Book Review: php|architect's Guide to PHP Security

Ilia Alshanetsky

ISBN: 0973862106, 197 pages, $32.99 US, $47.99 CA

php|architect nanobooks

Link to image file - http://www.amazon.com/gp/product/images/0973862106/104-2726526-6184739

Security is constantly being brought to our attention - at every turn it seems another flaw is discovered in this or that application or operating system. Because PHP is being ever more widely used, it was no surprise that security was a hot topic at the recent php|works conference (organized by php|architect). I missed hearing Ilia Alshanetsky, the author of this book, speak on security there was something equally attractive in the same time slot but I was impressed by his talks on web services and PHP performance.

As a core PHP developer responsible for a number of extensions, Mr. Alshanetsky has immediate credibility. Having Rasmus Lerdorf, PHP's originator, write the foreword to this book is certainly another vote of confidence. But implementing security in PHP requires more than just a good understanding of the language and this author's thorough knowledge of Apache, JavaScript, and using Linux from the command line make for a complete approach to the problem.

At this point we all know about the dangers of having “register globals'” turned on but don't be smug and skip any of the introductory material because you'll discover that there are lots of other simple ways to mess up. For instance, how many of you know when to use single vs. double quotation marks? And even if you do know, do you use them consistently?

The chapter on cross site scripting (XXS) is particularly appealing because you can immediately attempt some of the examples just to pop up a JavaScript alert, not to mount a real attack of course. Given the variety of attacks possible you'll also find that prevention is a little more difficult than simply stripping HTML tags.

The topic of SQL injection is given thorough coverage, including a discussion of the use of “magic quotes”, prepared statements, and the inherent dangers of allowing errors to go untrapped. With prepared statements you not only get improved security because input is automatically escaped but you reap the benefits of improved performance as well. The release of the MySQL improved extension means that this capability is now also available for the database most commonly used with PHP. This chapter also contains a number of useful suggestions for securing database authentication credentials using Apache configuration settings or placing sensitive information outside the web path. Recognizing that most sites aren't hosted on a dedicated server, the author also addresses issues associated with shared web hosting.

Properly implemented security is not simply a matter of locking down an application but doing so without adversely affecting performance. The author's easy familiarity with the internal workings of PHP, Apache, and Linux means he's particularly insightful on this front. For example, he steers us away from using the glob function to create a white list of file names and instead suggests a registry type technique. However, in some circumstances a performance hit may be inescapable.

Security may require compromises in other areas as well. Specifying a full path for include files is doubtless more secure but it is counter to a programmer's natural instinct, likewise with the use of hard coded values. Using $_SERVER['PHP_SELF'] is a great way to write portable code but there are security implications.

The author doesn't leave any stone unturned. He deals with the distinctly menacing sounding, daemonized attack scripts, the pros and cons of encryption, running PHP in safe mode and problems you may encounter when using open-source applications. I particularly liked the suggestion of setting up a sandbox and tracking suspicious activity using an SQLite database. This allows for easy analysis of data and keeps the sandbox separate from your main database.

The last chapter, “Securing your Applications” is a good summary of all the security issues and is very useful as a checklist for auditing existing applications. This chapter also includes some common sense suggestions. For example, it makes sense to turn on full error reporting when developing or reviewing an application but when doing so in a production environment make sure you log errors to file as we've already seen, displaying them to the user could itself be a breach of security.

An appendix of resources would have been a nice addition to this book but at $32.99 it is exceptionally well-priced and asking for more is perhaps a little greedy. O'Reilly's forthcoming title on this subject, “Essential PHP Security”, has approximately the same page count but is priced at $39.95.

All in all, this book gives a very thorough treatment of PHP security but I do have one major area of concern. By showing how to thwart attacks it indirectly shows how to make them. Perhaps this book should only be sold to a white list of trusted buyers so that “Dr. Evil” can never get his hands on a copy.

About the Author

Peter Lavin runs a Web Design/Development firm in Toronto, Canada. He has been published in a number of magazines and online sites, including UnixReview.com, php|architect and International PHP Magazine. He is a contributor to the recently published O'Reilly book, *PHP Hacks* and is also the author of *Object Oriented PHP*, published by No Starch Press.