Securing PHP Apps

Ben Edmunds

Apress®

Securing PHP Apps

Ben Edmunds
Brooklyn, New York, USA

ISBN-13 (pbk): 978-1-4842-2119-8 ISBN-13 (electronic): 978-1-4842-2120-4
DOI 10.1007/978-1-4842-2120-4

Library of Congress Control Number: 2016948186

Managing Director: Welmoed Spahr
Lead Editor: Steve Anglin
Technical Reviewer: Massimo Nardone
Editorial Board: Steve Anglin, Pramila Balan, Laura Berendson, Aaron Black, Louise Corrigan, Jonathan Gennick, Robert Hutchinson, Celestin Suresh John, Nikhil Karkal, James Markham, Susan McDermott, Matthew Moodie, Natalie Pao, Gwenan Spearing
Coordinating Editor: Mark Powers
Copy Editor: Mary Bearden
Compositor: SPi Global
Indexer: SPi Global
Artist: SPi Global

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail rights@apress.com, or visit www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at www.apress.com/bulk-sales.

Any source code or other supplementary materials referenced by the author in this text are available to readers at www.apress.com/9781484221198. For detailed information about how to locate your book's source code, go to www.apress.com/source-code/. Readers can also access source code at SpringerLink in the Supplementary Material section for each chapter.

Printed on acid-free paper

Contents at a Glance

Contents

About the Author

Ben Edmunds [1] leads development teams to create cutting-edge web and mobile applications. He is an active leader, developer, and speaker in various development communities. He has been developing software professionally for over 10 years and in that time has worked on everything from robotics to government projects.

Ben is the CTO at Mindfulware, PHP Town Hall podcast co-host, CodeIgniter Framework Security Counsel member, open source advocate, human.

Ben offers security auditing and consulting on a limited basis each year. If you're interested, please get in touch. He can be reached via e-mail at `consulting@benedmunds.com`.

[1] `http://benedmunds.com`

About the Technical Reviewer

Massimo Nardone holds a master's of science degree in computing science from the University of Salerno, Italy. He has worked as a project manager, software engineer, research engineer, chief security architect, information security manager, PCI/SCADA auditor, and senior lead IT security/cloud/SCADA architect for many years. He currently works in the Chief Information Security Office (CISO) for Cargotec Oyj. He has more than 22 years of work experience in IT including security, SCADA, cloud computing, IT infrastructure, mobile, security, and WWW technology areas for both national and international projects. He worked as a visiting lecturer and supervisor for exercises at the Networking Laboratory of the Helsinki University of Technology (Aalto University). He has been programming and teaching how to program with Android, Perl, PHP, Java, VB, Python, C/C++, and MySQL for more than 20 years. He holds four international patents (PKI, SIP, SAML, and Proxy areas).

He is the co-author of *Pro Android Games* (Apress, 2015).

Constructor

Several years ago I was writing a web application for a client in the CodeIgniter PHP framework, shudder, but CodeIgniter didn't include any type of authentication system built in. I, of course, did what any good/lazy developer would do and went on the hunt for a well-made library to supply authentication capabilities. To my chagrin, I discovered that there weren't any clean, concise libraries that fit my needs for authentication in CodeIgniter. Thus began my journey of creating Ion Auth, a simple authentication library for CodeIgniter, and a career-long crusade for securing web applications as well as helping other developers do the same.

Here we are years later, a lot of us have moved on to other frameworks or languages, but I still repeatedly see basic security being overlooked. So let's fix that. I want to make sure that you'll never have to live the horror of leaking user passwords, have someone inject malicious SQL into your database, or experience the suite of other "hacks" that could have been easily avoided. Let's make sure we all get home on time and sleep well at night.

The intended audience for this book is someone who knows PHP and has developed for the web before. A large breadth of knowledge is not needed, however, and this will be applicable for a junior developer through to a senior developer. This will be a framework-agnostic guide to help you learn the basics of securing web applications built into PHP and learning about the common security pitfalls that a senior developer usually acquires over years of experience. This book will be a quick read with handbook-style references to specific items you can act on. It is meant to be something you can read in a couple hours and then reference later as needed. I'll also try to make sure we have some fun in the process.

Format

All code samples in the indented blocks can be assumed to be in PHP unless otherwise noted. Line numbers are shown on coding blocks for reference.

Lines starting with a dollar sign

```
$ ls -al
```

are examples of using the command line as a normal user. Lines starting with a pound sign

```
# ls -al
```

are examples of using the command line as the root user. Server command-line examples will assume some type of *nix (centos, redhat, ubuntu, osx, etc.) operating system.

I'm trying to keep the code examples from wrapping where possible so method arguments will be on their own lines. This may seem odd but it is much easier to read than wrapped code with this book format.

Errata

If you find any errors don't hesitate to get in touch with me via e-mail.[2]

Sample Code

All of the examples are in PHP unless otherwise noted. I will use native PHP code where possible, even if it creates more boilerplate. If something requires too much work to succinctly explain in native PHP I will use the Laravel framework because it has an elegant syntax and should be easy to understand.

Some of the code examples are broken up for explanation. To view complete code examples you can reference the GitHub repository[3] or download the source code from the book's apress.com product page, located at www.apress.com/9781484221198.

Let's do this.

[2] feedback@buildsecurephpapps.com
[3] https://github.com/benedmunds/Building-Secure-PHP-Apps-Examples