

Securing the PHP Environment with PHPSecInfo

Ed Finkler
coj@cerias.purdue.edu

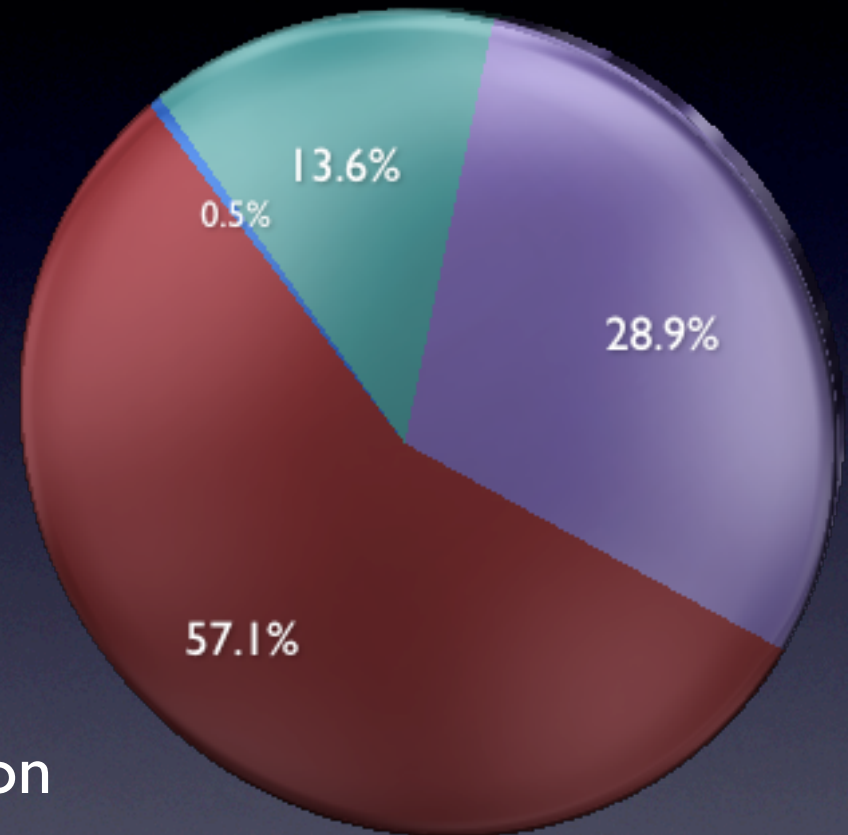
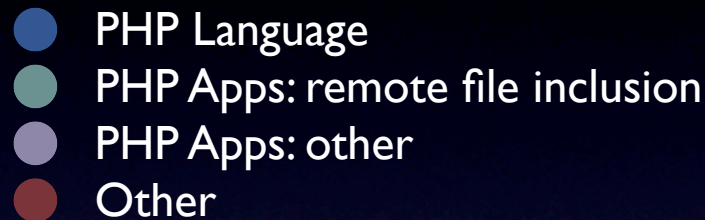
The ubiquity of PHP

- PHP is very, very popular
 - Nearly impossible to find a hosting service that doesn't support PHP in some form
 - About 34% of all domains report using PHP
- PHP is very easy to learn
- PHP provides results quickly
 - Time between setup and seeing results is very short

The ubiquity of PHP

- PHP powers many busy, high-profile sites
 - Wikipedia
 - Facebook
 - Wordpress.com
 - Digg
 - Flickr
 - Yahoo!

NIST NVD: 2006 data



- 6604 total entries
- 2803 PHP applications
- 895 PHP app remote file inclusion
 - Almost blocked by disabling `allow_url_fopen` (`allow_url_include` in 5.2+)

What does this tell us?

- How popular PHP is
- How much a target web apps are
- How many PHP developers are incapable of writing secure apps
- How many sysadmins don't secure their PHP environments

The parties involved

- **The System Administrator**
 - Directly responsible for PHP environment security
 - Tendency to lower security of environment to reduce application compatibility complaints



The parties involved



I can create web pages and install new computer equipment. Sometimes I even help my parents figure out why they can't get their email! Computers are fascinating to me and I love learning how to use them. If I believe in myself and follow my dream, someday I'll be writing computer programs. I have the power to be somebody!

EVERYBODY IS SOMEBODY

**I AM A
COMPUTER
PROGRAMMER**

RELATED CAREERS
Mathematics • Information • Computer Systems Analysis • Software Systems

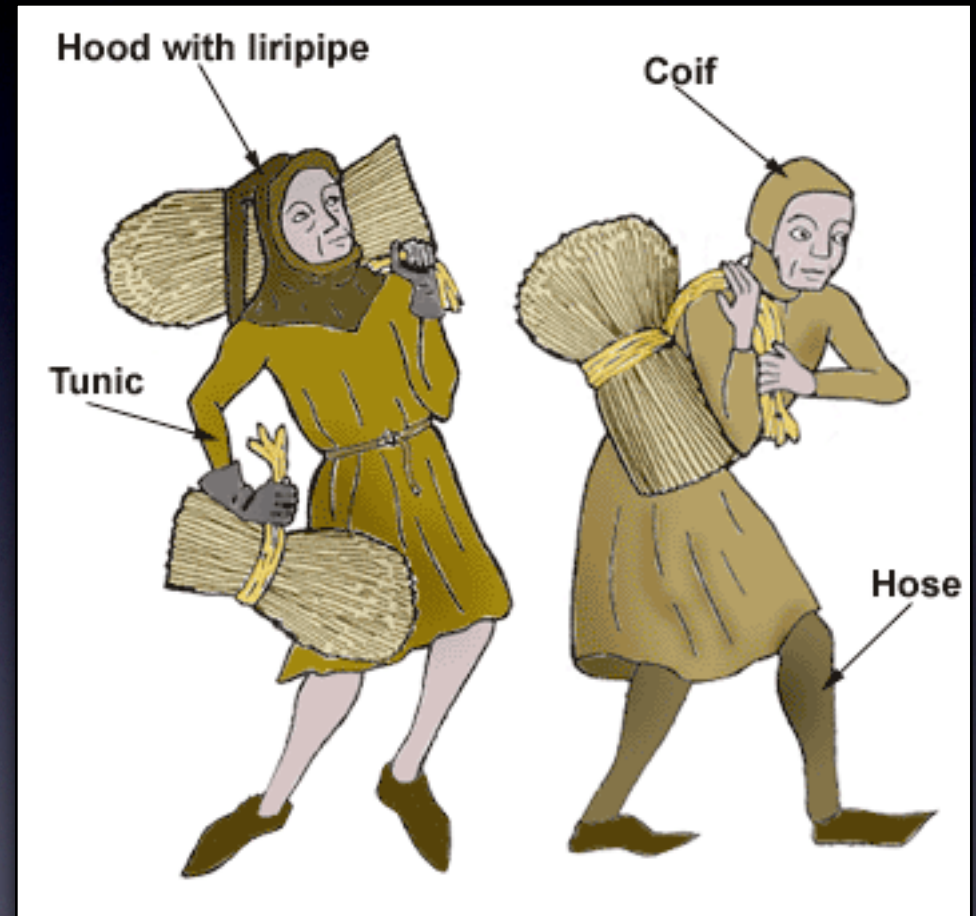
- **The PHP Developer**

- Must be aware of the environment and how it impacts app development
- Will write apps assuming certain features are enabled, despite security risks

The parties involved

- **The PHP “Deployer”**

- By far the largest portion of the audience
- Uses PHP apps on a web site, but not a coder
- Not capable of assessing security of an app
- At the mercy of the SysAdmin and Developer



Requirements of PHPSecInfo

- A security auditing tool accessible to the “Deployer”
 - Compatible
 - Support PHP4 (75%) and PHP5 (25%)
 - Easy to install
 - Unzip and Upload
 - Easy to execute (little or no config)
 - Runs upon upload; single function call

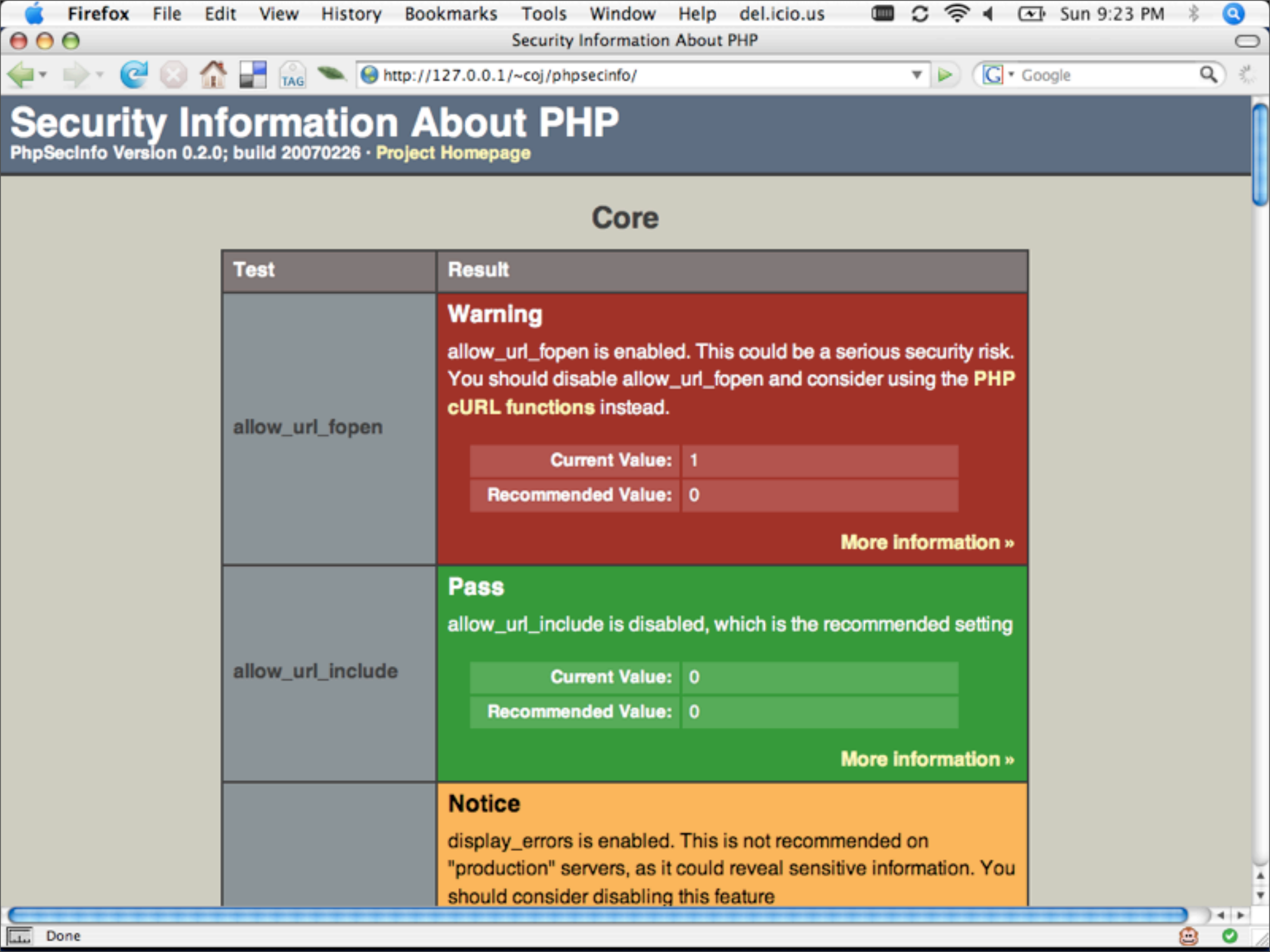
Requirements of PHPSecInfo

- Easy to understand
 - Clear, unambiguous results; color coding
- Encourage further exploration
 - Offer extended explanations with links to more info

Executing PHPSecInfo

1. Unzip
2. Upload
3. View in Browser

```
<?php require_once('PhpSecInfo/PhpSecInfo.php'); ?>  
<?php phpsecinfo(); ?>
```



Security Information About PHP

PhpSecInfo Version 0.2.0; build 20070226 · [Project Homepage](#)

Core

Test	Result				
allow_url_fopen	Warning allow_url_fopen is enabled. This could be a serious security risk. You should disable allow_url_fopen and consider using the PHP cURL functions instead. <table border="1"><tr><td>Current Value:</td><td>1</td></tr><tr><td>Recommended Value:</td><td>0</td></tr></table> <p>More information »</p>	Current Value:	1	Recommended Value:	0
Current Value:	1				
Recommended Value:	0				
allow_url_include	Pass allow_url_include is disabled, which is the recommended setting <table border="1"><tr><td>Current Value:</td><td>0</td></tr><tr><td>Recommended Value:</td><td>0</td></tr></table> <p>More information »</p>	Current Value:	0	Recommended Value:	0
Current Value:	0				
Recommended Value:	0				
	Notice display_errors is enabled. This is not recommended on "production" servers, as it could reveal sensitive information. You should consider disabling this feature				

Test Suite

- 17 tests for commonly exploited security vulnerabilities in PHP environment
- Each test result shows:
 - Current Setting
 - Recommended Setting
 - Result (color-coded)
 - Explanation
 - Link to further info
- Simple metrics output

use_trans_sid is disabled, which is the recommended setting

Current Value:	0
Recommended Value:	0

[More information »](#)

Tests Not Run

Test	Result
CGI::force_redirect	Not Run You don't seem to be using the CGI SAPI More information »

Test Results Summary

Test	Result
Notice	9 out of 17 (52.94%)
Pass	7 out of 17 (41.18%)
Warning	1 out of 17 (5.88%)



Consortium News

Fri, 20 Oct 2006
PHP Security Consortium
Launches New Project -
PHPSecInfo

Wed, 16 Nov 2005
Daniel Convisor Elected
as Principal

Promotional Links

Please support us by
providing a link to the PHP
Security Consortium on
your web site. You can
also use our promotional
image:



PhpSecInfo Test Information

allow_url_fopen

Test Description

This test checks to see if allow_url_fopen is enabled.

Security Implications

If enabled, allow_url_fopen allows PHP's file functions -- such as file_get_contents() and the include and require statements -- can retrieve data from remote locations, like an FTP or web site. Programmers frequently forget this and don't do proper input filtering when passing user-provided data to these functions, opening them up to code injection vulnerabilities. A large number of code injection vulnerabilities reported in PHP-based web applications are caused by the combination of enabling allow_url_fopen and bad input filtering.

allow_url_fopen is on by default.

Recommendations

You should disable allow_url_fopen in the php.ini file:

```
; Disable allow_url_fopen for security reasons
allow_url_fopen = 'off'
```

The setting can also be disabled in apache's httpd.conf file:

```
# Disable allow_url_fopen for security reasons
php flag allow url fopen off
```

Get PhpSecInfo

Download Now »

- Version: 0.2.0 20070226
md5 hash

Documentation

- View README
View CHANGELOG
View generated documentation

Mailing List »

Screenshots:



PHPSecInfo encourages accountability

Sorry, we can't support your app because it requires an insecure config!

Sysadmins

Our hosting is secure – PHPSecInfo says so!

Here's what's wrong with your PHP setup – fix it before you run our app!

Developers

Why doesn't your hosting service provide a secure PHP environment?

Deployers

Why does your application require an insecure configuration?

For advanced users

- Still a useful tool for evaluating PHP environments
 - Part of an auditing toolkit for web app security experts
- Extensible test framework
 - Create custom tests specific to an environment
 - Full generated documentation available



Zend_Environment Security Module

- Part of Zend Framework
- PHP5-only
- Zend_Environment offers programatic access to PHP environment information
- Z_E security module based on PHPSecInfo
 - Offers better (for now) programatic access to test results
 - More flexible output (HTML, Text, etc)
 - Part of a full-featured development framework



Zend Environment Info : 18th March, 2007

Security

Group	Name	Result	Current Value	Rec. Value	Details	More Info
cgi	force_redirect	notrun	0	1	You don't seem to be using the CGI SAPI	More Info >
core	allow_url_fopen	warning	1	0	allow_url_fopen is enabled. This could be a serious security risk. You should disable allow_url_fopen and consider using the PHP cURL functions instead.	More Info >
core	allow_url_include	ok	0	0	allow_url_include is disabled, which is the recommended setting	More Info >
core	display_errors	notice	1	0	display_errors is enabled. This is not recommended on "production" servers, as it could reveal sensitive information. You should consider disabling this feature	More Info >
core	expose_php	notice	1	0	expose_php is enabled. This adds the PHP "signature" to the web server header, including the PHP version number. This could attract attackers looking for vulnerable versions of PHP	More Info >
core	file_uploads	notice	1	0	file_uploads are enabled. If	More Info >

What the future may bring

- New view system & new output formats (xml, console, html themes, etc)
- Better IIS support
- Instantiate and obtain results programmatically for embedding in apps
 - Security testing during installation process, et al

More Information

phpsecinfo.com
phpsec.org
cerias.purdue.edu
framework.zend.com

Slides: funkatron.com

