

do_action('hack_me')





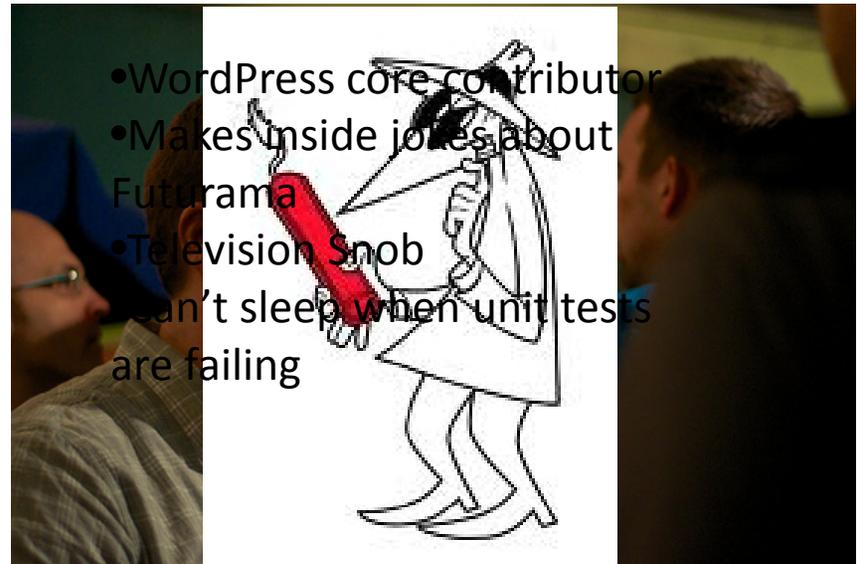
Who are we and why are we talking about this?

ABOUT US

Joshua Hansen



Kurt Payne





Reasons your site gets hacked

WHY ME?

Money

- Phishing Site
- Malware Downloads
- Warez/Piracy
- SEO
- Spam
- DDOS





What are they exploiting?

COMMON VULNERABILITIES

OWASP Top Ten

- Injection
- Cross-Site Scripting
- Broken Authentication and Session Management
- Insecure Direct Object References
- Cross-Site Request Forgery
- Security Misconfiguration
- Insecure Cryptographic Storage
- Failure to Restrict URL Access
- Insufficient Transport Layer Protection
- Unvalidated Redirects and Forwards



Explanation and Example

REMOTE FILE INCLUSION

DEMO



Prevention

REMOTE FILE INCLUSION

- Never trust user supplied input!
- Sanitize your data -
[http://codex.wordpress.org/Data Validation](http://codex.wordpress.org/Data_Validation)
- Verify length
- Verify type
- Test against regex
- Compare against whitelist
- Validate inputs



Explanation and Example

CROSS-SITE SCRIPTING (XSS)

DEMO



Prevention

CROSS-SITE SCRIPTING (XSS)

- Never trust user supplied input!
- Sanitize your data -
[http://codex.wordpress.org/Data Validation](http://codex.wordpress.org/Data_Validation)
- Escape based on context
- Use the methods provided by WordPress



Explanation and Example

SQL INJECTION

DEMO



Prevention

SQL INJECTION

- Never trust user supplied input!
- Sanitize your data -
[http://codex.wordpress.org/Data Validation](http://codex.wordpress.org/Data_Validation)
- Use Prepared Statements – `wpdb::prepare()`
- Escape user supplied input
- Least Privilege
- Whitelist input validation



Tools and Resources

WHAT CAN I DO?

W3AF



w3af

Web Application Attack and Audit Framework

RIPS

path / file: subdirs

verbosity level: vuln type:

code style:

RIPS 0.32

Scanned 30 lines in 1 files for 177 functions in 0.019 seconds.

File: e:/scantest/types.php

File Disclosure

```
7: fread $data = fread($fh, 1024);
4: $fh = fopen($evil, 'w+');
  • 3: $evil = $_GET['userinput'];
```

File Manipulation

```
10: fwrite fwrite($fh, $evil);
4: $fh = fopen($evil, 'w+');
  • 3: $evil = $_GET['userinput'];
  • 3: $evil = $_GET['userinput'];
```

SQL Injection

```
13: mysql_query $query = mysql_query("SELECT * FROM users WHERE id = $evil");
  • 3: $evil = $_GET['userinput'];
```

Command Execution

```
19: system system($evil);
  • 3: $evil = $_GET['userinput'];
```

CodeViewer - e:/scantest/types.php

```
1 <?php
2
3 $evil = $_GET['userinput'];
4 $fh = fopen($evil, 'w+');
5
6 // file disclosure
7 $data = fread($fh, 1024);
8
9 // file manipulation
10 fwrite($fh, $evil);
11
12 // SQL Injection
13 $query = mysql_query("SELECT * FROM users WHERE id = $evil");
```

Vulnerability Databases

The screenshot shows the OSVDB (Open Source Vulnerability Database) search interface. The search query is 'test' and the results are sorted by score. The table lists various vulnerabilities with their IDs, disclosure dates, CVE identifiers, and titles.

ID	Disc Date	CVE	Title
59056	2009-10-19		AjaxChat Component for Joomla! components/com_ajaxchat/tests/ajaxuser.php mosConfig_absolute_path Parameter Remote File Inclusion
59222	2009-10-14	2009-3612	Linux Kernel Netlink Subsystem net/sched/ds_aplc.tcf_fil_node Function Local Memory Disclosure
58293	2009-09-22	2009-3501	BBowertouse.BPSstudents.students.php.test Parameter SQL Injection
57821	2009-09-02		Linux Kernel tc_fil_tclass() Function Kernel Memory Disclosure
59070	2009-09-02	2009-3228	Linux Kernel tc_Subsystem net/sched/sch_aplc.tc_fil_tclass Function Local Memory Disclosure
57428	2009-08-27	2009-3002	Linux Kernel proto_ops.getname Function Arbitrary Kernel Memory Disclosure
58254	2009-08-21	2009-2741	IBM WebSphere Business Events Test Servlet wberuntimevar Application Unspecified Arbitrary Code Execution
57697	2009-08-19	2009-3043	Linux Kernel drivers/char/tty_klbc.c tty_klbc_hangup Function Local DoS
57737	2009-08-17	2009-2695	Linux Kernel Multiple mmio Operations Local Privilege Escalation
56992	2009-08-13	2009-2692	Linux Kernel Multiple Protocol proto_ops().Inbalisation.NULL Pointer Dereference Local Privilege Escalation
57133	2009-08-13	2009-2768	Linux Kernel Flat Subsystem fs/binfat_flat.c load_flat_shared_library Function Local DoS
57204	2009-08-13	2009-2852	WP-Syntax Plugin for Wordpress test/index.php.test_filter(wp_null) Array Parameter Arbitrary PHP Code Execution
56822	2009-08-03	2009-2767	Linux Kernel kernel/poix-timers.c init_poix_timers Function NULL Dereference Local DoS
57071	2009-08-01	2009-3424	MAXcms.modulmod.test.php.thCMS_root Parameter Remote File Inclusion
55807	2009-07-13	2009-1895	Linux Kernel PER_CLEAR_ON_SETID Mask Local Security Restriction Bypass
55758	2009-06-29	2009-2398	PHP-Sugar test/index.php.t Parameter Traversal Arbitrary File Access
55511	2009-06-26	2009-2392	Virtue Online Test Generator test.php.tid Parameter SQL Injection
55512	2009-06-26	2009-2391	Virtue Online Test Generator test.php.tid Parameter XSS
55770	2009-06-26	2009-2393	Virtue Online Test Generator admin/index.php Admin Authentication Bypass
54892	2009-06-02	2009-1395	Linux Kernel e1000 drivers/net/e1000/e1000_main.c e1000_clean_irq Function Underflow DoS
54554	2009-05-15		Linux Kernel KVM Guest Machine Exit Both Local DoS
54498	2009-05-14	2009-1633	Linux Kernel CFS String Conversion Multiple Local Overflows
54188	2009-05-04	2009-1527	Linux Kernel ptrace_attach() Function cred_exec_mutex Handling Local Privilege Escalation

The screenshot shows the Exploit Database website. The main heading is 'EXPLOIT DATABASE' with a logo of a flask. Below the heading is a banner that reads 'IS YOUR PORT 80 WIDE OPEN? HACKERS ARE LOOKING.' The website features a navigation menu, a search bar, and a section titled 'The Exploit Database' with a description of the site's purpose. There is also a 'Last Posts' section and a 'Remote Exploits' table.

Date	Description	Views	Rate	Author
2009-01-27	✓ Win7-SP1-MS09-005-Remote-Exec	1000	100%	0x0x0x0
2009-01-26	✓ FreeBSD 5.0-RELEASE-ARMED Vulnerability	989	100%	0x0x0x0
2009-01-24	✓ Linux 2.6.30-rc7-1.2-ARMED-ARMED Vulnerability	960	100%	0x0x0x0
2009-01-24	✓ SQL Server 2008-SP2-ARMED-ARMED Vulnerability	455	100%	0x0x0x0
2009-01-23	✓ MS-Exchange 2007-SP2-ARMED-ARMED Vulnerability	479	100%	0x0x0x0
2009-01-23	✓ Microsoft SQL Server 2008-SP2-ARMED-ARMED Vulnerability	462	100%	0x0x0x0
2009-01-23	✓ Microsoft Exchange 2007-SP2-ARMED-ARMED Vulnerability	408	100%	0x0x0x0

OWASP



Links

- W3AF - <http://w3af.sourceforge.net/>
- RIPS - <http://rips-scanner.sourceforge.net/>
- NVD - <http://nvd.nist.gov/>
- OSVDB - <http://osvdb.org/>
- Exploit-DB - <http://www.exploit-db.com/>
- OWASP - <https://www.owasp.org>
- Hardening WordPress - [http://codex.wordpress.org/Hardening WordPress](http://codex.wordpress.org/Hardening_WordPress)
- WordPress SQLi - [http://codex.wordpress.org/Class Reference/wpdb#Protect Queries Against SQL Injection Attacks](http://codex.wordpress.org/Class_Reference/wpdb#Protect_Queries_Against_SQL_Injection_Attacks)
- WordPress Data Validation – [http://codex.wordpress.org/Data Validation](http://codex.wordpress.org/Data_Validation)