OWASP Top Ten 2010

FINAL Release

Christian Heinrich
christian.heinrich@owasp.org
"Google Hacking" Project Leader

OWASP – Sydney Chapter
April 2010

Previously presented at:
- AISA Annual Seminar Day 2009 and;
- OWASP Melbourne Chapter - December 2009

The OWASP Foundation
http://www.owasp.org/

#whoami

OWASP Testing Guide v3

- 4.2.1 “Spiders/Robots/Crawlers”
- 4.2.2 “Search Engine Reconnaissance”

OWASP “Google Hacking” Project

- “Download Indexed Cache” PoC

Presented at

- .au, EU and USA OWASP Conferences
- London (.uk) and Melbourne (.au) OWASP Chapters

http://www.owasp.org/index.php/user:cmlh

OWASP Top Ten 2010

1. What is the OWASP Top Ten?
2. Additions from the OWASP Top Ten 2007
   - A.6 Security Misconfiguration
   - A.8 Unvalidated Redirects and Forwards
3. OWASP Top Ten Risk Rating Methodology
4. Timeline from Release Candidate (RC) to Final
5. When **Not** to Cite the OWASP Top Ten?
   - Application Security Verification Standard (ASVS)
6. Politics of the OWASP Top Ten

What is the OWASP “Top Ten”?

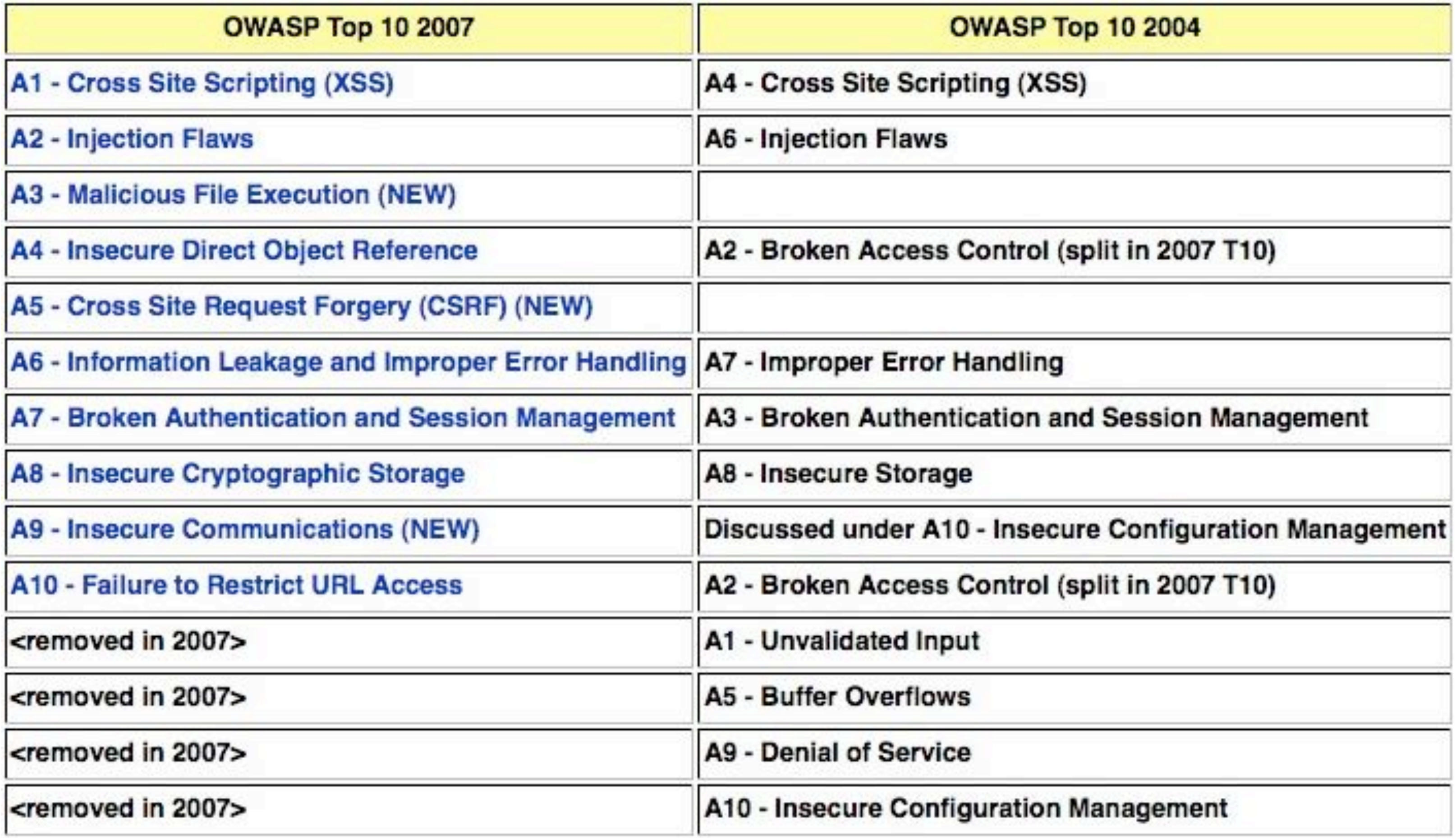
Ten most common WebAppSec **risks**:

- Based on the “OWASP Risk Rating Methodology.
- Intended Audience is Executive Level.
- Prior T10 Releases on **prevalence and severity**.

Statistics of vulnerabilities contributed by:

- Aspect Security
- MITRE
- White Hat

Differences between 2004 and 2007

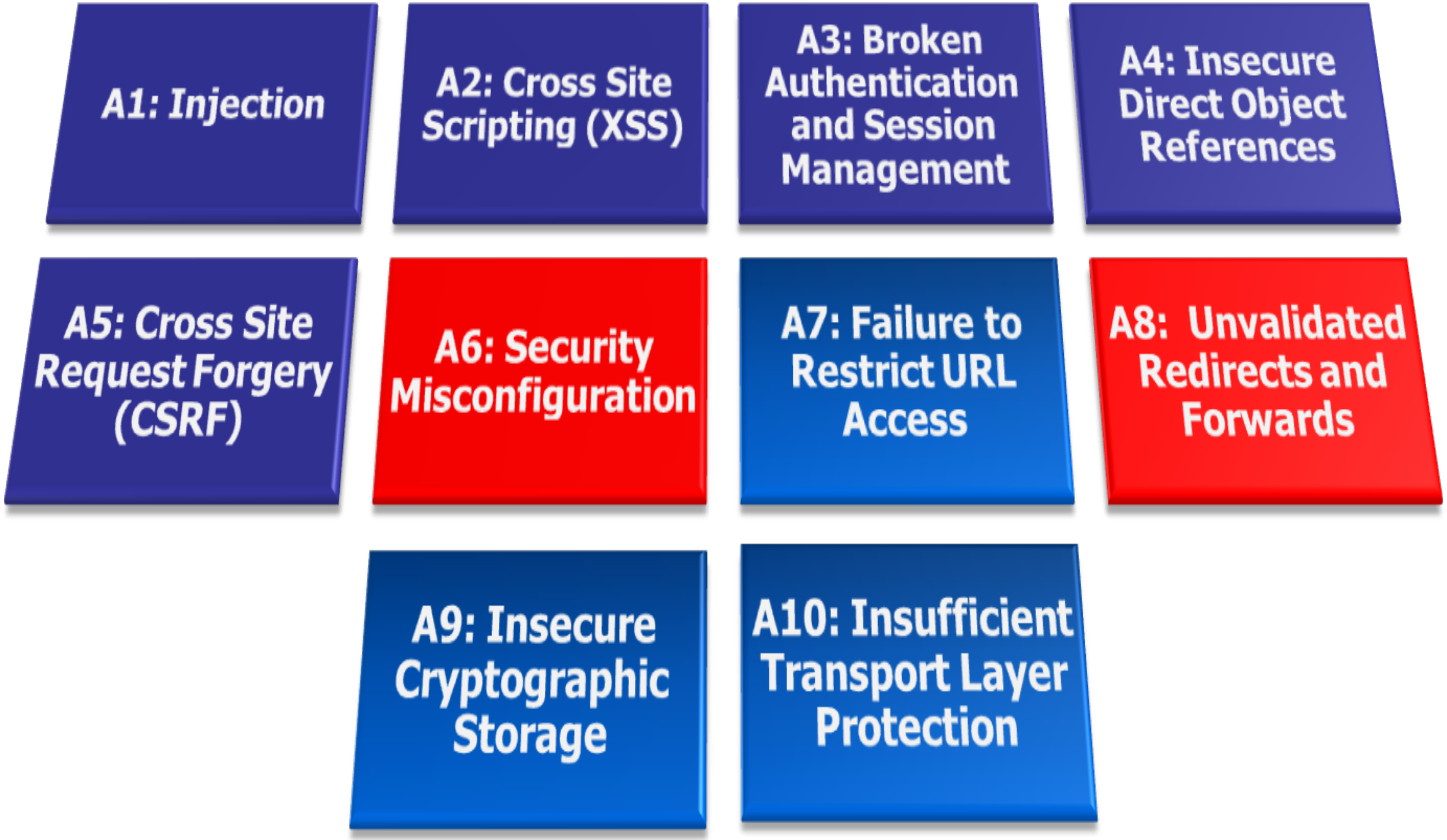


OWASP Top 10 2007	OWASP Top 10 2004
A1 - Cross Site Scripting (XSS)	A4 - Cross Site Scripting (XSS)
A2 - Injection Flaws	A6 - Injection Flaws
A3 - Malicious File Execution (NEW)	
A4 - Insecure Direct Object Reference	A2 - Broken Access Control (split in 2007 T10)
A5 - Cross Site Request Forgery (CSRF) (NEW)	
A6 - Information Leakage and Improper Error Handling	A7 - Improper Error Handling
A7 - Broken Authentication and Session Management	A3 - Broken Authentication and Session Management
A8 - Insecure Cryptographic Storage	A8 - Insecure Storage
A9 - Insecure Communications (NEW)	Discussed under A10 - Insecure Configuration Management
A10 - Failure to Restrict URL Access	A2 - Broken Access Control (split in 2007 T10)
<removed in 2007>	A1 - Unvalidated Input
<removed in 2007>	A5 - Buffer Overflows
<removed in 2007>	A9 - Denial of Service
<removed in 2007>	A10 - Insecure Configuration Management

Differences between 2007 and 2010

OWASP Top 10 – 2007 (Previous)		OWASP Top 10 – 2010 (New)
A2 – Injection Flaws	↑	A1 – Injection
A1 – Cross Site Scripting (XSS)	↓	A2 – Cross Site Scripting (XSS)
A7 – Broken Authentication and Session Management	↑	A3 – Broken Authentication and Session Management
A4 – Insecure Direct Object Reference	=	A4 – Insecure Direct Object References
A5 – Cross Site Request Forgery (CSRF)	=	A5 – Cross Site Request Forgery (CSRF)
<was T10 2004 A10 – Insecure Configuration Management>	+	A6 – Security Misconfiguration (NEW)
A10 – Failure to Restrict URL Access	↑	A7 – Failure to Restrict URL Access
<not in T10 2007>	+	A8 – Unvalidated Redirects and Forwards (NEW)
A8 – Insecure Cryptographic Storage	↓	A9 – Insecure Cryptographic Storage
A9 – Insecure Communications	↓	A10 – Insufficient Transport Layer Protection
A3 – Malicious File Execution	-	<dropped from T10 2010>
A6 – Information Leakage and Improper Error Handling	-	<dropped from T10 2010>

OWASP Top Ten 2010



A1: Injection

A2: Cross Site Scripting (XSS)

A3: Broken Authentication and Session Management

A4: Insecure Direct Object References

A5: Cross Site Request Forgery (CSRF)

A6: Security Misconfiguration

A7: Failure to Restrict URL Access

A8: Unvalidated Redirects and Forwards

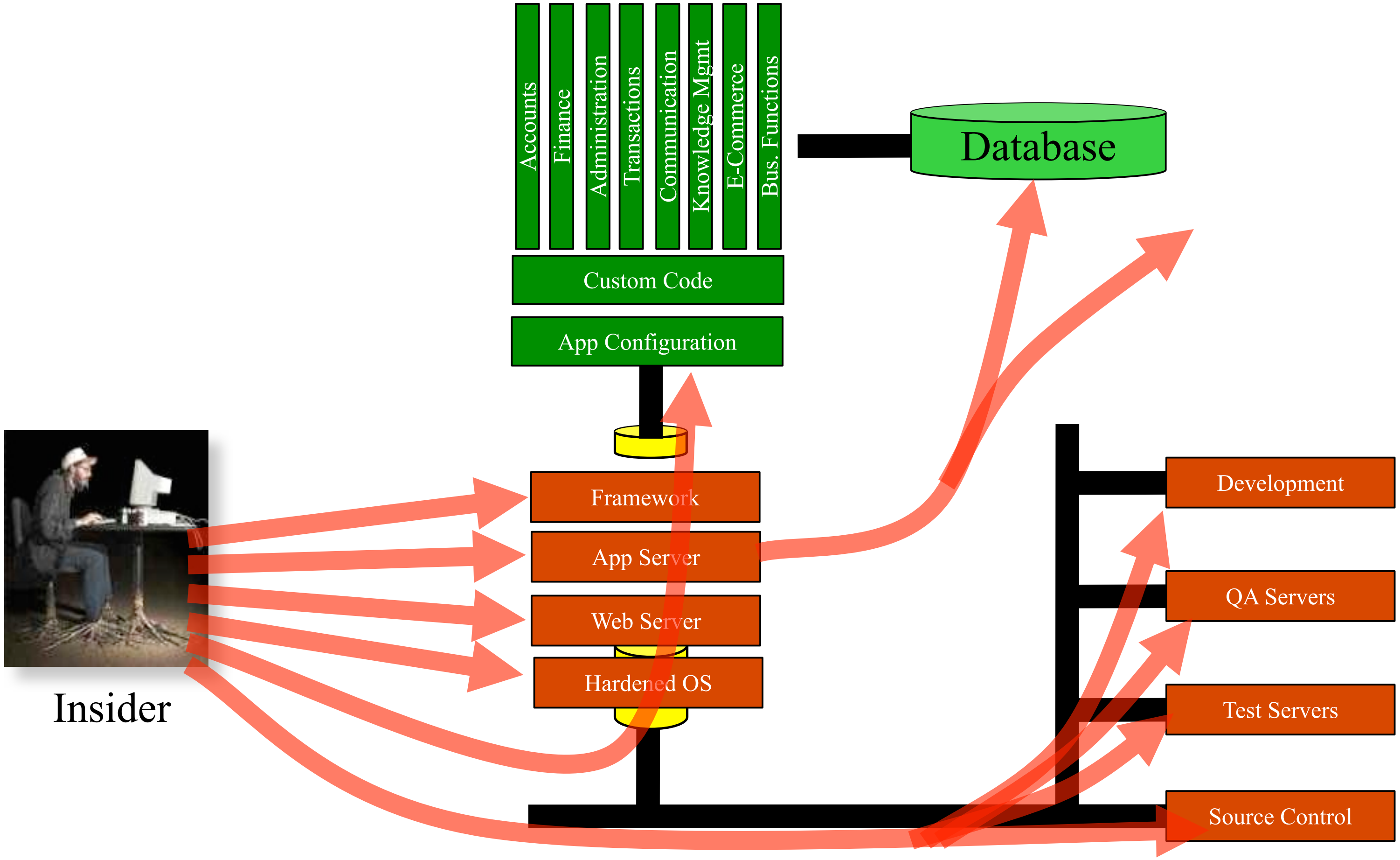
A9: Insecure Cryptographic Storage

A10: Insufficient Transport Layer Protection

Comparison of 2004, 2007 and 2010 Releases

OWASP Top Ten Entries (Unordered)	Releases		
	2004	2007	2010
Unvalidated Input	A1	×	×
Buffer Overflows	A5	×	×
Denial of Service	A9	×	×
Injection Flaws	A6	A2	A1[1]
Cross Site Scripting (XSS)	A4	A1	A2
Broken Authentication and Session Management	A3	A7	A3
Insecure Direct Object Reference	A2	A4[2]	A4
Cross Site Request Forgery (CSRF)	×	A5	A5
Security Misconfiguration	A10[3]	×	A6
Failure to Restrict URL Access	A2	A10[4]	A7
Unvalidated Redirects and Forwards	×	×	A8
Information Leakage and Improper Error Handling	A7[5]	A6	×
Malicious File Execution	×	A3	×
Insecure Cryptographic Storage	A8[6]	A8	A9
Insecure Communications	A10	A9[7]	A10

Added “A.6 – Security Misconfiguration”



Avoiding Security Misconfiguration

- Verify your system's configuration management
  - Secure configuration "hardening" guideline
    - Automation is REALLY USEFUL here
  - Must cover entire platform and application
  - Keep up with patches for ALL components
    - This includes software libraries, not just OS and Server applications
  - Analyze security effects of changes

- Can you "dump" the application configuration
  - Build reporting into your process
  - If you can't verify it, it isn't secure

- Verify the implementation
  - Scanning finds generic configuration and missing patch problems

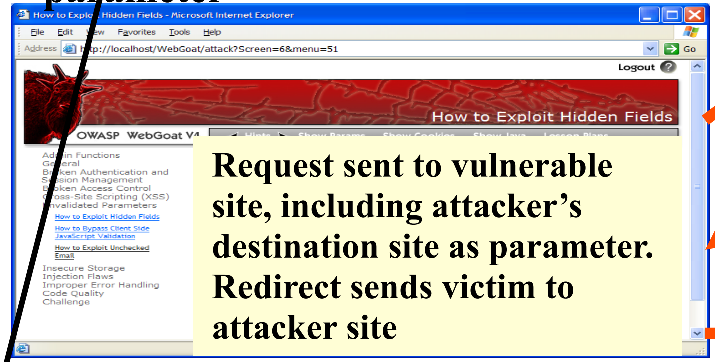
Added “A.10 - Unvalidated Redirect”

1 Attacker sends attack to victim via email or webpage

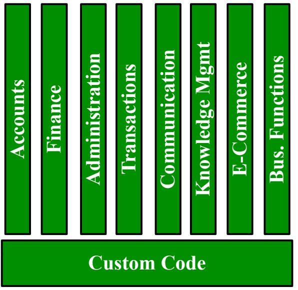
From: Internal Revenue Service
Subject: Your Unclaimed Tax Refund
Our records show you have an unclaimed federal tax refund. Please click here to initiate your claim.

3 Application redirects victim to attacker’s site

2 Victim clicks link containing unvalidated parameter


How to Explo Hidden Fields - Microsoft Internet Explorer
File
Edit
ew
Favorites
Tools
Help
Address
tp://localhost/WebGoat/attack?Screen=6&menu=51
Go
Logout
How to Exploit Hidden Fields
OWASP WebGoat V4
Ad in Functions
Ge eral
Br ken Authentication and
S ssion Management
B oken Access Control
C oss-Site Scripting (XSS)
nvalidated Parameters
How to Exploit Hidden Fields
How to Bypass Client Side JavaScript Validation
How to Exploit Unchecked Email
Insecure Storage
Injection Flaws
Improper Error Handling
Code Quality
Challenge

Request sent to vulnerable site, including attacker’s destination site as parameter. Redirect sends victim to attacker site



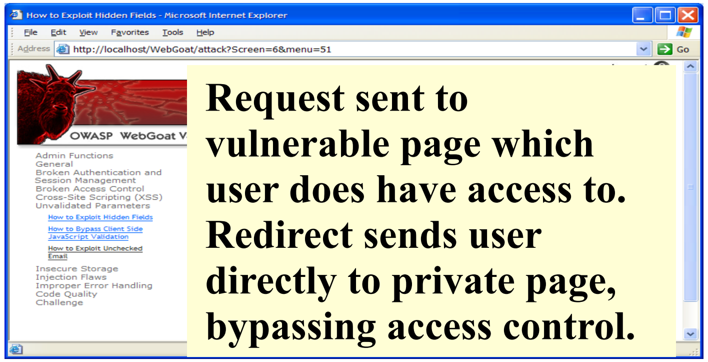
Evil Site

4 Evil site installs malware on victim, or phish’s for private information

http://www.irs.gov/taxrefund/claim.jsp?year=2006& … &dest=www.evilsite.com

Added “A.10 - Unvalidated Forward”

1 Attacker sends attack to vulnerable page they have access to


How to Exploit Hidden Fields - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://localhost/WebGoat/attack?Screen=6&menu=51
Go
OWASP WebGoat V
Admin Functions
General
Broken Authentication and Session Management
Broken Access Control
Cross-Site Scripting (XSS)
Unvalidated Parameters
How to Exploit Hidden Fields
How to Bypass Client Side JavaScript Validation
How to Exploit Unchecked Email
Insecure Storage
Injection Flaws
Improper Error Handling
Code Quality
Challenge

Request sent to vulnerable page which user does have access to. Redirect sends user directly to private page, bypassing access control.

```
public void sensitiveMethod
( HttpServletRequest request,
HttpServletResponse response) {
    try {
          // Do sensitive stuff here.
          ...
    }
    catch ( ...
```

2 Application authorizes request, which continues to vulnerable page

Filter

3 Forwarding page fails to validate parameter, sending attacker to unauthorized page, bypassing access control

```
public void doPost( HttpServletRequest request,
HttpServletResponse response) {
    try {
         String target = request.getParameter( "dest" ) );
         ...
         request.getRequestDispatcher( target ).forward
         (request, response);
    }
    catch ( ...
```

Avoiding Unvalidated Redirects and Forwards

- There are a number of options
  1. Avoid using redirects and forwards as much as you can
  2. If used, don't involve user parameters in defining the target URL
  3. If you 'must' involve user parameters, then either
     a) Validate each parameter to ensure its valid and authorized for the current user, or
     b) (preferred) – Use server side mapping to translate choice provided to user with actual target page
  - Defense in depth: For redirects, validate the target URL after it is calculated to make sure it goes to an authorized external site
  - ESAPI can do this for you!!
    - See: SecurityWrapperResponse.sendRedirect(URL)
    - http://owasp-esapi-java.googlecode.com/svn/trunk_doc/org/owasp/esapi/filters/SecurityWrapperResponse.html#sendRedirect(java.lang.String)

- Some thoughts about protecting Forwards
  - Ideally, you'd call the access controller to make sure the user is authorized before you perform the forward (with ESAPI, this is easy)
  - With an external filter, like Siteminder, this is not very practical
  - Next best is to make sure that users who can access the original page are ALL authorized to access the target page.

Politics of “Unvalidated Redirects and Forwards”

Two totally separate and different vulnerabilities

- Executives confuse both to refer to redirects.

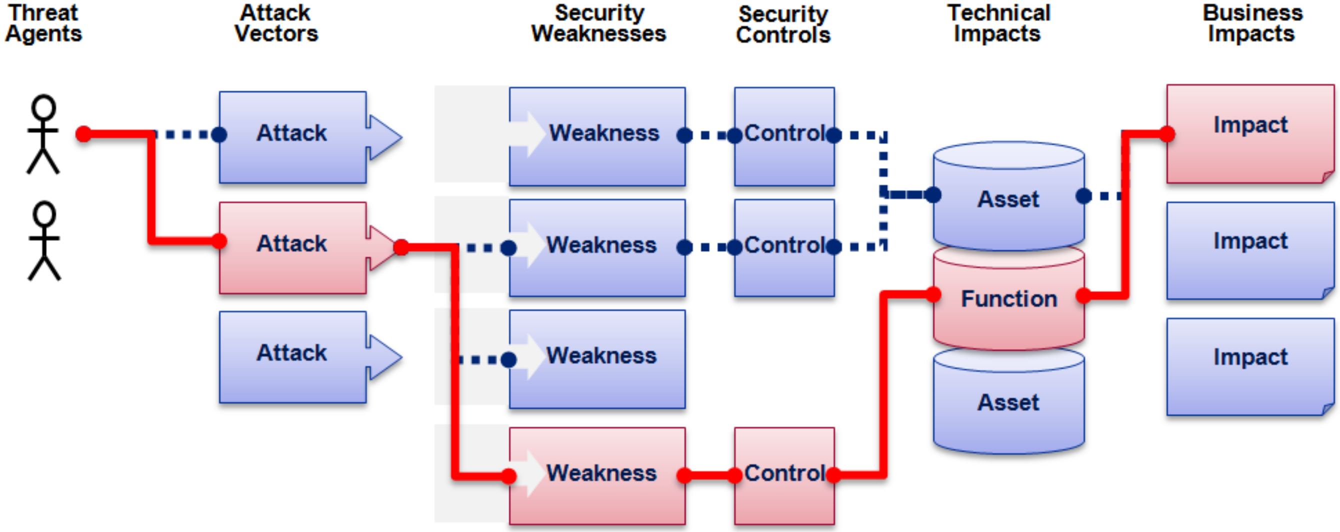
Solution list as T10 separate apart entries e.g.

8. “Unvalidated Forwards”

…

x. “Unvalidated Redirects”

OWASP Top 10 Risk Rating Methodology



Threat Agent	Attack Vector	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact
?	1 Easy	Widespread	Easy	Severe	?
	2 Average	Common	Average	Moderate	
	3 Difficult	Uncommon	Difficult	Minor	
	2	1	1	2	
		1.3	*	2	

XSS Example

2.6 weighted risk rating

Politics of OWASP Risk Rating Methodology

Not recommended by OWASP Threat Modeling.

- Others e.g. STRIDE, DREAD, etc not used either.

ASPECT SECURITY “donated” this to OWASP.

Application Security Specialists

- Perceived Conflict of Interest.

Timeline from Release Candidate (RC) to Final

1. Closed Peer Review
2. RC unveiled at AppSecDC on **13 Nov 2009**
3. Public Comment until **31 Dec 2009**
4. Final released planned for ~~January~~ April ~~15~~ 19
   - Due to competition with SANS Top 25 (2010) released in Feb
   - Press Release dated Saturday 17 April
   - Moved FINAL Release to Google Docs due to download demand

Politics of the OWASP T10 vs SANS Top 25

SANS Top 25 (2009) attempted “steal” but PR failed.

- Now a residual risk to the “Awareness” of Top Ten.
- Not much difference i.e.
  - “Buffer Overflows” vs “Security Misconfiguration”

MITRE CWE publishes more then 700 types of vuln

T10 2010 Release Date was pushed back and forward

When *Not* to Cite the OWASP Top Ten?

PCI DSS and PA-DSS

- Cited (incorrectly) as OWASP "Guide"
- Payment Applications (PA) are TANDEM, etc based.
  - Exception is Web Server within LPAR

"Platform Security – Facebook Developer Wiki"

When *Not* to Cite the OWASP Top Ten?

Web Application Firewall (WAF) and other Vendors:

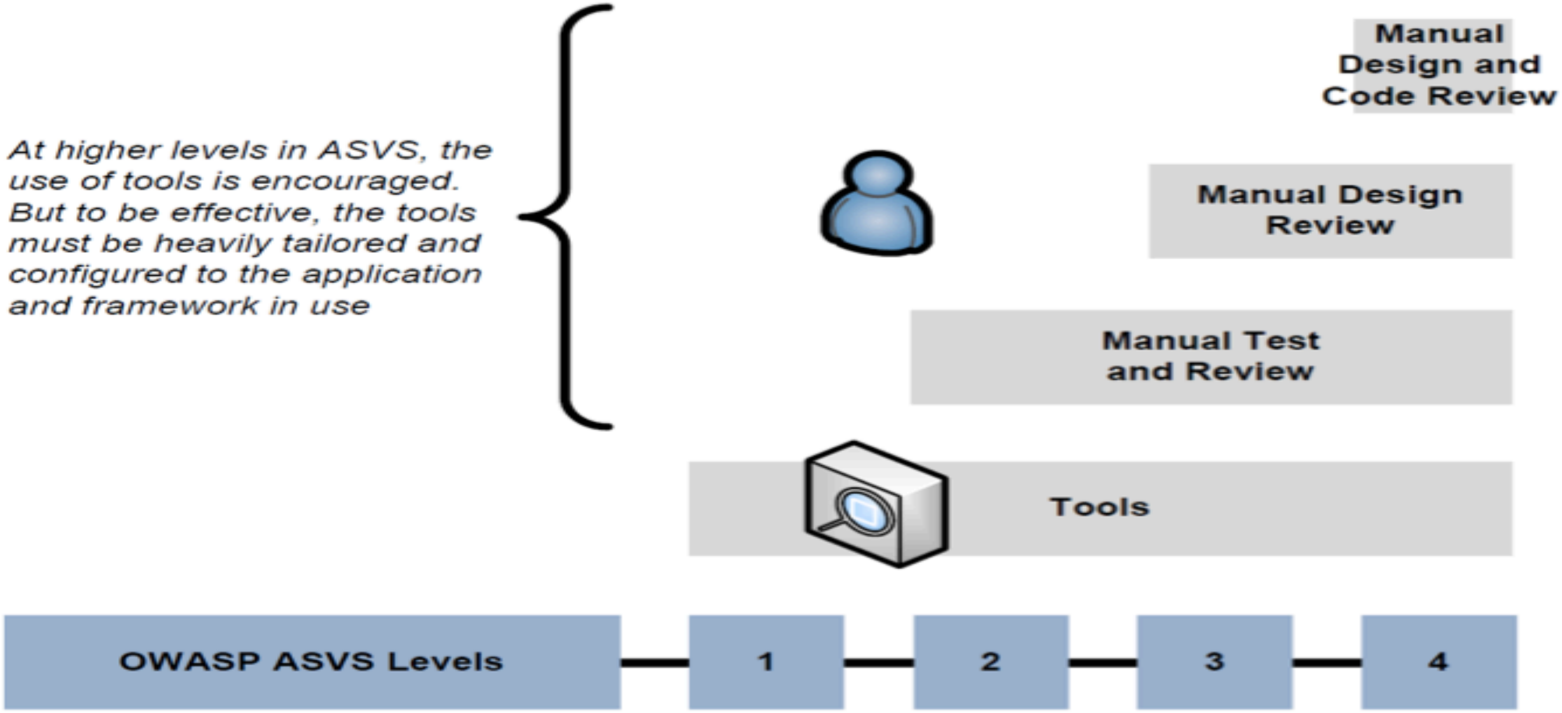
- WAF don’t address root causes
- Mark Curphey (OWASP Founder) raised abuse issue.
- AvdS suggested OWASP T10 Certification Scheme

webappsec “blackbox” or “whitebox” pen testing RFTs

Application Security Verification Standard

Consider ASVS instead of OWASP Top 10

- Some issues when implemented in practice.


At higher levels in ASVS, the use of tools is encouraged. But to be effective, the tools must be heavily tailored and configured to the application and framework in use
Manual Design and Code Review
Manual Design Review
Manual Test and Review
Tools
OWASP ASVS Levels
1
2
3
4

Internal OWASP Politics of the Top Ten

Against OWASP “Builders not Breakers” Directive

Justified as “Awareness” for Executive audience

- ASPECT SECURITY Application Security Specialists generate “not for profit” revenue

In Summary

It's About Risks, Not Just Vulnerabilities

- New title is: "The Top 10 Most Critical Web Application Security Risks"

OWASP Top 10 Risk Rating Methodology

- Based on the OWASP Risk Rating Methodology, used to prioritize Top 10

2 Risks Added, 2 Dropped

- **Added: A6 – Security Misconfiguration**
  - Was A10 in 2004 Top 10: Insecure Configuration Management
- **Added: A8 – Unvalidated Redirects and Forwards**
  - Relatively common and VERY dangerous flaw that is not well known
- **Removed: A3 – Malicious File Execution**
  - Primarily a PHP flaw that is dropping in prevalence
- **Removed: A6 – Information Leakage and Improper Error Handling**
  - A very prevalent flaw, that does not introduce much risk (normally)

Further Information

URLs Published by OWASP

http://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

http://lists.owasp.org/mailman/listinfo/owasp-topten

URLs Aggregated by cmlh

http://deli.cio.us/cmlh/OWASP.Top.Ten

Copyright Notices

Slides and Notes Licensed as:

- **AU Creative Commons 2.5**
  - **Attribution-Non Commercial-No Derivative Works**

Attribution for Images:

- AppSec_DC_2009_-_OWASP_Top_10_-_2010_rc1.pptx
- About_OWASP_ASVS.ppt

Thanks

OWASP “Top Ten” Project

- Dave Wichers and Jeff Williams
- Andrew van der Stock (T10 2010 Reviewer)
- All other T10 2010 Reviewers

Thanks

Jean-Marie Abighanem

- OWASP – Melbourne Chapter

Audrey Lyon and Drazen Drazic

- AISA

Paul Theriault

- OWASP – Sydney Chapter

In Closing

Slides are Published on slideshare

http://www.slideshare.net/cmlh

christian.heinrich@owasp.org

http://www.owasp.org/index.php/user:cmlh

OWASP Top Ten 2010

FINAL Release

Christian Heinrich

christian.heinrich@owasp.org

"Google Hacking" Project Leader

OWASP – Sydney Chapter
April 2010

Previously presented at:
- **AISA Annual Seminar Day 2009 and;**
- **OWASP Melbourne Chapter - December 2009**

The OWASP Foundation
http://www.owasp.org/