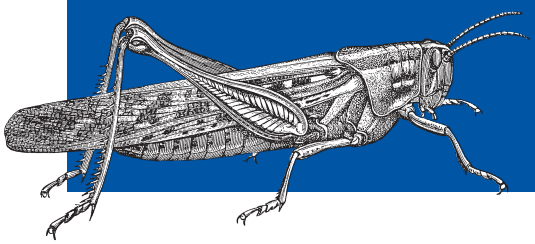
Руководство для системных администраторов

5-е издание
включает BIND 9.3

DNS и BIND



Символ®
O'REILLY®

Крикет Ли и Пол Альбитц

DNS and BIND

Fifth Edition

Cricket Liu and Paul Albitz

O'REILLY®

DNS и BIND

Пятое издание

Крикет Ли и Пол Альбитц

Санкт-Петербург — Москва
2008

Крикет Ли, Пол Альбитц

DNS и BIND, 5-е издание

Перевод М. Зислиса

Главный редактор	*А. Галунов*
Зав. редакцией	*Н. Макарова*
Редактор	*В. Овчинников*
Корректор	*О. Макарова*
Верстка	*О. Макарова*

Ли К., Альбитц П.
DNS и BIND, 5-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2008. – 712 с., ил.
ISBN-10: 5-93286-105-3
ISBN-13: 978-5-93286-105-9

Книга «DNS и BIND» стала библией для системных администраторов. Она уникальна по полноте изложения материала, что в сочетании с прекрасным авторским стилем делает ее незаменимой и актуальной для каждого, кто хочет наладить эффективную работу DNS. В пятом издании обсуждаются BIND 9.3.2 (последняя версия в ветви BIND 9) и BIND 8.4.7. BIND 9.3.2 включает усовершенствования безопасности и поддержки IPv6, а также ряд новых возможностей, таких как ENUM, SPF и использование имен доменов, содержащих буквы национальных алфавитов.

Рассмотрены следующие темы: функциональность и принципы работы DNS; структура пространства доменных имен; установка и настройка серверов имен; применение MX-записей для маршрутизации почты; настройка узлов на работу с DNS; разделение доменов на поддомены; обеспечение безопасности DNS-сервера; расширения системы безопасности DNS (DNSSEC) и подписи транзакций (TSIG); распределение нагрузки между DNS-серверами; динамические обновления, асинхронные уведомления об изменениях зоны, пошаговая передача зон; разрешение проблем (nslookup и dig, чтение отладочной диагностики); программирование при помощи функций библиотеки DNS-клиента.

ISBN–10: 5–93286–105–3
ISBN–13: 978–5–93286–105–9
ISBN 0–596–10057–4 (англ)

Издательство «Символ-Плюс». 199034, Санкт-Петербург, 16 линия, 7,
тел. (812) 324-5353, www.symbol.ru. Лицензия ЛП N 000054 от 25.12.98.
Налоговая льгота – общероссийский классификатор продукции
ОК 005-93, том 2; 953000 – книги и брошюры.

Подписано в печать 28.01.2008. Формат 70x100 1/16 . Печать офсетная.
Объем 44,5 печ. л. Тираж 2000 экз. Заказ N
Отпечатано с готовых диапозитивов в ГУП «Типография «Наука»
199034, Санкт-Петербург, 9 линия, 12.

Оглавление

Предисловие

Возможно, вам не так уж много известно о системе доменных имен (Domain Name System), но, работая в Интернете, вы неизбежно ее используете. Всякий раз, отправляя сообщения электронной почты или исследуя просторы World Wide Web, вы полагаетесь на DNS – систему доменных имен.

Дело в том, что люди предпочитают запоминать *имена* компьютеров, а компьютерам больше нравится обращаться друг к другу по числовым адресам. В Интернете этот адрес имеет разрядность 32, то есть может быть числом от нуля до четырех с хвостиком миллиардов.[1] Компьютеры с легкостью запоминают такие вещи, потому что обладают большими объемами памяти, идеально подходящей для хранения чисел, но для людей эта задача не в пример сложнее. Попробуйте случайным образом выбрать из телефонной книги десять номеров и запомнить их. Непросто? Теперь вернитесь к началу телефонной книги и сопоставьте каждому номеру случайный код района. Примерно настолько же сложно будет запомнить 10 произвольных интернет-адресов.

Отчасти именно по этой причине необходима система доменных имен. DNS занимается двунаправленным отображением имен узлов, подходящих для запоминания людьми, и интернет-адресов, с которыми работают компьютеры. По сути дела, DNS в сети Интернет является не только средством работы с адресами, но и стандартным механизмом для предоставления и получения разнообразной информации об узлах сети. DNS нужен практически для каждой программы, обеспечивающей сетевое взаимодействие, в том числе программам для работы с электронной почтой, терминальным клиентам (например, ssh), средствам передачи файлов, таким как ftp, и, разумеется, веб-броузерам, таким как Microsoft Internet Explorer.

Другой важной особенностью DNS является способность системы распространять информацию об узле по всей сети Интернет. Хранение доступной информации об узле на единственном компьютере полезно лишь для тех, кто пользуется этим компьютером. Система доменных имен обеспечивает получение информации из любой точки сети.

Более того, DNS позволяет распределять управление информацией об узлах между многочисленными серверами и организациями. Нет необ-

---

[1] А в системе IP-адресации версии 6 адреса имеют колоссальную длину – 128 бит, что позволяет охватить десятичные числа от 0 до 39-значных.

ходимости передавать данные на какой-то центральный сервер или регулярно синхронизировать свою базу данных с «основной». Достаточно убедиться, что ваш раздел, называемый *зоной*, соответствует действительности на ваших *DNS-серверах*. А они, в свою очередь, сделают информацию о зоне доступной всем остальным DNS-серверам сети.

Поскольку база данных DNS является распределенной, в системе должна быть предусмотрена возможность поиска нужной информации путем опроса множества возможных источников ее получения. Система доменных имен наделяет DNS-серверы способностью находить нужные источники информации и получать сведения по любой зоне.

Разумеется, система DNS не лишена недостатков. К примеру, в целях избыточности базы данных система позволяет хранить зональную информацию на более чем одном сервере, но при этом возникает опасность десинхронизации копий зональной информации.

Но *самая большая* проблема, связанная с DNS, несмотря на широкое распространение в сети Интернет, – это реальное отсутствие хорошей документации по работе с системой. Большинство администраторов сети Интернет вынуждены обходиться лишь той документацией, которую считают достаточной поставщики используемых программ, а также тем, что им удается выудить из соответствующих списков интернет-рассылок и конференций Usenet.

Такой дефицит документации означает, что понимание предельно важной интернет-службы, одной из монументальных основ сегодняшней сети Интернет, либо передается от администратора к администратору как ревностно хранимая семейная тайна, либо постоянно изучается повторно отдельными программистами и разработчиками. Новые администраторы зон повторяют ошибки, уже бесчисленное число раз сделанные другими.

Цель этой книги – изменить сложившуюся ситуацию. Мы осознаем, что не у каждого читателя есть время и желание становиться специалистом по DNS. У большинства из вас есть достаточно других занятий помимо управления зонами и DNS-серверами: системное администрирование, разработка сетевых инфраструктур или разработка программного обеспечения. Заниматься исключительно DNS может только сотрудник безумно большой организации. Мы постарались предоставить информацию, достаточную для решения основных рабочих задач, будь то управление небольшой зоной или целой международной системой, работа с единственным сервером имен или наблюдение за сотней серверов. Извлеките из книги нужный вам минимум и возвращайтесь к ней по мере необходимости.

DNS – это сложная тема, настолько сложная, что взяться за нее пришлось не одному, а двум авторам; но мы постарались представить систему настолько прозрачно и доступно, насколько это возможно. В первых двух главах содержится теоретический обзор и достаточный для

применения объем практической информации, а в последующих главах использование системы доменных имен рассмотрено более подробно. С самого начала мы предлагаем читателям нечто вроде дорожной карты, чтобы каждый мог выбрать собственный путь изучения книги, соответствующий рабочим задачам или интересам.

Когда речь пойдет о программах, обеспечивающих работу DNS, мы практически целиком сконцентрируемся на инструменте под названием BIND, Berkeley Internet Name Domain, который является наиболее популярной (и наиболее нами изученной) реализацией спецификаций DNS. Мы старались представить в этой книге выжимку из нашего опыта управления и поддержки зон с помощью BIND. (Так получилось, что некоторое время одна из наших зон являлась самой большой зоной сети Интернет; правда, это было очень давно). Где это было возможно, мы включали реальные программы, используемые нами в администрировании; многие из них переписаны на языке Perl с целью достижения большей скорости работы и повышения эффективности.

Надеемся, эта книга поможет вам познакомиться с системой DNS и инструментом BIND, если вы еще новичок, лучше понять их работу, если вы с ними уже знакомы, и приобрести ценное понимание и опыт, даже если вы уже знаете DNS и BIND как свои пять пальцев.

Версии

Четвертое издание этой книги затрагивает новые версии BIND – 9.3.2 и 8.4.7, а также более старые версии BIND 8 и 9. Несмотря на то, что на момент написания этой книги версии 9.3.2 и 8.4.7 являются наиболее свежими, они пока не получили широкого распространения в составе UNIX-систем – отчасти потому, что обе версии были выпущены недавно, а многие поставщики настороженно относятся к использованию новых программ. Мы время от времени упоминаем и другие версии BIND, поскольку многие поставщики продолжают распространять программы, содержащие код, основанный на более старых версиях, в составе своих UNIX-разработок. Если определенная возможность доступна только в версии 8.4.7 или 9.3.2 либо существуют различия в поведении версий, мы постараемся четко определить, что именно работает и для какой версии BIND.

В наших примерах мы очень часто прибегаем к служебной программе DNS – *nslookup*. Мы пользуемся *nslookup* из комплекта поставки BIND версии 9.3.2. Более старые версии *nslookup* обеспечивают большую часть функциональности (но не всю) *nslookup* версии 9.3.2. В большинстве примеров мы использовали команды, доступные почти во всех версиях *nslookup*; случаи, когда это было невозможно, отмечены отдельно.

Что нового в пятом издании?

Текст книги был обновлен, чтобы соответствовать наиболее поздним версиям BIND; добавлен следующий новый материал:

- Описание технологии SPF (Sender Policy Framework) – в главе 5.
- Более подробное рассмотрение динамических обновлений и механизма NOTIFY, включая и подписываемые динамические обновления (signed dynamic updates), а также описание нового для BIND 9 механизма *update-policy* – в главе 10.
- Поэтапная передача зоны – также в главе 10.
- Зоны ретрансляции, поддерживающие передачу по условию (conditional forwarding), – в главе 10.
- Прямое и обратное отображение адресов в контексте технологии IPv6 с использованием записей новых типов AAAA и ip6.arpa – в конце главы 10.
- Новый механизм подтверждения подлинности транзакций – транзакционные подписи (transaction signatures, известные также как TSIG) – описан в главе 11.
- Более подробное рассмотрение вопросов обеспечения безопасности DNS-серверов – в главе 11.
- Более подробное рассмотрение работы с брандмауэрами в сети Интернет – в главе 11.
- Описаны обновленные расширения DNS, связанные с безопасностью (DNS Security Extensions или DNSSECbis), представляющие собой механизм цифровой подписи зональных данных, – все в той же 11 главе.
- Новая глава 16 посвящена развертыванию полноценной архитектуры DNS в масштабах организации.
- В главе 17 описывается ENUM, технология для отображения телефонных номеров в формате стандарта E.164 в URI-адреса.
- Стандарт кодирования символов Unicode в именах доменов (IDN, Internationalized Domain Names) описан в главе 17.
- Обновлен раздел, посвященный совместной работе Active Directory и BIND, – в главе 17.

Структура

Порядок следования глав настоящей книги приблизительно соответствует возможному развитию зоны и росту знаний ее администратора. В главах 1 и 2 обсуждается теория системы доменных имен. В главах с 3 по 6 рассматриваются вопросы, связанные с принятием решений по созданию собственных зон, а также действия администратора в случае необходимости создать зону. Следующая часть книги, главы с 7 по 11,

посвящена сопровождению зон, настройке узлов для использования DNS-серверов, планированию развития зон, созданию доменов различных уровней и безопасности серверов. Наконец, главы с 12 по 16 посвящены разрешению сложностей, возникающих при работе с различными инструментами, общим проблемам и забытому искусству программирования с применением библиотек DNS-клиента. Глава 16 сводит знания в единый архитектурный ансамбль. Перечислим темы по главам:

Глава 1 «Основы»

Описывает исторический фон создания системы, посвящена проблемам, приведшим к созданию DNS, а также собственно обзору теории системы доменных имен.

Глава 2 «Как работает DNS»

Посвящена более подробному рассмотрению теоретических основ DNS, в частности организации пространства имен в системе DNS, доменов, зон и DNS-серверов. Там же рассматриваются такие важные понятия, как разрешение адресов и кэширование.

Глава 3 «С чего начать?»

Рассматриваются получение пакета BIND в случае его отсутствия, применение пакета, когда он уже у вас в руках, определение и выбор доменного имени, а также установление связи с организацией, которая обладает полномочиями делегировать выбранную зону.

Глава 4 «Установка BIND»

Подробное рассмотрение того, как установить два первых DNS-сервера на основе BIND, включая создание базы данных серверов, запуск и диагностику их работы.

Глава 5 «DNS и электронная почта»

Рассказывает о записи DNS типа MX, которая позволяет администраторам задавать альтернативные узлы, которым передается на обработку почта для определенных адресов. В этой главе описаны стратегии маршрутизации почты для различных типов сетей и узлов, включая сети с интернет-брандмауэрами и узлы, не имеющие прямого подключения к сети Интернет. В этой главе также повествуется о технологии Sender Policy Framework, позволяющей использовать DNS для авторизации отправления почты с определенных почтовых адресов.

Глава 6 «Конфигурирование узлов»

Рассказывает о том, как настраивать клиентскую часть (*resolver*) BIND, а также об особенностях реализаций клиента, применяемых на платформах Windows.

Глава 7 «Работа с BIND»

Посвящена регулярным действиям администратора, выполнение которых необходимо для поддержания устойчивой работы зон, находящихся под его началом, в частности проверке состояния DNS-сервера и вопросам, касающимся авторитетных серверов зоны.

Глава 8 «Развитие домена»

Рассказывает о планировании роста и эволюции зон, включая вопросы о том, как вырасти большим, а также о планировании переездов и перебоев в работе.

Глава 9 «Материнство»

О радостях, связанных с обретением потомства. Мы расскажем, когда имеет смысл заводить детей (создавать поддомены), как их называть, как их заводить (!) и как присматривать за ними.

Глава 10 «Дополнительные возможности»

Рассказывает о параметрах настройки сервера имен, которые используются не очень часто, но могут помочь в настройке производительности DNS-сервера и упростить процесс администрирования.

Глава 11 «Безопасность»

Посвящена обеспечению безопасности и тем настройкам DNS-сервера, которые относятся к работе с интернет-брандмауэрами, а также двум новым технологиям DNS, связанным с безопасностью: DNS Security Extensions и подписям транзакций (Transaction Signatures).

Глава 12 «nslookup и dig»

Подробно рассказывает о самых популярных инструментах DNS-отладки и содержит описания способов извлечения неявной информации из удаленных DNS-серверов.

Глава 13 «Чтение отладочного вывода BIND»

Это Розеттский камень[1] отладочной информации BIND. Глава поможет разобраться в таинственной отладочной информации, создаваемой пакетом BIND, а это, в свою очередь, поможет лучше понять, как работает DNS-сервер.

Глава 14 «Разрешение проблем DNS и BIND»

Содержит описания и способы разрешения многих распространенных проблем, связанных с использованием DNS и BIND, а также

[1] Розеттский камень – черная базальтовая плита с трехъязычной надписью, обнаруженная в 1799 г. при сооружении форта Сен-Жюльен на берегу Розеттского рукава Нила. Расшифровка иероглифического текста в 1822 г. стала началом изучения египетской иероглифической письменности. – *Примеч. ред.*

рассказывает о более редких случаях, связанных с ошибками, диагностика которых может вызывать затруднения.

Глава 15 «Программирование с использованием библиотечных функций»

Рассказывает о том, как использовать функции библиотеки клиента BIND для опроса DNS-серверов и получения информации в программе на языке C или Perl. Приводится исходный текст полезной (как мы надеемся) программы, которая проверяет работоспособность DNS-серверов и их авторитетность.

Глава 16 «Архитектура»

Описывает полноценную инфраструктуру DNS, включающую внешние DNS-серверы, ретрансляторы, а также внутренние DNS-серверы.

Глава 17 «Обо всем понемногу»

Посвящена незатронутым темам. Она содержит описание использования масок (wildcards) в DNS, принципов работы с узлами и сетями, не имеющими постоянного подключения к сети Интернет, кодировки сетевых имен, дополнительных типов записей ENUM и IDN, а также работы с Active Directory.

Приложение A «Формат сообщений DNS и RR-записи»

Содержит предельно подробный справочник по форматам, используемым в запросах и ответах DNS, а также полный перечень определенных в настоящее время типов RR-записей (resource records).

Приложение B «Таблица совместимости BIND»

Перечисление наиболее важных особенностей самых распространенных версий BIND.

Приложение C «Сборка и установка BIND на Linux-системах»

Содержит пошаговые инструкции по сборке BIND версии 9.3.2 в Linux.

Приложение D «Домены высшего уровня»

Перечисление существующих в настоящее время доменов высшего уровня сети Интернет.

Приложение E «Настройка DNS-сервера и клиента BIND»

Содержит справочник по синтаксису и семантике каждого из существующих параметров настройки серверов и библиотек клиента.

Для кого эта книга

Прежде всего эта книга предназначена для системных и сетевых администраторов, которым приходится управлять зонами и одним или несколькими DNS-серверами, но она содержит материал, который будет

интересен проектировщикам сетей, почтовым администраторам и многим другим людям. Не все главы одинаково интересны для столь разношерстной аудитории, и, конечно же, читателю нет смысла копаться во всех семнадцати главах, чтобы найти интересующий его материал. Мы надеемся, что следующая карта поможет выстроить правильный путь по главам книги.

Системным администраторам, впервые столкнувшимся с вопросами сопровождения зон

Следует прочесть главы 1 и 2, чтобы получить теоретическую подготовку по DNS, главу 3 – в целях получения информации о первых шагах и выборе подходящего доменного имени, главы 4 и 5 – чтобы узнать, как происходит настройка зоны «с нуля». Глава 6 объясняет, как настроить узлы для работы с новыми DNS-серверами. Затем следует обратиться к главе 7, в которой объясняется, как «подкачать» объем, добавляя серверы и данные в зону. Главы с 12 по 14 содержат описание инструментов и методов, помогающих в устранении проблем.

Опытным администраторам

Может быть полезно прочитать главу 6, чтобы узнать, как настраивать DNS-клиенты на различных узлах, и главу 7, чтобы получить информацию о том, как грамотно сопровождать зоны. В главе 8 содержатся инструкции, связанные с планированием роста и развития зоны, которые должны быть особенно полезны людям, занятым в администрировании больших зон. Глава 9 рассказывает о том, как стать родителем, то есть о создании поддоменов, и является учебником *этикета*, обязательным к прочтению теми, кто планирует совершить этот трудный шаг. В главе 10 рассмотрены многие новые возможности BIND версий 9.3.2 и 8.4.7. Глава 11 посвящена обеспечению безопасности DNS-серверов, и для опытных администраторов может представлять особенный интерес. Главы с 12 по 14 содержат описание инструментов и действий, которые помогут устранить возникшие проблемы; эти главы могут оказаться занимательным чтением даже для очень опытных администраторов. Глава 16 поможет администраторам осмыслить общее положение дел.

Системным администраторам сетей, не имеющих постоянного подключения к сети Интернет

Рекомендуется прочесть главу 5, чтобы изучить процесс настройки маршрутизации почты в таких сетях, и главы 11 и 17, которые содержат описание создания независимой инфраструктуры DNS.

Программистам

В целях освоения теории DNS предлагается прочесть главы 1 и 2, а затем главу 15, в которой содержится подробное рассмотрение программирования при помощи библиотечных функций BIND.

Сетевым администраторам, которые напрямую не вовлечены в процесс сопровождения зон

Рекомендуется прочесть главы 1 и 2 в целях освоения теории DNS, главу 12, чтобы научиться использовать *nslookup* и *dig*, а затем главу 14, чтобы узнать о способах разрешения возникающих сложностей.

Почтовым администраторам

Следует прочесть главы 1 и 2 в целях освоения теории DNS, главу 5, чтобы узнать, как сосуществуют DNS и электронная почта, и главу 12, в которой описаны инструменты *nslookup* и *dig*; эта глава научит извлекать информацию о маршрутизации почты из пространства доменных имен.

Заинтересованные пользователи

Могут прочесть главы 1 и 2 в целях освоения теории DNS, а затем любые главы по желанию!

Мы предполагаем, что читатель знаком с основами администрирования UNIX-систем, сетевым взаимодействием TCP/IP, а также программированием на уровне простых сценариев командного интерпретатора или языка Perl. При этом никаких других специальных знаний не требуется. При появлении новых терминов и понятий они насколько возможно подробно объясняются в тексте книги. По возможности мы использовали аналогии с системами UNIX (и реальным миром), чтобы облегчить читателю восприятие новых для него концепций.

Примеры программ

Исходные тексты программ-примеров, приводимых в книге[1], доступны для загрузки по протоколу FTP по следующим адресам:

- *ftp://ftp.uu.net/published/oreilly/nutshell/dnsbind/dns.tar.Z*
- *ftp://ftp.oreilly.com/published/oreilly/nutshell/dnsbind/*

В обоих случаях извлечь файлы из архива можно командой:

```
% zcat dns.tar.Z | tar xf -
```

На системах System V необходимо использовать следующую *tar*-команду:

```
% zcat dns.tar.Z | tar xof -
```

Если команда *zcat* недоступна в системе, следует использовать отдельные команды *uncompress* и *tar*.

Если не удается получить тексты примеров напрямую по сети Интернет, но существует возможность посылать и получать сообщения элек-

[1] Примеры также доступны по адресу *http://examples.oreilly.com/dns5*.

тронной почты, можно воспользоваться службой *ftpmail*. Чтобы получить справку по использованию службы *ftpmail*, необходимо отправить сообщение на адрес *ftpmail@online.oreilly.com*. Следует оставить пустым поле темы сообщения; тело письма должно содержать единственное слово – «help».

Как с нами связаться

Комментарии и вопросы, связанные с этой книгой, можно направлять непосредственно издателю:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800 998-9938 (в США или Канаде)
707 829-0515 (международный/местный)
707 829-0104 (факс)

Издательством O'Reilly создана веб-страница, посвященная этой книге, на которой доступна информация о найденных ошибках и будут появляться разнообразные дополнительные сведения. Страница доступна по адресу:

http://www.oreilly.com/catalog/dns5

Если у вас есть технический вопрос или комментарий, связанный с этой книгой, задайте его, отправив сообщение по адресу:

bookquestions@oreilly.com

На веб-сайте издательства O'Reilly доступна дополнительная информация о книгах, конференциях, программном обеспечении, источниках информации и сети O'Reilly (O'Reilly Network):

http://www.oreilly.com

Типографские соглашения

Использованы следующие соглашения по шрифту и формату для команд, инструментов и системных вызовов UNIX:

- Выдержки из сценариев или конфигурационных файлов оформлены моноширинным шрифтом:

```
if test -x /usr/sbin/named -a -f /etc/named.con
then
    /usr/sbin/named
fi
```

- Примеры диалоговых сеансов, отображающие ввод в командной строке и соответствующую реакцию системы, оформлены моноши-

ринным шрифтом, причем ввод пользователя отмечен **жирным** выделением:

```
% cat /var/run/named.pid
78
```

- Если команда должна вводиться суперпользователем (администратором системы, или пользователем root), она предваряется символом диеза (#):

```
# /usr/sbin/named
```

- Заменяемые элементы кода оформлены *моноширинным курсивом*.
- Имена доменов, файлов, функций, команд, названия страниц руководства UNIX, функции Windows, URL-адреса, фрагменты кода оформлены *курсивом*, если они расположены в основном тексте.

Это подсказка, предложение или совет общего характера.

Это предупреждение или предостережение.

Использование кода примеров

Эта книга должна помогать вам в работе. Как правило, код из этой книги вы можете использовать в своих программах и документации. Наше разрешение не требуется, за исключением случаев, когда вы собираетесь воспроизвести значительный объем кода. К примеру, написание программы, использующей несколько фрагментов кода из этой книги, разрешения не требует. Продажа или распространение компакт-диска с примерами книг O'Reilly *требует* разрешения. Разрешение не требуется, если вы отвечаете на вопросы, приводя цитаты и примеры кода из этой книги. Разрешение *требуется*, если вы включаете большой объем кода примеров из этой книги в документацию к своему продукту.

Мы не настаиваем, чтобы вы ссылались на первоисточник, но будем признательны, если вы не забудете это сделать. Ссылка обычно включает название, имя автора, издательство и номер ISBN. Например: «DNS and BIND, Fifth Edition, by Cricket Liu and Paul Albitz. Copyright 2006 O'Reilly Media, Inc., 0-596-10057-4».

Если вам кажется, что вы используете примеры более вольно, чем предполагается приведенными выше примерами, или выходите за рамки свободного использования (fair use), свяжитесь с нами по адресу *permissions@oreilly.com*.

Safari® Enabled

Если на обложке книги есть пиктограмма «Safari® Enabled», это означает, что книга доступна в сети Интернет посредством технологии O'Reilly Network Safari Bookshelf (Safari, книжная полка сети O'Reilly.)

Safari предлагает решение, превосходящее электронные книги. Это виртуальная библиотека, которая позволяет выполнять поиск в тысячах лучших технических книг, копировать примеры кода, загружать главы книг на свой компьютер и быстро находить ответы, когда требуется самая точная и свежая информация. Нашу технологию можно бесплатно опробовать по адресу *http://safari.oreilly.com*.

Цитаты

Цитаты из Льюиса Кэррола в каждой из глав приводятся по версии 2.9 издания Millenium Fulcrum электронного текста «Алисы в Стране чудес» из библиотеки проекта Гутенберга (Project Gutenberg) и по изданию 1.7 текста «Алиса в Зазеркалье». Цитаты в главах 1, 2, 5, 6, 8 и 14 из «Алисы в Стране чудес», а цитаты в главах 3, 4, 7, 9–13, 15–17 – из «Алисы в Зазеркалье».[1]

Благодарности

Авторы выражают благодарность Кену Стоуну (Ken Stone), Джерри Мак-Коллому (Jerry McCollom), Питеру Джеффу (Peter Jeffe), Хэлу Стерну (Hal Stern), Кристоферу Дарему (Christopher Durham), Биллу Уизнеру (Bill Wisner), Дэйву Керри (Dave Curry), Джеффу Окамото (Jeff Okamoto), Брэду Ноулзу (Brad Knowles), Роберту Эльцу (K. Robert Elz), а также Полу Викси (Paul Vixie) за их бесценный вклад в написание этой книги. Мы также хотели бы поблагодарить наших рецензентов Эрика Пирса (Eric Pearce), Джека Репенинга (Jack Repenning), Эндрю Черенсона (Andrew Cherenson), Дэна Тринкла (Dan Trinkle), Билла Лефевра (Bill LeFebvre) и Джона Секреста (John Sechrest) за их критику и предложения. Без помощи этих людей эта книга была бы совсем не такой (и была бы гораздо короче!).

За второе издание этой книги авторы выражают благодарность безупречной команде рецензентов: Дэйву Бэрру (Dave Barr), Найджелу Кэмпбеллу (Nigel Campbell), Биллу Лефевру, Майку Миллигану (Mike Milligan) и Дэну Тринклу.

[1] В русском издании цитаты даны в переводе Нины Демуровой (М.:, ПРЕССА, 1992). – *Примеч. ред.*

За третье издание книги авторы признательны команде мечты технических рецензентов: Бобу Хэлли (Bob Halley), Барри Марголину (Barry Margolin) и Полу Викси.

Долг благодарности за четвертое издание причитается Кевину Данлэпу (Kevin Dunlap), Эдварду Льюису (Edward Lewis) и Брайану Веллингтону (Brian Wellington), первоклассной команде рецензентов.

За помощь в работе над пятым изданием авторы благодарят блестящую команду технических рецензентов: Джоао Дамаса (João Damas), Мэтта Ларсона (Matt Larson) и Пола Викси (Paul Vixie), а также Сильвию Хаген (Silvia Hagen) за помощь с IPv6 в последнюю минуту.

Крикет хотел бы отдельно поблагодарить своего бывшего руководителя Рика Норденстена (Rick Nordensten), образцового современного высокопроизводительного менеджера, под присмотром которого была написана первая версия этой книги; своих соседей, которые терпели его эпизодическую раздражительность в течение многих месяцев, и конечно же свою жену Пэйдж за постоянную поддержку и за то, что она мирилась с непрекращающимся, даже во время ее сна, стуком клавиш. Что касается второго издания, Крикет хотел бы добавить слова благодарности в адрес своих бывших руководителей Регины Кершнер (Regina Kershner) и Пола Клоуда (Paul Klouda) за их поддержку работы Крикета с сетью Интернет. За помощь в работе над третьим изданием Крикет считает своим долгом поблагодарить своего партнера Мэтта Ларсона (Matt Larson), который участвовал в разработке Acme Razor; за четвертое он благодарит своих преданных пушистиков Дакоту и Энни – за их поцелуи и участие, а также замечательного Уолтера Б. (Walter B), который время от времени заглядывал в кабинет и проверял, как у папы дела. Что касается пятого издания, он должен упомянуть пополнение, замечательного малыша Джи (Baby G.), и передает благодарности друзьям и коллегам в Infoblox за их тяжелую работу и великодушную поддержку, а также за их компанию.

Пол благодарит свою жену Катерину за ее терпение, за многочисленные разборы полетов и за доказательство того, что она в свободное время может гораздо быстрее сшить стеганое одеяло, чем ее муж напишет свою половину книги.

1

Основы

Кролик надел очки.
– С чего начинать, Ваше Величество? – спросил он.
– Начни с начала, – важно ответил Король, –
продолжай, пока не дойдешь до конца.
Как дойдешь – кончай!

Чтобы понять DNS, необходимо обратиться к истории сети ARPAnet. Система DNS была создана с целью решения конкретных проблем этой сети, а сеть Интернет, выросшая из ARPAnet, сейчас является главным потребителем этих решений.

Опытные пользователи сети Интернет, вероятно, могут пропустить эту главу. Все прочие, мы надеемся, найдут здесь достаточно информации для понимания причин, которые привели к появлению DNS.

(Очень) краткая история сети Интернет

В конце шестидесятых Управление передовых исследований Министерства обороны США (Department of Defense's Advanced Research Agency, или ARPA) – позднее DARPA – открыло финансирование *ARPAnet*, экспериментальной глобальной компьютерной сети, объединившей важные исследовательские организации страны. Первоначальной целью создания этой сети было разделение дорогостоящих или дефицитных компьютерных ресурсов между государственными подрядчиками. Но с самого начала пользователи ARPAnet задействовали сеть и для совместной работы: они обменивались файлами и программами, сообщениями электронной почты (получившей теперь повсеместное распространение), объединяли усилия по разработке и исследованиям, используя разделяемые ресурсы компьютеров сети.

Набор протоколов TCP/IP (Transmission Control Protocol/Internet Protocol) был разработан в начале восьмидесятых годов и быстро получил статус стандарта для сетевого обмена информацией между узлами

ARPAnet. Включение этого набора протоколов в популярную операционную систему BSD UNIX, разработанную в Калифорнийском университете Беркли, сыграло определяющую роль в процессе демократизации и объединения сетей. Система BSD UNIX университетам была доступна практически бесплатно. Так работа с сетями и доступ к ARPAnet стали внезапно доступными и очень дешевыми для большого числа организаций, которые ранее никак не были связаны с ARPAnet. Машины, подключавшиеся к ARPAnet, входили также в состав локальных сетей, и в итоге это привело к объединению многочисленных разрозненных локальных сетей посредством ARPAnet.

Сеть разрослась с очень небольшого числа узлов до десятков тысяч. Первоначальная сеть ARPAnet стала основой объединения локальных и региональных сетей, работающих по протоколам TCP/IP. Это объединение носит название *Интернет*.

Однако в 1988 году организация DARPA пришла к заключению, что эксперимент окончен. Министерство обороны приступило к демонтажу сети ARPAnet. В этот момент несущим остовом для сети Интернет стала другая сеть, которая финансировалась национальным научным фондом (National Science Foundation) и носила название NSFNET.

Весной 1995 года сеть Интернет в очередной раз сменила главную магистраль, финансируемую обществом NSFNET; она уступила место целому ряду коммерческих магистралей, управляемых телекоммуникационными компаниями, такими как SBC и Sprint, а также такими опытными коммерческими сетевыми организациями, как MFS и UUNET.

Сегодня сеть Интернет объединяет миллионы узлов по всему миру. Большая часть не-PC машин подключена к сети Интернет. Некоторые из новых коммерческих информационных магистралей имеют пропускную способность, измеряемую гигабитами в секунду, что в десятки тысяч раз превышает пропускную способность когда-то существовавшей ARPAnet. Ежедневно десятки миллионов людей используют сеть для общения и совместной работы.

Интернет и интернет-сети

Следует сказать пару слов об Интернете и интернет-сетях. В тексте различие между названиями кажется незначительным: одно название всегда пишется с прописной буквы, второе всегда нет. Тем не менее разница в значении *существенна*. Интернет с заглавной буквы «И» – обозначение сети, которая началась с ARPAnet и существует сегодня, грубо говоря, как объединение всех TCP/IP-сетей, прямо или косвенно связанных с коммерческими информационными магистралями США. При внимательном рассмотрении это целый ряд различных сетей – коммерческих опорных сетей TCP/IP, корпоративных и правительственных сетей США, а также TCP/IP-сетей других стран. Сети объединены высокоскоростными цифровыми каналами. Начинающийся со

строчной буквы интернет – это просто любая сеть, объединяющая несколько сетей масштабом поменьше, причем с использованием все тех же протоколов межсетевого взаимодействия. Сеть интернет не всегда связана с сетью Интернет и не обязательно основана на сетевых протоколах TCP/IP. Существуют, к примеру, изолированные корпоративные интернет-сети.

Термин *intranet*, по сути, обозначает всего лишь сети на основе TCP/IP и используется, чтобы подчеркнуть применение технологий, обкатанных в Интернете, в рамках внутренних корпоративных сетей. С другой стороны, *extranet*-сети – это интернет-сети, объединяющие сотрудничающие компании либо компании с их агентами по продаже, поставщиками и клиентами.

История системы доменных имен

В семидесятых годах сеть ARPAnet представляла собой тесное сообщество из нескольких сотен узлов. Всю информацию по узлам, в частности необходимую для взаимных преобразований имен и адресов узлов ARPAnet, содержал единственный файл *HOSTS.TXT*. Известная UNIX-таблица узлов, */etc/hosts*, прямо унаследовала свою структуру от файла *HOSTS.TXT* (в основном с помощью удаления ненужных на UNIX-системах полей).

За файл *HOSTS.TXT* отвечал Сетевой информационный центр (NIC, Network Information Center) Стэнфордского исследовательского института (SRI, Stanford Research Insitute). В тот период времени единственным источником, распространявшим файл, являлся узел SRI-NIC.[1] Администраторы ARPAnet, как правило, просто посылали изменения электронной почтой в NIC и периодически синхронизировали свои файлы *HOSTS.TXT* с копией на узле SRI-NIC с помощью протокола FTP. Присылаемые ими изменения добавлялись в файл *HOSTS.TXT* один или два раза в неделю. Однако по мере роста сети ARPAnet эта схема стала неработоспособной. Размер файла рос пропорционально количеству узлов ARPAnet. Еще быстрее рос информационный поток, связанный с необходимостью обновления файла на узлах: появление одного нового узла приводило не только к добавлению строки в *HOSTS.TXT*, но и к потенциальной необходимости синхронизации данных каждого узла с данными SRI-NIC.

[1] Организация SRI International в настоящее время уже не связана жестко со Стэнфордским исследовательским институтом, расположенным в Менло-Парк (Калифорния); она проводит исследования во многих областях, включая и компьютерные сети.

После перехода ARPAnet на протоколы TCP/IP рост сети стал взрывным. Появился гордиев узел проблем, связанных с файлом *HOSTS.TXT*:

Информационные потоки и нагрузка

Нагрузка на SRI-NIC в смысле сетевого трафика и работы процессора, связанных с раздачей файла, приближалась к предельной.

Конфликты имен

Никакие два узла, описанные в файле *HOSTS.TXT*, не могли носить одинаковые имена. Организация NIC могла контролировать присваивание адресов способом, гарантирующим их уникальность, но не имела никакого влияния на имена узлов. Не было способа предотвратить добавление узла с уже существующим именем, при том что такое действие нарушало работу всей схемы. Так, добавление узла с именем, идентичным имени одного из крупных почтовых концентраторов, могло привести к нарушению работы почтовых служб большей части сети ARPAnet.

Синхронизация

Синхронизация файлов в масштабах быстро растущей сети становилась все более сложной задачей. К тому моменту, когда обновленный файл *HOSTS.TXT* достигал самых далеких берегов выросшей ARPAnet, адреса отдельных узлов успевали измениться или же появлялись новые узлы.

Основная проблема заключалась в том, что схема с файлом *HOSTS.TXT* не поддавалась масштабированию. По иронии судьбы, успех ARPAnet как эксперимента вел к моральному устареванию и провалу механизма *HOSTS.TXT*.

Административные советы ARPAnet начали исследование, которое должно было привести к созданию замены *HOSTS.TXT*. Целью его было создание системы, которая решила бы проблемы, присущие сводной таблице узлов. Новая система должна была позволить локальное управление данными, но делать эти данные доступными всем. Децентрализация администрирования решила бы проблемы с трафиком и нагрузкой, непосильными для единственного узла. Локальное управление данными упростило бы задачу обновления и синхронизации информации. В новой системе следовало использовать иерархическое пространство имен для присваивания идентификаторов узлам, что позволило бы гарантировать уникальность каждого отдельного имени.

Ответственным за разработку архитектуры новой системы стал Пол Мокапетрис, работавший тогда в Институте информационных наук (Information Sciences Institute). В 1984 году он издал документы RFC 882 и 883, в которых описывалась система доменных имен (Domain Name System, или DNS). Эти RFC-документы были обновлены документами RFC 1034 и 1035, которые и являются в настоящее время

действующей спецификацией DNS.[1] RFC 1034 и 1035 к настоящему времени дополняются многими другими подобными документами, в которых описаны потенциальные проблемы DNS с точки зрения сетевой безопасности, возможные трудности реализации, проблемы административного плана, механизмы динамического обновления DNS-серверов, обеспечение безопасности зональных данных и многое другое.

Система доменных имен в двух словах

DNS – это *распределенная* база данных. Такая структура дает возможность локально управлять отдельными сегментами общей базы, а также позволяет сделать данные каждого сегмента доступными всей сети посредством использования механизма «клиент-сервер». Надежность и адекватная производительность основаны на репликации и кэшировании.

Серверная часть клиент-серверного механизма DNS представлена программами, которые называются *DNS-серверами* (*name servers*, дословно – серверами имен). DNS-серверы владеют информацией о некоторых сегментах базы данных и делают ее доступной клиентам, которые называются *поисковыми анализаторами* (*resolvers*).[2] Как правило, DNS-клиент – это просто набор библиотечных функций, которые создают запросы и посылают их по сети серверу имен.

Структура базы данных DNS очень похожа на структуру файловой системы UNIX (рис. 1.1). Вся база данных (или файловая система) представлена в виде перевернутого дерева, корень (корневой узел) которого расположен на самом верху. Каждый узел дерева имеет прикрепленную текстовую метку, которая идентифицирует его относительно родительского узла по аналогии с «относительным путевым именем» в файловой системе (например, *bin*). Одна из меток, пустая, (она обозначается как “ ”) закреплена за корневым узлом дерева. В тексте корневой узел обозначается точкой (.). В файловой системе UNIX корень обозначается символом «слэш» (/).

Каждый узел является корнем новой ветви дерева. Каждая из ветвей (поддеревьев) является разделом базы данных – «каталогом» в интерпретации файловой системы UNIX, или *доменом* в интерпретации системы доменных имен. Каждый домен или каталог может быть раз-

[1] Документы RFC (Request for Comments, запрос комментариев) являются частью относительно неформальной процедуры введения новых технологий в сети Интернет. RFC-документы обычно свободно распространяются и содержат технические описания технологий, предназначенные в основном для разработчиков.

[2] Далее в тексте будут использоваться как термин «поисковый анализатор», так и «клиентская часть DNS», или просто «DNS-клиент», – в зависимости от контекста. – *Примеч. ред.*

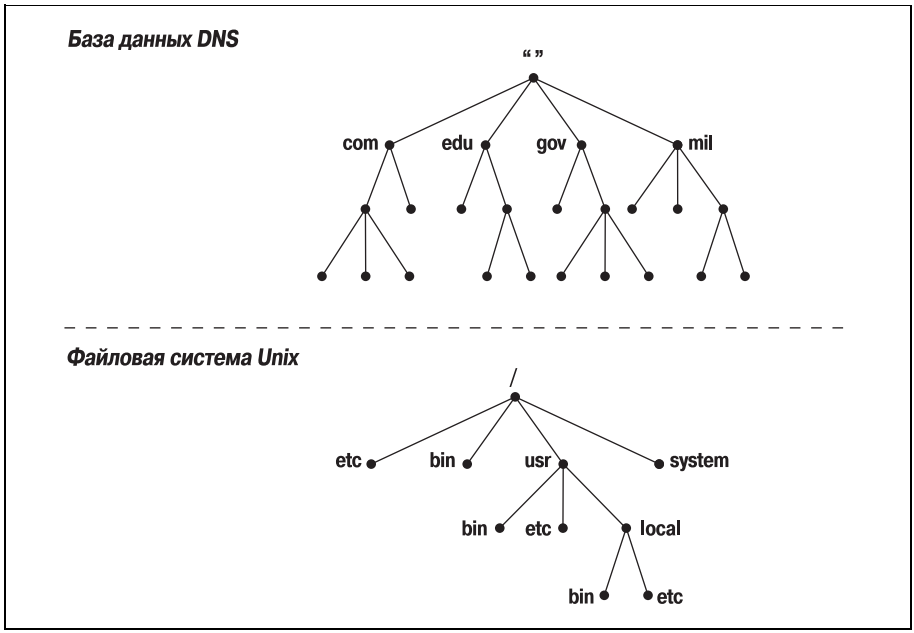


Рис. 1.1. *База данных DNS и файловая система UNIX*

бит на еще более мелкие подразделы, которые в DNS называются *поддоменами*, а в файловых системах – «подкаталогами». Поддомены, как и подкаталоги, изображаются как потомки соответствующих родительских доменов.

Имя домена, как и имя любого каталога, уникально. *Имя домена* определяет его расположение в базе данных, так же как «абсолютный путь к каталогу» однозначно определяет его расположение в файловой системе. Имя домена в DNS – это последовательность меток от узла, корневого для данного домена, до корня всего дерева; метки в имени домена разделяются точками. В файловой системе UNIX абсолютное путевое имя каталога – это последовательность относительных имен, начиная от корня дерева до конкретного узла (то есть чтение происходит в направлении, противоположном направлению чтения имен DNS; рис. 1.2), при этом имена разделяются символом «прямая наклонная черта» («слэш»).

В DNS каждый домен может быть разбит на поддомены, и ответственность за эти поддомены может распределяться между различными организациями. Допустим, организация EDUCAUSE сопровождает домен *edu* (*educational*, то есть образовательный), но делегирует ответственность за поддомен *berkeley.edu* Калифорнийскому университету Беркли (рис. 1.3). Это похоже на удаленное монтирование файловой системы: определенные каталоги файловой системы могут в действительности являться файловыми системами, расположенными на дру-

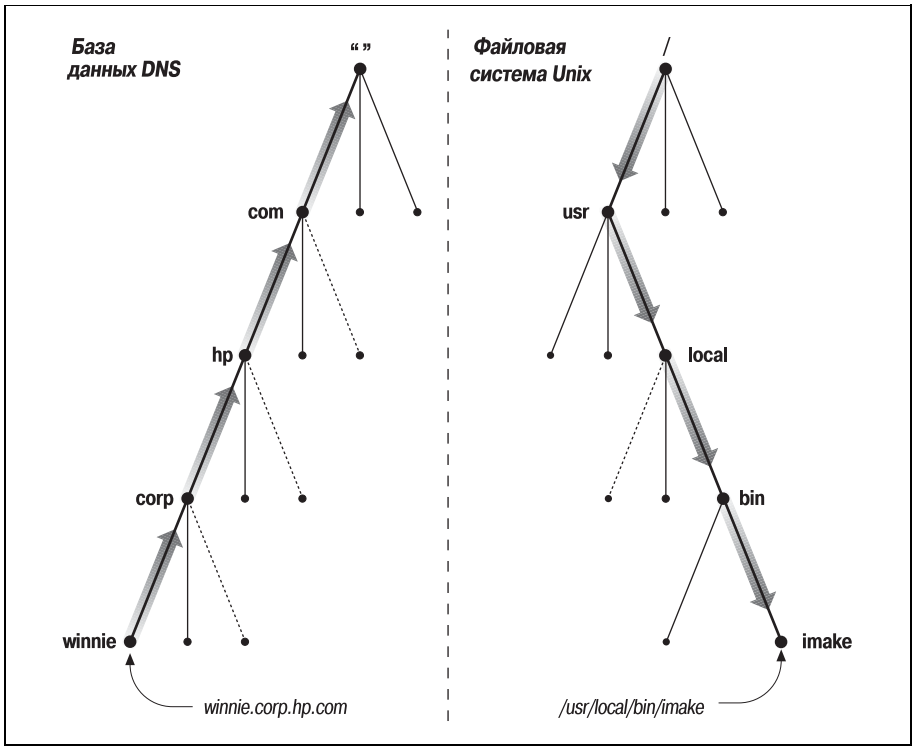


Рис. 1.2. *Чтение имен DNS и файловой системы UNIX*

гих узлах и смонтированными удаленно. К примеру, администратор узла *winken* (рис. 1.3) отвечает за файловую систему, которая на локальном узле выглядит как содержимое каталога */usr/nfs/winken*.

Делегирование управления поддоменом *berkeley.edu* Калифорнийскому университету Беркли приводит к созданию новой *зоны* – независимо администрируемой части пространства имен. Зона *berkeley.edu* теперь не зависит от *edu* и содержит все доменные имена, которые заканчиваются на *berkeley.edu*. С другой стороны, зона *edu* содержит только доменные имена, оканчивающиеся на *edu*, но не входящие в делегированные зоны, такие, например, как *berkeley.edu*. *berkeley.edu* может быть поделен на поддомены с именами вроде *cs.berkeley.edu*, и некоторые из этих поддоменов могут быть выделены в самостоятельные зоны, если администраторы *berkeley.edu* делегируют ответственность за них другим организациям. Если *cs.berkeley.edu* является самостоятельной зоной, зона *berkeley.edu* не содержит доменные имена, которые заканчиваются на *cs.berkeley.edu* (рис. 1.4).

Доменные имена используются в качестве индексов базы данных DNS. Данные DNS можно считать «привязанными» к доменному имени. В файловой системе каталоги содержат файлы и подкаталоги. Анало-

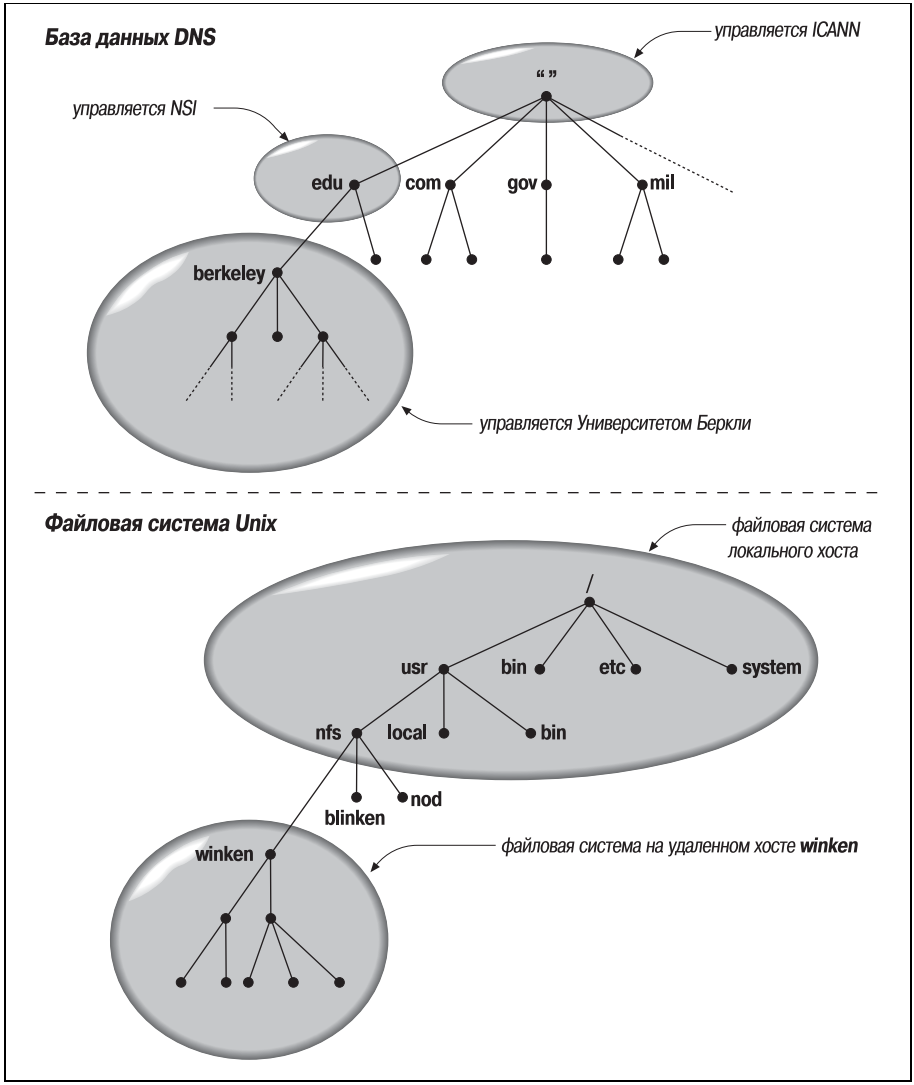


Рис. 1.3. *Удаленное управление доменами разных уровней и файловыми системами*

гичным образом домены могут содержать узлы и поддомены. Домен включает в себя те узлы и поддомены, доменные имена которых расположены в принадлежащей этому домену части иерархии имен.

У каждого узла в сети есть доменное имя, которое является указателем на информацию об узле (рис. 1.5). Эта информация может включать IP-адреса, информацию о маршрутизации почтовых сообщений и другие данные. Узел может также иметь один или несколько *псевдонимов доменного имени*, которые являются просто указателями на ос-

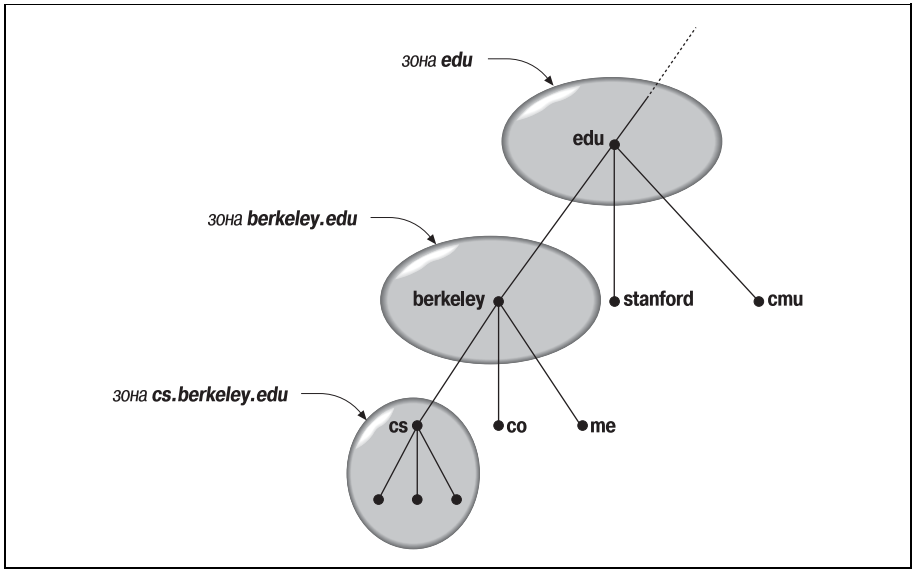


Рис. 1.4. *Зоны edu, berkeley.edu и cs.berkeley.edu*

новное (официальное, или *каноническое*) доменное имя. На рис. 1.5 *mailhub.nv...* – псевдоним канонического имени *rincon.ba.ca...*

Для чего нужна столь сложная структура? Чтобы решить проблемы, существовавшие при использовании *HOSTS.TXT*. К примеру, строгая иерархичность доменных имен устраняет угрозу конфликтов имен. Имя каждого домена уникально, так что организация, управляющая

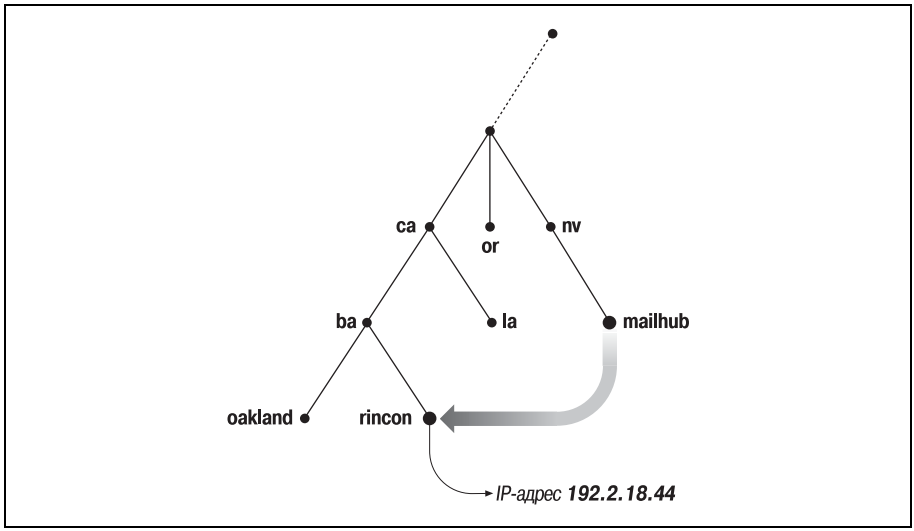


Рис. 1.5. *Псевдоним в DNS, ссылающийся на каноническое имя*

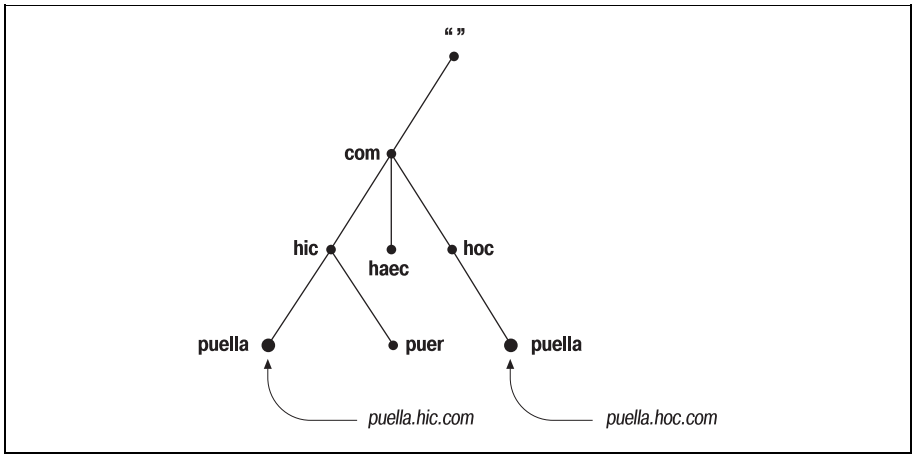


Рис. 1.6. *Решение проблемы конфликтов имен*

доменом, вольна придумывать имена поддоменов, входящих в этот домен, самостоятельно. Независимо от выбранных имен, имена эти не будут конфликтовать с доменными именами других организаций, поскольку заканчиваются уникальным именем домена, сопровождаемого только этой организацией. Так, организация, ответственная за домен *hic.com*, может дать узлу имя *puella* (рис. 1.6), поскольку известно, что доменное имя узла будет заканчиваться уникальным доменным именем *hic.com*.

История пакета BIND

Первая реализация системы доменных имен называлась JEEVES и была разработана самим Полом Мокапетрисом (Paul Mockapetris). Более поздняя реализация носит название BIND, это аббревиатура от *Berkeley Internet Name Domain*, и была разработана для операционной системы 4.3 BSD UNIX (Беркли) Кевином Данлэпом. В настоящее время развитием и сопровождением пакета BIND занимается Internet Systems Consortium.[1]

На пакете BIND мы и сосредоточимся в данной книге, поскольку именно BIND является сегодня наиболее популярной и распространенной реализацией DNS. Пакет доступен на большинстве разновидностей системы UNIX и входит в стандартную конфигурацию систем от большинства поставщиков UNIX. BIND также был портирован на платформу Microsoft Windows NT, Windows 2000 и Windows Server 2003.

[1] Более подробно об организации Internet Systems Consortium и ее разработках в области BIND можно узнать по адресу *http://www.isc.org/sw/bind/*.

Надо ли мне использовать DNS?

Несмотря на очевидную пользу DNS, существуют случаи, в которых ее применение неоправданно. Помимо DNS существуют и другие механизмы разрешения имен, некоторые из которых могут быть составной частью операционной системы. Иногда затраты сил и времени, связанные с сопровождением зон и DNS-серверов, превышают все возможные выгоды. С другой стороны, возможна ситуация, когда нет другого выбора, кроме как установить и поддерживать DNS-серверы. Вот некоторые указания, которые помогут сориентироваться и принять решение:

Если вы подключены к сети Интернет...

...DNS является жизненной необходимостью. DNS можно считать общепринятым языком сети Интернет: почти все сетевые службы в Интернете, включая Web, электронную почту, удаленный терминальный доступ и передачу файлов, используют DNS.

С другой стороны, подключение к сети Интернет вовсе не означает, что не удастся избежать самостоятельной установки и сопровождения нужных пользователю зон. В случае ограниченного числа узлов всегда можно найти уже существующую зону и стать ее частью (подробнее – в главе 3 «С чего начать?») или найти кого-то, кто позаботится о размещении зоны. Если пользователь платит интернет-провайдеру за подключение, обычно существует возможность разместить свою зону на технологических мощностях этого провайдера. Существуют также компании, предоставляющие подобную услугу за отдельную плату.

Если же узлов много или очень много, то, скорее всего, понадобится самостоятельная зона. Если вы хотите иметь непосредственный контроль над зоной и серверами имен, то вступайте на путь администрирования и сопровождения. Читайте дальше!

Если у вас интернет-сеть на основе протоколов TCP/IP...

...то DNS, вероятно, не помешает. В данном случае под интернет-сетью мы не подразумеваем простую сеть из одного сегмента Ethernet и нескольких рабочих станций, построенную на протоколах TCP/IP (если вы так подумали, обратитесь к следующему разделу), но достаточно сложную «сеть сетей», например несколько десятков Ethernet-сегментов, объединенных при помощи маршрутизаторов.

Если интернет-сеть является преимущественно гомогенной и узлы не нуждаются в службе DNS (скажем, если они вообще не используют TCP/IP), вполне возможно, что можно обойтись без нее. Но в случае разнородных узлов, в особенности если некоторые из них работают под управлением UNIX, DNS пригодится. Система упростит распространение информации об узлах и избавит администра-

тора от необходимости выдумывания своей схемы распространения таблиц узлов.

Если у вас собственная локальная сеть...

...и эта сеть не соединена с большей сетью, вполне возможно обойтись без DNS. Можно попробовать использовать службу Windows Internet Name Service (WINS) от Microsoft, таблицы узлов или Network Information Service (NIS) от Sun.

В случаях, когда требуется распределенное администрирование либо присутствуют сложности с синхронизацией данных в сети, использование DNS может иметь смысл. И если планируется подключение вашей сети к другой, скажем к корпоративной интернет-сети либо к Интернету, стоит заранее заняться настройкой собственных зон.