Apache Security

Ivan Ristić
To my dear wife Jelena,
who makes my life worth living.
# Table of Contents

Preface to Digital Reprint ........................................................................................................... xi

Preface ........................................................................................................................................ xiii
  Audience ................................................................................................................................... xiii
  Scope ....................................................................................................................................... xiv
  Contents of This Book ................................................................................................................ xv
  Online Companion ..................................................................................................................... xvii
  Conventions Used in This Book ................................................................................................. xvii
    Programming Conventions ........................................................................................................ xviii
    Typesetting Conventions ........................................................................................................... xviii
  Using Code Examples .................................................................................................................. xviii
  Acknowledgments ....................................................................................................................... xix

1. Apache Security Principles ....................................................................................................... 1
  1.1. Security Definition ................................................................................................................ 1
    1.1.1. Essential Security Principles ......................................................................................... 3
    1.1.2. Common Security Vocabulary ....................................................................................... 4
    1.1.3. Security Process Steps ................................................................................................... 4
    1.1.4. Threat Modeling ............................................................................................................ 5
    1.1.5. System-Hardening Matrix .............................................................................................. 7
    1.1.6. Calculating Risk ............................................................................................................. 10
  1.2. Web Application Architecture Blueprints ............................................................................. 11
    1.2.1. User View ...................................................................................................................... 11
    1.2.2. Network View ............................................................................................................... 13
    1.2.3. Apache View ................................................................................................................. 13

2. Installation and Configuration ................................................................................................... 17
  2.1. Installation ............................................................................................................................ 18
    2.1.1. Source or Binary ............................................................................................................ 18
    2.1.2. Static Binary or Dynamic Modules ............................................................................... 21
    2.1.3. Folder Locations ............................................................................................................ 22
    2.1.4. Installation Instructions .................................................................................................. 23
2.2. Configuration and Hardening
  2.2.1. Setting Up the Server User Account 29
  2.2.2. Setting Apache Binary File Permissions 29
  2.2.3. Configuring Secure Defaults 30
  2.2.4. Enabling CGI Scripts 33
  2.2.5. Logging 34
  2.2.6. Setting Server Configuration Limits 35
  2.2.7. Preventing Information Leaks 37
2.3. Changing Web Server Identity
  2.3.1. Changing the Server Header Field 40
  2.3.2. Removing Default Content 42
2.4. Putting Apache in Jail
  2.4.1. Tools of the chroot Trade 45
  2.4.2. Using chroot to Put Apache in Jail 48
  2.4.3. Using the chroot(2) Patch 52
  2.4.4. Using mod_security or mod_chroot 53
3. PHP
  3.1. Installation 57
    3.1.1. Using PHP as a Module 57
    3.1.2. Using PHP as a CGI 59
    3.1.3. Choosing Modules 60
  3.2. Configuration 61
    3.2.1. Disabling Undesirable Options 62
    3.2.2. Disabling Functions and Classes 64
    3.2.3. Restricting Filesystem Access 65
    3.2.4. Setting Logging Options 66
    3.2.5. Setting Limits 66
    3.2.6. Controlling File Uploads 67
    3.2.7. Increasing Session Security 68
    3.2.8. Setting Safe Mode Options 69
  3.3. Advanced PHP Hardening 72
    3.3.1. PHP 5 SAPI Input Hooks 72
    3.3.2. Hardened-PHP 73
4. SSL and TLS
  4.1. Cryptography 76
    4.1.1. Symmetric Encryption 77
    4.1.2. Asymmetric Encryption 79
    4.1.3. One-Way Encryption 80
    4.1.4. Public-Key Infrastructure 81
4.1.5. How It All Falls into Place 84
4.2. SSL 85
  4.2.1. SSL Communication Summary 86
  4.2.2. Is SSL Secure? 86
4.3. OpenSSL 89
4.4. Apache and SSL 92
  4.4.1. Installing mod_ssl 92
  4.4.2. Generating Keys 93
  4.4.3. Generating a Certificate Signing Request 94
  4.4.4. Signing Your Own Certificate 95
  4.4.5. Getting a Certificate Signed by a CA 96
  4.4.6. Configuring SSL 96
4.5. Setting Up a Certificate Authority 99
  4.5.1. Preparing the CA Certificate for Distribution 102
  4.5.2. Issuing Server Certificates 102
  4.5.3. Issuing Client Certificates 104
  4.5.4. Revoking Certificates 105
  4.5.5. Using Client Certificates 105
4.6. Performance Considerations 106
  4.6.1. OpenSSL Benchmark Script 106
  4.6.2. Hardware Acceleration 108
5. Denial of Service Attacks ......................................................... 109
  5.1. Network Attacks 111
    5.1.1. Malformed Traffic 111
    5.1.2. Brute-Force Attacks 111
    5.1.3. SYN Flood Attacks 112
    5.1.4. Source Address Spoofing 113
    5.1.5. Distributed Denial of Service Attacks 114
    5.1.6. Reflection DoS Attacks 115
  5.2. Self-Inflicted Attacks 116
    5.2.1. Badly Configured Apache 116
    5.2.2. Poorly Designed Web Applications 118
    5.2.3. Real-Life Client Problems 119
  5.3. Traffic Spikes 120
    5.3.1. Content Compression 121
    5.3.2. Bandwidth Attacks 121
    5.3.3. Cyber-Activism 122
    5.3.4. The Slashdot Effect 122
  5.4. Attacks on Apache 123
5.4.1. Apache Vulnerabilities 123
5.4.2. Brute-Force Attacks 124
5.4.3. Programming Model Attacks 125

5.5. Local Attacks 125
5.5.1. Process Limits 127
5.5.2. Process Accounting 127
5.5.3. Kernel Auditing 128

5.6. Traffic-Shaping Modules 128
5.7. DoS Defense Strategy 129

6. Sharing Servers .......................................................... 131
6.1. Sharing Problems 131
   6.1.1. File Permission Problems 132
   6.1.2. Dynamic-Content Problems 134
   6.1.3. Sharing Resources 139
   6.1.4. Same Domain Name Problems 139
   6.1.5. Information Leaks on Execution Boundaries 141
6.2. Distributing Configuration Data 144
6.3. Securing Dynamic Requests 146
   6.3.1. Enabling Script Execution 146
   6.3.2. Setting CGI Script Limits 148
   6.3.3. Using suEXEC 148
   6.3.4. FastCGI 155
   6.3.5. Running PHP as a Module 157
6.4. Working with Large Numbers of Users 157
   6.4.1. Web Shells 158
   6.4.2. Dangerous Binaries 158

7. Access Control ............................................................................................................. 161
7.1. Overview 161
7.2. Authentication Methods 163
   7.2.1. Basic Authentication 163
   7.2.2. Digest Authentication 165
   7.2.3. Form-Based Authentication 166
7.3. Access Control in Apache 168
   7.3.1. Basic Authentication Using Plaintext Files 168
   7.3.2. Basic Authentication Using DBM Files 170
   7.3.3. Digest Authentication 171
   7.3.4. Certificate-Based Access Control 171
   7.3.5. Network Access Control 172
   7.3.6. Proxy Access Control 174
12.2.2. More Configuration Advice 368
12.2.3. Deployment Guidelines 371
12.2.4. Detecting Common Attacks 374
12.2.5. Advanced Topics 379

A. Tools .................................................................................................................. 385
  A.1. Learning Environments 385
    A.1.1. WebMaven 385
    A.1.2. WebGoat 386
  A.2. Information-Gathering Tools 388
    A.2.1. Online Tools at TechnicalInfo 388
    A.2.2. Netcraft 388
    A.2.3. Sam Spade 389
    A.2.4. SiteDigger 390
    A.2.5. SSLDigger 391
    A.2.6. Httprint 392
  A.3. Network-Level Tools 393
    A.3.1. Netcat 393
    A.3.2. Stunnel 394
    A.3.3. Curl 395
    A.3.4. Network-Sniffing Tools 396
    A.3.5. SSLDump 396
  A.4. Web Security Scanners 397
    A.4.1. Nikto 397
    A.4.2. Nessus 398
  A.5. Web Application Security Tools 399
    A.5.1. Paros 399
    A.5.2. Commercial Web Security Tools 400
  A.6. HTTP Programming Libraries 401

Index ....................................................................................................................... 403
Preface to Digital Reprint

*Apache Security* was originally published by O’Reilly in 2005, and it was very well received. Soon after publication, it was heralded as the best Apache security resource, according to many. Contrary to my expectations, it also aged very gracefully, which is probably why it continues to be popular. As much as I wanted to release an update, I struggled for years to justify a second edition. When I finally could, it turned out that O’Reilly was not too keen on the idea.

That was an opportunity for me to do things differently. As much as I enjoyed working on *Apache Security* a few years ago, the process was traditional and slow. It was a new digital age and we had all the advanced technology at our fingertips, yet we were still producing books the old-fashioned way. I wanted more. Above all, I wanted the ability to update my books whenever I felt the need. Unable to find a publisher that supported the process I wanted, I started my own publishing company. Feisty Duck, as my wife and I named it, is a publisher of computer books, with special focus on continuous publishing and digital delivery.

We are now releasing what is pretty much the original *Apache Security*, in digital format only, in order to establish a starting point for the second edition, which will be published by Feisty Duck at some point in the future. The known errors in the book have been fixed. If further errors are discovered, they will be fixed, too, and the digital version will be updated.

You may wonder whether the first edition of *Apache Security* is still worth paying for. After all, it has been five years since the first edition. Here’s what I think:

- The only part of the book that is completely obsolete is the ModSecurity chapter. I have only myself to blame for that, because I completely rewrote ModSecurity itself in 2006. If ModSecurity is what you’re after, you should look at my other book, *Mod-Security Handbook* (Feisty Duck, 2010). You will find more information about it at https://www.feistyduck.com.

- Chapter 10, “Web Application Security,” was the best introduction to the topic at the time of the original publication. It remains a good introduction, but there have been many advances and discoveries since it was written. These days, you actually have to
read several books to get decent coverage of web application security, and complete coverage is virtually impossible.

- The same can be said for Chapter 11, “Web Security Assessment”: it’s still good, but it’s just not enough any more.

- The rest of the book remains pretty solid. Five years later, some aspects are not entirely accurate, but what is in the book is still very useful. You will find that the general principles of web server security haven’t changed at all.

To conclude, Apache Security is still a good book, although it will no longer serve all audiences equally well. To paraphrase a recent Amazon.com reviewer, if you are at the beginner or intermediate levels, it will work for you. If you are an advanced user, it may not. If you are not sure, the best thing to do is decide by looking at the table of contents.
Preface

There is something about books that makes them one of the most precious things in the world. I’ve always admired people who write them, and I have always wanted to write one myself. The book you are now holding is a result of many years of work with the referenced Internet technologies and almost a year of hard work putting the words on paper. The preface may be the first thing you are reading, but it is the last thing I am writing. And I can tell you it has been quite a ride.

Aside from my great wish to be a writer in the first place, which only helped me in my effort to make the book as good as possible, there is a valid reason for its existence: a book of this profile is greatly needed by all those who are involved with web security. I, and many of the people I know, need it. I’ve come to depend on it in my day-to-day work, even though at the time of this writing it is not yet published. The reason this book is needed is that web security is affected by some diverse factors, which interact with each other in web systems and affect their security in varied, often subtle ways. Ultimately, what I tried to do was create one book to contain all the information one needs to secure an Apache-based system. My goal was to write a book I could safely recommend to anyone who is about to deploy on Apache, so I would be confident they would succeed provided they followed the advice in the book. You have, in your hands, the result of that effort.

Audience

This book aims to be a comprehensive Apache security resource. As such, it contains a lot of content on the intermediate and advanced levels. If you have previous experience with Apache, I expect you will have no trouble jumping to any part of the book straight away. If you are completely new to Apache, you will probably need to spend a little time learning the basics first, perhaps reading an Apache administration book or taking one of the many tutorials available online. Since Apache Security covers many diverse topics, it’s likely that no matter what level of experience you have you are likely to have a solid starting point.
This book does not assume previous knowledge of security. Security concepts relevant for discussion are introduced and described wherever necessary. This is especially true for web application security, which has its own chapter.

The main thing you should need to do your job in addition to this book, is the Apache web server’s excellent reference documentation (http://httpd.apache.org/docs/).

The book should be especially useful for the following groups:

**System administrators**
Their job is to make web systems secure. This book presents detailed guidance that enables system administrators to make informed decisions about which measures to take to enhance security.

**Programmers**
They need to understand how the environment in which their applications are deployed works. In addition, this book shows how certain programming errors lead to vulnerabilities and tells what to do to avoid such problems.

**System architects**
They need to know what system administrators and programmers do, and also need to understand how system design decisions affect overall security.

**Web security professionals**
They need to understand how the Apache platform works in order to assess the security of systems deployed on it.

**Scope**

At the time of this writing, two major Apache branches are widely used. The Apache 1.x branch is the well-known, and well-tested, web server that led Apache to dominate the web server market. The 2.0.x branch is the next-generation web server, but one that has suffered from the success of the previous branch. Apache 1 is so good that many of its users do not intend to upgrade in the near future. A third branch, 2.2.x will eventually become publicly available. Although no one can officially retire an older version, the new 2.2.x branch is a likely candidate for a version to replace Apache 1.3.x. The Apache branches have few configuration differences. If you are not a programmer (meaning you do not develop modules to extend Apache), a change from an older branch to a newer branch should be straightforward.

This book covers both current Apache branches. Wherever there are differences in the configuration for the two branches, such differences are explained. The 2.2.x branch is configured in practically the same way as the 2.0.x branch, so when the new branch goes officially public, the book will apply to it equally well.
Many web security issues are directly related to the operating system Apache runs on. For most of this book, your operating system is irrelevant. The advice I give applies no matter whether you are running some Unix flavor, Windows, or some other operating system. However, in most cases I will assume you are running Apache on a Unix platform. Though Apache runs well on Windows, Unix platforms offer another layer of configuration options and security features that make them a better choice for security-conscious deployments. Where examples related to the operating system are given, they are typically shown for Linux. But such examples are in general very easy to translate to other Unix platforms and, if you are running a different Unix platform, I trust you will have no problems with translation.

## Contents of This Book

While doing research for the book, I discovered there are two types of people: those who read books from cover to cover and those who only read those parts that are of immediate interest. The book’s structure (12 chapters and 1 appendix) aims to satisfy both camps. When read sequentially, the book examines how a secure system is built from the ground up, adding layer upon layer of security. However, since every chapter was written to cover a single topic in its entirety, you can read a few selected chapters and leave the rest for later. Make sure to read the first chapter, though, as it establishes the foundation for everything else.

Chapter 1, presents essential security principles, security terms, and a view of security as a continuous process. It goes on to discuss threat modeling, a technique used to analyze potential threats and establish defenses. The chapter ends with a discussion of three ways of looking at a web system (the user view, the network view, and the Apache view), each designed to emphasize a different security aspect. This chapter is dedicated to the strategy of deploying a system that is created to be secure and that is kept secure throughout its lifetime.

Chapter 2, gives comprehensive and detailed coverage of the Apache installation and configuration process, where the main goal is not to get up and running as quickly as possible but to create a secure installation on the first try. Various hardening techniques are presented along with discussions of the advantages and disadvantages of each.

Chapter 3, discusses PHP installation and configuration, following the same style established in Chapter 2. It begins with a discussion of and installation guidance for common PHP deployment models (as an Apache module or as a CGI), continues with descriptions of security-relevant configuration options (such as the safe mode), and concludes with advanced hardening techniques.

Chapter 4, discusses cryptography on a level sufficient for the reader to make informed decisions about it. The chapter first establishes the reasons cryptography is needed, then introduces SSL and discusses its strengths and weaknesses. Practical applications of SSL for Apache are covered through descriptions and examples of the use of `mod_ssl` and OpenSSL.
This chapter also specifies the procedures for functioning as a certificate authority, which is required for high security installations.

Chapter 5, discusses some dangers of establishing a public presence on the Internet. A denial of service attack is, arguably, one of the worst problems you can experience. The problems discussed here include network attacks, configuration and programming issues that can make you harm your own system, local (internal) attacks, weaknesses of the Apache processing model, and traffic spikes. This chapter describes what can happen, and the actions you can take, before such attacks occur, to make your system more secure and reduce the potential effects of such attacks. It also gives guidance regarding what to do if such attacks still occur in spite of your efforts.

Chapter 6, discusses the problems that arise when common server resources must be shared with people you may not trust. Resource sharing usually leads to giving other people partial control of the web server. I present several ways to give partial control without giving too much. The practical problems this chapter aims to solve are shared hosting, working with developers, and hosting in environments with large numbers of system users (e.g., students).

Chapter 7, discusses the theory and practice of user identification, authentication (verifying a user is allowed to access the system), and authorization (verifying a user is allowed to access a particular resource). For Apache, this means coverage of HTTP-defined authentication protocols (Basic and Digest authentication), form-based and certificate-based authentication, and network-level access control. The last part of the chapter discusses single sign-on, where people can log in once and have access to several different resources.

Chapter 8, describes various ways Apache can be configured to extract interesting and relevant pieces of information, and record them for later analysis. Specialized logging modules, such as the ones that help detect problems that cause the server to crash, are also covered. The chapter then addresses log collection, centralization, and analysis. The end of the chapter covers operation monitoring, through log analysis in batch or real-time. A complete example of using mod_status and RRDtool to monitor Apache is presented.

Chapter 9, discusses a variety of security issues related to the environment in which the Apache web server exists. This chapter touches upon network security issues and gives references to web sites and books in which the subject is covered in greater detail. I also describe how the introduction of a reverse proxy concept into network design can serve to enhance system security. Advanced (scalable) web architectures, often needed to securely deploy high-traffic systems, are also discussed here.

Chapter 10, explains why creating safe web applications is difficult, and where mistakes are likely to happen. It gives guidance as to how these problems can be solved. Understanding the issues surrounding web application security is essential to establish an effective defense.
Chapter 11, establishes a set of security assessment procedures. Black-box testing is presented for assessment from the outside. White-box and gray-box testing procedures are described for assessment from the inside.

Chapter 12, builds on the material presented in previous chapters to introduce the concept of web intrusion detection. While the first part of this chapter discusses theory, the second part describes how Apache and mod_security can be used to establish a fully functional open source web intrusion detection system.

The Appendix, Tools, describes some of the more useful web security tools that save time when time is at a premium.

**Online Companion**

A book about technology cannot be complete without a companion web site. To fully appreciate this book, you need to visit [http://www.apachesecurity.net](http://www.apachesecurity.net), where I am making the relevant material available in electronic form. Some of the material available is:

- Configuration data examples, which you can copy and paste to use directly in your configuration.
- The tools I wrote for the book, together with documentation and usage examples. Request new features, and I will add them whenever possible.
- The links to all resources mentioned in the book, grouped according to their appearance in chapters. This will help you avoid retyping long links. I intend to maintain the links in working order and to provide copies of resources, should they become unavailable elsewhere.

I hope to expand the companion web site into a useful Apache security resource with a life on its own. Please help by sending your comments and your questions to the email address shown on the web site. I look forward to receiving feedback and shaping the future book releases according to other people’s experiences.

**Conventions Used in This Book**

Throughout this book certain stylistic conventions are followed. Once you are accustomed to them, you will distinguish between comments, commands you need to type, values you need to supply, and so forth.

In some cases, the typeface of the terms in the main text and in code examples will be different. The details of what the different styles (italic, boldface, etc.) mean are described in the following sections.
Programming Conventions

In command prompts shown for Unix systems, prompts that begin with # indicate that you need to be logged in as the superuser (root username); if the prompt begins with $, then the command can be typed by any user.

Typesetting Conventions

The following typographical conventions are used in this book:

- *Italic* indicates new terms, URLs, email addresses, filenames, file extensions, pathnames, directories, usernames, group names, module names, CGI script names, programs, and Unix utilities.
- *Constant width* indicates commands, options, switches, variables, functions, methods, HTML tags, HTTP headers, status codes, MIME content types, directives in configuration files, the contents of files, code within body text, and the output from commands.
- *Constant width bold* shows commands or other text that should be typed literally by the user.
- *Constant width italic* shows text that should be replaced with user-supplied values.

Note

This icon signifies a tip, suggestion, or general note.

Warning

This icon indicates a warning or caution.

Using Code Examples

This book is here to help you get your job done. In general, you may use the code in this book in your programs and documentation. You do not need to contact us for permission unless you’re reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing a CD-ROM of examples from our books does require permission. Answering a question by citing this book and quoting example code does not require permission. Incorporating a significant amount of example code from this book into your product’s documentation does require permission.

If you feel your use of code examples falls outside fair use or the permission given above, feel free to contact us at <contact@feistyduck.com>.
Acknowledgments

This book would not exist, be complete, or be nearly as good if it were not for the work and help of many people. My biggest thanks go to the people believing in the open source philosophy, the Apache developers, and the network and application security communities. It is a privilege to be able to work with you. A book like this cannot exist in isolation. Others have made it possible to write this book by allowing me to stand on their shoulders. Much of their work is referenced throughout the book, but it is impossible to mention it all.

Some people have had a more direct impact on my work. I thank Nathan Torkington and Tatiana Diaz for signing me up with O’Reilly and giving me the opportunity to have my book published by a publisher I respect. My special thanks and gratitude go to my editor, Mary Dageforde, who showed great patience working with me on my drafts. I doubt the book would be nearly as useful, interesting, or accurate without her. My reviewers, Rich Bowen, Dr. Anton Chuvakin, and Sebastian Wolfgarten were there for me to give words of encouragement, very helpful reviews, and a helping hand when it was needed.

I would like to thank Robert Auger, Ryan C. Barnett, Mark Curphey, Jeremiah Grossman, Anders Henke, and Peter Sommerlad for being great people to talk to and work with. My special thanks goes to the merry members of #port80, who were my first contact with the web security community and with whom I’ve had great fun talking to.

My eternal gratitude goes to my wife Jelena, for inspiring me to lead a better life, and encouraging me to do more and go further. She deserves great credit for putting up with me in the months I did nothing else but work on the book. Finally, I’d like to thank my parents and my family, for bringing me up the way they have, to always seek more but to be at peace with myself over where I am.