Setting Up LAMP: Getting Linux, Apache, MySQL, *and* PHP Working Together

Setting Up LAMP: Getting Linux, Apache, MySQL, *and* PHP Working Together

Eric Rosebrock
Eric Filson

San Francisco · London

Associate Publisher: Joel Fugazzotto
Acquisitions Editor: Tom Cirtin
Developmental Editor: Tom Cirtin
Production Editor: Lori Newman
Technical Editor: Sean Schluntz
Copyeditor: Sharon Wilkey
Compositor: Laurie Stewart, Happenstance Type-O-Rama
Proofreaders: Nancy Riddiough, Laurie O'Connell
Indexer: Nancy Guenther
Cover Designer: Caryl Gorska, Gorska Design
Cover Photographer: Peter Samuels, Tony Stone

Copyright © 2004 SYBEX Inc., 1151 Marina Village Parkway, Alameda, CA 94501. World rights reserved. No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photograph, magnetic, or other record, without the prior agreement and written permission of the publisher.

Library of Congress Card Number: 2004104101

ISBN: 0-7821-4337-7

SYBEX and the SYBEX logo are either registered trademarks or trademarks of SYBEX Inc. in the United States and/or other countries.

Transcend Technique is a trademark of SYBEX Inc.

TRADEMARKS: SYBEX has attempted throughout this book to distinguish proprietary trademarks from descriptive terms by following the capitalization style used by the manufacturer.

The author and publisher have made their best efforts to prepare this book, and the content is based upon final release software whenever possible. Portions of the manuscript may be based upon pre-release versions supplied by software manufacturer(s). The author and the publisher make no representation or warranties of any kind with regard to the completeness or accuracy of the contents herein and accept no liability of any kind including but not limited to performance, merchantability, fitness for any particular purpose, or any losses or damages of any kind caused or alleged to be caused directly or indirectly from this book.

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

SOFTWARE LICENSE AGREEMENT: TERMS AND CONDITIONS

The media and/or any online materials accompanying this book that are available now or in the future contain programs and/or text files (the "Software") to be used in connection with the book. SYBEX hereby grants to you a license to use the Software, subject to the terms that follow. Your purchase, acceptance, or use of the Software will constitute your acceptance of such terms.

The Software compilation is the property of SYBEX unless otherwise indicated and is protected by copyright to SYBEX or other copyright owner(s) as indicated in the media files (the "Owner(s)"). You are hereby granted a single-user license to use the Software for your personal, noncommercial use only. You may not reproduce, sell, distribute, publish, circulate, or commercially exploit the Software, or any portion thereof, without the written consent of SYBEX and the specific copyright owner(s) of any component software included on this media.

In the event that the Software or components include specific license requirements or end-user agreements, statements of condition, disclaimers, limitations or warranties ("End-User License"), those End-User Licenses supersede the terms and conditions herein as to that particular Software component. Your purchase, acceptance, or use of the Software will constitute your acceptance of such End-User Licenses.

By purchase, use or acceptance of the Software you further agree to comply with all export laws and regulations of the United States as such laws and regulations may exist from time to time.

Reusable Code in This Book

The author(s) created reusable code in this publication expressly for reuse by readers. Sybex grants readers limited permission to reuse the code found in this publication, its accompanying CD-ROM or available for download from our website so long as the author(s) are attributed in any application containing the reusable code and the code itself is never distributed, posted online by electronic transmission, sold, or commercially exploited as a stand-alone product.

Software Support

Components of the supplemental Software and any offers associated with them may be supported by the specific Owner(s) of that material, but they are not supported by SYBEX. Information regarding any available support may be obtained from the Owner(s) using the information provided in the appropriate read.me files or listed elsewhere on the media.

Should the manufacturer(s) or other Owner(s) cease to offer support or decline to honor any offer, SYBEX bears no responsibility. This notice concerning support for the Software is provided for your information only. SYBEX is not the agent or principal of the Owner(s), and SYBEX is in no way responsible for providing any support for the Software, nor is it liable or responsible for any support provided, or not provided, by the Owner(s).

Warranty

SYBEX warrants the enclosed media to be free of physical defects for a period of ninety (90) days after purchase. The Software is not available from SYBEX in any other form or media than that enclosed herein or posted to `www.sybex.com`. If you discover a defect in the media during this warranty period, you may obtain a replacement of identical format at no charge by sending the defective media, postage prepaid, with proof of purchase to:

SYBEX Inc.
Product Support Department
1151 Marina Village Parkway
Alameda, CA 94501
Web: `http://www.sybex.com`

After the 90-day period, you can obtain replacement media of identical format by sending us the defective disk, proof of purchase, and a check or money order for $10, payable to SYBEX.

Disclaimer

SYBEX makes no warranty or representation, either expressed or implied, with respect to the Software or its contents, quality, performance, merchantability, or fitness for a particular purpose. In no event will SYBEX, its distributors, or dealers be liable to you or any other party for direct, indirect, special, incidental, consequential, or other damages arising out of the use of or inability to use the Software or its contents even if advised of the possibility of such damage. In the event that the Software includes an online update feature, SYBEX further disclaims any obligation to provide this feature for any specific duration other than the initial posting.

The exclusion of implied warranties is not permitted by some states. Therefore, the above exclusion may not apply to you. This warranty provides you with specific legal rights; there may be other rights that you may have that vary from state to state. The pricing of the book with the Software by SYBEX reflects the allocation of risk and limitations on liability contained in this agreement of Terms and Conditions.

Shareware Distribution

This Software may contain various programs that are distributed as shareware. Copyright laws apply to both shareware and ordinary commercial software, and the copyright Owner(s) retains all rights. If you try a shareware program and continue using it, you are expected to register it. Individual programs differ on details of trial periods, registration, and payment. Please observe the requirements stated in appropriate files.

Copy Protection

The Software in whole or in part may or may not be copy-protected or encrypted. However, in all cases, reselling or redistributing these files without authorization is expressly forbidden except as specifically provided for by the Owner(s) therein.

I would like to dedicated this book to my Wife and children who have given me patience, encouragement, love, and support during the past 11 years of my life as I progressed on the paths that I have chosen. They have been nothing less than supportive of everything I have chosen to do. Tracy, Summer, and Gage, I love you all and I thank you for being there for me.

—Eric Rosebrock

I would like to dedicate this book to my loving sister Amy. She has always worked diligently to achieve her goals and I know myself she can accomplish anything she sets her mind to. Although we have spent many years apart, I feel closer to her now than ever. She has been an inspiration in my life and has always had the courage to stand up for what she believes in regardless of the consequences. It is a powerful and rare individual who lays their own credibility and name on the line for the betterment of ideals and morals in today's society. Thank you for being who you are.

—Eric Filson

Acknowledgments

I would like to acknowledge some of my friends and associates who have supported me with enthusiasm, kind words, and encouragement throughout the writing of this book. Thanks to Eric Filson, Genesis Font, Dustin Whittle, Chris Kozlowski, Calvin Fraites, Richard Blundell, and Jeraimee Hughes.

I would also like to thank Sybex for their support and for believing in me as an author. The Sybex team: Tom Cirtin, Joel Fugazzotto, Lori Newman, Sean Schluntz, and the rest of the folks at Sybex who have worked on our book. Sybex is a wonderful publishing company, and I only wish our readers could truly understand what an outstanding experience it is to work with professionals of this level.

Finally, I would like to thank the open source developers, community, and end users. This community has fueled my desire to progress through the learning phases and allowed me to contribute everything I can for sharing this excellent concept. Nowhere have I ever found a group of online communities that dedicate every spare moment to helping other people with learning the ways of this masterful puzzle. This has truly been the most excellent computer experience I have ever found, and it is because of you.

—Eric Rosebrock

In addition to echoing the excellent words spoken about the outstanding team at Sybex who made this book possible and the open source community who fuels the fire of the future, I would like to add a few thanks of my own. To do so, I'll need to start at the beginning with those who have been a main inspiration in my life.

First, to Roberta and John Williams, the creator and developer, respectively, of King's Quest I (among many others). This game was released in 1984 and was my main inspiration at the age of five for dedicating my life to computers. They taught me it was possible to imbue technology with creativity, giving birth to original works that can force the world to pay attention and change the way people think. I will never forget what they unwittingly gave me.

In addition, I would like to thank key individuals who stand out in my mind as supporting my drives and teaching me the ways of the world. To Rick Kelly, who showed me the behind-the-scenes of computers in the '80s. To Kenny Whisenhunt, who helped me into the world of BBSs and furthered my knowledge of computers. To Shaun Hopkins, who taught me the mystical art of business and sales. Especially to Jeremy Thompson, who has been a friend in times of need and who has shown me the way to the fine art of higher-level programming. And most recently to Eric Rosebrock, who invited me to team with him to produce this great book. Of course my

family cannot go without my thanks for their enduring push to always continue my education. And once again, many thanks to the readers who now continue their pursuit of education through this book.

A very special thanks to Kevin Anderson, his beautiful wife, Susan, and darling baby girl, Eva, for their support of my work and enduring friendship; I am forever grateful. Additional thanks to Travis Turner, Mason Ambrose, and Nathan Chase.

—Eric Filson

Contents at a Glance

Introduction *xix*

Chapter 1: Introducing LAMP **1**

Chapter 2: Installing Linux **11**

Chapter 3: Using Linux **33**

Chapter 4: Linux Administration **71**

Chapter 5: Network Connectivity **107**

Chapter 6: Linux Security **133**

Chapter 7: Electronic Mail **161**

Chapter 8: Apache Web Server: Installation and Configuration **193**

Chapter 9: MySQL: Installation and Administration **243**

Chapter 10: PHP: Installation and Configuration **283**

Chapter 11: Testing Your LAMP Installation **319**

Appendix A: LAMP Quick Installation **327**

Appendix B: MySQL Configuration Directives **349**

Appendix C: Getting Support **367**

Index *375*

Contents

Introduction *xix*

Chapter 1 Introducing LAMP **1**

What Is LAMP? 2
L—Linux Operating System *3*
A—Apache Web Server *3*
M—MySQL Database Server *4*
P—PHP Scripting Language and Engine *5*
Why LAMP? 6
Why LAMP Is Free 6
LAMP versus Other Solutions 7
Who Else Is Using LAMP? 8
Understanding LAMP Checklist 9

Chapter 2 Installing Linux **11**

Choosing Which Linux to Use 12
Meeting Hardware Requirements 14
Obtaining Fedora Linux 16
Installing Fedora Linux 17
Pre-Installation 17
Installation Type Selection 18
Hard Disk Partitioning 20
Selecting a Boot Loader 22
Configuring the Network 23
Firewall Configuration 24
Miscellaneous Settings 25
Package Selection 25
Package Installation 27
Boot Diskette Creation 27
Installation Complete! 27

Booting Up 27
Post-Installation Setup 28
Changing the Bootup Type 30
Linux Installation Checklist 32

Chapter 3 Using Linux 33

Booting Linux 34
Your First Package Installation 35
A Closer Look at */etc/inittab* 36
Linux Initialization Scripts 38
The *rc* Script and Its Kill and Startup Files 40
Run Level Scripts 41
Login Process 43
Exploring the Linux Shell 43
Understanding the Bash Environment 46
Navigating through Linux 48
Understanding the Linux Filesystem Layout 52
`/bin` *53*
`/boot` *53*
`/dev` *53*
`/etc` *54*
`/home` *64*
`/lib` *64*
`/lost+found` *65*
`/mnt` *65*
`/opt` *65*
`/proc` *65*
`/root` *68*
`/sbin` *68*
`/tmp` *68*
`/usr` *69*
`/var` *70*
Using Linux Checklist 70

Chapter 4 Linux Administration 71

Managing Users and Groups 72
The */etc/passwd* File 73
The */etc/group* File 74
Linux Passwords 76
User Administration 76
Group Administration 78
Modifying Users or Groups 79
Managing Services and Processes 80
Gathering System Information 80
Creating Disk Quotas 85
Starting and Stopping the System and Services 88
Controlling Access to Services 89
Managing Software 91
Traditional Source Tarballs 91
Source Code versus Binary Packages 94
RPM and RPM Source Packages 95
Performing System Backup and Recovery 100
Critical Data 100
Backup Media 101
Backing Up Your System 101
System Restoration 105
Linux Administration Checklist 105

Chapter 5 Network Connectivity 107

Using TCP/IP 109
Internet Protocol 109
Transmission Control Protocol 110
User Datagram Protocol 110
Ports 110
Addressing 111
Configuring Your Address 113
Static IP 114
DHCP 115

Setting Up Hardware 116
Network Cards 117
Hubs 118
Switches 118
Routers 119
Routing Switches 120
Bridges 120
Repeaters 120
Understanding Firewalls 121
The Linux Kernel as a Firewall 122
Why Close Ports 122
How a Firewall Can Increase Security 123
Using DNS 123
The Domain Name Space 124
DNS Operation 124
The */etc/resolv.conf* File 128
The */etc/hosts* File 129
Record Types 129
Network Connectivity Checklist 130

Chapter 6 Linux Security 133

Disabling Unwanted Services 134
Utilizing *chkconfig* 134
Utilizing *ntsysv* 136
Determining the Purpose of a Service 137
Disabling Services 139
Staying Up-to-Date 140
Using up2date 140
Using APT Package Management 141
Using Yum Package Management 141
Controlling Root Access 142
Creating a Standard User Account 142
Disabling Remote Root Login 143
Configuring the Firewall 144
Creating the Firewall Rules Script 144
Configuring the Firewall to Run at Startup 150

Monitoring the *iptables* Log 153
Don't Panic, Just Drop It! 153
Using Network Monitoring and Testing Applications 154
Nmap 154
Snort 156
Ping 157
Traceroute 158
Linux Security Checklist 159

Chapter 7 Electronic Mail 161

Understanding How E-Mail Works 162
The E-Mail Message 162
Electronic Mail Protocols 164
Electronic Mail Transport Agents 165
Installing the qmail MTA 166
Obtaining the Source Files 167
Preparing for Installation 169
Installing qmail 170
Installing qmail Add-Ons 179
Opening Firewall Ports for the E-Mail Server 188
Managing Your qmail Server 188
Monitoring Logs 189
Understanding qmail Actions 189
Managing Virtual Domains and E-Mail Accounts 189
Electronic Mail Checklist 190

Chapter 8 Apache Web Server: Installation and Configuration 193

Choosing a Version: Apache 1.3 vs. Apache 2.0 194
New Features in Apache 2.0 195
Module Enhancements in Apache 2.0 196
Which Version of Apache to Use 197
Understanding Apache 1.3 197
Apache 1.3 Features 197
Apache 1.3 Modules 206
Preparing to Install Apache Web Server 209
Removing Pre-Installed Apache Web Server RPMs 209

Downloading and Unpacking Apache 210
Allowing HTTP Server Access through the Firewall 211
Installing Apache Web Server 1.3 211
Using the Non-SSL Dynamic Shared Objects Installation Method 211
Using the Non-SSL Traditional Installation Method 214
Using the SSL Dynamic Shared Object Installation Method 215
Exploring Apache Directories 218
Becoming Familiar with Apache Programs 219
Understanding the *httpd.conf* File 222
Using Apache Virtual Hosts 228
Directives Supported by Virtual Hosts 229
Configuring Apache Virtual Hosts 230
A Lesson in Testing Configuration File Changes 234
Performing Other Apache Configurations 235
Enabling Directory Listings 235
Password-Protecting Web Directories 236
Configuring *cgi-bin* Directories 238
Using *.htaccess* Files for Local Directory Configurations 239
Configuration File Summary 239
Starting Apache During Boot 240
Apache Web Server Installation and Configuration Checklist 241

Chapter 9 MySQL: Installation and Administration 243

Understanding MySQL and Database Structure 244
Differences between Flat-File and Relational Databases 245
Advantages and Limitations of MySQL 246
MySQL Version Differences 248
Downloading MySQL 250
Installing MySQL 251
Preparing the System 251
Using Common Configuration Directives 253
Installing MySQL Server and Client 255
Configuring MySQL after Installation 256
Editing the Configuration Files 257
Configuring MySQL to Start During Boot 259
Enhancing Security 260

Performing MySQL Administration 263
Using Command-Line Tools 263
Creating and Administering Databases 269
Creating and Administering Users 272
Backing Up and Restoring Databases 274
Performance and Replication 275
Understanding How Replication Threads Work 276
Introduction to Replication Application 278
MySQL Installation and Administration Checklist 281

Chapter 10 PHP: Installation and Configuration 283

Why Use PHP? 284
PHP Versions 285
Installing PHP 285
Determining Configuration Options and Extensions 286
Downloading and Preparation 293
Compiling and Installing PHP 293
Configuring Apache to Handle PHP 295
Testing the PHP Installation 295
The PHP INI File 297
PHP Installation and Configuration Checklist 317

Chapter 11 Testing Your LAMP Installation 319

Setting Up the Apache Virtual Host 320
Preparing the MySQL Database and User 321
Testing Apache, PHP, and MySQL 321
Database Connection Script 322
Data Insertion Script 322
Data Extraction and Formatting Script 323
Troubleshooting 324
Testing Your LAMP Installation Checklist 325

Appendix A LAMP Quick Installation 327

Linux 328
Security 329
E-Mail 333

Vpopmail 339
Courier-IMAP 339
POP3 340
Clam AntiVirus 341
qmail-Scanner 342
Apache Web Server 343
MySQL 345
PHP 346

Appendix B MySQL Configuration Directives 349

MySQL Configuration Directives 350
Installation Configuration Directives 362

Appendix C Getting Support 367

Books on Open Source Technologies 368
Linux 368
Apache 368
MySQL 369
PHP 369
Additional Software 369
Local Resources 370
Internet Resources 370
Websites 370
Mailing Lists 372
Newsgroups 372
Internet Relay Chat 373

Index *375*

Introduction

Welcome to *Setting Up LAMP: Getting Linux, Apache, MySQL, and PHP Working Together.* This book is written for those people out there who either struggle with this technology or just want a simple solution that is compiled by the experts. We have taken years of experience, problem solving, experimentation, and most importantly *implementation*, and bundled it all together as an easy-to-understand practical guide for you to utilize as you set up your LAMP web server. In this book you will find the following topics thoroughly explained:

Linux Throughout this book you will learn about Linux. We discuss where it came from and how it developed, which flavor of Linux to use, how to install it, and, most importantly, how to use it. Each chapter is a practical guide and exercise on everyday Linux techniques. If you are new to Linux, simply following the examples in this book will definitely bring you up to a level of understanding and application through real-world techniques that will hone your skills as a Linux user or administrator.

Apache Web Server We discuss how to download, build, install, administer, and configure the Apache Web Server. We show you the best methods of how to properly build your server, understand the configuration directives, configure virtual hosts, set up password-protected directories, and much more. You will have a firm understanding of and should feel comfortable running your own Linux-based Apache Web Server when you have completed this book.

MySQL This book discusses how to download, build, install, administer, and configure MySQL. We show you the best methods for running your own MySQL server, protecting it from the outside world, and creating user accounts, databases, and tables. We cover MySQL 5, which is the newest version, and we cover its excellent features in depth.

PHP This book discusses how to download, build, install, administer, and configure the PHP: Hypertext Preprocessor language for your web server. Additionally, we detail how to understand PHP. We do not show you how to develop websites with PHP, however; we focus on the server-side aspects of configuring, securing, and running it.

qmail E-mail Server In this book, we tackle one of the most popular e-mail servers: qmail. qmail is a robust replacement for the old Sendmail application, and we show you how to download, build, install, and configure it to run efficiently with spam filters, antivirus software, IMAP, POP3, and SMTP protocols. By the time you are finished with this book, you will have a top-notch, high-quality e-mail server that can be used in any application, large or small.

Conventions Used in This Book

This book uses certain typographic styles in order to help you quickly identify important information, and to avoid confusion over the meaning of words such as onscreen prompts. In particular:

- A normal, proportionally spaced font is used for the bulk of the text in the book.
- *Italicized text* indicates technical terms that are introduced for the first time in a chapter. (Italics are also used for emphasis.)
- `A monospaced font` is used to indicate the contents of configuration files, messages displayed at a text-mode Linux shell prompt, filenames, commands, and Internet URLs.
- *`Italicized monospaced text`* indicates variables—information that differs from one system or command run to another, such as the name of a client computer or a process ID number.
- **`Bold monospaced text`** is information that you're to type into the computer, usually at a Linux shell prompt. This text can also be italicized to indicate that you should substitute an appropriate value for your system.

In addition to these text conventions, which can apply to individual words or entire paragraphs, a few conventions are used to highlight segments of text:

NOTE A Note indicates information that's useful or interesting, but that's somewhat peripheral to the main discussion. A Note might be relevant to a small number of networks, for instance, or refer to an outdated feature.

TIP A Tip provides information that can save you time or frustration, and that might not be entirely obvious. A Tip might describe how to get around a limitation or how to use a feature to perform an unusual task.

WARNING Warnings describe potential pitfalls or dangers. If you fail to heed a Warning, you could end up spending a lot of time recovering from a bug or even restoring your entire system from scratch.

Sidebars

A Sidebar is like a Note but addresses a larger topic. The information in a Sidebar is useful, but it doesn't fit into the main flow of the discussion.

Help Us Help You

In the open source world, things change rapidly. In fact, things change so rapidly that information found in this book will likely become outdated over time. We ask that you help us locate and identify these changes and report them to us as quickly as possible. However, we ask that you validate the information beforehand so that we can focus on realistic information and not a simple misunderstanding on the part of the reader. It is possible that this book might have typographical errors as well due to its size and nature. If you find an error or a change that needs to be submitted, please send an e-mail to `support@sybex.com`, and they will route the issue to the appropriate channels (editorial staff or the authors).

Chapter 1

Introducing LAMP

- What is LAMP?
- Why LAMP?

If you are a webmaster, a web developer, or a company that is interested in running your own web server, look no further: LAMP is here. *LAMP*, which stands for *Linux*, *Apache*, *MySQL*, *and PHP*, is a robust combination of applications driven on the powerful Linux operating system. This combination of technologies has a proven track record of being efficient, secure, and always on the leading edge of the ever so popular Internet.

The goal of this book is to teach you how to install and configure a web server by using the LAMP technology. You can rest assured that by the time you are finished reading this book, you will have a strong, powerful, and efficient web server to host your personal websites, powerful e-commerce or business-based websites, and anything in between.

The software used in this book is completely free to use and will not cost you any money in licensing or purchasing, unless you choose to purchase discs. Additionally, all of the software is easily downloaded through the Internet.

If you are ready to save yourself time and money by running and managing your own server, then read on!

What Is LAMP?

LAMP is a proven, efficient set of software that works well as a system. The open architecture of each of these elements allows for smooth and seamless integration with one another and results in a powerful combination. Early adopters of these technologies back in 1997 were seen as radical, but today the open source movement is on the rise, and both large and small enterprises are adopting the LAMP method of development. Steering away from the high costs of implementing licensed server and client software is becoming increasingly beneficial because the stability of each application is surpassing that of its far more expensive brethren. In recent events, some governments have decided to make the jump to open source software, touting its reliability, efficiency, and substantial cost savings over proprietary solutions.

Besides reaching this dependability level capable of being embraced by entire governments, the major advantage seen by LAMP adopters is *speed.* Each component of LAMP exhibits benchmarks that far exceed those of their competitors, and as with any equation the sum is much more than its parts. Let's take a look at just the Linux/Apache combination for a moment. This outstanding combination is capable of serving more pages to its users than any other commercial or open source solution. "What about MySQL?" you might ask. MySQL is the fastest open source database available, with speed comparable to that of Oracle's. This fact alone pushed NASA (the National Aeronautics and Space Administration) to switch to MySQL in 2000. With MySQL, the level of functionality offered to its consumers grows in leaps and bounds—most notably by offering *stored procedures*, a system found only in mature databases.

Which brings us to the power of PHP. PHP is the fastest server-side scripting program on the planet. Faster than a one-legged chicken on a downhill slope, PHP whisks by Active Server Pages (ASP), Java, .Net, and ColdFusion, allowing a greater maximum user count per server while providing the same amount of functionality—taking into account proper programming methods of course.

Now that we've touched on a few of the major advantages of choosing LAMP, let's take a brief look at each of its elements.

L—Linux Operating System

Linux is the operating system that runs the applications. It is specifically noted for its speed, minimal hardware requirements, security, and remote administration. Another great key point about Linux is that it's free! Linux is a fully featured operating system that doesn't cost you anything to use. You can download it directly from the Internet, install it, and use it without ever spending a dime (except for your own hardware of course). If you're curious about how this is possible, see the "Why LAMP Is Free" section later in this chapter.

Another major advantage of Linux is its ability to run with or without a graphical user interface (GUI), depending on your needs. (The non-GUI interface could almost be related to DOS mode for those Windows users who are not yet familiar with Linux.)

Linux is a project that began as a hobby by mastermind Linus Torvalds while he was a student at the University of Helsinki in Finland. Torvalds, originally working with Minix (a small Unix system), decided to create an operating system that would exceed the Minix standards. He began his development in 1991, and his first public release was version 0.02. Development of Linux continues even now with updates released as enough major changes are made to justify a new *version* release. Now that Torvalds has gained a much larger development team, releases are becoming more and more frequent. He is also the one who chose Tux (the penguin) to be the mascot of Linux.

Because Linux is published under the GNU (GNU stands for *GNU's Not Unix*) General Public License (GPL), many companies and individuals have taken the source code and adapted it to their needs. Some (most notably Red Hat) have turned huge profits by offering support for their releases and are now closing their source to commercialize their distributions.

A—Apache Web Server

Developed by the Apache Software Foundation (ASF), *Apache* is an open source web server solution that is packed with features, is extremely fast, and works well with the Linux operating system. With the Apache web server, you can create *virtual hosts* that enable you to run multiple websites on a single server, and it has many more awesome features. The Apache web server is available for the Windows environment as well; however, your system will suffer from decreased

performance because of Microsoft's memory management (a.k.a. *leaks*) and architectural differences. Therefore it is *highly* recommended that you use Linux for all your web serving needs.

A quick overview of Apache's features would include items such as enhanced logging, bandwidth throttling, directory access protection, Common Gateway Interface (CGI) support, Secure Sockets Layer (SSL) support, and a handful of other built-in modules that enable you to do all sorts of neat things with your website.

Apache is rapidly growing in popularity and is currently the number one web server solution according to Netcraft (`www.netcraft.com`) surveys. It has held the number one position, with a large margin, for eight years. Taking the number one spot in 1996, Apache has grown to dominate the market with more than a 69 percent saturation. Coming in a rough second place is Microsoft, with 20 percent, which has stayed about the same since 1997. Next is Sun ONE with 3 percent.

NOTE If you are interested in more statistics about Apache, take a look at the current Netcraft Web Server Survey located at `http://news.netcraft.com/archives/web_server_survey.html`.

M—MySQL Database Server

MySQL is a powerful, robust database manager that enables you to store and retrieve data with a scripting language such as PHP. You can store various types of data, such as Boolean operators, text, integers, images, binary digits, and BLOBs (binary large objects) quickly and efficiently with minimal effort. Using a database is important for creating dynamic sites. The term "dynamic site" is derived from being able to utilize a single page of code to display different information based on a user's interaction. This would be virtually impossible without the use of a database and a scripting language such as PHP to manipulate the data.

MySQL is packed full of features such as data replication, table locking, query limiting, user accounts, multiple databases, persistent connections, and—as of MySQL 5—stored procedures, triggers, and views. These features will be explained in more detail later, but for now you should be aware of some of the benefits you will enjoy from implementing such a great database manager.

MySQL, developed by MySQL AB, originated from a need for the founders to use mSQL to connect to their own fast, low-level (Indexed Sequential Access Method, or ISAM) routines. After testing these procedures and functions, they were found to be neither fast nor flexible enough and so MySQL was born: a new system from almost the same Application Protocol Interface (API) as mSQL, so that any third-party code that might be written for mSQL could easily be ported to MySQL. The mascot for MySQL is the dolphin seen in their logo. Her name is Sakila, and she was given her name from a Name the Dolphin contest held by MySQL.

P—PHP Scripting Language and Engine

PHP is a recursive acronym that stands for *PHP: Hypertext Preprocessor*. This widely used general-purpose scripting language is especially suited for Web development and can be embedded into HTML. What this means to you is that it's a simple scripting language that can greatly enhance your website. You simply learn the code, apply the logic, and create a dynamic website that can interact with your users on many levels greater than the traditional "flat file" HTML methods of the Internet.

PHP's initial inception in 1995 was a simple set of Perl scripts for tracking Rasmus Lerdorf's online résumé. As time went on, Lerdorf began to write a much larger C implementation to handle the increased amount of functionality he needed, including database connectivity. Lerdorf then decided to send out an initial release, open source style, called PHP/FI for anyone to use and to improve upon. Back in the day, this stood for Personal Home Page/Forms Interpreter. By 1997, the second release was distributed (PHP/FI 2.0) and had started to gain a following of several thousand from around the globe. Although several individuals were contributing code, it was still Lerdorf who continued the majority of all development.

Mid-1997 saw the dawn of a new age of PHP: PHP 3. This version was a complete rewrite of PHP/FI 2.0 by Andi Gutmans and Zeev Suraski, who needed more juice than was previously available for a university project they had been working on. To capitalize on PHP's growing user base, Lerdorf, Gutmans, and Suraski decided to release this new creation under the PHP name and so started what we know today as PHP. In the winter of 1998, PHP 4 development was begun by Gutmans and Suraski. They released the first official version in May of 2000. PHP 4 boasted much higher performance and pushed new technologies to its ever growing fan base with HTTP sessions, output buffering, and more secure ways of handling user input.

We believe that PHP 5 will knock the butterfly off its flower when it hits. A new object-oriented model coupled with the Zend Engine 2, stack tracing, and exception handling is expected to push a wider acceptance across the planet. At the same time, an introduction of integration with external object-oriented models, such as COM and Java, will throw a wildcard into the mix. For the first time, the ability of other communities to integrate seamlessly with PHP will be available. This means that prewritten APIs will be able to be much more easily integrated with PHP, destroying the last remaining reasons for these other communities to *not* use PHP.

NOTE Don't worry if you do not yet have a complete understanding of the LAMP technologies from these brief descriptions. Throughout the rest of the book, we will cover in depth how to install, configure, and utilize them.

Why LAMP?

If you are wondering, "Why choose LAMP?" then the answer is something you must consider for yourself. The combination has been proven on many popular websites, and the technology is free to use. These factors are why many people and businesses have chosen this combination for launching their production-based websites.

With LAMP, you have full control over your server. Most important, you have remote access, which enables you to easily administer your Linux server from anywhere in the world. Linux enables you to run the services required without running a GUI, and therefore it uses less system resources—resources that could be used to speed up the process of delivering web pages to your audience.

Why LAMP Is Free

The core of the Linux operating system (the kernel) is under the GNU General Public License (GPL). The reason Linux is licensed under the GPL is simple: you are authorized to make modifications of the software, and in turn, release your version to the public, as long as you release the source code along with it. This keeps everything under the GPL "open source" and allows other people to modify the work that you have done, and so on.

Although it is not necessary to license any open source software under GPL, the GPL does prevent the code from becoming closed to the public. Others cannot capitalize on an open source code foundation by improving upon it and then close the source, thereby making it difficult if not impossible for the open source community to develop the same functionality. This is a "remember your roots" type license and it is how great operating systems such as Linux and most of the open source software evolve. This type of peer review is the basis for what's called the *Open Source Movement*.

The GPL is written and maintained by the Free Software Foundation. If you are interested in learning more about GPL, you can visit the Free Software Foundation's website at `www.gnu.org/copyleft/gpl.html`.

Apache, MySQL, and PHP are some of the other free software applications we will be using throughout this book. Although they are not licensed under the GPL, each of these applications is under a similar license. This enables us to not only share in the work of countless numbers of developers across the globe but also to freely use and distribute these applications as we choose. From our basic Linux installation all the way to our mail server (the application responsible for sending and receiving our e-mail on our server), we will be using free software brought to us by those who love to do one thing: write software.

NOTE Some fees might be associated with free software—for example, for tech support—but these fees can be avoided by making use of the open source community itself. There are hundreds of forums and thousands of sites devoted to helping you find answers to questions that might arise, and these sites can be accessed for free. Although these sites might not answer your questions *immediately,* they should become an invaluable resource for your newfound, or newly rekindled, love of open source.

It's a safe bet that on any given day you use at least one piece of free software, even if you don't know it. In fact, this book was written on free word processing applications that come with Linux. Now, you have the opportunity to learn more about free software that you will use by following the examples we set for you in this book to create your own LAMP solution.

LAMP versus Other Solutions

Other solutions such as Microsoft's Internet Information Server (IIS) are popular. However, the lack of security and potential higher cost of hardware and maintenance keep them out of reach of many small companies. Rather than spend millions in licenses and administrative costs for an inferior product, we choose to run a Free Software solution.

In a recent study by the Robert Frances Group of Westport, Connecticut, it was found that the cost of running Linux was roughly 40 percent of that of Microsoft Windows and 14 percent of Sun Microsystems' Solaris operating systems. This alone will make any company think twice about deploying a Web-based application on Microsoft Windows. In that same study, it was also found that the typical Windows administrator managed an average of only 10 machines each, whereas Linux administrators managed many times that number.

Speed is another significant factor to most of us. We need our Web-based applications to respond quickly and remain snappy throughout the user's experience. With proper coding techniques, PHP is many times faster than Microsoft's ASP or Sun Microsystems' Java platform.

Another great feature of PHP is the ability to create command-line scripting that can perform tasks to your system at scheduled intervals by using CRON jobs within Linux. You can have PHP clean up your MySQL database weekly, or have it create backups of your website every few days, and so on. Although many PHP versus ASP debates could be spawned from these statements, the truth stands that ASP probably will never be as robust as PHP.

In the end, it's you that makes LAMP better than other platforms. Through the use of the LAMP combination, you directly contribute to the success of the software on your server.

Who Else Is Using LAMP?

The question of "Who is using LAMP?" is often asked. Because of the size of the Internet and the number of servers out there, we do not have the ability to give you a full report; however, here's a list of some of the more popular sites using LAMP:

LinuxForum.com A free Linux support website.

ApacheFreaks.com A free Apache web server support website.

MySQLFreaks.com A free MySQL database manager support website.

PHPFreaks.com A free PHP support website.

Winamp.com The Nullsoft Winamp site is built around PHP.

DevShed.com The open source development site Developer Shed is completely PHP driven.

MP3.com The single largest MP3 Web-based server is built using PHP.

SpeedTV.com SPEED TV, the television station owned by Fox, has been completely developed in PHP with a few Flash/ColdFusion elements thrown in.

Yahoo.com Even though Yahoo does not show the `.php` extension on their files, it has been confirmed by many sources that they are using it. Here's our source: `public.yahoo.com/~radwin/talks/yahoo-phpcon2002.htm`.

Nearly 70 percent of the websites you visit are using Linux as their operating system, and MySQL AB and Zend Technologies report that "over 10 million Web-based applications have been built using MySQL and PHP through low-cost open source software stacks such as LAMP" (see `www.mysql.com/press/release_2004_05.html`). This number continues to grow as Linux and other open source projects gain momentum within the industry. Every passing year adds more credibility, more features, and more support to the LAMP solution.

If you're curious to see how many domains are using PHP, you can check the Usage Stats for PHP at `www.php.net/usage.php`. At the time of this writing, according to SecuritySpace (`www.securityspace.com/s_survey/data/index.html`), the current usage is about 4.6 million domain names that are PHP enabled.