700 Мб дискового пространства.

Инсталлятор системы предложил выбрать компоненты, разделив их по своей функциональной направленности:
Printer Support (Поддержка принтера), X Window system (Система X Window), Mail/WWW/News Tools (средства электронной поты, WWW и новостей), DOS/Windows connectivity (Средства взаимодействия с Windows и DOS), File Managers (Менеджеры файлов), Graphics Manipulation (Графические средства), X Games (Игровые программы под X Window), X Multimedia support (Мультимедийная поддержка X Window), Console multimedia (Мультимедийная поддержка консольных режимов), Networked workstation (Сетевая станция), DialUp Workstation (Станция удаленного доступа), News Server (Сервер новостей), NFS Server (Сервер сетевой файловой системы), SMB/Samba connectivity (Программный пакет Samba), IPX/Netware connectivity (Комплект поддержки протокола IPX и сетей Netware), Anonimous FTP server (Анонимный FTP-сервер), Web server (Web-сервер), DNS Name Server (пакет сервера DNS), Postgres (SQL) server (Сервер баз данных PostgreSQL), Network Managment Workstation (средства станции управления сетью), TeX document formating (Издательский набор TeX), Emacs (редактор Emacs), Emacs with X Window (редактор Emacs под X Window), C Development (Средства разработки на языке C), C++ Development (Средства разработки на C++), X Development (средства разработки X Window приложений), Extra Documentation (специальная документация).

Наряду с делением пакетов по функциональной направленности существует возможность индивидуального выбора продуктов, расположенных по алфавитному порядку и разбитых на группы Applications (Приложения), Base (Базовый набор системы), Daemons (Демоны), Development (Средства разработки), Documentation (Документация), Extensions (Расширения), Games (Игровые программы), Libraries (Библиотеки), Networking (Сетевые средства), Shells (Оболочки), Utilities (Утилиты), X11 (Компоненты системы X Window). Каждая из перечисленных групп представлена огромным количеством программ, библиотек, исходных текстов, документации и проч. Например, группа Applications, в свою очередь, делится на подгруппы Mail (почта), Math (математические программы), Editors (Редакторы), News (Новости), Graphics (Графика), Publishing (Издательство), Networking (Сетевые приложения), Communication (Средства коммуникации).

Подобный индивидуальный выбор компонент позволяет опытному системному администратору установить лишь необходимые для конкретной станции продукты и сэкономить дисковое пространство.

К числу необходимых компонент автором отнесены следующие продукты:

- Компоненты ядра (kernel -- Version 2.0.36, 4,726K, kernel-headers -- Version 2.0.36, 1,551K, kernel-ibcs -- Version 2.0.36, 219K, kernel-source -- Version 2.0.36, 30,727K);

- Сетевые программы (tcp_wrappers -- Version 7.6, 242K, net-tools -- Version 1.46, 190K, anonftp -- Version 2.6, 1,046K, arpwatch -- Version 2.1a4, 117K, lynx -- Version 2.8.1, 2,031K, ncftp -- Version 2.4.3, 170K, tcpdump -- Version 3.4, 215K, wget -- Version 1.5.2, 352K, ipfwadm -- Version 2.3.0, 85K);

- Коммуникационные приложения (minicom -- Version 1.82, 289K);

- Редакторы (ed -- Version 0.2, 103K, emacs -- Version 20.3, 17,337K, emacs-X11 -- Version 20.3, 5,839K, joe -- Version 2.8, 282K);

- Графические компоненты (ghostscript -- Version 4.03, 2,737K, ghostscript-fonts -- Version 4.03, 3,679K);

- Компоненты и приложения X Window (gimp -- Version 1.0.1, 7,347K, netscape-common -- Version 4.06, 6,601K, netscape-navigator -- Version 4.06, 6,795K, xterm-color -- Version 1.1, 191K, xv -- Version 3.10a, 4,480K, gnome-core -- Version 0.20.1, 1,019K, gtk+ -- Version 1.0.6, 1,175K, gtk+-devel -- Version 1.0.6, 1,688K, gnome-linuxconf -- Version 0.14, 96K, Xconfigurator -- Version 3.79, 266K, xmailbox -- Version 2.5, 30K, AfterStep -- Version 1.5, 3,384K, AfterStep-APPS -- Version 1.5, 715K, WindowMaker -- Version 0.20.1, 2,977K, fvwm -- Version 1.24r, 550K, fvwm2 -- Version 2.0.46, 1,471K, fvwm2-icons -- Version 2.0.46, 599K, wmakerconf -- Version 1.1.1, 515K, X11R6-contrib -- Version 3.3.2, 446K, XFree86-100dpi-fonts -- Version 3.3.2.3, 1,228K, XFree86 -- Version 3.3.2.3, 12,040K, XFree86-100dpi-fonts -- Version 3.3.2.3, 1,228K, XFree86-75dpi-fonts -- Version 3.3.2.3, 1,060K, XFree86-S3V -- Version 3.3.2.3, 3,793K, XFree86-SVGA -- Version 3.3.2.3, 4,585K, XFree86-Xnest -- Version 3.3.2.3, 1,985K, XFree86-Xvfb -- Version 3.3.2.3, 2,415K, XFree86-libs -- Version 3.3.2.3, 1,863K, gnome-libs -- Version 0.20, 556K, xinitrc -- Version 1.6, 9K) ;

- Программы-демоны (SysVinit -- Version 2.74, 141K, at -- Version 3.1.7, 60K, gpm -- Version 1.13, 193K, procmail -- Version 3.10, 179K, sendmail-cf -- Version 8.8.7, 611K, vixie-cron -- Version 3.0.1, 54K, sysklogd -- Version 1.3, 105K);

- Средства разработки (autoconf -- Version 2.12, 524K, automake -- Version 1.3, 777K, libtool -- Version 1.2b, 485K, make -- Version 3.76.1, 247K, egcs-c++ -- Version 1.0.3a, 1,780K, egcs-objc -- Version 1.0.3a, 1,490K, gcc -- Version 2.7.2.3, 2,041K, tcl -- Version 8.0.3, 5,464K, python -- Version 1.5.1, 5,320K, tk -- Version 8.0.3, 5,227K);

- Библиотеки (glibc -- Version 2.0.7, 15,608K, glibc-profile -- Version 2.0.7, 10,010K, libstdc++-devel -- Version 2.8.0, 1,090K);

- Утилиты (binutils -- Version 2.9.1.0.14, 4,537K, bison -- Version 1.25, 154K, gettext -- Version 0.10.35, 824K, gzip -- Version 1.2.4, 227K, tar -- Version 1.12, 471K, zip -- Version 2.1, 206K, unzip -- Version 5.31, 335K, SVGATextMode -- Version 1.8, 845K, open -- Version 1.4, 12K, file -- Version 3.25, 198K, fileutils -- Version 3.16, 868K, findutils -- Version 4.1, 155K, smbfs -- Version 2.0.1, 50K, symlinks -- Version 1.2, 98K, control-panel -- Version 3.7, 177K, MAKEDEV -- Version 2.3.1, 24K, glint -- Version 2.6.1, 227K, isapnptools -- Version 1.15a, 175K, helptool -- Version 2.4, 23K, kbd -- Version 0.96a, 1,080K, lilo -- Version 0.20, 1,437K, linuxconf -- Version 1.12r5, 7,324K, logrotate -- Version 2.6, 41K, lpr -- Version 0.33, 168K, man -- Version 1.5f, 89K, mingetty -- Version 0.9.4, 31K, mkisofs -- Version 1.12b4, 138K, modemtool -- Version 1.21, 15K, mount -- Version 2.8a, 107K, netcfg -- Version 2.19, 165K, procinfo -- Version 14, 41K, quota -- Version 1.55, 80K, rhs-printfilters -- Version 1.46, 90K, rpm -- Version 2.5.5, 1,015K, setserial -- Version 2.14, 40K, grep -- Version 2.2, 258K, patch -- Version 2.5, 94K и другие);

- Командные оболочки (bash -- Version 1.14.7, 1,327K, mc -- Version 4.1.35,

869K, tcsh -- Version 6.07.09, 487K) ;

- Документация (различные файлы HOWTO, FAQ, файлы мануалов);

Процесс инсталляции, деинсталляции, контроля целостности и учета производится при помощи утилиты RPM (RedHat Pakage Manager). RPM ведет собственную базу установленных приложений и компонент. В настоящее время огромное количество продуктов поставляется в rpm-формате. Установка таких пакетов чрезвычайно проста и не требует “ковыряния” в конфигурационных файлах.

Помимо непосредственно rpm, существует X Window приложение Pakage Management, отображающее установленные пакеты в графическом окне и позволяющее деинсталлировать ненужные пакеты и добавлять необходимые с дистрибутивного диска RedHat (См. Приложение 5).

Однако, не все необходимые пргораммные продукты содержаться на инсталляционном диске, но практически все можно получить с анонимных FTP-серверов. Так, например, Apache 1.3.4 rus, netatalk 1.4b2+asun, ifmail 2.14, последняя версия Netscape Communicator были найдены и сгружены с соответствующих серверов. Данное программное обеспечение распространяется бесплатно.

2.5 Стратегия администрирования и управления

2.5.1 Категории пользователей

Кратко основную идею принадлежности объектов и процессов в UNIX-системах можно выразить так: “Пользователь должен иметь доступ только к тому, что необходимо”.

Административное управление в системе UNIX отделено от общепользовательского доступа. Привелегированный пользователь в UNIX - это “полубог“, который может выполнять такие важные задачи, как управление процессами и подключение устройств.

На процессы и файлы в ОС UNIX распространяется единая система принадлежности объектов. Преимущественное право контроля над файлом или процессом принадлежит владельцу. Права владельца могут отменяться только привелегированным пользователем.

У каждого файла есть владелец и группа. Владелец файла имеет только одну привелегию, которая другим пользователям системы не доступна: он может изменять права доступа к файлу. В частности, владелец может установить права доступа так, что никто, кроме него, не сможет обращаться к данному файлу.

Владелец файла - всегда один пользователь. Группа - один или несколько. Информация о группах хранится в файле /etc/group. Вот несколько строк из этого файла:

```
root::0:root
bin::1:root,bin,daemon
daemon::2:root,bin,daemon
wheel::10:root,dima,backup
mail::12:mail
news::13:news
users::100:al,mount,unmount,dima,yura,little,irina,am,nadja,macmaster
floppy:x:19:
pppusers:x:230:dima,pppuser
popusers:x:231:
slipusers:x:232:
postgres:x:233:
netware::503:root,yura,dima,little
web::504:
mailonly::505:
quoted:x:507:yura,little,macmaster
dba:x:500:oracle,dba
```

Владелец файла определяет, какие операции могут совершать над файлом члены группы. Помимо этого владелец файла определяет права доступа к файлу всех остальных пользователей, не входящих в группу. Подобная организация допускает коллективное использование файлов членами одной группы. Рассмотрим это на примере содержания системного раздела эмулятора NetWare, установленного на Linux-машине:

```
[root@unix sys]# ls -l
total 10
drwxr-xr-x   2 dima     netware      1024 Apr 26 17:46 dima
drwxr-xr-x   6 little   netware      1024 Apr  1 12:47 little
drwxr-xr-x   2 root     root         1024 Aug 19  1998 login
drwxr-xr-x  20 root     root         1024 Mar 24 19:31 mail
drwxr-xr-x   2 nadja    netware      1024 May 31 14:36 nadja
drwxr-xr-x   3 root     root         1024 Mar  3 16:08 print
drwxrwxr-x  27 dima     netware      1024 Apr 13 18:39 public
-rw-r--r--   1 root     root          138 Aug 19  1998 readme.txt
drwxr-xr-x   2 root     root         1024 Aug 19  1998 system
drwxr-xr-x   2 yura     netware      1024 Feb 27 22:38 yura
```

Права к служебным каталогам login, print, mail и system принадлежат суперпользователю root. Каталоги yura, little, dima, nadja принадлежат соответствующим пользователям. Заметим, что остальные члены группы netware имеют право на исполнение и чтение файлов в этих каталогах. В то же время, каталог общего пользования public имеет биты записи, чтения и исполнения.

UNIX отслеживает не символьные имена владельцев и групп (которые введены исключительно для удобства), а их идентификаторы. Идентификаторы пользователей (сокращенно UID) и соответствующие им имена пользователей хранятся в файле /etc/passwd, а идентификаторы групп (GID) и имена групп, отвечающие им, - в файле /etc/group.

С учетом конкретной ситуации в сети фирмы необходимо и логическое деление пользователей Linux-машины. В этой системе высшую иерархию занимают пользователи, имеющие доступ к командному shell. Их немного, а возможность доступа к командной оболочке объясняется исключительной необходимостью управлять определенными процессами и файлами (разумеется, в рамках их полномочий). На ступень ниже - категория обычных пользователей. Их бюджеты существуют для поддержания почтовых ящиков и для осуществления процессов аутентификации, например, при обращении к ftp или smb серверу. И, наконец, третья группа - специальные бюджеты. Они выделяются для осуществления специфических действий. Отличительной чертой таких пользователей (скорее, именно бюджетов, а не пользователей) является указание в последнем поле строки файла /etc/passwd нестандарных исполняемых модулей. Рассмотрим пример:

```
mount:x:514:514::/home/mount:/usr/sbin/cdmount
unmount:x:515:515::/home/unmount:/usr/sbin/cdunmount
```

Здесь mount и unmount - фиктивные пользователи, необходимые для удаленного монтирования CD-ROM-накопителя сервера, соответственно /usr/sbin/cdmount и /usr/sbin/cdmount - специально написанные системным администратором сценарии, содержащие команды монтирования и отмонтирования диска.

Вот другой пример:

```
pppuser:x:507:508::/home/pppuser:/etc/ppp/pppplogin
```

Положительная аутентификация псевдо-пользователя pppuser приводит к

исполнению сценария /etc/ppp/ppplogin и к установлению PPP-соединения между сервером и удаленным компьютером.

С привилегированным пользовательским доступом сопряжены три проблемы: безграничные полномочия, отсутствие учета, наличие операций, производить которые в праве только он (необходимость некоторых из которых может возникнуть в отсутствие root) . Поскольку полномочия привелегированного пользователя распределить нельзя, то трудно предоставить кому-то возможность снятия резервных копий (что должно делаться под именем root) не давая при этом полной свободы действий в системе.

Программа sudo позволяет разрешать обычным пользователям выполнение (запуск) конкретно оговоренных задач, полномочиями на которые обладает суперпользователь root. Команда sudo обращается к файлу /etc/sudoers, содержащему список пользователей, имеющих полномочия на ее применение, и перечень команд, которые они имеют право использовать на конкреной машине. Если предлагаемая команда разрешена, sudo приглашает пользователя ввести его собственный пароль и выполняет команду как root. Все действия пользователей, осуществляемые с использованием sudo записываются в специальном регистрационном файле. В качестве конкреного примера следует упомянуть, что менеджеру компьютерного центра разрешено монтирование файловых разделов дополнительных жестких дисков и осуществлять внеочередные операции резервного копирования.

Институт пользовательских бюджетов, групп и закрепленных за ними прав на процессы и файловые ресурсы позволяют системному администратору системы гибко управлять ресурсами сервера.

2.5.2 Файловые системы и управление дисковым пространством

Файловая система ОС Linux - ext2fs. Загрузочная часть файловой системы должна обязательно являться ext2fs, хотя Linux и может работать с различными файловыми системами.

В терминологии Linux дисковые разделы называются следующим образом:

hda1,hda2,hda3... - разделы одного физического носителя, подключенного к первому IDE-контроллеру в качестве ведущего (master);

hdb1,hdb2,hdb3... - разделы одного физического носителя, подключенного к первому IDE-контроллеру в качестве ведомого (slave);

hdc1,hdc2,hdc3... - разделы одного физического носителя, подключенного ко второму IDE-контроллеру в качестве ведущего (master);

hdd1,hdd2,hdd3... - разделы одного физического носителя, подключенного ко второму IDE-контроллеру в качестве ведомого (slave);

Стандарным средством разбиения дискового пространства является программа fdisk. Она позволяет изменять тип дисковых разделов, создавать и удалять их. Вот таблица разбиения основного жесткого диска нашей Linux-системы:

```
Disk /dev/hda: 128 heads, 63 sectors, 621 cylinders
Units = cylinders of 8064 * 512 bytes

   Device Boot    Start     End   Blocks   Id  System
/dev/hda1             1     605  2439328+  83  Linux native
/dev/hda2           606     621    64512   82  Linux swap
```

Физический жесткий диск стандарта IDE объемом около 2.5 Гб, подключенный к IDE-контроллеру системной платы как primary master (т.е. hda в терминологии Linux), разбит на два раздела. Первый раздел инициализирован как ext2fs (Linux Native), на нем и располагается основная файловая система ОС Linux. Второй раздел hda2 объемом 64 Мб отведен под swap-партицию. Он используется при нехватке системы физической оперативной памяти; часть оперативных данных простаивающих в конкретный момент разделов выгружается из оперативной памяти и размещается в виртуальной памяти swap-раздела.

Программа fdisk распознает и может инициализировать файловые системы различных стандартов (DOS FAT, HPFS OS/2, Win95 FAT 16/32, BSDI fs и другие).

Помимо основного дискового раздела hda1 в системе используется вспомогательный жесткий диск объемом около 500 Мб, подключенный как secondary master и представленный как /hdc1 в терминологии Linux:

```
Disk /dev/hdc: 16 heads, 63 sectors, 1057 cylinders
Units = cylinders of 1008 * 512 bytes
```

```
  Device Boot    Start     End   Blocks   Id  System
/dev/hdc1             1    1057   532696+  83  Linux native
```

Раздел также отформатирован как ext2fs (Linux native, тип 83) и используется для временного хранения больших объемов оперативных данных. Помимо упомянутых двух физических накопителей, к системе можно подключать еще до двух накопителей.

Раздел, с которого происходит загрузка Linux, определяется конфигурационным файлом /etc/lilo.conf и соответсвующей ему записи MBR (Master Boot Record) на primary накопителе. Вообще, система начальной загрузки LILO (LInux LOader) позволяет выбирать до четырех различных систем во время начальной загрузки. Следует отметить, что в отличие, например, от Windows95, корень основной файловой системы Linux может рапологаться не на основном ведущем разделе (primary master).

Помимо низкоуровневого средства разбиения и первичной инициализации разделов fdisk, в стандартной поставке Linux представлены средства проверки целостности файловых систем и их восстановления при обнаружении ошибок. Это такие утилиты как fsck, fsck.minix, fsck.ext2, e2fsck, fsck.xiafs. Вот пример проверки целостности файловой ситемы раздела /dev/hdc1:

```
[root@unix /root]# fsck.ext2 /dev/hdc1
e2fsck 1.12, 9-Apr-98 for EXT2 FS 0.5b, 95/08/09
/dev/hdc1 has reached maximal mount count, check forced.
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/hdc1: 3665/133640 files (0.7% non-contiguous), 295059/532481 blocks
```

Форматирование (собственно создание файловой системы на новом разделе) производится при помощи утилиты mkfs (Make fs).

Информация о файловых разделах и параметрах их монтирования содержится в файле /etc/fstab:

```
[root@unix init.d]# less /etc/fstab
/dev/hda1       /          ext2        exec,dev,suid,rw,usrquota,grpquota      1 1
/dev/hda2     none         swap        exec,dev,suid,rw                        0 0
/dev/hdc1     /mnt/hdc     ext2        exec,dev,suid,rw,usrquota,grpquota      0 0
/dev/fd0      /mnt/floppy  ext2              noauto                            0 0
/dev/cdrom    /mnt/cdrom   iso9660     noauto,user,ro                          0 0
none           /proc          proc       defaults                            0 0
```

Во второй колонке указаны точки монтирования файловых систем, в первой - устройтсва им соответствующие. Третья колонка определяет тип файловой системы. Четвертая колонка описывает опции монтирования. Рассмотрим

некоторые из них подробнее: usrquota и groupquota означают, что файловые квоты на данных разделах включены, опции rw и ro означают режимы чтения и записи и только чтения соответственно, параметр noauto запрещает автоматическое монтирование системы. Параметр user в строке, отвечающей за монтирование CD ROM, означает возможность монтирования устройства обычным пользователем. Все остальные системы монтируются и отмонтируются привелегированным пользователем root.

Весьма интересна последняя строка файла: proc - виртуальная системная файловая система, предназначенная для хранения информации о процессах. Так, например, файл /proc/mounts содержит информацию о смонтированных на данный момент разделах:

```
[root@unix /proc]# less ./mounts
/dev/root / ext2 rw 0 0
/proc /proc proc rw 0 0
/dev/hdc1 /mnt/hdc ext2 rw 0 0
```

Ручное монтирование и отмонтирование файловых систем совершается командами mount и umount соответственно. Разумеется, отмонтирование раздела, файлы которого в текущим момент по какой-либо причине открыты, невозможно. Приведем пример использования команды mount для монтирования CD ROM:

```
mount -t iso9660 /dev/cdrom /mnt/cdrom
```

Первый параметр команды указывает на тип файловой системы, второй - физическое устройство, третий и последний - точку монтирования системы.

Зачастую пользователи выходят за рамки разумных пределов использования дискового пространства сервера. Для этого в UNIX (да и в большинстве сетевых ОС) существует механизм квот. Квоты позволяют системному администратору ограничить количество индексных дескрипторов и дисковых блоков, которое может быть выделено каждому пользователю. Количество индексных дескрипторов грубо определяет, сколько файлов может принадлежать одному пользователю. Лимит дисковых блоков ограничивает общий объем пространства файловой системы, которое предоставляется для работы конкретному пользователю. В Linux система квот реализвона не только на уровне пользователей, но и на уровне групп, что позволяет более удобно управлять квотированием.

Каждое из ограничений задается в виде пары чисел: нестрого лимита, по достижении которого пользователь предупреждается о возможном нарушении квоты, и строгого лимита, который определяет абсолютное ограничение данного ресурса.

Команда edquota позволяет редактировать квоты пользователей и групп, команда quota позволяет просматривать текущее состояние квотируемых ресурсов:

```
[root@unix pam.d]# quota little
Disk quotas for user little (uid 509):
   Filesystem blocks  quota  limit  grace  files  quota  limit  grace
    /dev/hda1  74426 100000 150000         1264      0      0
```

Команды quotaon, quotaoff позволяют, соответсвенно, включать и выключать дисковые квоты. Команда repquota выводит сводную таблицу дисковых квот по всем пользователям/группам и файловым системам.

Файлы сводки дисковых квот quota.group и quota.user размещаются в корне файловых систем.

На практике управление дисковыми квотами осуществляется при помощи системной утилиты linuxconf, которая обеспечивает удобство управление, наглядность и снабжена графическим интерфейсом (См. Приложение 7).

2.5.3 Учет системных ресурсов и анализ производительности

Учет системных ресурсов системы - одна из забот системного администратора. Ядро Linux и различные системные программы ведут учетные записи о времени использования центрального процессора, количества сеансов работы пользователей, времени работы принтеров, модемов и ряда других ресурсов системы. Обычно файлы регистрации находятся в каталоге /var/log.

Учетные файлы выполняют две главные задачи. Первая из них - информационная помощь в отладке работы системных и прикладных процессов. Второй причиной использования файлов регистрации является необходимость выявления несанкционированного доступа к системе.

Основа учета системных ресурсов - система syslog. Специальные демоны sys-logd и klogd регистрирует сообщения ядра и системных процессов в файлах регистрации /var/log/messages, /var/log/secure, /var/log/maillog, /var/log/spooler. Многие системные и пользовательские программы ведут собственные файлы регистрации. Вот пример одного из регистрационных файлов Web-сервера Apache:

150.254.173.2 - - [26 / Mar / 1999:00:58:14 +0300] "GET /pcprice.html HTTP / 1.0" 404 275
195.34.34.69 - - [26 / Mar / 1999:06:35:06 +0300] "GET /robots.txt HTTP / 1.1" 404 285
195.34.34.69 - - [26 / Mar / 1999:06:37:06 +0300] "GET /miha_test HTTP / 1.1" 404 284
195.34.34.69 - - [26 / Mar / 1999:06:37:06 +0300] "GET /csc.html HTTP / 1.1" 404 283
195.34.34.69 - - [26 / Mar / 1999:06:37:07 +0300] "GET /macprice.html HTTP / 1.1" 404 288

В системе RedHat реализована система автоматической ротации файлов регистрации. Файлы регистрации совершают периодические перемещения: раз в неделю файлы переименовываются, в результате чего более старые данные сдвигаются в конец цепочки:

secure
secure.1
secure.2
secure.3
secure.4

Система ротации представляет из себя совокупность shell-сценариев каталога /etc/logrotate.d, вызываемых системным шедулером cron, и конфигурационный файл /etc/logrotate.conf.

Одновременно с этим раз в неделю происходит резервное копирование файлов регистрации вместе с остальной важной информацией системы (См. разделы 2.5.6 и 3.4, посвященные резервному копированию).

Производительность системы во многом определяется эффективностью распределения и коллективного использования ее ресурсов. В первом приближении серьезное влияние на производительность системы оказывают четыре вида ресурсов:

время центрального процессора;

память;
скорость обмена с жестким диском при операциях ввода-вывода;
пропускная способность сетевого адаптора;

Каждый процесс потребляет определенную часть ресурсов системы. Если после того, как активные процессы взяли все, что им нужно, остались свободные ресурсы, можно сказать, что производительность системы удовлетворительна. Если ресурсов недостаточно, процессы должны выстраиваться в очередь. Процесс, не имеющий немедленного доступа к необходимым ресурсам, должен ждать, ничего при этом не делая. Бесполезная трата времени на ожидания - одна из основных причин ухудшения производительности.

Для анализа производительности существует много способов.Один из них - команда vmstat:

```
[root@unix log]# vmstat 5 5
procs                  memory    swap      io     system         cpu
r b w  swpd  free  buff cache  si  so   bi   bo   in   cs  us  sy  id
1 0 0 13324 10236  1036 12168   5   1    6    1  138   59  86   1  14
2 0 0 13324 10236  1036 12168   0   0    0    0  109   46 100   0   0
1 0 0 13324 10236  1036 12168   0   0    0    1  110   47 100   0   0
1 0 0 13324 10236  1036 12168   0   0    0    0  108   45  99   1   0
1 0 0 13324 10236  1036 12168   0   0    0    0  109   46 100   0   0
```

Команда top дает более подробную информацию об использовании системных ресурсов системы в целом и об активных процессах:

```
 3:30am  up  4:52,  3 users,  load average: 0.31, 0.88, 0.96
47 processes: 46 sleeping, 1 running, 0 zombie, 0 stopped
CPU states:  0.5% user,  0.9% system,  0.0% nice, 98.6% idle
Mem:   30944K av,  18632K used,  12312K free,   5656K shrd,   1100K buff
Swap:  64508K av,  11668K used,  52840K free                 12340K cached

  PID USER     PRI  NI  SIZE  RSS SHARE STAT  LIB %CPU %MEM   TIME COMMAND
 7117 root      16   0   724   724   560       R     0    1.3    2.3    0:01    top
  973 root       1   0   292   180   128       S     0    0.1    0.5    0:01    in.telnetd
.....
```

Мгновенный “снимок” активности процессов можно получить при помощи команды ps:

```
[root@unix log]# ps axu
USER       PID %CPU %MEM  SIZE   RSS TTY STAT START   TIME COMMAND
bin        256  0.0  0.0   752     4 ?   S    22:38   0:00 (portmap)
daemon     287  0.0  0.3   784   112 ?   S    22:38   0:00 /usr/sbin/atd
dima       974  0.0  0.1  1264    52 p0  S    22:59   0:00 (bash)
news       467  0.0  2.8  2696   868 ?   S    22:39   0:00 /usr/lib/news/bin/act
news       469  0.0  4.6  7204  1428 ?   S    22:39   0:00 /usr/sbin/innd -p4 -i
news       471  0.0  0.0  1212    24 ?   S    22:39   0:00 (rc.news)
```

```
news      488  0.0  0.0   912    0 ?  SW   22:39   0:00 (crosspost)
news      489  0.0  0.0   912    0 ?  SW   22:39   0:00 (overchan)
news      490  0.0  1.6  1100  504 ?  S    22:39   0:00 /usr/lib/news/bin/inn
news      514  0.0  1.3  1256  404 ?  S    22:40   0:03 sh /usr/lib/news/bin/
news     7332  0.0  0.8   716  276 ?  S    03:29   0:00 sleep 600
nobody    395  0.0  0.3  1624  116 ?  S    22:39   0:00 (httpd)
nobody    972  0.0  0.1  1096   52 ?  S    22:57   0:00 afpd -G -g nobody -c
```

Поскольку в Linux используется виртуальная память, реализованная в виде раздела подкачки (swap partition), то скорость обмена с диском непосредственно связана с емкостью памяти. В сильно загруженной системе с ограниченной емкостью оперативной памяти для получения чистой страницы виртуальной памяти часто приходится записывать ее содержимое на диск. Такая ситуация снижения производительности наблюдается во время работы с большим числом программ, требующих много памяти. В нашем конкретном случае это, например, приложения X window и Oracle server.

Команда free позволяет контролировать распределение памяти:

```
[root@unix log]# free -t
                     total       used       free     shared    buffers     cached
Mem:                 30944      18532      12412       5356       1100      12356
-/+ buffers/cache:              5076      25868
Swap:                64508      11664      52844
Total:               95452      30196      65256
```

2.5.4 Планирование процессов

Ключ к сохранению постоянного контроля над системой - автоматизация максимально возможного количества задач. Планирование процессов является неотъемлемой частью функционирования UNIX-систем.

В ОС UNIX периодическим управлением процессов управляет демон crond. Он запускается во время начальной загрузки системы и остается в активном состоянии, пока система не выключена. Демон cron читает файл конфигурации, содержащий последовательности командных строк, расписание их вызова и регистрационные имена, под которыми они должны выполняться. Командные строки обрабатываются shell-интерпретатором, поэтому почти все, что можно сделать в командной оболочке вручную, можно перепоручить системному шедулеру cron.

Главный конфигурационный файл программы cron - файл /etc/crontab:

```
[root@unix log]# less /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 13 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
30 07 * * * root run-parts /etc/cron.nologin
04 21 * * * root run-parts /etc/cron.login
```

Дата вызова той или иной последовательности сценариев каждой строчки задается следующим образом: минуты, часы, день, месяц и день недели. Так, в нашем примере совокупность shell-процедур, находящихся в каталгое /etc/cron.monthly, будет выполняться 1-го числа каждого месяца в 4 часа 42 минуты. При запуске того или иного процесса из cron производится соответствующая запись в файл регистрации /var/log/cron.

В нашей системе при помощи cron вызываются и осуществляются следующие периодические процессы: ротация регистрационных файлов (см. раздел “2.5.3 Учет системных ресурсов и анализ производительности”, установка и сброс флага-разрешения снятия модемом трубки при организации сервера удаленного доступа (см. раздел ”2.3.6 Сервер удаленного доступа”), обновление статей News-конференций, ежедневное базы имен файлов и еженедельное резервное копирование (см. раздел “3.4 Резервное копирование”).

2.5.5 Информационная безопасность

Проблемы информационной безопасности охватывают широчайший диапазон административных, этических, правовых, технических вопросов.

В рассмотрение данного раздела попадают причины исчезновения, порчи, изменения и утечки информации.

Проблемы, возникающие в этой области по своим причинам делятся на явления человеческого, личностного происхождения (здесь следует выделить халатность или некомпетентность сотрудников, злонамеренные попытки несанкционерованного доступа к информационным ресурсам, умышленное заражение вирусами и их написание и другое) и угрозы не связанные с деятельностью человека (отказ оборудования, сбои файловых систем, стихийные бедствия, скачки напряжения и др.)

Задача системного администратора - стараться свести к минимуму возможность потери и/или кражи информации. Задача это усложняется тем фактом, что внутренняя сеть офиса соотносится с внешим миром (во-первых, через канал подключения к Интернет, а во-вторых, через сервер удаленного доступа).

Комплекс мер, предпринимаемых системным администраторам не должен сводиться лишь к программно-техническим средствам защиты, во многом следует уделять внимание административным мерам (правила, инструктажи. распоряжения, обучение и оповещение сотрудников). По возможности нужно стараться максимально автоматизировать обработку и хранение информации для снижения угрозы человеческого фактора. Например, автоматизированная система резервного копирования (см. Раздел “2.5.6 Резервное копирование”) позволила решить вопрос несознательности, недисциплинированности и некомпетентности сотрудников, которые решительно пренебрегали копированием критичной информации своих рабочих станций на файловые разделы сервера. Мало того, в какой-то степени такая автоматизация снизила вероятность потери данных вследствии халатности или ошибки самого системного администратора.

В сети фирмы строго выполняется правило четкого разграничения прав доступа. Этот принцип реализуется как на горизонтальном уровне (права доступа к файлам сервера и разделенным сетевым ресурсам), так и на вертикальном (различные категории пользователей, выделение групповых полномочий). В двух словах этот принцип можно выразить так: “Пользователь должен иметь доступ только к тому, к чему ему иметь НЕОБХОДИМО”.

Так, например, пользователи, владеющие собственными почтовыми ящиками и имеющие доступ к файловым сервисам mars_nwe,samba и ftp-серверу, совершенно необязательно должны иметь доступ к командному shell UNIX-системы. Вместо стандарной оболочки bash в строчках файла /etc/passwd указан исполняемый файл-”пустышка” /bin/nonexistent.

В свою очередь, пользователи, имеющие доступ к командному shell, не имеют прав доступа к файлам бюджетов, конфигурационным и регистрационным файлам и системным процессам. В этом смысле, система UNIX с ее институтом принадлежности файлов и процессов сильно упрощает задачу.

К средствам и методам защиты сетей следует отнести файрволлинг (от “fire-wall” - огненная стена, англ.), использование криптованных протоколов обмена, анализ регистрационных файлов, своевременное обнаружение и устранение прорех в защите, разделение прав на определенные виды сетевого взаимодействия извне

или с конкретных хостов. Так, например, система XDM была сконфигурирована таким образом, что право на соединение X сервера с Linux имеет лишь рабочая станция Macintosh bigmac.px.podolsk.ru.

С помощью системы PAM (Pluggable Authentication Modules) можно ограничить возможность аутентификации пользователя с определенного хоста или сети, установить время разрешенной работы конкретных пользователей или группы и проч. (О применении системы PAM будет подробно рассказано в разделе “3.5 Меры по обеспечению информационной безопасности”).

Мощнейшим средством защиты сети от аттак и несанкционированного доступа извне является firewall. Firewall это совокупность компонент или система, которая располагается между двумя сетями и обладает следующими свойствами:

Весь трафик из внутренней сети во внешнюю и из внешней сети во внутреннюю должен пройти через эту систему;

Только трафик, определенный локальной стратегией защиты, может пройти через эту систему;

Система надежно защищена от проникновения.

Обычно все firewalls осуществляют фильтрацию IP пакетов средствами фильтрующих маршрутизаторов. Фильтрация пакетов, проходящих через интерфейсы маршрутизатора, основана на наборе правил, которые устанавливаются, базируясь на стратегии защиты. Фильтрующие маршрутизаторы обычно могут фильтровать IP пакеты, основываясь на некоторых или всех следующих критериях:

1. IP адрес источника,
2. IP адрес назначения,
3. TCP/UDP порт источника,
4. TCP/UDP порт назначения.

Фильтрация может использоваться, чтобы блокировать соединение на определенные хосты или сети, а также блокировать соединение с определенными портами. Например, можно блокировать соединения от определенных адресов хостов или сетей, которые рассматриваются как враждебные или незаслуживающие доверие. Также можно блокировать соединение от всех внешних адресов, исключая, например, только SMTP для получения электронной почты.

Добавление фильтрации TCP или UDP портов к фильтрации IP адресов дает большую гибкость в стратегии защиты. Сервисы, такие как TELNET демон, обычно располагаются на определенном порту (для TELNET порт 23). Эти сервисы можно блокировать на все хосты и разрешать их только на определенные системы. Например, можно блокировать все входные соединения, но разрешить только определенные сервисы, такие как SMTP, для одного хоста и TELNET или FTP соединения для другого хоста.

Следующие сервисы наиболее уязвимы и обычно блокируются в firewall:

tftp, порт 69, trivial FTP, используются для загрузки бездисковых станций, терминальных серверов и маршрутизаторов.

X Window, OpenWindows, порт 6000, порт 2000, может пропускать информацию от X Window дисплеев, включая все нажатия клавиш.

RPC, порт 111, Remoute Procedure Calls, включая NIS и NFS, которые могут быть использованы для захвата системной информации, такой как пароли и для чтения и записи файлов.

rlogin, rsh и rexec, порты 513, 514 и 512, сервисы, которые при неправильной конфигурации могут разрешать несанкционированный доступ в систему.

Другие сервисы менее опасные обычно фильтруют и по возможности ограничивают их доступ только к тем системам, которые нуждаются в них:

TELNET, порт 23.
FTP, порт 20 и 21.
SMTP, порт 25.
RIP, порт 520.
DNS, порт 53.
UUCP, порт 540.
NNTP, порт 119.
gopher, http, порты 70 и 80.

В сети фирмы в качестве firewall-хоста выступает Linux-машина. Firewall реализован с использованием ipfwadm - средства, регулирующего правила фильтрации IP-пакетов на уровне ядра Linux. Собственно, firewall выполнен в виде одноименной shell-процедуры, состоящей из последовательных команд ipfwadm с определенными параметрами, которые и задают правила фильтрации.

Данная процедура вызывается в соответствующих уровнях исполнения при загрузке Linux-машины.

К сожалению, описание всех применяемых методик и средств защиты информации выходит далеко за рамки дипломной работы.

2.5.6 Резервное копирование

Правильный подход к операциям резервного копирования позволяет свести к минимуму потерю важных для предприятия данных. Среди потенциальных причин таких потерь следует упомянуть:

выход из строя дисковых систем;
ошибки и крахи файловых систем;
стихийные бедствия, пожары, кража компьютерной техники и проч.;
злонамеренные действия хакеров;
ошибки действий пользователей рабочих станций и системного администратора;

Практика администрирования сети фирмы показала, что все бремя забот о резервном копировании лежит на системном администраторе. При обычной общей недисциплинированноти, плохой осведомленности и безответственности пользователей рабочих станций фирмы, лучшим решением проблемы сохранности данных является автоматизация всех процессов резервного копирования.

Данная система была впервые полностью разработана в рамках данного проекта.

Файловые ресурсы, подлежащие резервному копированию следует разделить на две составляющие: файлы и каталоги рабочих станций офисной сети и собственные файловые ресурсы сервера фирмы. Общая схема системы резервного копирования представлена в Приложении 9.

Суть работы системы сводится к периодическому вызову специально написанных shell-процедур. Главная из них - процедура main (см. полный текст процедуры в Приложении 12), которая осуществляет вызов процедуры архивации необходимых разделов сетевых рабочих станций, архивацию собственных ресурсов сервера, вызов сценария ротации архивов и уведомление оператора резервного копирования о выполненных операциях.

В приложении 9 представлен текст вспомогательной процедуры winbackup, осуществляющей доступ к необходимым разделенным каталогам станций Windows95. Отметим, что данный shell-сценарий вызывается с тремя параметрами: сетевое имя машин, имя разделенного ресурса и пароль доступа к данному ресурсу. Внутри процедуры осуществляется контроль правильности передачи параметров, доступности в настоящий момент удаленных сетевых ресурсов, архивирование соответствующих разделов, сжатие полученных архивов и информирование об осуществленных действиях оператора резервного копирования при помощи электронной почты.

Архивация собственных критичных к потере файлов и директорий сервера производится внутри процедуры main согласно списку ресурсов, подлежащих резервному копированию. Список находится в файле /usr/local/bin/backup/locallist и составляется системным администратором фирмы.

Полученные архивы подвергаются трехступенчатой ротации.

После получения по электронной почте уведомления о произведенной операции резервного копирования, оператор резервного копирования обязан скопировать полученные архивы на станцию Macintosh и произвести запись на записываемые CD ROM.

Более подробно настройка системы резервного копирования рассмотрена в разделе “3.4 Система резервного копирования”.

2.5.7 Сетевая печать

Печатающие устройства фирмы можно разделить на средства сестевой печати (лазерный принтер Hewlett Packard LaserJet 5M) и локальные устройства печати рабочих станций (лазерный принтер Hewlett Packard LaserJet 4L и цветной струйный принтер Hewlett Packard 870 Cxi). Хотя локальные печатающие устройства могут быть доступны другим рабочим станциям как разделенный сетевой ресурс (что время от времени используется), основную печатную мощность фирмы составляет сетевой офисный принтер HP LJ 5M.

Принтер HP LaserJet 5M обладает следующими характеристиками:

- Механизм печати со скоростью 12 страниц в минуту использует методы ускоренной печати (Accelerated Printing Technologies), обеспечивающие высокую скорость печати.
- Принтер печатает с реальной разрешающей способностью, равной 600 точек на дюйм, используя технологию увеличения разрешающей способности (REt), микрозернистый тонер (MicroFine) и обеспечивающая 120 уровней полутонов (lpi).
- Принтер поддерживает языки PCL6 и Adobe PostScrip level 2, HP-GL/2, негативную печать и растровые шрифты.
- Принтер может производить распечатку на бумаге различных размеров и веса, а также на конвертах, наклейках и прозрачной пленке. Лоток 1 настраивается на закладку конвертов и бумаги различных размеров, которые подаются вручную или автоматически из стопы. Лоток 2 вмещает до 250 листов.
- Оперативная память принтера - 6 Мб.
- Метод расширения памяти (MEt) позволяет распечатывать большинство документов с использованием памяти принтера стандартного объема. Метод MEt производит автоматическое сжатие данных при распечатке сложных страниц.
- В состав программного обеспечения входят утилиты управления принтером.
- Принтер обладает следующими интерфейсами: последовательный порт RS-232, параллельный порт Centronics, принтерный порт для компьютеров Macintosh (DIN 8), коаксиальный (BNC) сетевой разъем 10Base2, сетевой разъем RJ-45 стандарта 10BaseT.
- Используемый в принтере режим PowerSave обеспечивает экономию электроэнергии за счет существенного сокращения ее потребления, когда принтер находится в неактивном состоянии в течении заданного интервала времени.
- Режим экономии тонера (EconoMode) сокращает затраты тонера на 66% при выводе черновых копий.

Плата JetDirect, входящая в состав принтера обеспечивает непосредственное подсоединение к локальной сети, что ускоряет процесс печати. Плата HP JetDirect позволяет производить печать с различных платформ и автоматически переключает протоколы, что позволяет использовать один принтер при одновременной работе со многими сетями.

Собствеено говоря, описанные возможности HP LJ 5M приближают его к полноценному принт-серверу.

Время подтвердило правильность такого выбора, а продолжительная и интенсивная (до 30.000 копий в месяц) эксплуатация, сочетающаяся с удобством работы, оправдала достаточно высокую цену устройства (более 2000 долларов).

Рабочие станции, работающие под управлением Windows 95/98 осуществляют печать на сетевом принтере посредством специального диспетчера сетевой печати JetAdmin фирмы Hewlett Packard, поставляемой в составе самой операционной системы. Данная утилита предоставляет пользователям рабочих станций широкий выбор средств управления конфигурациями сетевых принтеров, очередями печати и параметрами печати. JetAdmin содержит так же дополнительные средства диагностики, позволяющие выявлять ошибки настройки, аварии и сбои сетевой печати (к счастью, автору за весь период эксплуатации не приходилось к ним прибегать).

При печати с Windows-станций используется протокол IPX 802.3.В качестве драйвера принтера используется программное обеспечение, поставленное вместе с аппаратом.

Печать с рабочих станций Macintosh производится с использованием PostScript-драйвера печати AdobPS 8.5.1 OEM и собственного профайла принтера посредством ethernet-реализации протокола AppleTalk.

Сетвеая плата JetDirect обладает собственным сервисом telnet для удаленного конфигурирования с использованием протокола TCP/IP:

```
> > ?
===JetDirect Telnet Configuration===

    Configured Parameters
    IP Address      : 195.133.132.20
    MAC Address     : 00:60:b0:11:a8:63
    Subnet Mask     : 255.255.255.240
    Default Gateway : 195.133.132.17
    Syslog Server   : 195.133.132.17
    Idle Timeout    : 90 Seconds
    Set Cmnty Name  :
    Passwd : disabled
    Port [ 1] Banner page: enabled

    To Change / Configure Parameters Enter:
    Parameter-name: value <Carriage Return>
    Parameter-name  Type of value
    ip:             IP-address in dotted notation
    subnet-mask:    address in dotted notation
    default-gw:     address in dotted notation
    syslog-svr:     address in dotted notation
    idle-timeout:   seconds in integers
    set-cmnty-name: alpha-numeric string (32 chars max)
    banner: 0 to disable, 1 to enable
    type passwd to change passwd

 Type "?" for HELP Or "quit" to save-and-exit
 Or type "exit" to exit without saving configuration parameter entries
>
```

Как видно из примера, принтере работает как независимое сетевое устройств. Печать с рабочих станций может осуществляться как посредством собственной очереди принтера, так и в 2 очереди печати (PostScript и очередь с собственным профайлом принтера для печати не PostScript-документов) сетевой ОС Linux.

Сетевая печать на два других принтера, являющихся разделенными ресурсами рабочих станций осуществляется в особых случаях (например, при необходимости вывода результатов упражнений учащихся Учебного центра в цвете).

3. ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ

3.1 Инсталляция LINUX

3.1.1 Системные и программные компоненты

Для инсталляции ОС Linux был выбран жесткий диск объемом 2.5 Гб. Дисковое пространство было разбито на основную часть (hda1) и swap-партицию, размер которой рекомендуется выбирать как удвоенное количество оперативной памяти (в нашем случае это 32x2=64). Операция разбиения была выполнена в ходе процесса инсталляции при помощи Linux-версии программы fdisk:

```
Command (m for help): m
Command action
   a   toggle a bootable flag
   b   edit bsd disklabel
   c   toggle the dos compatibility flag
   d   delete a partition
   l   list known partition types
   m   print this menu
   n   add a new partition
   o   create a new empty DOS partition table
   p   print the partition table
   q   quit without saving changes
   t   change a partition's system id
   u   change display/entry units
   v   verify the partition table
   w   write table to disk and exit
   x   extra functionality (experts only)

Command (m for help): p

Disk /dev/hda: 128 heads, 63 sectors, 621 cylinders
Units = cylinders of 8064 * 512 bytes

   Device Boot    Start       End    Blocks   Id  System
/dev/hda1             1       605   2439328+  83  Linux native
/dev/hda2           606       621     64512   82  Linux swap
```

Достаточно современный Bios материнской платы позволил произвести начальную загрузку системы и запуск программы-инсталлятора с дистрибутивного диска RedHat 5.2, т.е. создания загрузочных дискет не потребовалось. Следует отметить удобство и простоту процедуры инсталляции. Определенного уровня знаний требовал оптимальный выбор пакетов.

Так, например, к числу необходимых к установке отнесены следующие компоненты системы:

X Window
система Bind
компилятор GCC

базовые системные библиотеки glibc
исходные тексты ядра и модулей
текстовые редакторы joe, vi
коммуникационная программа minicom
командные оболочки Bash, csh

Как уже говорилось, в ходе процесса инсталляции было предложено подготовить дисковое пространство. Операцию было возможно выполнить либо утилитой fdisk, либо более удобной и наглядной в обращении утилитой DiskDruid. После выделения соответсвующих разделов было произведено их форматирование (на hda1 - ex2fs, стандартная файловая система Linux) с проверкой поверхности диска на наличие т.н. bad blocks (поврежденных блоков). По завершению данной процедуры программа-инсталлятор предложила сконфигурировать LILO (программу начальной загрузки LInux LOader, позволяющую помимо Linux иметь на одной машине несколько операционных систем таких как MS DOS, Windows или OS/2, при загрузке выбирая одну из них). Раздел hda1 выбран загрузочным с точкой монтирования корневой файловой системы, вспомогательный диск hdc1 емкостью 500 Мб, отформатированный как Linux extended предполагалось монтировать к точке /mnt/hdc.

После копирования и инсталляции ядра (по умолчанию) и выбранных компонент системы на раздел hda1, инсталлятор предложил сконфигурировать типовые базовые параметры системы, такие как параметры сетевой карты, IP-адрес системы, домен, имя хоста, пароль пользователя root и проч. была произведена перезагрузка компьютера.

После загрузки ОС в терминальном окне появилось название и версия системы и приглашение к вводу имени пользователя (login). Было введено имя root и пароль, введенный в процессе инсталляции. После успешного процесса аутентифкации был получен доступ к командному shell (Bash).

3.2.2 Пересборка ядра и модулей

Прежде всего необходимо было пересобрать ядро системы, оптимальное для нашей конкретной конфигурации и реализующий необходимый набор свойств. Инсталлированный набор исходных текстов и библиотек дал автору такую

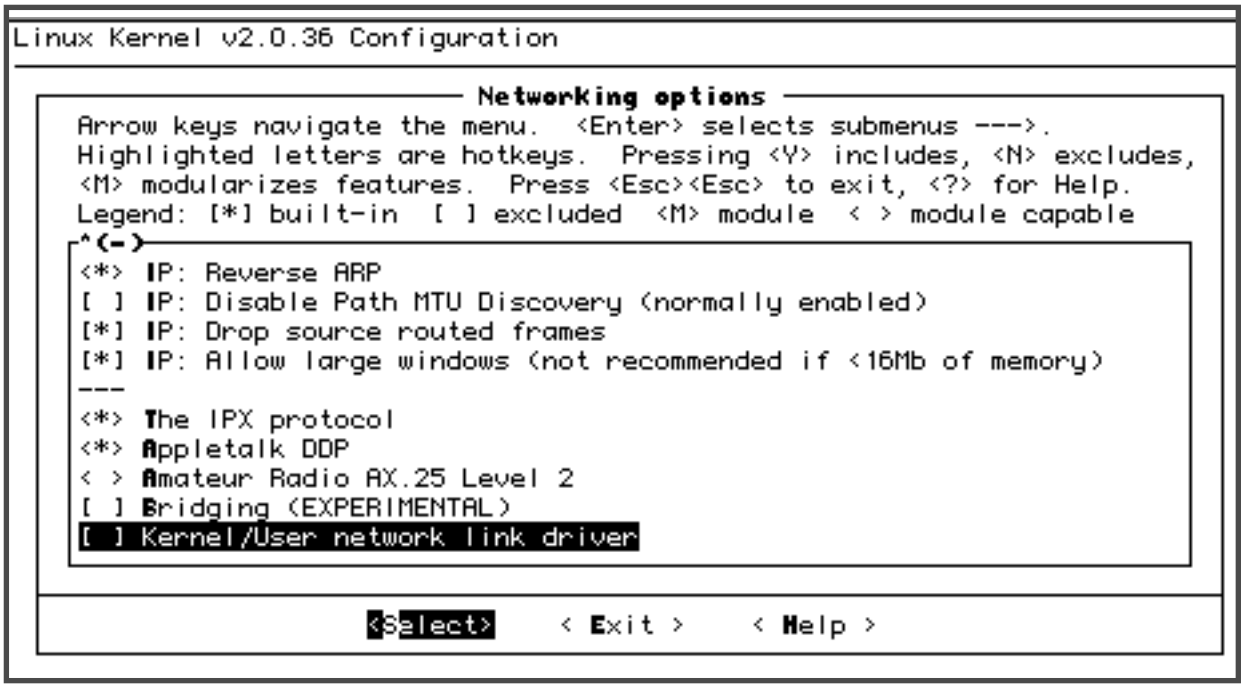


Рис. 14

возможность.

Конфигурирование ядра возможно производить двумя путями: командой make config, последовательно отвечая на вопросы и командой make menuconfig при установленной библиотеке ncurces3 в режиме последовательно окрывающихся меню. Второй вариант, безусловно, является более предпочтительным, поскольку позволяет возвращаться к уже установленным параметрам и является более наглядным и удобным в обращении.

Конфигуратор ядра включал в себя несколько больших разделов:

Code maturity level options
Loadable module support
General setup
Floppy, IDE, and other block devices
Networking options
SCSI support
Network device support
ISDN subsystem
CD-ROM drivers (not for SCSI or IDE/ATAPI drives)
Filesystems
Character devices
Sound

Kernel hacking

На Рис. 14 приведен пример выбора сетевых опций и параметров ядра.

Большинство парметров и опций имеет три состояния: включено, выключено и модуль. Последнее определяет, следует ли включать поддерку того или иного протокола, устройства и проч. непосредственно в ядро или же реализовывать данное свойство в виде отделного загружаемого модуля. Среди важнейших компонент, безусловно подлежащих включению в ядро, следует перечислить поддержку протоколов TCP/IP (с полным набором свойств), IPX, AppleTalk и PPP, оптимизацию ядра под процессор Pentium. Напротив, поддержка звуковой карты, контроллеров SCSI, накопителей на магнитной ленте и сетевых плат помимо NE2000-совместимых была выключена. При выходе из конфигуратора была сохранена новая конфигурация ядра.

Была выполнена команда make dep, позволяющая убедиться в том, что соблюдены все зависимости (такие, например, как включаемые файлы), необходимые для последующей компиляции ядра и модулей. Следует отметить, что вышеперечисленные и последующие команды make выполяются в директории /usr/src/linux-2.0.36.

После успешного выполнения make dep была дана команда make clean, результатом действия которой является удаление объектных файлов и прочего, оставшегося после предыдущего процесса компиляции ядра.

Командами make modules и make modules_install, соответственно, была произведена сборка и инсталляция выбранных модулей. Собранные модули находятся в каталоге /lib/modules/2.0.36-0.7.

Перед финальной сборкой ядра был откорректирован файл /etc/lilo.conf для последующей загрузки вновь собранного ядра из корня файловой системы:

```
[root@unix /etc]# less ./lilo.conf
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
image=/vmlinuz
      label=linux
      root=/dev/hda1
      read-only
```

Далеее, собственно, основная операция: make zlilo. Операция собственно сборки ядра заняла около 25 минут. После успешного завершения была произведена общая перезагрузка системы.

3.2.3 Пользовательские бюджеты

Одной из необходимых процедур был переход к системе теневых паролей, хранящихся в файле /etc/shadow. Вот некоторые улучшения, внесенные в систему бюджетов пакетом Shadow-Suite:

- добавлен файл предворительных установок процесса login (/etc/login.defs)
- добавлены утилиты добавления и модификации пользовательских бюджетов и групп;
- появилось свойство устаревания бюджетов и паролей;
- добавлено опциональное свойство групповых теневых паролей;
- появилось опциональное свойство удвоенной длины пароля;
- улучшен контроль над выбором паролей пользователей;
- добавлены Dial-up пароли;

После собственно установки пакета при помощи утилиты pwconv файл /etc/passwd вместо своей исходной структуры *username:Npge08pfz4wuk:503:100:Full Name: /home /username: /bin /sh* приобрел следующий вид: *username:x:503:100:Full Name: /home /username: /bin /sh*. Появился файл /etc/shadow следующей структуры: *username:passwd:last:may:must:warn:expire:disable:reserved*, где: username - имя пользователя, passwd - зашифрованный пароль, last - количество дней с 1 Января 1970, прошедших с последнего изменения пароля, may - количество дней, по прошествии которых пароль можнт быть изменен, must - количество дней, по прошествии которых парол должен быть изменен,warn - количество дней перед предупреждением о необходимости замены пароля, expire - количество дней, после которых просроченный бюджет должен быть выключен, disable - количество дней после 1 Января 1970 года, после которого бюджет выключается, reserved - зарезервированное поле. Вот пример записи файла /etc/shadow:

username:Npge08pfz4wuk:9479:0:10000::::

Добавление пользователей системы производилось при помощи команды adduser, пароль задавался командой passwd [имя пользователя]. Дисковые квоты пользователей были установлены при помощи утилиты конфигурироваия и настройки linuxconf (установка и контроль дисковых квот может производиться и штатными средствами UNIX - с помощью команд quota, quotactl, edquota, quotacheck, quotaon, repquota.) Вот пример включенного квотирования диска пользователя little:

```
[root@unix /root]# quota little
Disk quotas for user little (uid 509):
   Filesystem blocks  quota  limit  grace  files  quota  limit  grace
    /dev/hda1  74426 100000 150000         1262      0      0
```

3.2.4. Сетевые настройки

Часть сетевых настроек выполнена еще в процессе инсталляции. Программа-инсталлятор Linux запросил такие первичные параметры как параметры сетевой карты, IP-адрес ethernet-адаптора (195.133.132.17), имя хоста (unix.px.podolsk.ru), адрес шлюза по умолчанию, статический роутинг иадрес DNS-сервера (195.133.132.17).

3.2.4.1 Протоколы уровня ядра

Низкоуровневые компоненты протоколов TCP/IP, PPP, AppleTalk и IPX реализованы в ядре Linux в процессе его пересборки. Вот некоторые сообщения ядра, иллюстрирующие данный факт:

NET3: Unix domain sockets 0.13 for Linux NET3.035.
Swansea University Computer Society TCP/IP for NET3.034
IP Protocols: IGMP, ICMP, UDP, TCP
Linux IP multicast router 0.07.
Swansea University Computer Society IPX 0.34 for NET3.035
IPX Portions Copyright (c) 1995 Caldera, Inc.
Appletalk 0.17 for Linux NET3.035
....
PPP: version 2.2.0 (dynamic channel allocation)
TCP compression code copyright 1989 Regents of the University of California
PPP Dynamic channel allocation code copyright 1995 Caldera, Inc.
PPP line discipline registered.
SLIP: version 0.8.4-NET3.019-NEWTTY (dynamic channels, max=256).
CSLIP: code copyright 1989 Regents of the University of California.
SLIP linefill/keepalive option.
ne.c:v1.10 9/23/94 Donald Becker (becker@cesdis.gsfc.nasa.gov)
*NE*000 ethercard probe at 0x280: 00 00 b4 3a cb f9*
eth0: NE2000 found at 0x280, using IRQ 12.

3.2.4.2 Настройка сетевых интерфейсов

Прочие сетевые настройки выполнены при помощи утилиты linuxconf, предоставляющей удобный интерфейс конфигурирования сетевых протоколов и интерфейсов. (См. Приложение 7) Собствеенно, файлы этих настроек располагаются в различных частях системы. Файлы и сценарии, отвечающие за инициализацию сетевых интерфейсов ethernet, ppp, lo, slip находятся в каталоге /etc/sysconfig/network-scripts:

chat-ppp0, chat-ppp1, ifcfg-eth0, ifcfg-lo, ifcfg-ppp0, ifcfg-ppp1,
ifdhcpc-done, ifdown, ifdown-post, ifdown-ppp, ifdown-sl, ifup, ifup-aliases,
ifup-ipx, ifup-plip, ifup-post, ifup-ppp, ifup-routes, ifup-sl,
network-functions

Эти файлы используются при вызове сценариев начальной инициализации при загрузке ОС в соответствующем уровне исполнения.
Так, файл /etc/rc.d/init.d/inet используется для запуска, контроля и остановки сетевых сервисов TCP/IP, он определяет имя хоста, запускает супердемон inetd, устанавлеивает параметры IP-роутинга и проч. Сценарий /etc/rc.d/init.d/network отвечает за начальную инициализацию сетевых протоклов.

3.2.4.3 Порты TCP/IP

Файл /etc/services отвечает за соотвестствие определенных сетевых интерфейсов портам TCP/IP. В частности, для функционирования пакета netatalk необходимо было расширить данный описательный файл строчками, закрепляющими порты TCP за протоколами AppleTalk:

```
rtmp        1/ddp       # Routing Table Maintenance Protocol
nbp         2/ddp       # Name Binding Protocol
echo        4/ddp       # AppleTalk Echo Protocol
zip         6/ddp       # Zone Information Protocol

afpovertcp    548/tcp       # AFP over TCP
afpovertcp    548/udp
```

Процесс Listener, “прослушивающий” запросы к серверу баз данных Oracle описан следующим образом:

```
listener      1521/tcp #for Oracle
```

Файл /etc/inetd.conf определяет работу супердемона inetd и порождаемых им процессов сетевого взаимодействия (например, in.ftpd, ipop3d) при поступающих на конкретные порты запросах.

3.2.4.4 Диагностика и отладка

Правильность работы сетевых протоколов и диагностика их работы проведена (и проводится) с помощью следующих утилит:

- netstat позволяет просматривать существующие сетевые соединения на текущий момент, проводить анализ информации о конфигурации интерфейсов, выводить таблицу маршрутизации и получать статические данные о различных сетевых протоколах;
- ifconfig выводит интегрированную статистику, помимо этого, при помощи ifconfig можно на ходу инициализировать и деинициализировать сетевые интерфейсы, задавать IP-адреса и проч.:

```
...
eth0      Link encap:Ethernet  HWaddr 00:00:B4:3A:CB:F9
          inet addr:195.133.132.17  Bcast:195.133.132.31  Mask:255.255.255.240
```

IPX / Ethernet 802.3 addr:00000022:0000B43ACBF9
EtherTalk Phase 2 addr:66 / 6
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:86822 errors:0 dropped:0 overruns:0 frame:0
TX packets:77966 errors:0 dropped:0 overruns:0 carrier:0
collisions:30
Interrupt:12 Base address:0x280
...

В данном примере выведена статистика сетевого интерфейса eth0.

- ping - простейшая утилиты проверки возможности установления IP-соединения:

```
[root@unix /etc]# ping win
PING win.px.podolsk.ru (195.133.132.25): 56 data bytes
64 bytes from 195.133.132.25: icmp_seq=0 ttl=32 time=18.1 ms
64 bytes from 195.133.132.25: icmp_seq=1 ttl=32 time=0.7 ms
64 bytes from 195.133.132.25: icmp_seq=2 ttl=32 time=0.7 ms
```

- route - утилита, определяющая статические маршруты и позволяющая выводить таблицу маршрутизации

- arp - данная команда обращается к таблице ядра, в которой задано соответствие IP-адресов аппаратным адресам:

```
[root@unix /etc]# arp -a
bigmac.px.podolsk.ru (195.133.132.18) at 00:05:9A:40:D1:DC [ether] on eth0
```

- tcpdump - утилита слежения за сетевым трафиком в сети;
- traceroute позволяет выявить последовательность шлюзов, через которую проходит IP-пакет на пути к пункту своего назначения;

В системе существует несколько файлов диагностики проколов и соединений IPX:
- /proc/net/ipx_interface - данный файл содержит информацию о IPX-интерфейсе, которые могут быть настроены вручную или же автоматическим детектированием и конфигурированием:

```
[root@unix rpm]# less /proc/net/ipx_interface
Network   Node_Address   Primary  Device    Frame_Type
00000022  0000B43ACBF9   Yes     eth0      802.3
```

- /proc/net/ipx_route данный файл отвечает за таблицу маршрутизации IPX:

```
[root@unix rpm]# less /proc/net/ipx_route
Network   Router_Net   Router_Node
00000022  Directly     Connected
```

- /proc/net/ipx - распечатывает таблицу IPX-сокетов, открытых для пользования;

Большинство сетевых настроек RedHat 5.2 Linux позволяет сделать утилита linuxconf, внешний вид которой представлен в Приложении 7.

3.2.4.5 Настройка DNS

Рассмотрим настройку собственного сервера DNS, реализована на системе BIND.

Как уже рассказывалось в разделе 2.3.5.3, DNS является распределенной базой данных, отвечающей за преобразование численных значений IP-адресов в символьные и обратно. DNS реализована в виде системы BIND версии 8.1.2, входящей в состав дистрибутива Linux RedHat 5.2. Система основана на :

1. Демоне named
2. Главном конфигурационном файле /etc/named.conf
3. Файлах, задающих прямые и реверсивные зоны сетей, обслуживаемых данным DNS (/var/named)

Документация системы (главный документ - BIND-HOWTO) дала практически все ответы на вопросы настройки и запуска системы.

Соответствующие файлы конфигурации приведены в Приложении 10.

Собственное пространство IP-адресов сети состоит из двух сетей 195.133.132.16, subnet mask 255.255.255.240, которая была выделена провайдером Интернет и интранет сети 172.16.200.1, состоящей из двух адресов 172.16.200.1 (here.ppp.px.podolsk.ru) и 172.16.200.2 (there.ppp.px.podolsk.ru).

Запись

```
zone "0.0.127.in-addr.arpa" {
      type master;
      file "named.local";
};
```

файла /etc/named.conf и файл named.local отвечают за преобразование служебного адреса закольцованного интерфейса localhost.

Прямая зона домена px.podolsk.ru определяется записью:

```
zone "px.podolsk.ru"{
      type master;
      file "named.hosts";
};
```

Соотвествующий файл named.hosts определяет преобразование символьных имен машин внутренней сети офиса в численные:

```
[root@unix /etc]# nslookup win
Server:  unix.px.podolsk.ru
Address:  195.133.132.17

Name:    win.px.podolsk.ru
Address:  195.133.132.25
```

Обратное преобразование задается реверсивной зоной:

```
zone "132.133.195.in-addr.arpa" {
      notify no;
      type master;
     file "16.132.133.195.rev";
          };
```

Проверим работу реверсивной зоны командой nslookup:

```
[root@unix /etc]# nslookup 195.133.132.18
Server:  unix.px.podolsk.ru
Address:  195.133.132.17

Name:    bigmac.px.podolsk.ru
Address:  195.133.132.18
```

Следует добавить, что запуск демона named совершается стандартным скриптом /etc/rc.d/init.d и линками на него из директорий соответствующих уровней запуска системы. После запуска named он начинает отвечать на запросы DNS, приходящие на 53 порт TCP.

3.2.5 Прочие индивидуальные настройки

К прочим настройка относятся, например, написание собственных shell-сценариев редактирование уровней выполнения (init), русификация консолей, установка шрифтов и прочее.

Система уровней исполнения системы реализована в RedHat достаточно элегантным образом. Рассмотрим содержимое каталога /etc/rc.d:

init.d
rc
rc.local
rc.news
rc.serial
rc.sysinit
rc0.d
rc1.d
rc2.d
rc3.d
rc4.d
rc5.d
rc6.d

Помимо начальных загрузочных процедур rc.serial, rc.sysinit и rc (последний файл отвечает за переходы из одного уровня исполнения в другой) в каталоге присутствуют каталоги rc x .d, каждый из которых определяет свой набор процедур старта и остановки соответствующих ему программ. Собственно, стандартные скрипты запуска полного набора программ находятся в каталоге initd.d, а в каталогах rc x.d находятся лишь псевдонимы (ссылки) на определенные сценарии каталога init.d.

Уровни исполнения удобно редактировать из X-приложения contol panel. Так, например, требовался запуск установленного в системе Web-сервера Apache на стандартном третьем уровне исполнения. Для этого соответствующий скрипт старта демона httpd, находящийся в init.d был продублирован в виде символьной ссылки в каталоге rc3.d.

3.2 Компоненты сервера

3.2.1 Установка и настройка системы X Window

Установка системы X Window состояла из дух частей: настройки локального X-сервера, приложений и Window-менеджеров и настройки удаленного X-сервера на рабочей станции Macintosh и системы xdm.

Следующие компоненты ситемы X Window были утановлены на Linux-машину:

- сервер XFree86 -S3V, локальный X-сервер с поддержкой видеокарт типа S3Trio Virge;
- различные билиотеки системы X Window;
- билиотеки gnome-0.20, используемые X-утилитами конфигурирования системы (control panel, xlinuxconf, pakage manager и др.);
- средства поддержки протокола X11R6;
- различные Window-менеджеры (оболочки), такие как AfterStep - 1.5 , WindowMaker -0.20.1, fvwm2 -2.0.46;
- собственно система XFree86 и шрифты для нее;
- средства конфигурирования режимов работы сервера Xconfigurator и XF86Setup;
- различные системные и пользовательские X-приложения;

Следующей фазой было настройка работы X-сервера XFree86 - S3V. Выбор видеокарты, установка параметров монитора и настройка X-сервера была произведена при помощи специальных утилит Xconfigurator и xf86config. Полученная конфигурация видеорежимов размещается в файле XF86Config. При помощи ручного редактирования этого файла реальное разрешение экрана было поставлено в соответствие виртуальному экрану X Window.

Автором проекта не подразумевалась работа в системе X Window на локальной Linux-машине, поэтому встала необходимость настройки клиент-серверных приложений X Window- приложений, соединяющихся с помощью локальной сети.

В качестве X -сервера была выбрана программа eXodus 6.1.2 фирмы WhitePine Inc. Установка ее не была связана с какими либо трудностями.

На клиентской стороне (а именно, Linux-машины) необходимо было произвести настройку системы xdm (X Window Display Manager). Именно это приложение должно разрешать соединение клиентских и серверных частей, производить аутентификацию пользователей и диспетчеризацию X-приложений.

Файлы конфигурации xdm расположены в каталоге /etc/X11/xdm:

GiveConsole
TakeConsole
Xaccess
Xresources
Xservers
Xsession

```
Xsetup_0
chooser
xdm-config
```

Файл Xaccess отвечает за разрешение установления соединения по TCP/IP:

```
# Access control file for XDMCP connections
#
# To control Direct and Broadcast access:
#
 *.px.podolsk.ru
# 195.133.132.17
.....
```

Строка с именем домена px.podolsk.ru разрешает соединение с серверной стороны только с рабочих станций локальной офисной сети.

Файл Xservers жестко закрепляет за рабочей станцией bigmac.px.podolsk.ru X-дисплей с номером 0:

```
# This file should contain an entry to start the server on the
# local display; if you have more than one display (not screen),
# you can add entries to the list (one per line). If you also
# have some X terminals connected which do not support XDMCP,
# you can add them here as well. Each X terminal line should
# look like:
#       XTerminalName:0 foreign
#
# :0 local /usr/X11R6/bin/X
bigmac:0 foreign
....
```

В файле Xresourcess определяется параметры процесса аутентификации пользователя и внешний вид графического окна login:

```
[root@unix xdm]# less Xresources
!$XConsortium: Xresources /main/8 1996/11/11 09:24:46 swick $
xlogin*login.translations: #override\
      Ctrl<Key>R: abort-display()\n\
      <Key>F1: set-session-argument(failsafe) finish-field()\n\
      Ctrl<Key>Return: set-session-argument(failsafe) finish-field()\n\
      <Key>Return: set-session-argument() finish-field()
xlogin*borderWidth: 5
xlogin*greeting: Welcome to PROMEXPORT XDM Service
xlogin*namePrompt: login:\040
xlogin*fail: Fuck U! Your authentification failed!
#ifdef COLOR
```

```
xlogin*greetColor: #ff00ff
xlogin*failColor: red
*Foreground: #000000
*Background: #fffff0
#else
xlogin*Foreground: black
xlogin*Background: white
#endif

XConsole.text.geometry: 480x130
XConsole.verbose:       true
XConsole*iconic:        true
XConsole*font:          fixed

Chooser*geometry:               700x500+300+200
Chooser*allowShellResize:       false
Chooser*viewport.forceBars:     true
Chooser*label.font:             *-new century schoolbook-bold-i-normal-*-240-*
Chooser*label.label:            XDMCP Host Menu  from CLIENTHOST
Chooser*list.font:              -*-*-medium-r-normal-*-*-230-*-*-c-*-iso8859-1
Chooser*Command.font:           *-new century schoolbook-bold-r-normal-*-180-*
```

В Приложении 4 приводится вид окна-приглашения к аутентификации.

Центральный файл конфигурации XDM - файл xdm-config:

```
[root@unix xdm]# less ./xdm-config
! $XConsortium: xdm-conf.cpp /main/3 1996/01/15 15:17:26 gildea $
DisplayManager.errorLogFile:    /var/log/xdm-error.log
DisplayManager.pidFile:         /var/run/xdm.pid
DisplayManager.keyFile:         /etc/X11/xdm/xdm-keys
DisplayManager.servers:         /etc/X11/xdm/Xservers
DisplayManager.accessFile:      /etc/X11/xdm/Xaccess
! All displays should use authorization, but we cannot be sure
! X terminals will be configured that way, so by default
! use authorization only for local displays :0, :1, etc.
DisplayManager._0.authorize:    true
DisplayManager._1.authorize:    true
! The following three resources set up display :0 as the console.
DisplayManager._0.setup:        /etc/X11/xdm/Xsetup_0
DisplayManager._0.startup:      /etc/X11/xdm/GiveConsole
DisplayManager._0.reset:        /etc/X11/xdm/TakeConsole
!
DisplayManager*resources:       /etc/X11/xdm/Xresources
DisplayManager*session:         /etc/X11/xdm/Xsession
DisplayManager*authComplain:    false
```

В данном файле задаются параметры менеджера дисплеев, местонахождение уже рассмотренных файлов кофигурации, разрешения авторизации, путь к

регистрационному файлу xdm и проч.

Настроенная система прекрасно эксплуатируется. Следует отметить, что данное решение позволяет обойтись без дорогостоящей видеоподсистемы сервера, используя при этом графическую станцию Macintosh S900 с 17-ти дюмовым монитором в качестве сетевого X-терминала.

3.2.2 Установка и настройка Apache

В качестве Web-сервера фирмы был (как это уже говорилось в разделе 2.3.4.4) использован свободно распространяемый программный продукт Apache 1.3.4PL28.9rus. В составе дистрибутива RedHat поставлялся Apache версии 1.3.3, но данная пакет не мог бы удовлетворять требованию по работе с русскоязычными HTML-документами без специальных исправлений.

В отличии от rpm-версии, поставляемой в дистрибутиве, русская версия Apache пока поставляется только в архиве исходных текстов. После распаковки архива с исходными текстами и изучения документации была выполнено редактирование файла конфигурации httpd.conf-dist в каталоге conf. Далее был выполнен обработчик конфигурациии ./configure:

```
[root@unix apache_1.3.4rusPL28.9]# ./configure
Configuring for Apache, Version 1.3.4
 + using installation path layout: Apache (config.layout)
Creating Makefile
Creating Configuration.apaci in src
Creating Makefile in src
 + configured for Linux platform
 + setting C compiler to gcc
 + setting C pre-processor to gcc -E
 + checking for system header files
 + adding selected modules
   o charset_module uses ConfigStart/End
 + doing sanity check on compiler and options
Creating Makefile in src/support
Creating Makefile in src/main
Creating Makefile in src/ap
Creating Makefile in src/regex
Creating Makefile in src/os/unix
Creating Makefile in src/modules/extra
Creating Makefile in src/modules/standard
```

Как мы видим, были созданы make-файлы в соответствующих каталогах, установлена платформа инсталляции (Linux), в качестве компилятора установлен gcc. Командами make и make install была произведена соответственно компиляция программы и модулей и установка компонент системы согласно выбранному месторасположению.

В качестве основного каталога Apache использован /usr/local/apache. Был выполнена настройка параметров сервера в основном файле конфигурации /usr/local/apache/conf/httpd.conf. В частности, было уменьшено количество демонов httpd, запускаемых при старте сервера.

Сценарий начальной загрузки Web-сервера был размещен в каталоге /etc/rc.d/init.d:

```
[root@unix apache]# less /etc/rc.d/init.d/apache
```

```
#!/bin/sh
#
# Apache 1.3.4 Startin', by MacDuck
#

# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0

# Check that httpd.conf exists.
[ -f /usr/local/apache/conf/httpd.conf ] || exit 0

# See how we were called.
        echo -n "Starting PROMEXPORT Web Server under Russian Apache..."
        daemon /usr/local/apache/httpd -f /usr/local/apache/conf/httpd.conf
        echo
exit 0
(END)
```

Установка Web-сервера была завершена созданием символических ссылок на сценарий запуска в директории соответствующих уровней исполнения и проверкой его функционирования. В настоящее время сервером обслуживается сайт-фирмы, и внутренний интранет-сервер документации.

3.2.3 Установка и настройка Netatalk

Одной из важнейших компонент файлового сервера фирмы является сервер AppleShareIP, реализуемый пакетом netatalk-1.4b2+asun2.1.0 (см. Раздел 2.3.1.3).

После неоднократных неудачных попыток собрать данный программный продукт из исходных тесктов, с базового ftp-сервера netatalk была сгружена RPM-версия пакета.

Инсталляция данного пакета (как и всех RPM) была выполнена стандартной командой rpm -ih ./netatalk-1.4b2+asun2.1.0-5.i386.rpm.

Демоны соответствующих процессов размещены в каталоге /usr/sbin (afpd, papd, atalkd), программы регистрации устройств-участников сети AppleTalk nbprgstr и nbpunrgstr находятся в каталоге /usr/bin.

Конфигурационные файлы пакета размещены в директории /etc/atalk:

AppleVolumes.default
AppleVolumes.system
afpd.conf
atalkd.conf
papd.conf

Первые два файла определяют директории файловой системы Linux, подлежащие выделению в качестве разделяемого сетевого ресурса AppleShare. В качестве сетвеого раздела Macintosh был выбран каталог /usr/UNIXatalk/

Строка eth0 -phase 2 -net 0-65534 -addr 66.6 в файле atalkd.conf определяет привязку AppleTalk phase 2 к интерфейсу eth0, диапазон допустимых в локальной ethernet-сети номеров сетей AppleTalk (0-65534); последний параметр определяет сосбственный сетевой EtherTalk-адрес сервера (зона 66, адрес 6).

Файл afpd.conf отвечает за название сервера в сети и параметры аутентификации на сервере.

Традиционное для RedHat расположение сценария запуска atalk.init и останова процесса - каталог /etc/rc.d/init.d. при этом процессы сервера запускаются и останавливаются командами atalk start и atalk stop. Ниже приводится пример системных сообщений о старте сервера:

Apr 17 19:27:19 unix atalkd [6434]: restart (1.4b2+asun2.1.0)
Apr 17 19:27:20 unix atalkd [6434]: zip_getnetinfo for eth0
Apr 17 19:27:39 unix last message repeated 2 times
Apr 17 19:27:49 unix atalkd [6434]: config for no router
Apr 17 19:27:50 unix atalkd [6434]: ready 0/0/0
Apr 17 19:28:02 unix papd [6445]: restart (1.4b2+asun2.1.0)
Apr 17 19:28:09 unix afpd [6454]: UNIXatalk:AFPServer@ started on 66.6:128 (1.4)*
Apr 17 19:28:09 unix afpd [6454]: ASIP started on 195.133.132.17:548(2) (1.4b2+a

Псевдонимы shell-сценария старта и останова размещены в каталоге 3-го, 4-го и 5-го уровней запуска системы, что обезпечивает автоматический запуск сервера при начальной загрузке системы.

Экспериментальным путем установлена примерная скорость копирования

файлов со станций Macintosh на сервер netatalk, равная, примерно, 600 килобайтам в секунду. Данный показатель весьма высок для сетей Macintosh, использующих технологию AppleShare.

Информационное окно системы MacOS, выдаваемое при монтировании раздела сетевого раздела UNIXatalk Linux-сервера представлено на Рис.15 :

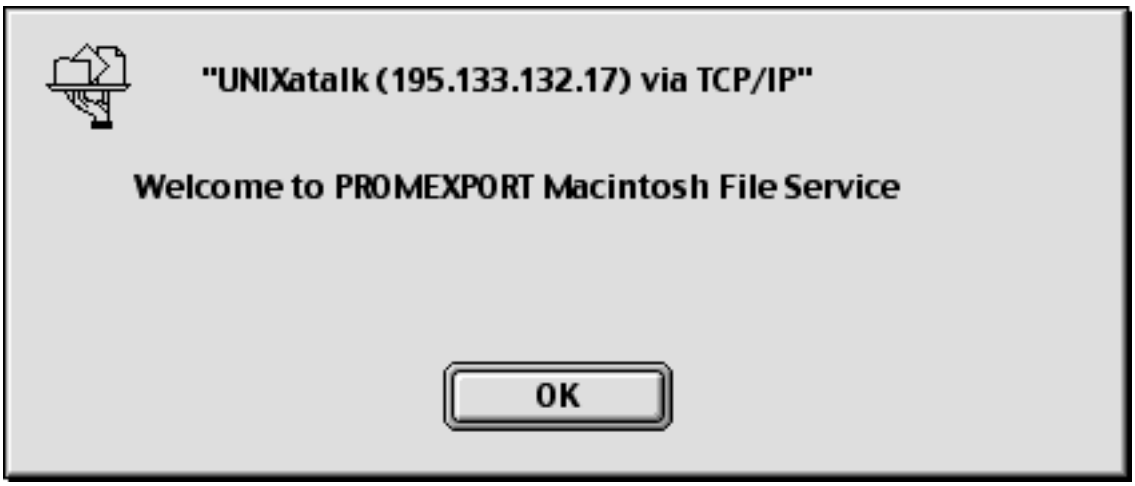


Рис. 15

3.2.4 Установка и настройка MarsNWE

В качестве эмулятора сервера NetWare 3.12 был взят программный пакет mars-nwe-0.99pl10. Об использовании данного пакета рассказано в разделе 2.3.1.2.

В состав дистрибутива RedHat 5.2 входит RPM-версия пакета mars-nwe-0.99pl10.

Соответственно, инсталляция пакета выполнена стандартной командой rpm -ih mars-nwe-0.99pl10-1.i386.rpm.

Базовый каталог установленного пакета - директория /var/mars_nwe:

bindery
sys

Каталог bindery отвечает за собственную аутентификационную базу эмулятора, а sys является его системным разделом. Основным конфигурационным файлом пакета mars_nwe является /etc/nwserv.conf. Данный файл имеет большой размер, позволю цитировать лишь отдельные строчки, измененные при настройке mars_nwe:

```
# This is the configuration-file for "mars_nwe", a free netware-emulator
# for Linux.
...
# Section 1: volumes (required)
# Use /var/netware for the SYS volume, and make it read-only
        1       SYS     /var/mars_nwe/sys       koO
        1     CDROM   /mnt/cdrom              kmr
        1       DOC     /usr/doc            rk
# Section 2: servername (optional)
    2     Netware   # name of the server would be "MARS"
# Section 4: IPX-devices (strongly recommended)
      4     0x22   eth0   802.3   1
# Section 7: password handling of DOS-clients (required)
# FLAG:
#      0     enforce encryption of _all_ passwords by the DOS-client
#            (default)
#      1     as "0", but allow the non-encrypted version of the
#            "change password"-routine.
#      7     allow all non-encrypted stuff but no empty nwe passwords.
#      8     allow all non-encrypted stuff and also allow empty
#            nwe-passwords.
#      9     use all non-encryted calls + "get crypt key" will allways fail
#            so the login program will use the old unencryted calls.
#            this will *not* work with all clients !! (OS2/client)
7     0
# Section 10: UID and GID with minimal rights
     10     99
     11     99
# Section 12: supervisor-login (required)
# Syntax:
```

```
#      12     NW_LOGIN       LINUX_LOGIN     [PASSWORD]
#      12     SUPERVISOR     nw-adm          top-secret
12  SUPERVISOR   adm       *
# Section 13: user-logins (optional)
13      dima    dima ****
13      little    little  ****
13      yura    yura    ****
13      guest  guest
# Section 15: automatic mapping of logins (decision required)
# ---------------------------------------------------------------------
# Syntax:
#      15     FLAG    DEFAULT_PASSWORD
# FLAG:
#      0      DON'T map the Linux-logins automatically to
#             "mars_nwe"-logins (default)
#      1      YES, DO the automatic mapping and provide every login
#             created this way with the common password given with
#             "DEFAULT_PASSWORD"
#      99     re-read the logins from  /etc /passwd and overwrite even the
#             already existing logins from the bindery (this will also
15   99
```

В данном примере реальные пароли пользователей заменены звездочками. Заметим, что секции описания разделов NetWare включены три ресурса: это директория /var/mars_nwe/sys (раздел sys), директория /mnt/cdrom (раздел CDROM) и директория /usr/doc с документацией Linux (раздел doc). Флаги r в строчках последних двух означают, что данные разделы доступны только для чтения.

Следует заметить, что после любого изменения пользовательских бюджетов mars_nwe или системы аутентификации в конфигурационном файле, необходимо удалить содержимое каталога bindery. Обновленная система аутентификации будет создана заново при рестарте сервера mars_nwe.

Стартовый скрипт /etc/rc.d/init.d/mars-nwe имеет четыре параметра start, stop, status, restart. Сервер автоматически загружается при общем старте в 3-м уровне исполнения. Работа mars_nwe представлена тремя демонами nwserv,nwbind и ncpserv плюс по одному процессу на каждый активный клиентский процесс.

Сервер обслуживает ограниченное число клиентов и преимущественно используется для долговременных архивов драйверов аппаратного обеспечения фирмы и частоупотребимого программных продуктов.

3.2.5 Установка Oracle 8

Необходимо заметить, что установка Oracle 8.0.5 for Linux оказалась весьма трудоемкой, сложной и длительной процедурой.

В качестве руководства по установке использовался документ [10].Шаги, предшествующие непосредственно инсталляции, заключались в следующем. Прежде всего необходимо было создать точки монтирования программных и табличных областей. Для этого был создан каталог /u04. Модель расположения структур Oracle OFA (Optimal Flexible Architecture) предусматривала как минимум четыре точки монтирования, в нашем же случае была использована простая модель с одной точкой монтирования.

Были созданы пользовательские бюджеты oracle и dba, принадлежащие группе dba. Далее в каталоге /mnt/cdrom был смонтирован инсталляционный диск Oracle 8. В профайл пользователя oracle были внесены необходимые переменные окружения:

```
#for Oracle
ORACLE_OWNER=oracle; export ORACLE_OWNER
ORACLE_BASE=/u04/app/oracle; export ORACLE_BASE
ORACLE_HOME=/u04/app/oracle/product/8.0.5; export ORACLE_HOME
LD_LIBRARY_PATH=/u04/app/oracle/product/8.0.5/lib; export LD_LIBRARY_PATH
ORACLE_SID=fuck; export ORACLE_SID
ORACLE_TERM=vt220; export ORACLE_TERM
ORA_NLS33=$ORACLE_HOME/ocommon/nls/admin/data; export ORA_NLS33
PATH="$PATH:/bin:/usr/bin:/usr/local/bin:$ORACLE_HOME/bin"
TMPDIR=/var/tmp; export TMPDIR
#end for Oracle
```

После этого необходимо было сформировать файл /etc/oratab. Для этого от пользователя root с предварительно установленной переменной окружения ORACLE_OWNER=oracle был запущен скрипт /mnt/cdrom/orainst/oratab.sh. Убедившись в том, что файл создан и имеет правильные права доступа и принадлежность, наступило время приступить непосредственно к инсталляции.

Главный инсталляционный скрипт /mnt/cdrom/orainst/orainst был запущен от имени пользователя oracle. Автором была выбрана выборочная инсталляция (Custom Install). Далее были выбраны точка монтирования, пути, некоторые опции установки и источник, с которго производить инсталляцию. Так в качестве языка в окне NLS (National Language Support) была выбрана поддержка русского. В результате этих действий программа-инсталлятор вывела на экран напоминание:

Post-installation steps that need to be run by root will be written to /u04/app/oracle/product/8.0.5/orainst/root.sh.

(Необходимые послеинсталляционные шаги, которые необходимо выполнить от пользователя root будут записаны shell-процедуре /u04/app/oracle/product/8.0.5/orainst/root.sh.)

Далее было предложено выбрать компоненты поставки, подлежащие инсталляции.

Вид экрана инсталлятора на этом шаге отображен на Рис.16

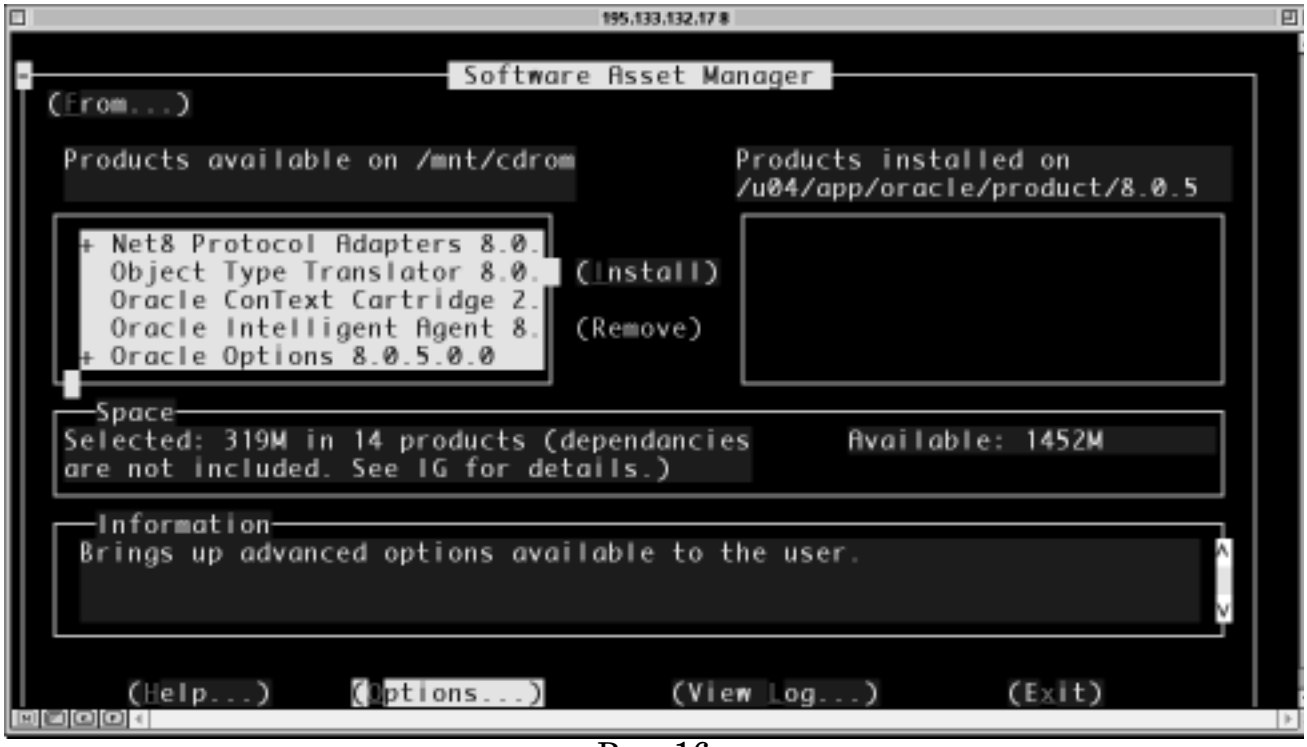


Рис. 16

После выверки зависимостей устанавливаемых компонент необходимо было выбрать тип хранения баз данных: основанный на существующей файловой системе или на специально выделенном устройстве. Естественным для нашего варианта является первое.

Далее необходимо было выбрать количество точек монтирования объектов баз данных; была выбрана единственная точка монтирования /u04. В опции NLS Support была выбрана кодировка внутреннего хранения баз данных CL8ISO8859P2, соотвествующая стандарту ISO 8859-5 Cyrillic 8-bit.

После двоекратного ввода системных паролей пользователей oracle и dba и указания количества конкурирующих пользователей класса dba, произведена конкретизация версий устанавливаемых драйверов JDK и JDBC. Далее следовала длинная череда уточнений путей, опций, типа устанавливаемой документации (html или pdf -фоматы); автор не считает нужным подробно останавливаться на этом этапе.

Экран инсталлятора в процессе копирования выбранных и сконфигурированных компонент представлен на Рис. 17

После выхода из инсталлятора необходимо было выполнить послеинсталляционные шаги и настройку компонент Oracle 8. К ним относятся: запуск shell-скрипта root.sh в каталоге /$ORACLE_HOME/orainst, устанавливающего соотвествующие права к файлам и директориям Oracle, запуск SQL-скрипта catrep.sql от пользователя sys в интерпретаторе SQLPlus, редактирование файла /etc/oratab, написание сценария автоматического старта и останова Oracle-сервера и создания соответствующих псевдонимов в уровнях запуска Linux.

Внесены необходимые изменения в файл /etc/services, описывающий соотвествие приложений портам TCP/IP (добавлена строка listener 1521/tcp #for

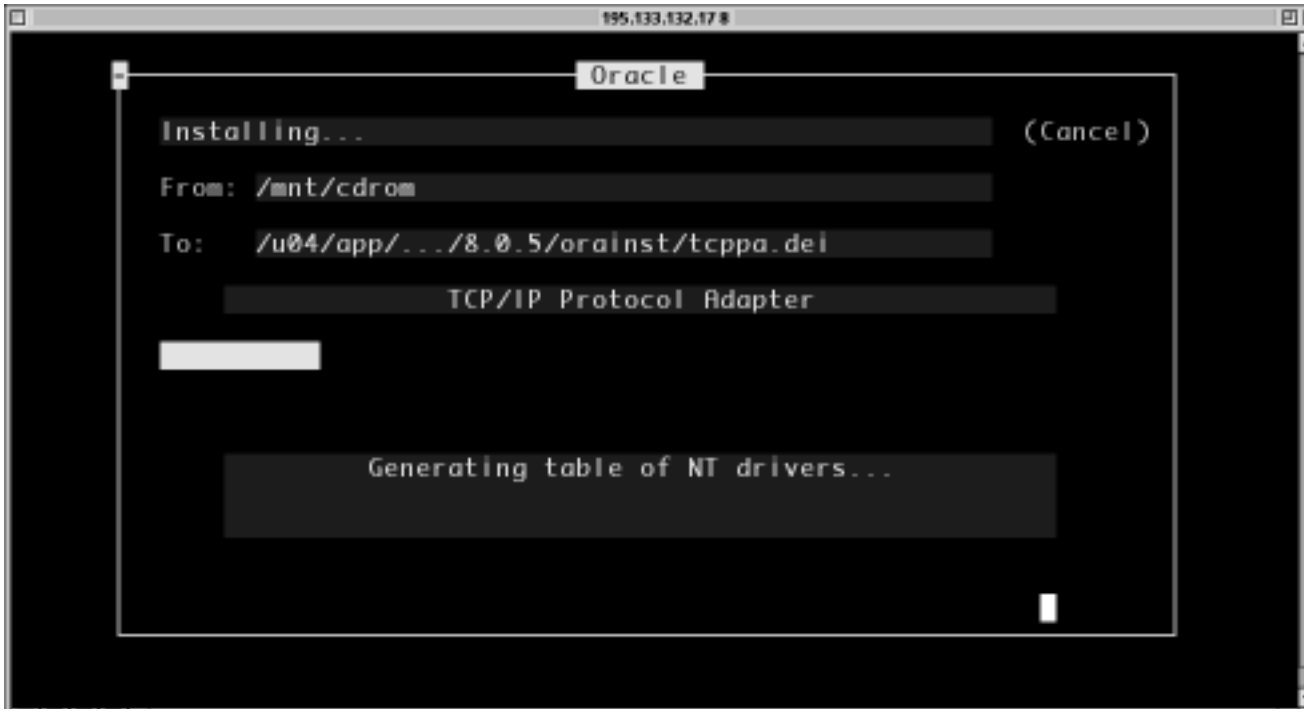


Рис.17

Oracle. Такая операция потребовала перезапуска супердемона inetd.

Произведена диагностика компоненты-демона Oracle, “прослушивающей” упомянутый 1521 порт TCP:

```
[oracle@unix admin]$ lsnrctl status
LSNRCTL for Linux: Version 8.0.5.0.0 - Production on 11-APR-99 16:23:13

(c) Copyright 1997 Oracle Corporation.  All rights reserved.
Connecting to (ADDRESS=(PROTOCOL=IPC)(KEY=fuck))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 8.0.5.0.0 - Production
Start Date                10-APR-99 22:32:48
Uptime                    0 days 17 hr. 50 min. 28 sec
Trace Level               off
Security                  OFF
SNMP                      OFF
Listener Parameter File   /u04/app/oracle/product/8.0.5/network/admin/listener.a
Listener Log File         /u04/app/oracle/product/8.0.5/network/log/listener.log
Services Summary...
  extproc                 has 1 service handler(s)
  fuck                    has 2 service handler(s)
The command completed successfully
```

Общий процесс установки был завершен проверкой работы скрипта пуска/запуска и инструктажем администратора баз данных фирмы.

3.2.6 Настройка сервера удаленного доступа

Настройка сервера удаленного доступа состояла из нескольких фаз:

- настройка протокола PPP;
- настройка модемного порта;
- установка пакета mgetty;
- настройка параметров PPP-соединения;
- настройка времени работы удаленного доступа;

Основным протоколом работы сервера удаленного доступа является PPP (См. разделы 2.2.4 и 2.3.6). Низкоуровневая компонента протокола реализована в ядре Linux (См. раздел "3.2.2 Пересборка ядра и модулей):

…
PPP: version 2.2.0 (dynamic channel allocation)
TCP compression code copyright 1989 Regents of the University of California
PPP Dynamic channel allocation code copyright 1995 Caldera, Inc.
PPP line discipline registered.
…

Остальные компоненты пакета ppp-2.3.5, такие как набор shell-скриптов работы с PPP-соединениями и демон pppd, были установлены во время инсталяции ОС LInux.

Строка в сценарии инициализации портов /etc/rc.d/rc.serial определила параметры модемного порта com3 (ttyS2 и cua2 в терминологии Linux):

….
 # These are the standard COM1 through COM4 devices
#
#${SETSERIAL} /dev/cua0 uart 16550A port 0x3F8 irq 4 ${STD_FLAGS}
#${SETSERIAL} /dev/cua1 uart 16550A port 0x2E8 irq 3 ${STD_FLAGS}
${SETSERIAL} /dev/cua2 uart 16550A port 0x3E8 irq 5 ${STD_FLAGS}
#${SETSERIAL} /dev/cua3 uart 16450 port 0x2E8 irq 3 ${STD_FLAGS}
…

Проверка правильности работы модема была проведена при помощи коммуникационной программы minicom:

[root@unix rc.d]# minicom

Welcome to minicom 1.82

OPTIONS: History Buffer, F-key Macros, Search History Buffer, I18n
Compiled on Oct 1 1998, 23:34:55.

Press CTRL-A Z for help on special keys

ati3
U.S. Robotics Sportster 33600 Voice V4.3.174

OK
atdp632015
CONNECT 14400 / ARQ / V32 / LAPM / V42BIS
Welcome to INET internet service :

free.podolsk.ru (Port ttyd4) !login:

Успешное соединение с модемом, находящимся на другом конце линии, подтвердило правильность настроек.

Следующим этапом была установка и настройка пакета работы с модемами и факсами mgetty-1.1.14. Данный набор компонент позволяет отправлять и принимать факсы, различать входящие звонки, самостоятельно производить дозвонку, а так же различать различные виды приветствований (hanshakes) при установлении модемного соединения. Пакет mgetty в RPM-формате был инсталлирован с установками по умолчанию, полностью отвечавшим функциональным потребностям.

В инициализационный файл /etc/inittab была добавлена следующая строка, определяющая запуск процесса mgetty:

S2:35:respawn:/sbin/mgetty -s 38400 -n 3 -i /etc/mgetty.issue -m '"" AT&F1M0 OK

(См. расшифровку опций строки в Разделе “2.3.6 Сервер удаленного доступа”)
Заметим, что при любых изменениях файла /etc/inittab необходимо перезапустить процесс init при помощи команды kill -HUP 1, где 1 - pid процесса init.

Удостоверимся при помощи команды ps, что mgetty находится среди работающих процессов Linux:

...
528 ? S 0:00 update (bdflush)
3189 ? S 0:00 /sbin/mgetty -s 38400 -n 3 -i /etc/mgetty.issue -m "" AT&F1M
3190 p0 R 0:00 ps ax
256 ? S 0:00 portmap
287 ? S 0:00 /usr/sbin/atd
.....

В принципе, на этом этапе удаленные пользователи после удачного соединения и аутентификации получают доступ к командному shell UNIX-сервера (разумеется, если они обладают правом доступа к shell, см. Раздел “ 2.5.1 Категории пользователей”), но это вряд ли можно было назвать полноценным сервером удаленного доступа.

Для возможности соединения с внутренней сетью фирмы по протоколу TCP/IP и используется туннельный протокол PPP.

Прежде всего, необходимо было установить суид-бит на файл pppd командой

chmod u+s /usr/sbin/pppd:

-rwsr-xr-x 1 root root 74224 Apr 28 07:17 /usr/sbin/pppd

Для запуска PPP был использован shell-сценарий запуска демона pppd пакета ppp-2.3.5 с соответствующими параметрами. В параметрах запуска сценария были указаны локальный и удаленный IP-адреса 172.16.200.1 и 172.16.200.2. Врочем, обойтись было можно и собственным коротким сценарием:

#!/bin/sh

exec /usr/sbin/pppd -detach crtscts lock proxyarp 172.16.201.1:172.16.201.2 /dev/ttyS2 38400 &

exit 0;

Был создан специальный пользователь pppuser, шеллом которого и был установлен упомянутый скрипт:

[root@unix ppp]# less /etc/passwd
....
pppuser:x:507:508::/home/pppuser:/etc/ppp/ppplogin
....

Таким образом, при положительной аутентификации пользователя pppuser при соединении удаленной системы с Linux-машиной, на обоих концах устанавливается PPP-соединение с фиксированными Intranet IP-адресами. При такой организации удаленный пользователь получает доступ ко всем сервисам внутренней сети фирмы, транспортом которых служит протокол TCP/IP.

3.2.7 Конфигурирование сервера новостей

Сервер INN функционирует на базе пакета INN 1.7.2. Данный пакет является стандарным для многих UNIX-систем и входит в большинство дистрибуций ОС Linux. Система INN 1.7.2 была исталлирована в общем процессе установки RedHat 5.2. В разделе 2.3.4.2 описывалась общая структура этого программного продукта и принципы его работы. Остановимся на конкретных настройках в конфигурационных файлах INN каталога /etc/news, вопросах тестирования и отладки news-сервера.

Файл inn.conf содержит информацию о принадлежности сервера:

```
[root@unix news]# less inn.conf
##  $Revision: 1.6 $
##  inn.conf -- inn configuration data
##  Format:
##      <parameter>:<whitespace><value>
organization:   P R O M E X P O R T
server:         localhost
```

Файл nnrp.access разрешает или запрещает соединение с сервером новостей. В данном примере сначала запрещаются любые действия со статьями сервера с любых хостов, предпоследняя строка разрешает чтение и написание статей во все локальные группы, кроме служебных групп test, junk,control и to:

```
[root@unix news]# less nnrp.access
# Default to no access
*:: -no- : -no- :!*
# Allow access from localhost
localhost:Read Post:::*
# 195.133.132.*:Read Post:::promexport*
# 195.133.132.*:Read Post:::junk*
195.133.132.*:Read Post:::*,!test,!junk,!control,!to
# 195.133.132.*:Read Post:::!test*
```

Файл newsfeeds содержит указания на то, какие статьи следует пересылать на нижестоящие узлы. Из-за, собственно, отсутствия нижестоящих узлов пересылки и сложности синтаксиса данного файла (структура файла охватывает самые разные случаи - от получения статей по UUCP-поставщиков до разовой посылки новостей по электронной почте), рассматривать его нет смысла.

В файлах host.nntp и host.nntp.nolimit описываются вышестоящие узлы рассылки, которым разрешено посылать ггруппы новостей и статьи на наш сервер (из-за ограниченности дискового пространства сервера и весьма избирательной заинтересованности пользователей в получении статей с вышестоящего уровня было решено не подписываться на конференции у вышестоящих рассыльщиков):

```
[root@unix news]# less hosts.nntp
##  $Revision: 1.7 $
##  hosts.nntp - names and addresses that feed us news
```

```
##  Format
##      <host>:
##      <host>:<password>
##  <host> can be a name or IP address; no wildcards.  Any hosts not
##  listed here are handed off to nnrpd.
localhost:
[root@unix news]# less hosts.nntp.nolimit
## $Revision: 1.1 $
##
## Any hosts listed in this file will be permitted to connect past the
## limits set up by the `-i' and `-X' flags to innd.
##
localhost
```

Файл expire.ctl используется программой expire, входящей в состав пакета INN и предназначенной для удаления статей с истекшим сроком хранения. После удаления статьи запись о ее идентификаторе остается в системе еще некоторое время, чтобы статью можно было отклонить, если поставщик предложит ее вновь. Это позволяет избежать дублирования публикаций.

```
/remember/:14

##  Keep for 1-10 days, allow Expires headers to work.
*:A:1:10:30
```

В строке remember задается срок , по истечении которого из ситемы удаляются идентификаторы старых статей. В нашем примере это 14 дней. В следующей командной строке указано, что все (*) конференции любого (A) типа (с модератором и обычные) хранятся минимально 1 день, по умолчанию 10 дней, а максимально 30. Каждая совокупность конференций может иметь свои параметры устаревания.

Файл cleanfeed.conf определяет прочие многочисленные параметры конференций, такие как разрешение или запрещение посылки бинарного файла в теле сообщения, максимальный размер сообщения, число строк, разрешение применения специальных фильтров, разрешения или запрещения инкапсулирования html-документов и другие. Файл достаточно громоздок и нашему рассмотрению не подлежит.

Файл /var/lib/news/active содержит перечень конференций, о которых известно ситеме:

```
[root@unix news]# less active
control 0000000002 0000000003 y
junk 0000000001 0000000002 y
test 0000000003 0000000004 y
to 0000000001 0000000002 y
promexport.local 0000000004 0000000005 y
pxtest 0000000000 0000000001 y
```

Кроме того, регистрируются номера хранящихся в системе статей по каждой конференции и статус приема и публикации по каждой из них.
Значения опций могут принимать следующие значения: y - конференция принимается и возможна публикация, n - конференция существует, но не на этом узле, прием и публикация невозможны, m - конференция с модератором, принимается, но публикация невозможна, x - конференция принимается, но публикация невозможна, j - статьи конференции не хранятся, а только передаются.

В директории /usr/lib/news/bin хранится основная часть программ пакета INN. Часть из них запускается из собственных shell-сценариев INN, другие допускают обычный пользовательский запуск. Так, например, программа innstatus выводит следущую информацию о состоянии системы:

```
[root@unix bin]# ./innstat
Server status:
Server running
Allowing remote connections
Parameters c 14 i 0 (0) l 75000 o 243 t 300 H 2 T 60 X 0 normal specified
Not reserved
Readers separate enabled
Perl filtering enabled
History cache: 1 lookups, 0 hits
Precommit cache: 2 lookups, 0 hits
Disk usage:
/dev/hda1          2359165 1492410   744789     67%   /
Batch file sizes:
Log file sizes:
  0 badcontrol.log     1 news             0 nntpsend.log
  0 errlog           0 news.crit         0 unwanted.log
  6 expire.log         0 news.err
  0 innfeed.log        0 news.notice
Lock files:
LOCK.innwatch
Server connections:
innfeed!:18:proc:519          crosspost:16:proc:517
overview!:17:proc:518
```

Центральным “пультом управления” пакета INN является интерактивная программа ctlinnd. Она позволяет останавливать и запускать сервер, менять аттрибуты конференций, временно преостанавливать работу сервера, удалять и создавать группы, пересчитывать файлы конфигурации и многое другое.

К сожалению, объем данной дипломной работы не позволяет подробно остановиться на всех вопросах функционирования INN.

3.2.8 Установка и настройка Samba

Как и многие программные продукты, используемые на Linux-машине, пакет Samba был установлен при инсталляции ОС Linux. При всей функциональной ценности пакета, он достаточно легко настраивается.

Функционирование сервера Smb обеспечивается работой демонов nmbd (обеспечивает клиентам netbios-поддержку имен) и smbd (обеспечивает клиентам smb-сервис). Эти программы запускаются скриптом начальной загрузки /etc/rc.d/init.d/smb. Упомянутый скрипт имеет четыре возможных значения опции вызова: start, stop, restart и status. Символьные ссылки на скрипт загрузки samba имеются в соответсвующих каталогах уровней исполнения.

Основной конфигурационный файл сервера - /etc/smb.conf. Файл достаточно большой и снабжен избыточными комментариями, поэтому заострим внимание только на ключевых моментах настроек:

```
[root@unix /etc]# less smb.conf
# This is the main Samba configuration file. You should read the
# workgroup = NT-Domain-Name or Workgroup-Name
  workgroup = Office
```

Здесь задано имя группы сети Microsoft

```
# server string is the equivalent of the NT Description field
  server string = Samba
```

Название сервера.

```
# the smb.conf man page
  hosts allow = 195.133.132. 127.
```

Разрешение обслуживания клиентов, принадлежащих к сети 195.133.132.

```
# this tells Samba to use a separate log file for each machine
# that connects
  log file = /var/log/samba/log.%m
```

Указание вести раздельные файлы регистрации по каждой клиентской машины.

```
# Put a capping on the size of the log files (in Kb).
  max log size = 50
```

Максимальный размер файлов регистрации.

```
# Security mode. Most people will want user level security. See
# security_level.txt for details.
  security = user
```

Пользовательский уровень безопасности.

```
# Unix users can map to different SMB User names
username map = /etc/smbusers
```

Файл соответствия пользовательских имен UNIX и клиентов сетей Microsoft.

```
#======ShareDefinitions ==============
```

Начало задания разделяемых ресурсов сервера

```
[homes]
  comment = Home Directories
  browseable = no
  writable = yes
```

Разрешение использования индивидуальных пользовательских каталогов UNIX.

```
# Set public = yes to allow user 'guest account' to print
  guest ok = no
  writable = no
  printable = yes
```

Разрешение гостевого доступа

```
[Linux CD]
  comment = cdrom
  path = /mnt/cdrom
  public = yes
  writable = no
  printable = no
  guest ok = yes
```

Разрешение общего и гостевого доступа к накопитетелю CD ROM (только чтение).

```
[public]
  path = /var/mars_nwe/sys/public
  public = yes
  only guest = yes
  writable = yes
  printable = no
```

Разрешение доступа к общему разделу эмулятора NetWare

Некоторое время существовала известная проблема, связанная с неправильным воспроизведением русских символов в именах файлов, объясняемая различем кодировок хранения имен файлов на Linux и Windows 95. Эта сложность была решена добавлением следующих строчек в файл smb.conf:

```
coding system = koi8-r
clientcodepage=866
```

Упомянутый в данном конфигурационном файле файл соответствий пользователей UNIX именам SMB-клиентов достаточно прост по своему строению:

```
[root@unix /etc]# less smbusers
# Unix_name = SMB_name1 SMB_name2 ...
root = administrator admin
nobody = guest pcguest smbguest
1 = guest
dima =dima
yura = yura
little = little
```

3.3 Настройка клиентских станций

3.2.1 Станции Windows

Настройка клиентских станций Windows95 - процедура достаточно простая и понятная среднему пользователю. Большинство действий выполняется в меню ”Сеть”.

Прежде всего, необходимо обнаружить сетевую карту, в большинстве случаев это действие выполняет сама Windows95. Если карта не относится к типу устройств plu’n’play и операционная система оказывается неспособной ее обнаружить, приходится устанавливать драйверы сетевой карты и вручную выставлять ее параметры (I/O adress, IRQ).

Далее производится добавление нужных сетевых протоколов и удаление ненужных. К необходимым протоколам относятся TCP/IP и IPX/SPX-совместимый протокол.

При настройке свойств TCP/IP выставляются следующие параметры:

- IP-адрес машины (195.133.132.xx);
- маска подсети (255.255.255.240);
- шлюз (195.133.132.17);
- имя машины;
- домен (px.podolsk.ru);
- сервер DNS (195.133.132.17);
- привязка (“Служба доступа к файлам и принтерам сетей Microsoft”, “Клиент для сетей Microsoft”);

Далее устанавливаются следующие клиент-серверные компоненты:

- “Служба доступа к файлам и принтерам сетей Microsoft”;
- “Клиент для сетей Microsoft”;
- “Клиент для сетей NetWare”;
- “HP JetAdmin” (Для доступа к сетевому принтеру офиса);

Проверка правильности настроек протокола TCP/IP проверяется при помощи команды ping [имя любого хоста сети].

Закладка “Сеть”/”Компьютер” позволяет установить имя данной машины в сетях Microsoft (Smb) и принадлежность ее к рабочей группе сети Microsoft.

Далее, пользователь машины (или системный администратор) должен определить те ресурсы, которые он собирается выделить для сетевого доступа другим станциям, и установить на них соответствующие права доступа.

Щелчком правой кнопки мыши на иконке файлового ресурса (директории или дискового раздела) необходимо вызвать окно “Свойства” и открыть его на закладке “Доступ”. В зависимости от конкретного ресурса устанавливаются свойства разделенного ресурса и пароль доступа.

3.2.2 Станции Macintosh

Большинство необходимых средств станций Macintosh поставляются в составе операционной системы MacOS, хотя они доступны и в виде отдельного набора OpenTransport/PPP. Средство MacTCP DNR является своеобразным ускорителем сетевого обмена.

Для настройки, прежде всего, необходимо убедиться в наличии соответствующих расширений системы (Extentions) и панелей управления (Control Panels) в соответствующих папках фолдера System.

Далее выполняется привязка протоколов AppleTalk и TCP/IP к соответствующему сетевому интерфейсу (помимо Ethernet, например, AppleTalk может быть привязан к последовательным портам станции Macintosh, а TCP/IP к адаптеру удаленного доступа по протоколу PPP).

Следует отметить, что ситема панелей управления сетевыми настройками взаимосвязана, интуитивна легка и не нуждается в подробном описании.

На Рис.17 представлен общий вид панели настройки протокола TCP/IP.

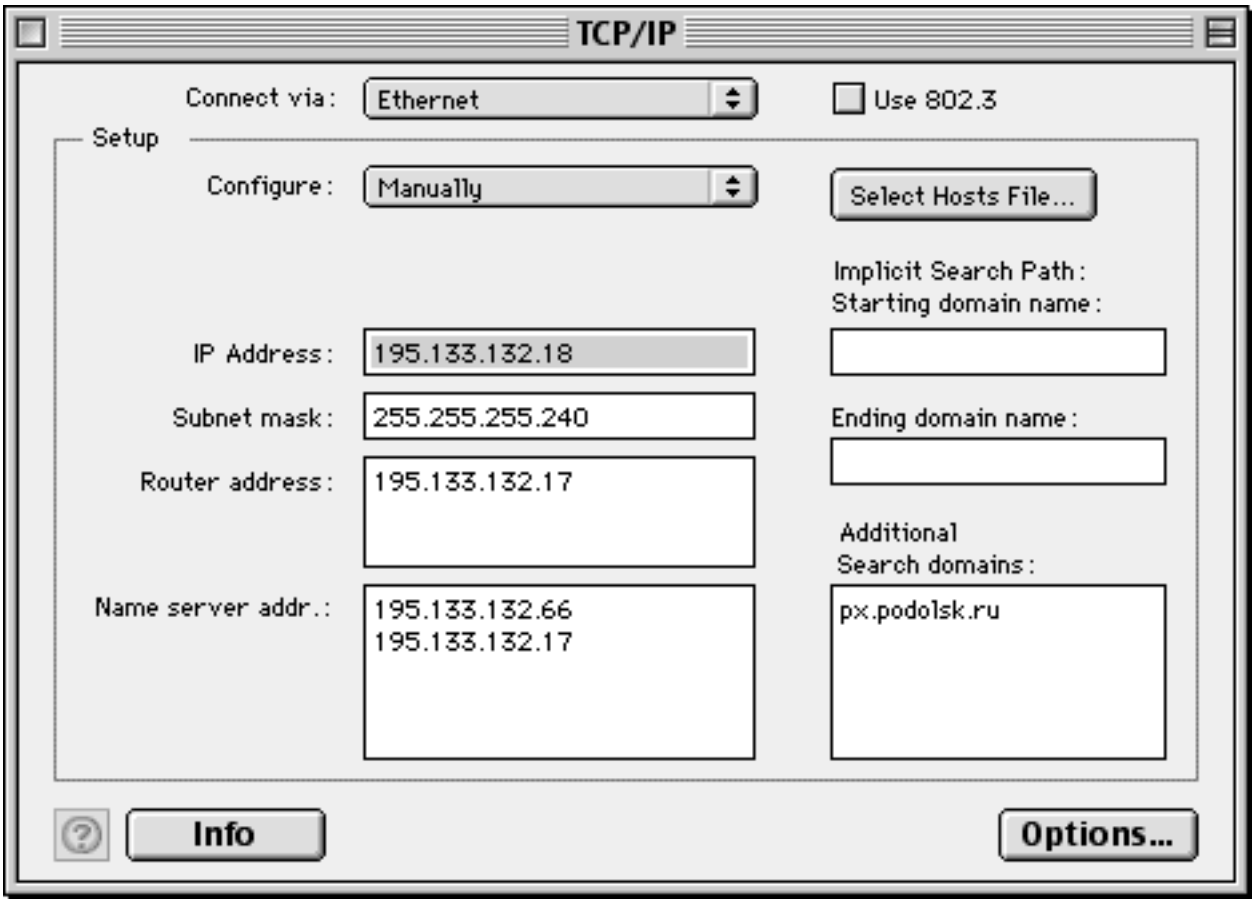


Рис. 17

Настройки пакета Dave и сетевое взаимодействие станций Macintosh достаточно подробно описаны в разделах 2.3.2.1 и 2.3.2.3

3.2.3 Internet/Intranet на рабочих местах

Для полноценного участия рабочих станций в сетях Internet/Intranet необходимы следующие компоненты:

- почтовый клиент (прием, пересылка и хранение электронной почты);
- Web-броузер (навигация по WoldWideWeb);
- клиент новостей (просмотр конференций новостей и написание статей);

Подавляющее большинство рабочих станций использует для этих целей Microsoft Internet Explorer (в качестве броузера), Microsoft Oulook Express (в качестве почтового клиента и клиента новостей) или Netscape Navigator, совмещающий в себе все три компоненты. В последнее время фирма Microsoft поставляет пакеты Outlook Express и Internet Explorer вместе со своими операционными системами, а фирма Apple - все три упомянутые компоненты в составе MacOS. Особенной качественной и функциональной разницы между продуктами Netscape и Microsoft не существует; предпочтение того или другого - дело вкуса пользователя.

Последовательно рассмотрим настройки каждого из трех компонент.
Для почтового клиента необходимо указать IP-адрес (символьный или численный) сервера входящей почты. Обмен с этим сервером происходит по протоколу pop3, хотя в последнее время широкое распространение получает протокол IMAP. Взаимодействие с сервером исходящей (отправляемой) почты происходит по протоколу SMTP (Simple Mail Transport Protocol). Вообще говоря, сервера входящей и исходящей почты разные, но на практике в большинстве случаев в их качестве выступает один почтовый сервер. На примере рассматриваемой сети фирмы - это сервер фирмы под управлением ОС Linux. В настройках необходимо указать имя (название бюджета) пользователя сервера входящей почты. Для удобства можно сохранить и пароль доступа, но это не рекомендуется по соображениям информационной безопасности. Помимо обязательных параметров можно указать обратный адрес отправителя, подпись и реквизиты владельца ящика.

Современные почтовые клиенты позволяют включать фильтрацию почты, т.е. автоматическую сортировку входящих сообщений по какому-то критерию. Это бывает удобно при больших объемах входящей корреспонденции. Программа Outlook Express позволяет даже заводить несколько почтовых бюджетов на различных серверах. Любой почтовый клиент позволяет сохранять полученные письма, удалять их, вести базу электронных адресов, пересылать письма с комментариями и сохранять копии исходящих.

Среди настроек Web-броузеров следует упомянуть адрес и порт кэширующего proxy-сервера, систему интернациональных перекодировок, шрифтовые настройки, базовую страницу, базовые поисковые сервера и прочее. Каждый из броузеров позволяет сохранять адреса интересных пользователю страниц; помимо этого они обладают собственным дисковым кэшем, что отчасти ускоряет загрузку часто посещаемых страниц.

К настройкам клиента новостей следует отнести указание адреса и порта TCP/IP News-сервера и, возможно, имя и пароль пользователя (для доступа к

закрытым серверам новостей). Корпоративный сервер новостей размещается на Linux-машине (unix.px.podolsk.ru). Аутентификация пользователей не требуется, т.к. сервер закрыт для обращений извне (см. Раздел “3.2.7 Конфигурирование сервера новостей”).

В заключение следует заметить, что почтовые клиенты и клиенты новостей плотно взаимодействуют. И в электронных сообщениях, и в статьях конференций новостей допускается использовать инкапсулированные HTML-документы.

3.3 Система резервного копирования

Система резервного копирования является своеобразной “жемчужиной” проекта. Ранее упоминалось, что автору потребовались собственные решения проблемы резервного копирования, наиболее подходящие для нужд предприятия.

Общая схема системы резервного копирования представлена в Приложении 9. Основа системы - shell-скрипт main, который находится в каталоге /usr/local/bin/backup:

```
#!/bin/sh
echo "Staring main Backup"
home=/usr/local/bin/backup

#Backin' Up Windows Machines
------------------------------------------------------------------------------ 1
echo " Windows Machines "
winbackup BUH buh_doc buhbuh
winbackup NALIM nal_reestr narreestr
winbackup NALIM nal_doc1 mardoc
winbackup NALIM nal_doc2 marshet
winbackup WIN csc_doc cdocsc
winbackup WIN csc_work cworksc
------------------------------------------------------------------------------ 2
#echo Backing Up Local UNIX Server, main files
date > "$home"/message.local
       tar cvzf "$home"/files/temp/local.tgz -T "$home"/locallist 2>> \
  "$home"/message.local 1>&2
[ $? -eq 0 ] || {
cat "$home"/message.local | mail -s "localhost backed up with errors!" backup;
exit $?
}
cat "$home"/message.local | mail -s "UNIX Server backuped OK" backup
rm -f "$home"/message.local
------------------------------------------------------------------------------ 3
"$home"/rotate
exit 0;
```

Скрипт можно разделить на три функциональные части. В первой части производятся вызовы специально написанной shell-процедуры winbackup, которая производит архивацию критически важных каталогов Windows-машин (вернее, разделенных smb-ресурсов). О процедуре winbackup будет рассказано позднее.

Вторая часть отвечает за копирование и архивирование ресурсов собственно Linux-сервера. Достигается это следующим образом: сначала во временный log-файл message.local записывается текущая дата и время. Далее при помощи утилиты tar локальные ресурсы машины, заданные в файле locallist, архивируются в файл files/temp/local.tgz. При этом файл сообщений message.local дополняется записями с абсолютными путями архивируемых файлов и каталогов:

```
[root@unix backup]# less message.local
Sun May 30 04:22:22 MSD 1999
tar: Removing leading `/' from absolute path names in the archive
etc/passwd
etc/named.conf
etc/smb.conf
etc/shadow
etc/group
etc/lilo.conf
.....
```

В этой же части выполняется проверка кода завершения архивации. Если по каким-либо причинам оператор tar закончил выполнение с ошибкой, то оператору резервного копирования backup посылается письмо с содержанием временного log-файла и полем Subj с сообщением об ошибках резервного копирования.В противном случае в теле письма, посылаемого оператору резервного копирования, содержится полный перечень заархивированных ресурсов; письмо отправляется с пометкой об успешном выполнении операции.

Третья часть скрипта - запуск процедуры ротации архивов. Изначально файлы, полученные при архивации разделов рабочих станций и локальных ресурсов помещаются в каталог /usr/local/bin/backup/files/temp. В каталоге /usr/local/bin/backup/files существуют три каталога “1”, “2” и “3”. Задача процедуры rotate - последовательно перемещать содержимое каталога temp в каталог 1, содержимое каталога 1 в каталог 2, содержимое каталога 2 в каталог 3, а содержимое каталога 3 уничтожать, осуществляя принцип очереди. Таким образом, при периодическом запуске основного скрипта резервного копирования соблюдается трехуровневая система хранения резервных копий.

Специальный пользователь backup, получив в письме уведомление об успешном (или не успешном) резервном копировании, обязан переместить архивы верхнего (а при не успешном - более низкого) уровня на станцию Macintosh для их последующей записи на пишущий CD ROM (CD Writer).

Устранение ошибок, связанных с резервным копированием - безусловная задача оператора резервного копирования.

Теперь вернемся к winbackup, процедуре резервного копирования ресурсов рабочих станций Windows. Полный текст данной shell-процедуры представлен в Приложении 11. Процедура должна вызываться с тремя параметрами: сетевое имя Windows-машины, имя раздела и пользовательский пароль. Первая часть скрипта проверяет корректность ввода параметров (вернее, их количество). На основе полученных параметров формируются некоторые служебные переменные, такие как имя флага, имя архива имя сообщения (log-файла). После этого производятся контрольные записи имени машины, названия раздела, даты и времени запуска процедуры в log-файл.

Возможность сетевого доступа к ресурсу проверяется при помощи программы smbclient, входящей с состав пакета Samba:

```
.....
# test availability of host/share:
smbclient -L "$win" | grep "$vol" > "$fl"
```

По результатам проверки, как можно заметить, формируется флаг доступности того или иного разделенного сетевого Windows-ресурса. В случае доступности ресурса производится формирование tar-архива при помощи соответствующей процедуры smbtar, написанной Martin Kraemer и Ricky Poulten и входящей в Samba Suite. Следует заметить, что smbtar - лишь сервисная процедура, позволяющая задавать всевозможные параметры эмулятора клиента SMB и передавать их программе smb-client.

Все операции создания и добавления файлов и каталогов удаленной машины регистрируются в log-файле. После закрытия tar-архива производится его сжатие при помощи GNU-утилиты gzip.

При невозможности достижения соответствующего сетевого ресурса в регистрационный файл, уже содержащий дату, время и параметры интересующего ресурса, производится запись о неудачной попытке резервного копирования упомянутого раздела.

Процедура завершается посылкой оператору резервного копирования содержания регистрационного log-файла и удалением флагового и log-файлов.

Описание системы резервного копирования следует завершить упоминанием еженедельного вызова основной процедуры main при помощи системного процесса cron.

3.5 Меры по обеспечению информационной безопасности

Ввиду ограниченности объемов пояснительной записки невозможно рассмотреть реализацию всех методов и средств информационной безопасности сети.

Рассмотрим систему подключаемых модулей аутентификации (PAM) системы Linux. PAM представляет из себя набор библиотек, позволяющих системному администратору Linux регулировать аутентификацию пользователей процессами.

Система основана на аутентификационных библиотеках, расположенных в каталоге /lib/security:

```
[root@unix pam.d]# ls -1 /lib/security/
pam_access.so
pam_cracklib.so
pam_deny.so
pam_env.so
pam_filter.so
pam_ftp.so
pam_group.so
pam_lastlog.so
pam_limits.so
pam_listfile.so
pam_mail.so
pam_nologin.so
pam_permit.so
pam_pwdb.so
pam_radius.so
pam_rhosts_auth.so
pam_rootok.so
pam_securetty.so
pam_shells.so
pam_stress.so
pam_tally.so
pam_time.so
pam_unix_acct.so
pam_unix_auth.so
pam_unix_passwd.so
pam_unix_session.so
pam_warn.so
pam_wheel.so
```

В каталоге /etc/pam.d распологаются файлы, соответсвующие различным процессам, которые сопровождаются процессом аутентификации:

chfn
chsh
ftp
imap

```
linuxconf
linuxconf-pair
login
netatalk
other
passwd
ppp
rexec
rlogin
rsh
samba
su
xdm
```

Рассмотрим процесс login, сопровождающий доступ к командому shell пользователя. Вот содержание этого файла:

```
[root@unix pam.d]# less login
#%PAM-1.0
auth       required     /lib/security/pam_securetty.so
auth       required     /lib/security/pam_pwdb.so shadow nullok
auth       required     /lib/security/pam_nologin.so
account    required     /lib/security/pam_pwdb.so
#by me using access.conf
account    required     /lib/security/pam_access.so
#
password   required     /lib/security/pam_cracklib.so
password   required     /lib/security/pam_pwdb.so shadow nullok use_authtok
session    required     /lib/security/pam_pwdb.so
session    required     /lib/security/pam_limits.so
```

Напротив определенных элементов процесса аутентификации стоят вызываемые билиотеки системы PAM. Строка, выделенная комментариями, добавлена для увеличения безопасности доступа к shell Linux. Библиоткека pam_access.so использует конфигурационный файл /etc/security/access.conf:

```
…
+:users dba pppusers:.px.podolsk.ru
+:dima:ALL
+:root:LOCAL
-:ALL:ALL
…
```

Расшифруем правила, заключенные в этих строчках: первая строка разрешает (+) попытку доступа к пользовательских бюджетам членам групп users, dba и pppusers с внутренних машин домена офисной сети px.podolsk.ru; вторая строка разрешает (+) доступ пользователю dima отовсюду, третья строка разрешает доступ суперпользователю root с локальной консоли Linux, последняя строка отбирает (-) права доступа ВСЕХ остальных пользователей, которые не пречислены предшествующих правилах.

Рассмотрим последнюю строчку файла login: она использует библиотеку /lib/security/pam_limits.so. Эта библиотека отвечает за выделение определенного набора системных ресурсов пользователю при его аутентификации. Следет с прискорбием констатировать тот факт, что что при установке системы Linux по умолчанию никакого предела отпускаемых ему ресурсам не существовало. Таким образом, любой пользователь, имеющий доступ к командной оболочке мог запустить процесс, порождающий бесконечное количество дочерних процессов, что неминуем привело бы к заполнению всех имеющихся системных ресурсов и полной “смерти” системы.

Рассмотрим конфигурационный файл /etc/security/limits.conf:

```
#       - core - limits the core file size (KB)
#       - data - max data size (KB)
#       - fsize - maximum filesize (KB)
#       - memlock - max locked-in-memory address space (KB)
#       - nofile - max number of open files
#       - rss - max resident set size (KB)
#       - stack - max stack size (KB)
#       - cpu - max CPU time (MIN)
#       - nproc - max number of processes
#       - as - address space limit
#       - maxlogins - max number of logins for this user
#<domain>     <type>  <item>         <value>
ftp             hard    nproc           8
#@student       -       maxlogins       4
@users          hard    nproc           30
@users          hard    data            500
@users          hard    fsize           100000
@users          hard    stack           200
@users          hard    rss             10000
@users          hard    core            30
```

Используя данный файл в сочетании с библиотекой pam_limits.so, мы можем устанавливать мягкие и жесткие ограничения на такие ресурсы пользователей или целых групп пользователей как максимальный размер файла, максимальное количество открытых файлов, максимальное количество памяти, резидентно занимаемого пользовательскими процессами, максимальное время использования центрального процессора, максимальный размер адресного пространства пользовательского процесса и проч.

Мы рассмотрели использование лишь двух библиотек системы PAM.

Система PAM с ее многообразием библиотек, конфигурационных файлов и опций является чрезвычайно гибким и мощным средством повышения безопасности UNIX-системы.

4. ЗАКЛЮЧЕНИЕ

Результатом данной дипломной работы явилось построение сети отдельно взятой фирмы (ПФГ “ПРОМЭКСПОРТ”, г. Подольск) как единой системы. Уникальность данной работы состоит в том, что автору удалось найти решения, позволяющие связать различные платформы (Macintosh, UNIX, WindowsPC) в рамках единой логически законченной системы большей частью на основе свободно распространяемых продуктов.

Экономия средств, достигнутых в результате внедрения решений данного проекта исчисляется тысячами, а возможно, и десятками тысяч долларов (достаточно сказать, что, например, только коммерческий Web-сервер Netscape Enterprise Server 3.6 стоит 1295 долларов).

В то же время, реализация проекта вывела информационные технологии офиса ПФГ “ПРОМЭКСПОРТ” на качественно новый уровень, позволила оптимизировать общую управляемость фирмы, улучшить документооборот, автоматизировать рутинный бумажный труд и увеличить производительность труда офисных сотрудников.

Получен богатый опыт по внедрению новых решений (в частности, системы автоматического резервного копирования, некоммерческого файлового сервера для сетей Macintosh, сетевое взаимодействие компьютеров WintelPC и Macintosh).

Следует еще раз подчеркнуть, что платформой этих решений послужил выбор ОС Linux в качестве серверной сетевой платформы. Именно благодаря чрезвычайной гибкости, мощности, внутренней логичности построения и открытости стали возможны упомянутые комплексные решения.

Уже не секрет, что деятельность Free Sofware Foundation, в частности, такие операционные системы как Linux и FreeBSD вызвали в компьютерном мире широкий резонанс и заставили многих профессионалов обратить свое пристальное внимание в область свободных программных продуктов.

Своеобразная GNU-революция, свидетелями которой мы являемся, не осталась без внимания автора данного проекта.

Дипломный проект доказывает, что серверные платформы на базе UNIX-систем не только не отмирают, но в лице открытых бесплатных систем (и прежде всего, Linux) вступают в новую фазу развития, получая все более широкую популярность и распространенность.

Автор не считает необходимым скрывать свое негативное отношение к коммерческим сетевым не-UNIX ОС и, прежде всего, к WindowsNT. В этом смысле проект является, пусть и небольшим, актом борьбы против “империи зла” Microsoft и ее агрессивной и лживой политики, за стремление к истинному прогрессу в области компьютерных технологий.

Выражаю свою признательность своему другу, руководителю проекта ***Алексею Моисееву*** *за неоценимую помощь в освоении UNIX-систем и сетевых технологий, его оптимизм и неукротимую энергию,* ***Андрею Эдемскому*** *за конструктивные идеи, советы, его настойчивость и методичность,* ***Стиву Джобсу****, основателю и руководителю Apple Computer, за революционные принципы в компьютерных технологиях и возрождение Apple,* ***Линусу Торвальдсу****,*

создателю ОС Linux, и всем тем, кто продолжает разрабатывать Linux как открытую, профессиональную и эффективную операционную систему, доступную для всех.

Работа оформлена на компьютерах Macintosh Apple PowerBook 540c, и UMAX SuperMac S900 с использованием программ BBEdit 5.0, TeachText. Графика подготовлена в программах Adobe Photoshop 5.02, Adobe Illustrator 8.0 Верстка осуществлена в пакете QuarkXpress 4.0 for PowerMacintosh.

5. СПИСОК ЛИТЕРАТУРЫ

1. “Unix руководство системного администратора” (“Unix system administration handbook”), второе издание оригинальное и второе переводное, Эви Немет, Гарт Снайдер, Скотт Сибасс, Трент Р.Хейн (Evi Nemeth, Garth Snyder, Scott Seebass, Trent R.Hein), Prentice Hall PTR (второе издание), BHV Киев, 1997

2. “Архитектура операционной системы Unix”(“THE DESIGN OF THE UNIX OPERATING SYSTEM”), Морис Дж. Бах (Maurice J. Bach), Prentice-Hall, 1986

3. “Essential System Administration”, AEleen Frisch, O’Reilly and Associates, 1991

4. “TCP/IP Network Administration”, Craig Hunt, O’Reilly and Associates, 1990

5. “ИНСТАЛЛЯЦИЯ LINUX И ПЕРВЫЕ ШАГИ” (“Linux Installation and Getting Started”), Matt Welsh, 1996, ТОО “Терем”

6. “UNIX system administration”, Frank G. Fiamingo, 1996 University Technology Services

7. “Linux: Руководство по операционной системе” (“Linux: The Complete Reference”), Ричард Петерсен (Richard Petersen), Osbourne McGraw-Hill 1996, BHV, Киев 1997

8. “AppleTalk networking Reference Guide”, Apple Computer, 1997, Cupertino, CA, USA

9. “ Macintosh TCP/IP Networking Bible”, Apple Computer, 1996, Cupertino, CA, USA

10. Oracle8 Intallation Guide Release 8.0.5 for Intel-LINUX, Reiko Nishi, Oracle Corp.

11. Руководство администратора сети в ОС Linux, Олаф Кирч (Olaf Kirch),Linux Documentation Project, 1994

12. ОС Linux. Руководство системного администратора, Ларс Вирженius (Lars Wirzenius), Linux Documentation Project, 1995

13. ПРОГРАММИРОВАНИЕ НА shell (UNIX) (Учебное пособие), А. Соловьев, www.linux.org.ru

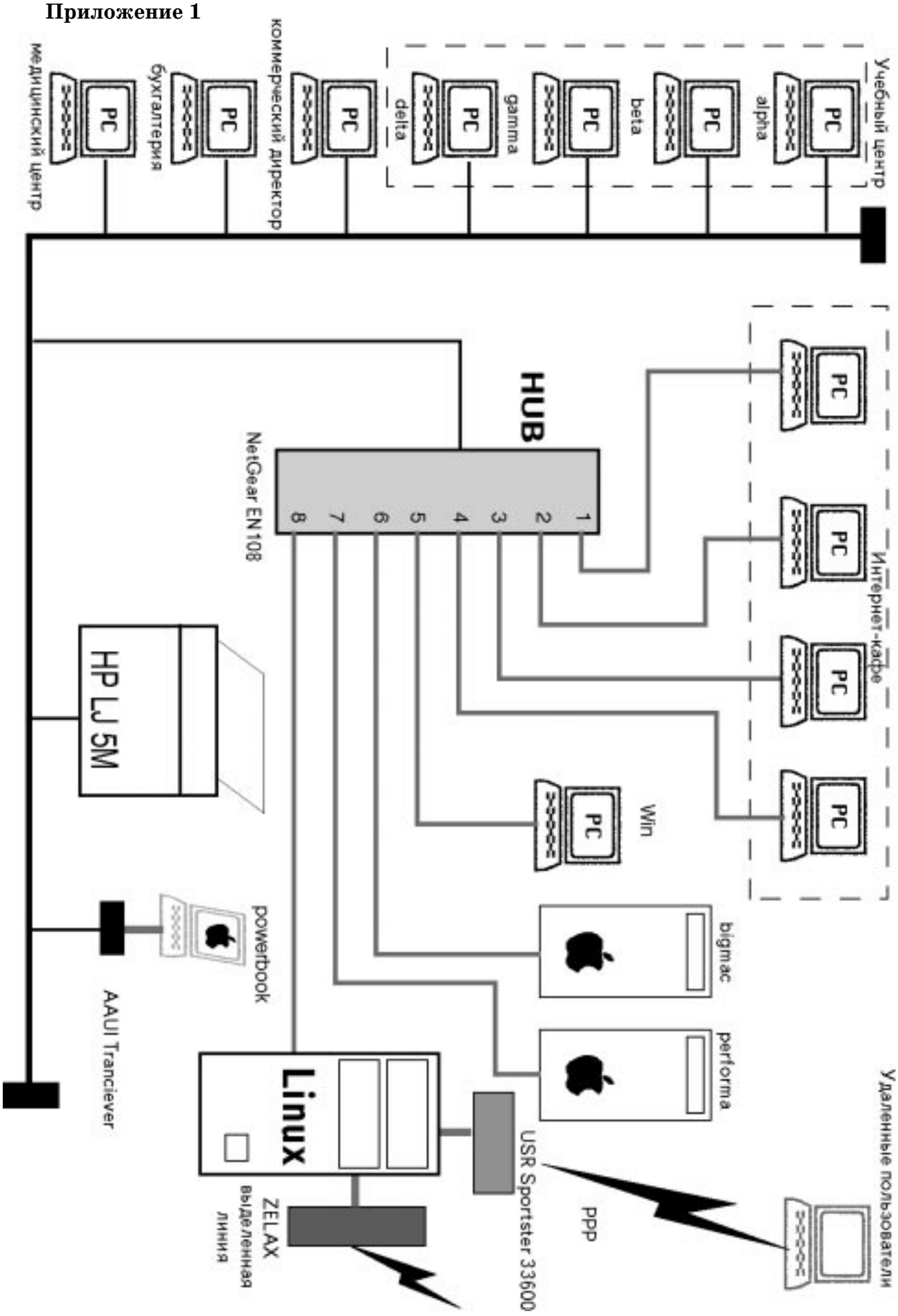
14. The Unix Programming Environment, Brian Kernighan and Bob Pike,Prentice-Hall, 1984

15. UNIX system administration, Frank G. Fiamingo, 1996 University Technology Services, The Ohio State University

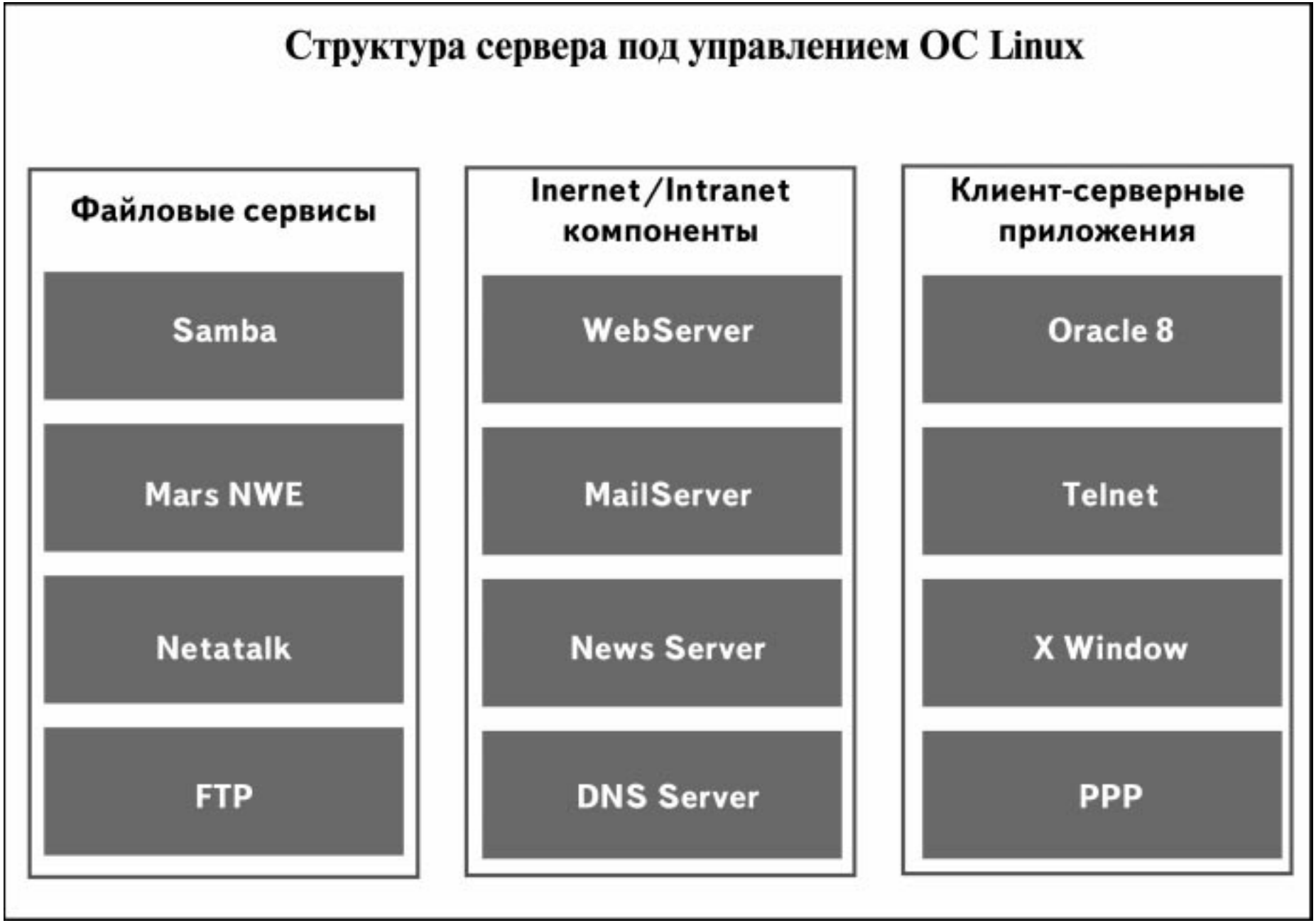
16. The X Window System: A User's Guide, Niall Mansfield, 1995, Addison-Wesley

6. Приложения

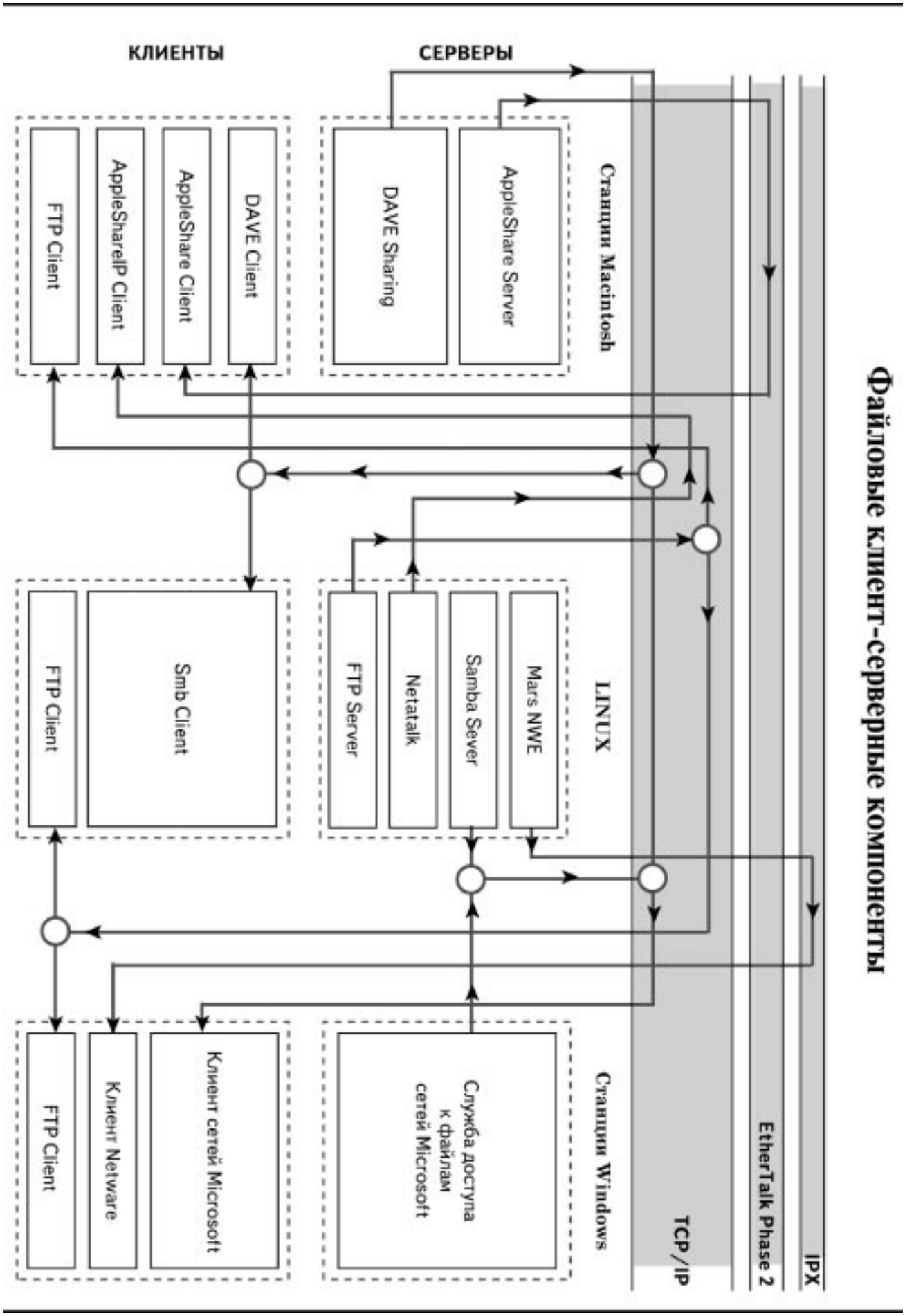
Приложение 1



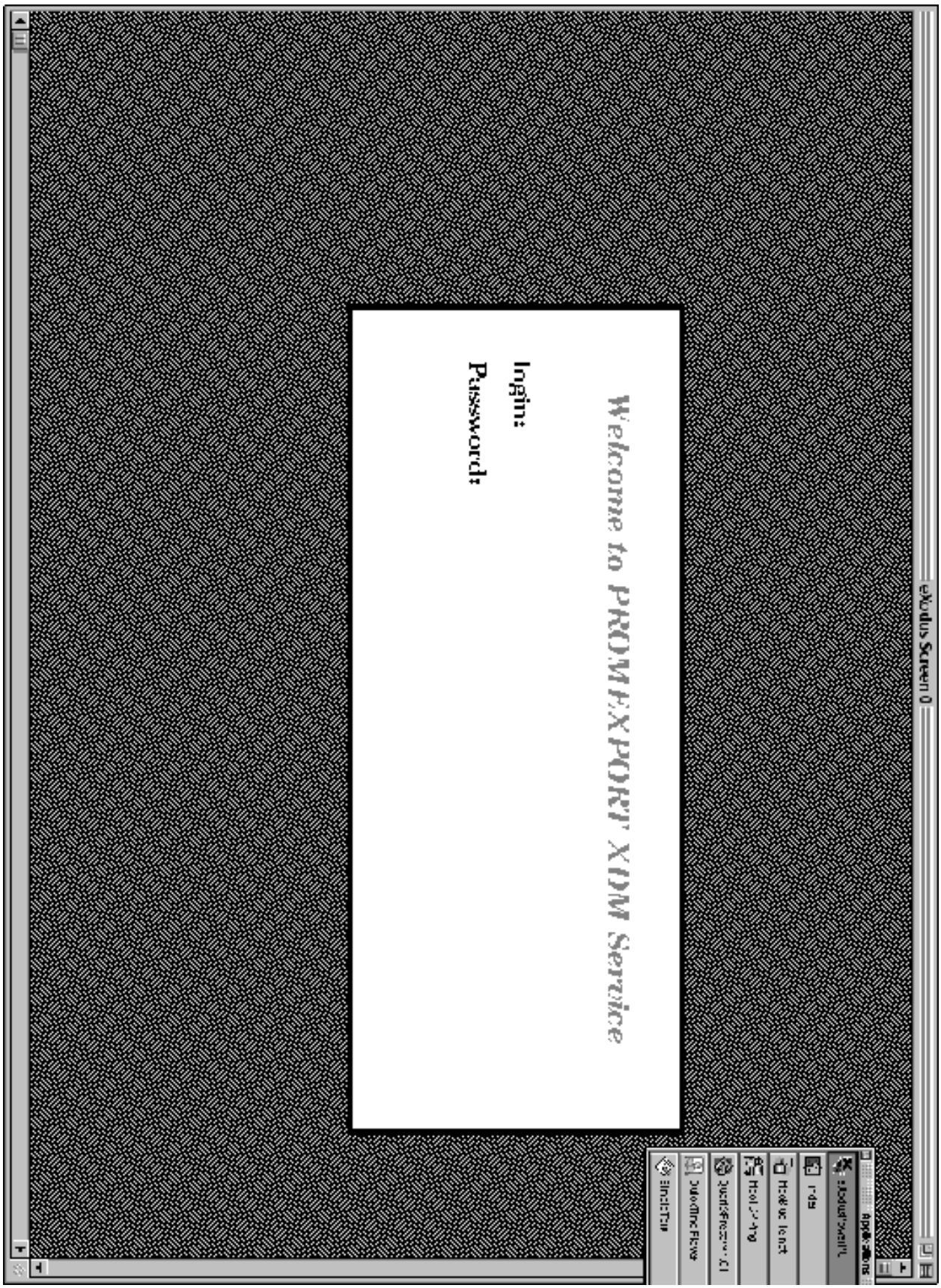
Приложение 2



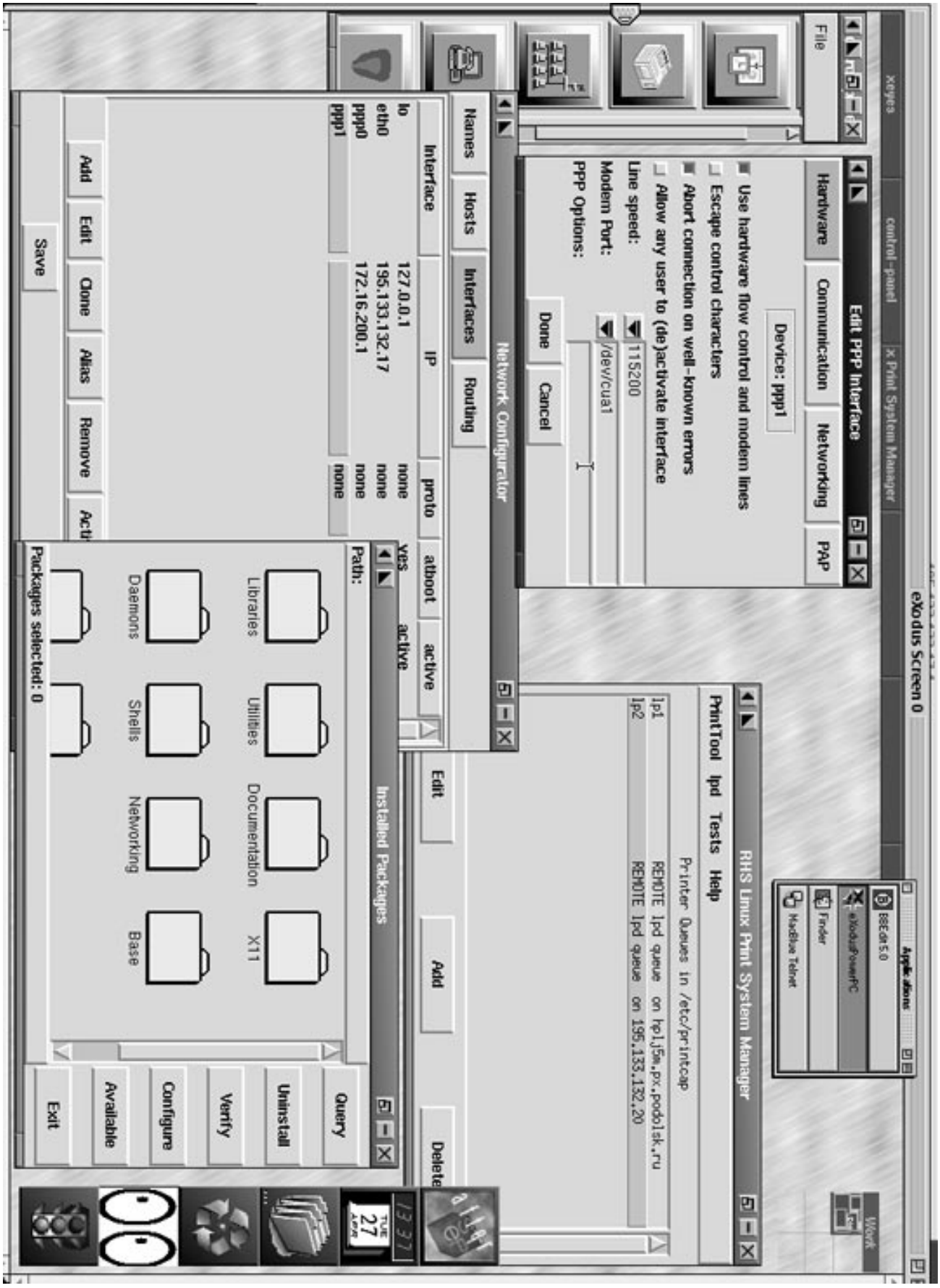
Приложение 3



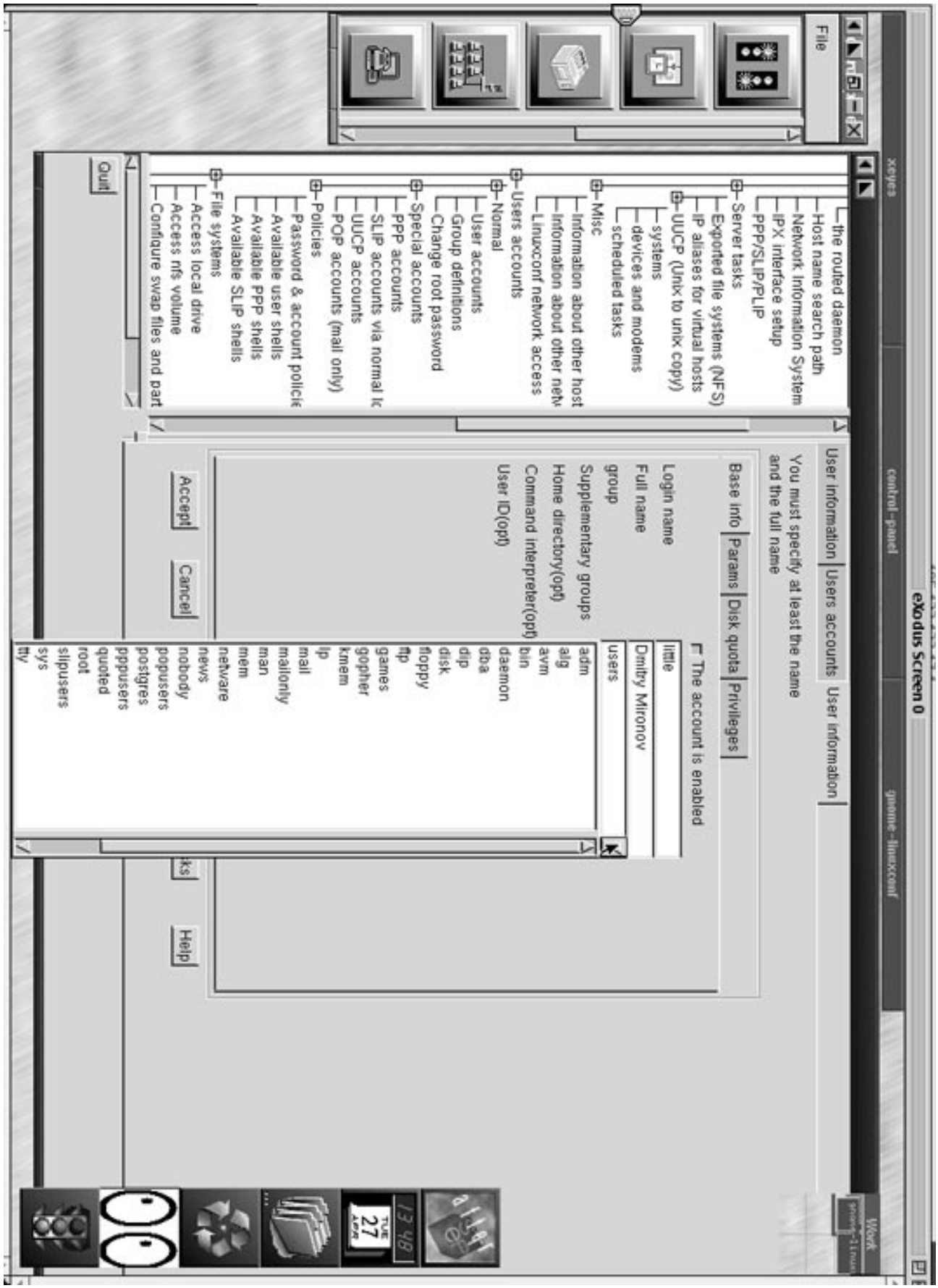
Приложение 4



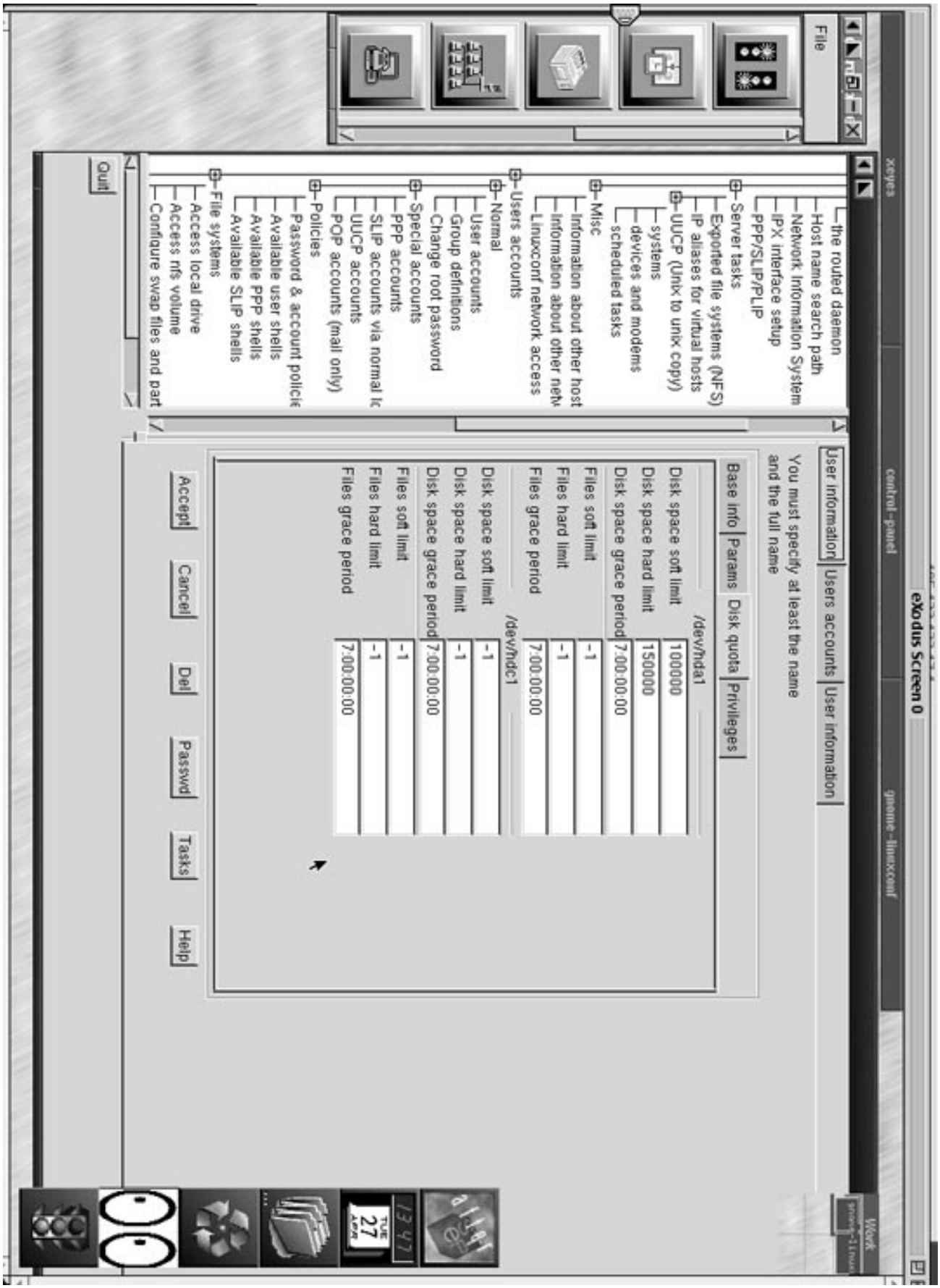
Приложение 5



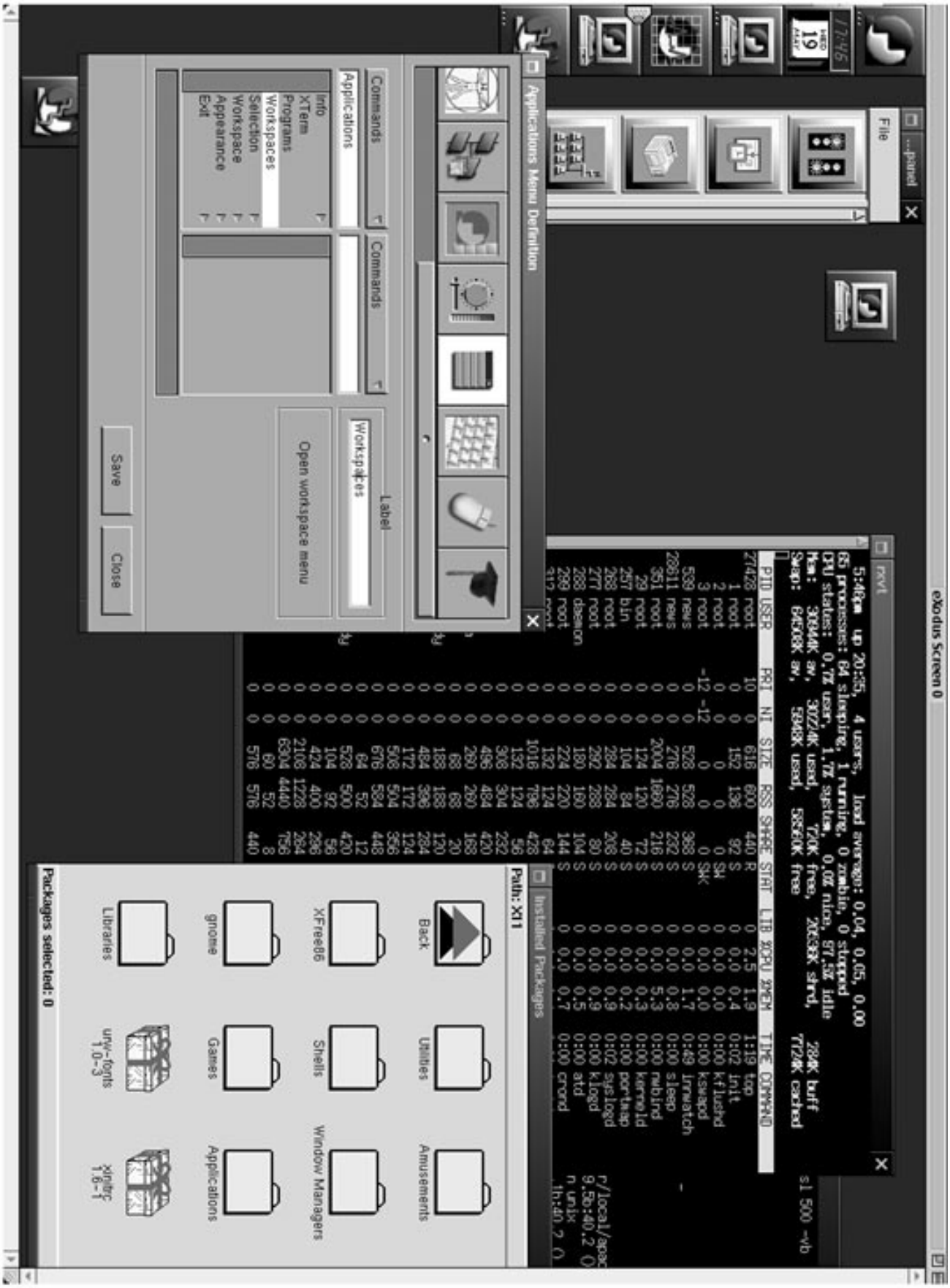
Приложение 6



Приложение 7

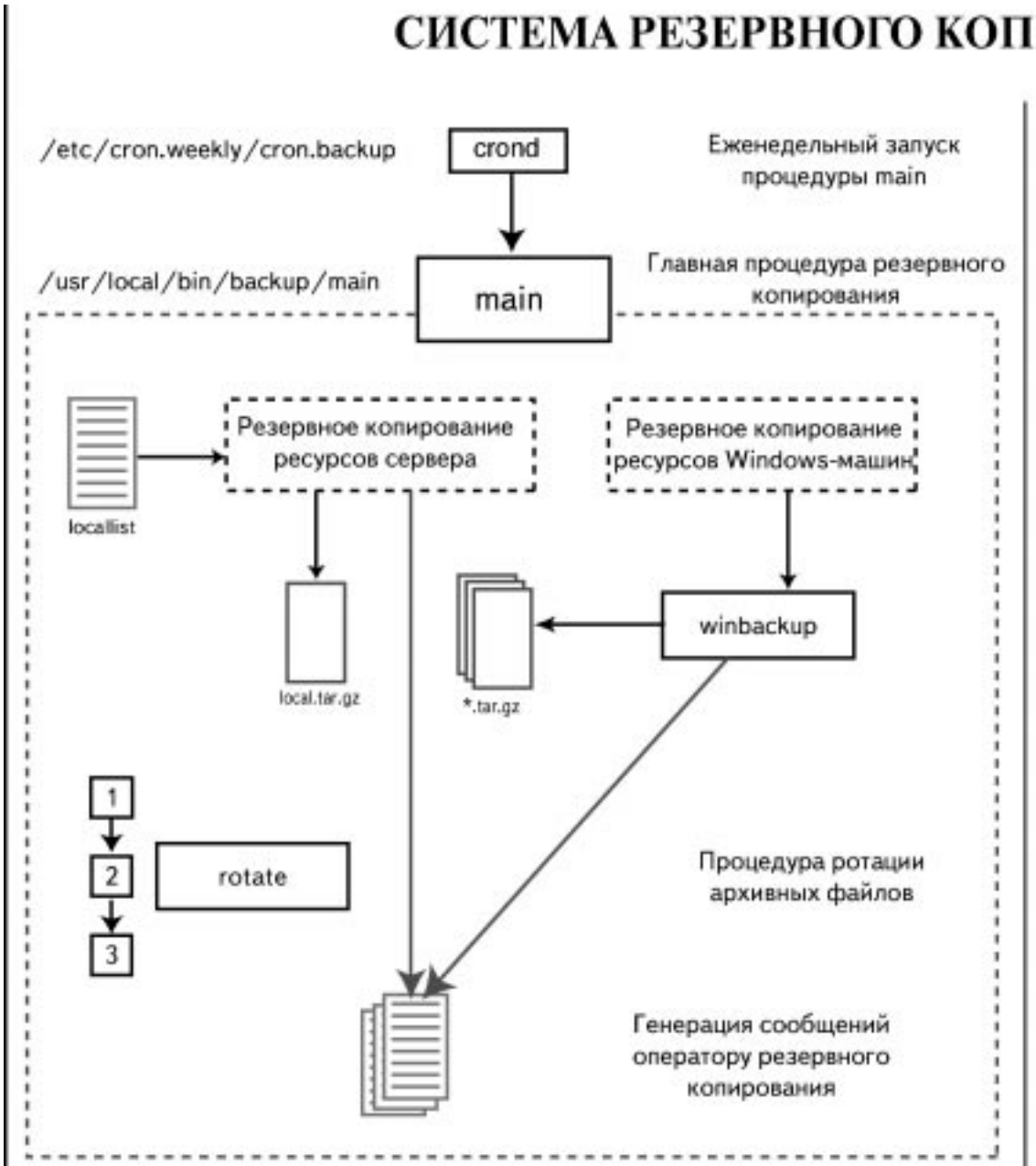


Приложение 8



Приложение 9

СИСТЕМА РЕЗЕРВНОГО КОПИРОВАНИЯ




```
#\/bin/sh
#     by Dmitry Seleznev, syscp of unix.px.podolsk.ru
#          ********************May, 1999*********************
echo " Starting Windows Machines BackUp Process "
if [ ! $# -eq 3 ];
then
echo "Syntax Errors"
echo " Usage: $0 hostname sharename passwd "
exit 1;
fi
home=/usr/local/bin/backup/
win=$1
vol=$2
pass=$3
filename="$win"_"$vol"
logfile=message."$filename"
fl=fl."$filename"
cd "$home"
#end of definitions

#Start of message file

echo "Machine: $win">$logfile
echo "Share: $vol">>$logfile
  date >> $logfile
  if [ -e "$fl" ]; then
                rm -f "$fl"
#               removing flag
  fi

# test availability of host/share:
smbclient -L "$win" | grep "$vol" > "$fl"
if [ -s "$fl" ]; then

#Executing TAR Script
smbtar -i -s "$win" -x "$vol" -p "$pass" -u backup -t files/temp/"$filename".tar
 >> $logfile

#Compressing tar file:
gzip "$home"files/temp/"$filename".tar
echo "Succesfully!" >>$logfile
else echo "Unseccessfully!" >>$logfile
fi
#Sending message to BackUp operator
cat $logfile | mail -s "$win Backup Result, Volume $vol" backup
#Removing the flag
rm -f "$fl"
rm -f "$logfile"
exit 0;
```

Приложение 10

1. Основной файл конфигурации BIND /etc/named.conf

```
[dima@unix dima]$ less /etc/named.conf
// generated by named-bootconf.pl

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;
forwarders
      {
195.133.132.66;
        };

};

//
// a caching only nameserver config
//
/*zone "." {
*       type hint;
*       file "named.ca";
*};
*/

zone "0.0.127.in-addr.arpa" {
        type master;
        file "named.local";
};

zone "px.podolsk.ru"{
        type master;
        file "named.hosts";
};

zone "ppp.px.podolsk.ru"
{
        type master;
        file "named.ppp";
};
```

```
zone "132.133.195.in-addr.arpa" {
        notify no;
        type master;
      file "16.132.133.195.rev";
};

zone "200.16.172.in-addr.arpa" {
        notify no;
        type master;
        file "200.16.172.rev";
};
```

2. Файл прямой зоны домена px.podolsk.ru

```
[dima@unix named]$ less ./named.hosts
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;     BIND configuration for the primary nameserver
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;
;     Domain PROMEXPORT.PODOLSK.RU host table
;
@          IN     SOA    unix.px.podolsk.ru.  hostmaster.unix.px.podolsk.
ru. (
                    102          ; Serial
                    10800        ; Refresh
                    1800         ; Retry
                    3600000      ; Expire
                    59200 ) ; Minimum ttl
           IN     NS           unix.px.podolsk.ru.
;          IN     NS           195.133.132.66
           IN     MX     10    unix.px.podolsk.ru.
           IN     MX     10    px.podolsk.ru.
           IN     A            195.133.132.17
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
unix        IN     A            195.133.132.17
           IN     HINFO        PC / Pentium-166 Linux
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
bigmac       IN     A            195.133.132.18
           IN     HINFO        PowerMac / 604e MacOS
alpha        IN     A            195.133.132.21
beta         IN     A            195.133.132.22
gamma        IN     A            195.133.132.23
delta        IN     A            195.133.132.24
win          IN     A            195.133.132.25
nalimov      IN     A            195.133.132.26
performa     IN     A            195.133.132.27
;hp          IN     A            195.133.132.28
```

```
www          IN    CNAME       unix
ftp          IN    CNAME       unix
hplj5m       IN    A           195.133.132.20
buh          IN    A           195.133.132.19
med          IN    A           195.133.132.29
powerbook    IN    A           195.133.132.30
             IN    HINFO       Apple PowerBook temporary
;
```

3. Файл конфигурации реверсивной зоны домена:

```
[dima@unix named]$ less 16.132.133.195.rev
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; BIND configuration for the primary nameserver
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;
; Revers 64.215.220.194.IN-ADDR.ARPAconfiguration
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
$ORIGIN 132.133.195.IN-ADDR.ARPA.
@        IN    SOA   unix.px.podolsk.ru.  root.unix.px.podolsk.ru. (
                102         ; Serial
                10800       ; Refresh
                1800        ; Retry
                3600000     ; Expire
                86400 )     ; Minimum ttl
         IN    NS    unix.px.podolsk.ru.
;        IN    NS    pgts.podolsk.ru.
;        IN    NS    ns.podolsk.ru.
;        IN    PTR   px-net.podolsk.ru.
;
17       IN    PTR   unix.px.podolsk.ru.
18       IN    PTR   bigmac.px.podolsk.ru.
19       IN    PTR   buh.px.podolsk.ru.
20       IN    PTR   hplj5m.px.podolsk.ru.
21       IN    PTR   alpha.px.podolsk.ru.
22       IN    PTR   beta.px.podolsk.ru.
23       IN    PTR   gamma.px.podolsk.ru.
24       IN    PTR   delta.px.podolsk.ru.
25       IN    PTR   win.px.podolsk.ru.
26       IN    PTR   nalimov.px.podolsk.ru.
27       IN    PTR   performa.px.podolsk.ru
29       IN    PTR   med.px.podolsk.ru
30       IN    PTR   powerbook.px.podolsk.ru
(END)
```

Приложение 11 Shell-процедура winbackup

```
[root@unix backup]# less winbackup
#!/bin/sh
#     by Dmitry Seleznev, sysop of unix.px.podolsk.ru
#        ******************May, 1999******************
echo " Starting Windows Machines BackUp Process "
if [ !$# -eq 3 ];
then
echo "Syntax Errors"
echo " Usage: $0 hostname sharename passwd "
exit 1;
fi

home= /usr/local/bin/backup/
win=$1
vol=$2
pass=$3
filename="$win"_"$vol"
logfile=message."$filename"
fl=fl."$filename"
cd "$home"
echo "Machine: $win">$logfile
echo "Share: $vol">>$logfile
  date >> $logfile
   if [ -e "$fl" ]; then
               rm -f "$fl"
#               removing flag
   fi
# test availability of host/share:
                  smbclient -L "$win" | grep "$vol"> "$fl"
                        if [ -s "$fl" ];
                              then
#Executing TAR Script
            smbtar -i -s "$win" -x "$vol" -p "$pass" -u backup -t files/temp/"$filename".tar
>> $logfile
#Compressing tar file:
            gzip "$home"files/temp/"$filename".tar
            echo "Succesfully!">>$logfile
                              else
            echo "Unseccesfully!">>$logfile
                        fi
#Sending message to BackUp operator
cat $logfile | mail -s "$win Backup Result, Volume $vol" backup
#Removing the flag and  logfile
rm -f "$fl"
rm -f "$logfile"
exit 0;
```

Приложение 12 Shell-процедура main

```
[dima@unix named]$ less /usr/local/bin/backup/main
#!/bin/sh
echo "Staring main Backup"
home=/usr/local/bin/backup

#Backin' Up Windows Machines
echo " Windows Machines "
"$home"/winbackup BUH BUH_DOC buhbuh
"$home"/winbackup NALIMOV NAL_REESTR narreestr
"$home"/winbackup NALIMOV NAL_DOC1 mardoc
"$home"/winbackup NALIMOV NAL_DOC2 marshet
"$home"/winbackup WIN CSC_DOC cdocsc
"$home"/winbackup WIN CSC_WORK cworksc
#"$home"/winbackup WIN TEMP macduck
#"$home"/winbackup MAC macos macos

echo Backing Up Local UNIX Server, main files
date > "$home"/message.local
        tar cvzf "$home"/files/temp/local.tgz -T "$home"/locallist 2>> \
   "$home"/message.local 1>&2
[ $? -eq 0 ] || {
cat "$home"/message.local | mail -s "localhost backed up with errors!" backup;
exit $?
}

cat "$home"/message.local | mail -s "UNIX Server backuped OK" backup
#rm -f "$home"/message.local
#tar -cf "$home"/files/temp/local.tar "$home"/localtmp/*
"$home"/rotate

exit 0;
```