

OWASP v3: nuovi projects and new goals

(**Matteo Meucci – CISA, CISSP – OWASP- Italy**)

INDEX PRESENTATION :

1. Presentazione di OWASP e del capitolo italiano
2. OWASP Projects and vision
3. OWASP Testing Guide v3

Who am I?

- Research

- OWASP-Italy Chair
- OWASP Testing Guide Lead



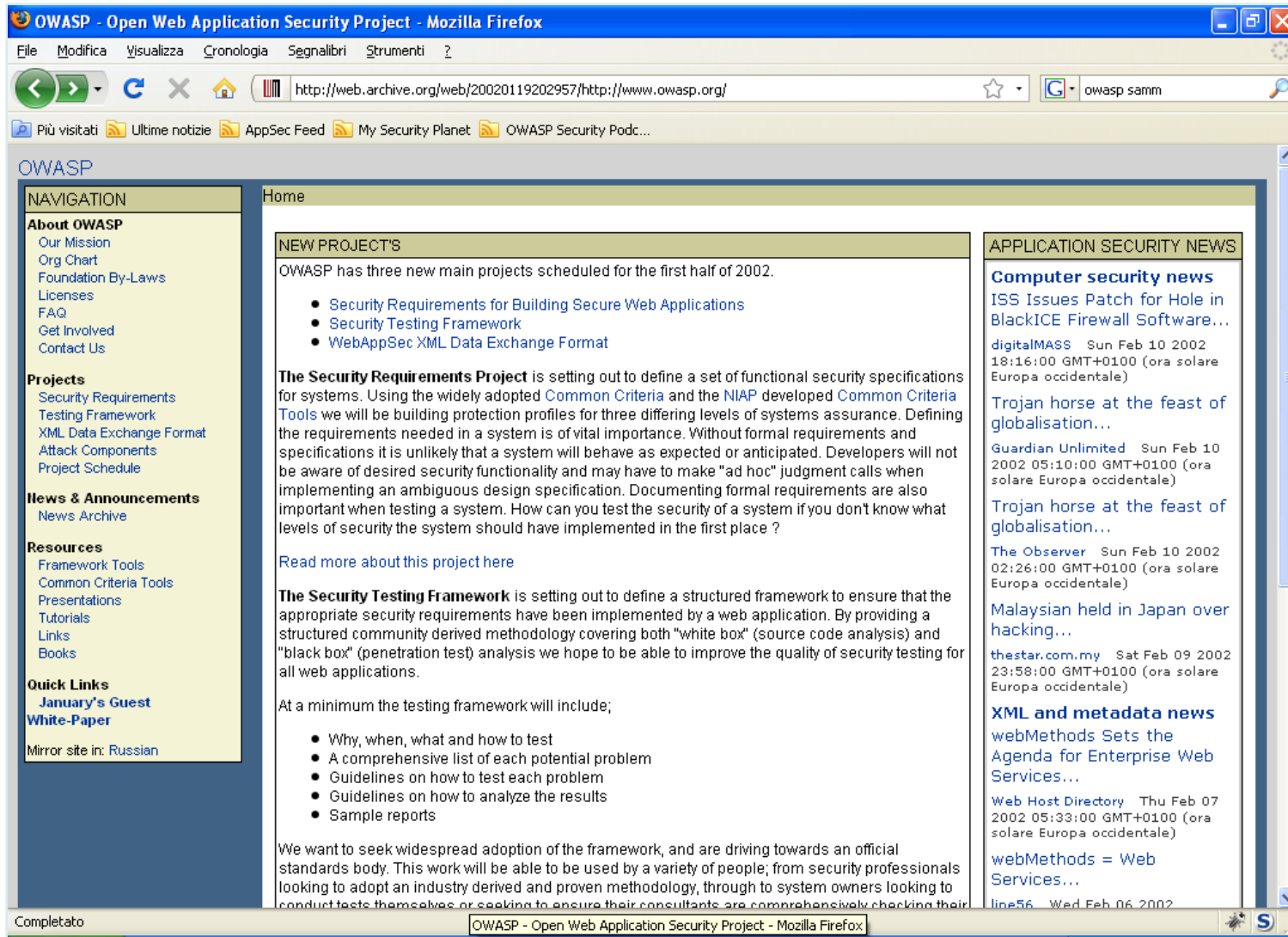
- Work

- CEO @ Minded Security
- Application Security Consulting
- 8+ years on Information Security
- focusing on Application Security
- www.mindedsecurity.com



- The Open Web Application Security Project (OWASP) is dedicated to finding and fighting the causes of insecure software. The OWASP Foundation is a 501c3 not-for-profit charitable organization that ensures the ongoing availability and support for our work.
- Participation in OWASP is free and open to all.
- Everything here is free and open source.
- Main objectives: producing tools, standards and documentations related to Web Application Security.
- Thousands active members, 100+ local chapters in the world
- Millions of hits on www.owasp.org

OWASP v1: start the community



OWASP v2: wiki and sharing

Main Page - OWASP - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://www.owasp.org/index.php/Main_Page

Google Search PageRank Check AutoLink AutoFill Subscribe Options Log in / create account

Main Page

GET A FREE Security Assessment **FORTIFY** Protect your code
Sponsored advertisement. OWASP does not endorse commercial products or services

Welcome to OWASP
the free and open application security community

About · Searching · Editing · New Article · OWASP Categories Statistics · Recent Changes

OWASP Overview

The Open Web Application Security Project (OWASP) is dedicated to finding and fighting the causes of insecure software. Everything here is free and open source. The OWASP Foundation is a 501c3 not-for-profit charitable organization that ensures the ongoing availability and support for our work. Participation in OWASP is free and open to all.

Join webappsec! The OWASP mail list. Get Started Find out more.

Contact OWASP owasp@owasp.org Become a Member Support our efforts.

Featured Story

Two free Java EE filters for CSRF, Reflected XSS, and Adobe XSS

OWASP contributors from **Aspect Security** have developed two new Java EE filters to protect against common web attacks. Just add a few lines to your web.xml file and enjoy the protection.

CSRF and Reflected XSS Filter for Java EE

This filter adds a random token to forms and URLs that prevent an attacker from executing both CSRF and reflected XSS attacks.

Adobe XSS Filter for Java EE

This filter protects against the recent XSS attacks on PDF files. By using a redirect and an encrypted

OWASP Community (add)

Click the map to find and join your local chapter

Feb 13 (18:00h) - Ireland chapter meeting
Feb 6 (18:00h) - Melbourne chapter meeting
Jan 31 (15:00h) - Mumbai chapter meeting
Jan 30 (11:30h) - Austin chapter meeting
Jan 25 (14:30h) - Italy@ISACA Rome
Jan 25 (18:00h) - San Francisco chapter meeting
Jan 24 (17:30h) - 6th OWASP Israel chapter meeting
Jan 23 (18:00h) - Belgium chapter meeting
Jan 22 (18:00h) - Rochester chapter meeting

Older events...

OWASP News (add)

Jan 16 - OWASP Newsletter 2

confirm: Which language do you prefer to display your English (USA)

Editing How to value the real risk (section) - OWASP - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

https://www.owasp.org/index.php/How_to_value_the_real_risk_ridaction=ridsection=15

Google Search PageRank Check AutoLink AutoFill Subscribe Options Log in / create account

Editing How to value the real risk (section)

Jeff Williams my talk my preferences my watchlist my contributions log out

article discussion edit history protect delete

Navigation

- Home
- News
- Projects
- Downloads
- Local Chapters
- Conferences
- Presentations
- Video
- Papers
- Mailing Lists
- About OWASP
- Membership

reference

- How To...
- Principles
- Threat Agents
- Attacks
- Vulnerabilities
- Countermeasures

asp-test

the list, Done

send email to owasp-testing@lists.owasp.org.

our existing subscription, in the sections below.

the following form. You will be sent email from gratuitously subscribing you. This is a hidden is available only to the list administrator.

use will be automatically generated once you've confirmed your subscription. your password when you edit your password will be emailed to you as a

reference

- How To...
- Principles
- Threat Agents
- Attacks
- Vulnerabilities
- Countermeasures
- Activities
- Technologies
- Glossary
- Code Snippets
- NET Project
- Java Project

search

Google Custom Search

toolbox

- What links here
- Related changes
- Upload file
- Special pages
- Printable version
- Permanent link

19 January 2007

- (diff) (hist) - & Phoenix/Tools, 22:18 - One (Talk) (contrib)
- (diff) (hist) - & Phoenix/Tools, 22:16 - One (Talk) (contrib)
- (diff) (hist) - & OWASP WebScarab NG Project, 15:43 - Jeff Williams (Talk) (contrib)
- (diff) (hist) - & No Long long ago, - 11:32 - EoinKeary (Talk) (contrib)
- (diff) (hist) - & OWASP Code Review Guide Table of Contents, 11:22 - EoinKeary (Talk) (contrib) [-Example by Vulnerability]
- (diff) (hist) - & OWASP Code Review Guide Table of Contents, 11:21 - EoinKeary (Talk) (contrib)
- (diff) (hist) - & Inner classes, 11:17 - Javatica (Talk) (contrib)
- (diff) (hist) - & Chapters Assigned, 10:10 - EoinKeary (Talk) (contrib) [-Example by Vulnerability]
- (diff) (hist) - & XSS Attacks, 10:09 - EoinKeary (Talk) (contrib)
- (diff) (hist) - & No Renaming code for XSS issues, 10:06 - EoinKeary (Talk) (contrib)
- (diff) (hist) - & OWASP Code Review Guide Table of Contents, 10:05 - EoinKeary (Talk) (contrib) [-Example by Vulnerability]
- (diff) (hist) - & Belgium, 08:31 - Staleenryder (Talk) (contrib)
- (diff) (hist) - & Phoenix/Tools, 02:24 - One (Talk) (contrib)
- (diff) (hist) - & Phoenix/Tools, 02:22 - One (Talk) (contrib)
- (diff) (hist) - & Phoenix/Tools, 02:17 - One (Talk) (contrib)
- (diff) (hist) - & Phoenix/Tools, 01:58 - One (Talk) (contrib)
- (diff) (hist) - & Phoenix/Tools, 01:57 - One (Talk) (contrib)
- (diff) (hist) - & Phoenix/Tools, 01:54 - One (Talk) (contrib)
- (diff) (hist) - & Phoenix/Tools, 01:51 - One (Talk) (contrib)
- (diff) (hist) - & Category OWASP Top Ten Project, 00:43 - OWASP (Talk) (contrib)
- (diff) (hist) - & Category OWASP Top Ten Project, 00:43 - OWASP (Talk) (contrib)
- (diff) (hist) - & Testing for Session Management, 00:33 - Newacco (Talk) (contrib) [-v1.5]

toolbox

- new user
- upload file
- Special pages

Done 2 Errors

OWASP v3: Improve Quality and Support

- Define Criteria for Quality Levels
 - Alpha, Beta, Release
- Encourage Increased Quality
 - Through Season of Code Funding and Support
 - Produce Professional OWASP books
- Provide Support
 - Full time executive director (Kate Hartmann)
 - Full time project manager (Paulo Coimbra)
 - Half time technical editor (Kirsten Sitnick)
 - Half time financial support (Alison Shrader)
 - Looking to add programmers (Interns and professionals)

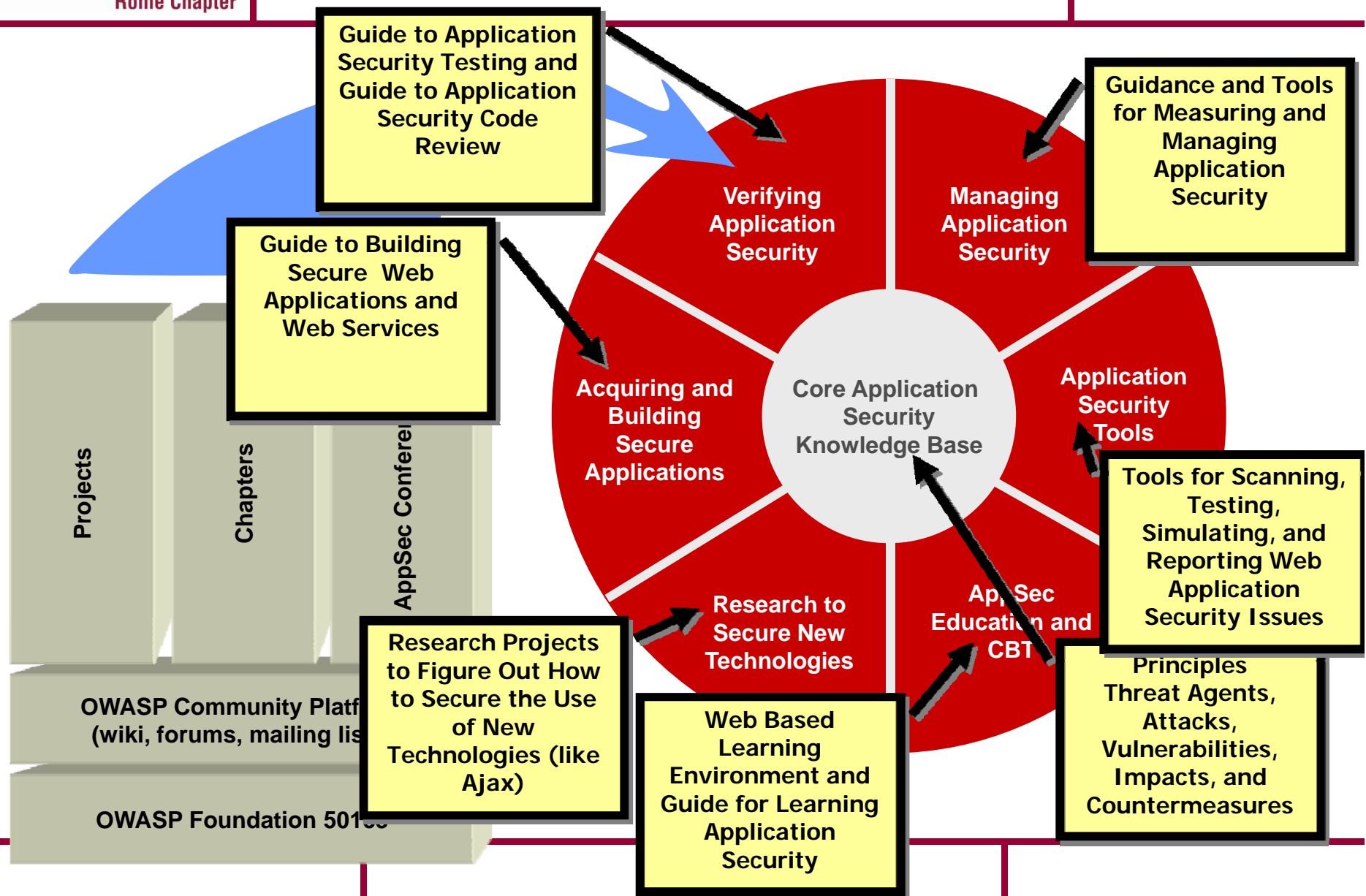


- > 30 project leaders
- OWASP GLOBAL COMMITTEES
 - Projects
 - Membership
 - Education
 - Conferences
 - Industry
 - Chapters

La comunità OWASP



OWASP Body of Knowledge



There are a lot of OWASP projects

https://www.owasp.org/index.php/Category:OWASP_Project

Category:OWASP Project

An OWASP project is a collection of related tasks that have a defined roadmap and team members. OWASP project leaders are responsible for defining the vision, roadmap, and tasks for the project. The project leader also promotes the project and builds the team.

If you would like to start a new project please review the [How to Start an OWASP Project](#) guide. Please send an email to owasp@owasp.org to discuss project ideas and how they might fit into OWASP. All OWASP projects must be free and open and have their homepage on the OWASP portal. You can read all the guidelines in the [Project Assessment Criteria](#).

Every project has an associated mail list. You can view all the lists, examine their archives, and subscribe to any of them on the [OWASP Project Mailing Lists](#) page.

Contents (hide)

- 1 Release Quality Projects
- 2 Current Season of Code Projects
- 3 Beta Status Projects
- 4 Alpha Status Projects
- 5 Inactive Projects
- 6 How to add a new OWASP Project article

Release Quality Projects

Release quality projects are generally the level of quality of professional tools or documents.

We have started the process of defining detailed guidelines which indicate what will be required from an OWASP Project in order for it to be classified an OWASP Release quality project (see [Project Assessment Criteria](#)). Please note that the projects below have **NOT** been evaluated under this criteria and might be re-classified once that process is completed.

Tools	Documentation
OWASP WebGoat Project an online training environment for hands-on learning about application security	OWASP AppSec FAQ Project FAQ covering many application security topics
OWASP WebScarab Project a tool for performing all types of security testing on web applications and web services	OWASP Guide Project a massive document covering all aspects of web application and web service security
	OWASP Legal Project a project focused on contracting for secure software
	OWASP Testing Guide a project focused on application security testing procedures and

Beta Status Projects

Beta quality projects are complete and ready to use with documentation.

We have started the process of defining detailed guidelines which indicate what will be required from an OWASP Project in order for it to be classified an OWASP Beta quality project (see [Project Assessment Criteria](#)). Please note that the projects below have **NOT** been evaluated under this criteria and might be re-classified once that process is completed.

Tools	Documentation
OWASP AntiSamy Project an API for validating rich HTML/CSS input from users without exposure to cross-site scripting and phishing attacks	OWASP CLASP Project a project focused on defining process elements that reinforce application security
OWASP CSRFGuard Project a J2EE filter that implements a unique request token to mitigate CSRF attacks	OWASP Code Review Project A project to capture best practices for reviewing code. This project is being sponsored by OWASP Summer of Code .
OWASP DirBuster Project DirBuster is a multi threaded java application designed to brute force directories and files names on web/application servers.	OWASP Tools Project The OWASP Tools Project's goal is to provide unbiased, practical information and guidance about application security tools.
OWASP Encoding Project a project focused on the development of encoding best practices for web applications.	
OWASP Enterprise Security API (ESAPI) Project a free and open collection of all the security methods that a developer needs to build a secure web application.	
OWASP LAPSE Project an Eclipse-based source-code static analysis tool for Java	
OWASP Live CD Education Project an educational supplement project containing tutorials, challenges and videos detailing the use of tools contained within the OWASP LiveCD - LabRat. This project was sponsored by OWASP Spring Of Code 2007 and Security Distro .	

Current Season of Code Projects

The projects placed in this category are under development. Project completion is expected by 15th September. After the Season Code being finished, all projects will be moved to the appropriate category - alpha, beta or release quality.

Tools	Documentation
GTK-GUI for w3af Project The main objective is to minimize the effort and learning curve of using w3af, providing a very usable graphical interface. This project is being sponsored by OWASP Summer of Code .	OWASP ASDR Project The ASDR is a reference volume that contains basic information about all the foundational topics in application security. This project is being sponsored by OWASP Summer of Code .
OWASP Access Control Rules Tester Project This project is intended to have two deliverables: research technical report (publication ready article) and an Access Control Rules Tester tool. This project is being sponsored by OWASP Summer of Code .	OWASP Application Security Verification Standard Project This is a new project created to define an evaluation framework that may be used to conduct OWASP Application Security Verification Standard certifications. This project is being sponsored by OWASP Summer of Code .
OWASP AntiSamy Project An API for validating rich HTML/CSS input from users without exposure to cross-site scripting and phishing attacks. This project is being sponsored by OWASP Summer of Code .	OWASP AppSensor Project A framework for detecting and responding to attacks from within the application. This project is being sponsored by OWASP Summer of Code .
OWASP Application Security Tool Benchmarking Environment and Site Generator Refresh Project The idea is to split destination web application technology from the three reusable libraries: library of navigational elements, library of vulnerabilities and library of language constructs. This project is being sponsored by OWASP Summer of Code .	OWASP Backend Security Project This is a new project created to improve and to collect the existent information about the backend security. This project is being sponsored by OWASP Summer of Code .
OWASP Code Crawler This tool is aimed at assisting code review practitioners. It is a static code review tool which searches for key topics within .NET and J2EE/JAVA code. The aim of the tool is to accompany the OWASP Code review Guide and to implement a total code review solution for "everyone". Where "everyone" means "more" companies performing secure software activities. This project is being sponsored by OWASP Summer of Code .	OWASP Book Cover & Sleeve Design This is a project of corporate design to develop a scalable book cover series strategy and a Book Sleeve. This project is being sponsored by OWASP Summer of Code .
OWASP Interceptor Project A testing tool for XML web service and Ajax interfaces. This project is being sponsored by OWASP Summer of Code .	OWASP Classic ASP Security Project It aims in creating a secure framework for Classic ASP application by complementing existing OWASP projects with documentation for this particular technology and the creation of security libraries. This project is being sponsored by OWASP Summer of Code .
OWASP JSP Testing Tool Project The goal of this project is to create an easy to use, freely available tool that can be used to quickly ascertain the level of protection that each	OWASP Code Review Project A project to capture best practices for reviewing code. This project is being sponsored by OWASP Summer of Code .
	OWASP Corporate Application Security Rating Guide This project will organize and structure publicly available data that large

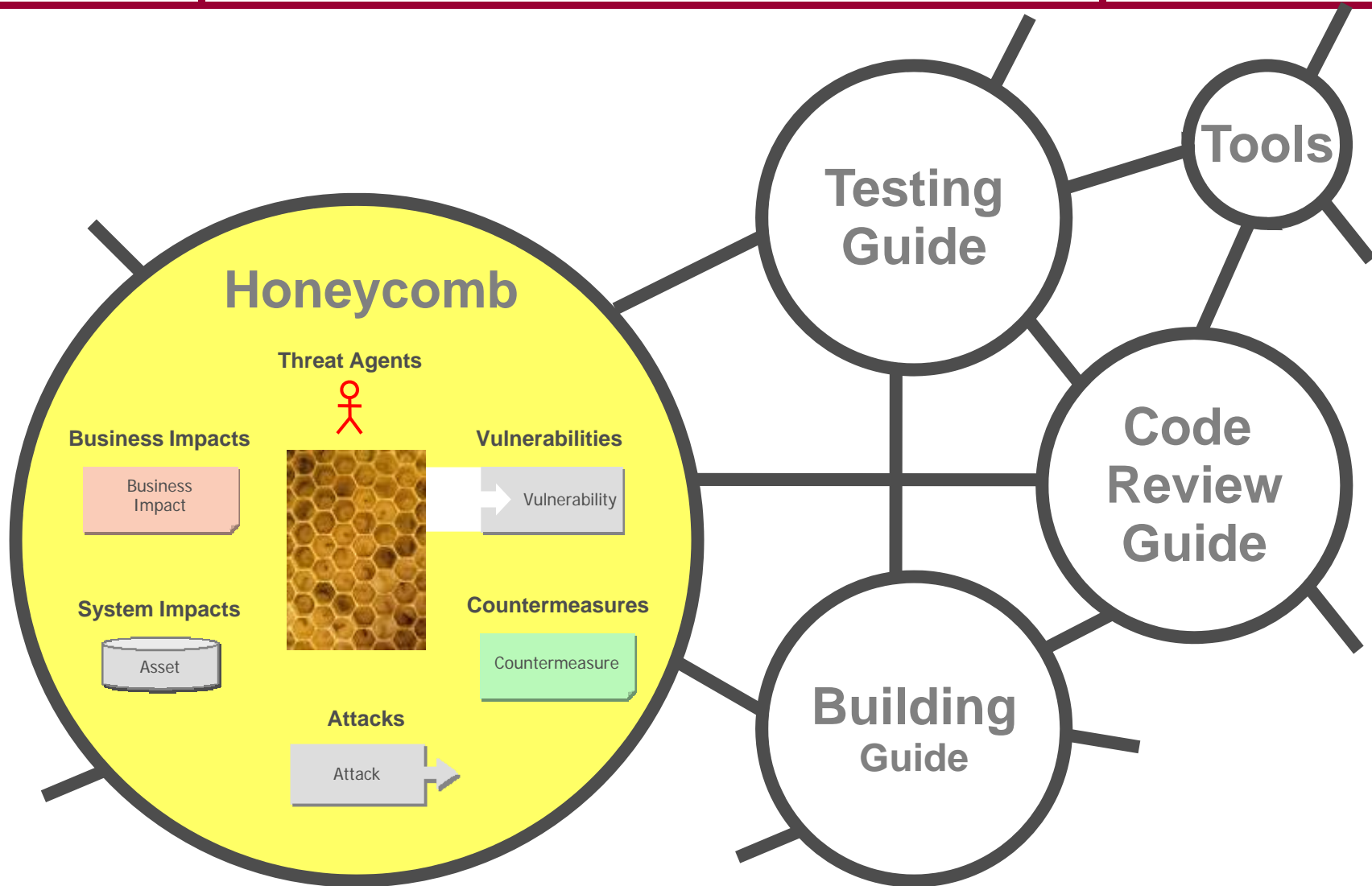
Alpha Status Projects

Alpha quality projects are generally usable but may lack documentation or quality review.

We have started the process of defining detailed guidelines which indicate what will be required from an OWASP Project in order for it to be classified an OWASP Alpha quality project (see [Project Assessment Criteria](#)). Please note that the projects below have **NOT** been evaluated under this criteria and might be re-classified once that process is completed.

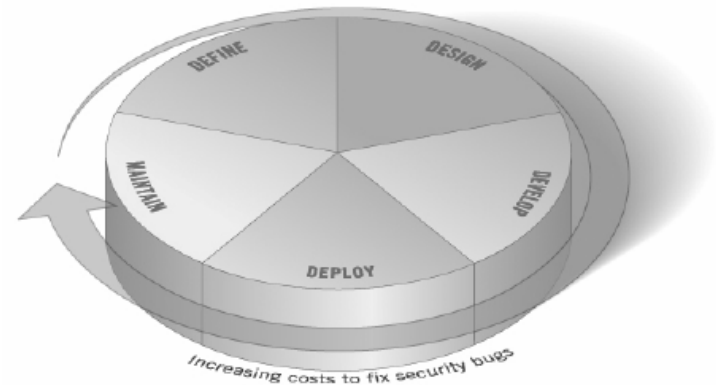
Tools	Documentation
OWASP CSRFTester Project gives developers the ability to test their applications for CSRF flaws	OWASP AIR Security Project investigating the security of AIR applications
OWASP EnDe Project This tool is an encoder, decoder, converter, transformer, calculator, for various codings used in the wild wide web.	OWASP AJAX Security Guide investigating the security of AJAX enabled applications
OWASP Google Hacking Project Google SOAP Search API with Perl	OWASP Application Security Assessment Standards Project establish a set of standards defining baseline approaches to conducting differing types/levels of application security assessment
OWASP Insecure Web App Project a web application that includes common web application vulnerabilities	OWASP Application Security Requirements
OWASP JBrofuzz Project a fuzzer application, supporting a number of automated security checks including basic cross site scripting checks (XSS) as well as basic SQL injection testing. This project was sponsored by OWASP Spring Of Code 2007	OWASP Application Security Metrics Project identify and provide a set of application security metrics that have been found by contributors to be effective in measuring application security
OWASP NetBouncer Project is secure by default centralised input/output validation library which combines security rules and business rules as well as escaping in the output level.	OWASP Career Development Project The OWASP Career Development project is focused on helping application security professionals understand the job market, roles, career paths, and skills to work in the field.
OWASP Open Review Project (ORPRO)	OWASP Certification Criteria Project
	OWASP Certification Project our challenge is to create a plan for certification: a set of OWASP Certification for Developers and Testers.

OWASP Guidelines: the big picture



Software development life cycle

- Software Development Life Cycle, SDLC:
 - Define
 - Design
 - Develop
 - Deploy
 - Maintain
- Which are the controls to implement?
 - Training
 - Policy Review
 - Guidelines
 - Code Review
 - Web Application Penetration Testing



SDLC & OWASP Guidelines in your organization

Before SDLC

Define&Design

Development

Deploy&Maintenance

**Policy and
Standards**

**Develop
metrics**



Awareness



Guidelines

**Security
Requirement**

**Threat
Modeling**



Building



Building Guide

**Code
Walkthrough**

Code Review



Review



Code Review Guide

**Application
Testing**

**Management
reviews**

Health checks



Test

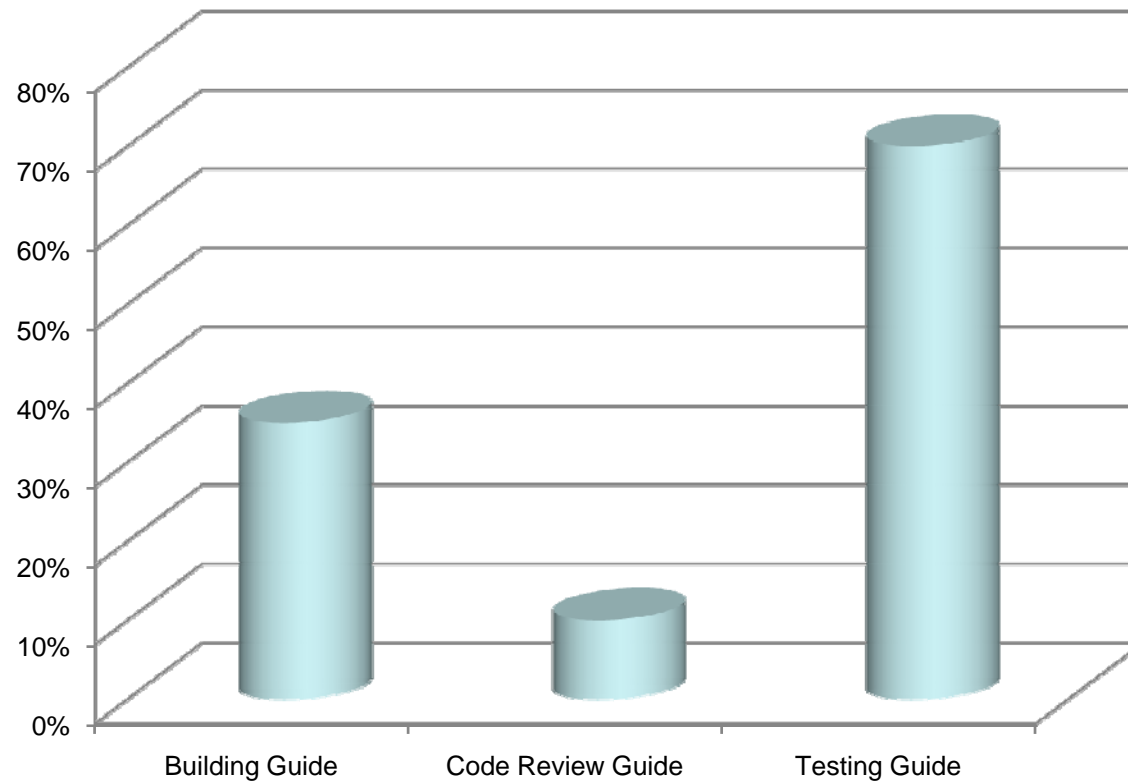


Testing Guide

Security in SDLC

OWASP Framework

OWASP guidelines in the italian companies



For a total of 15 Companies (Finance, Banking and Telco)

Source: Minded Security 2008

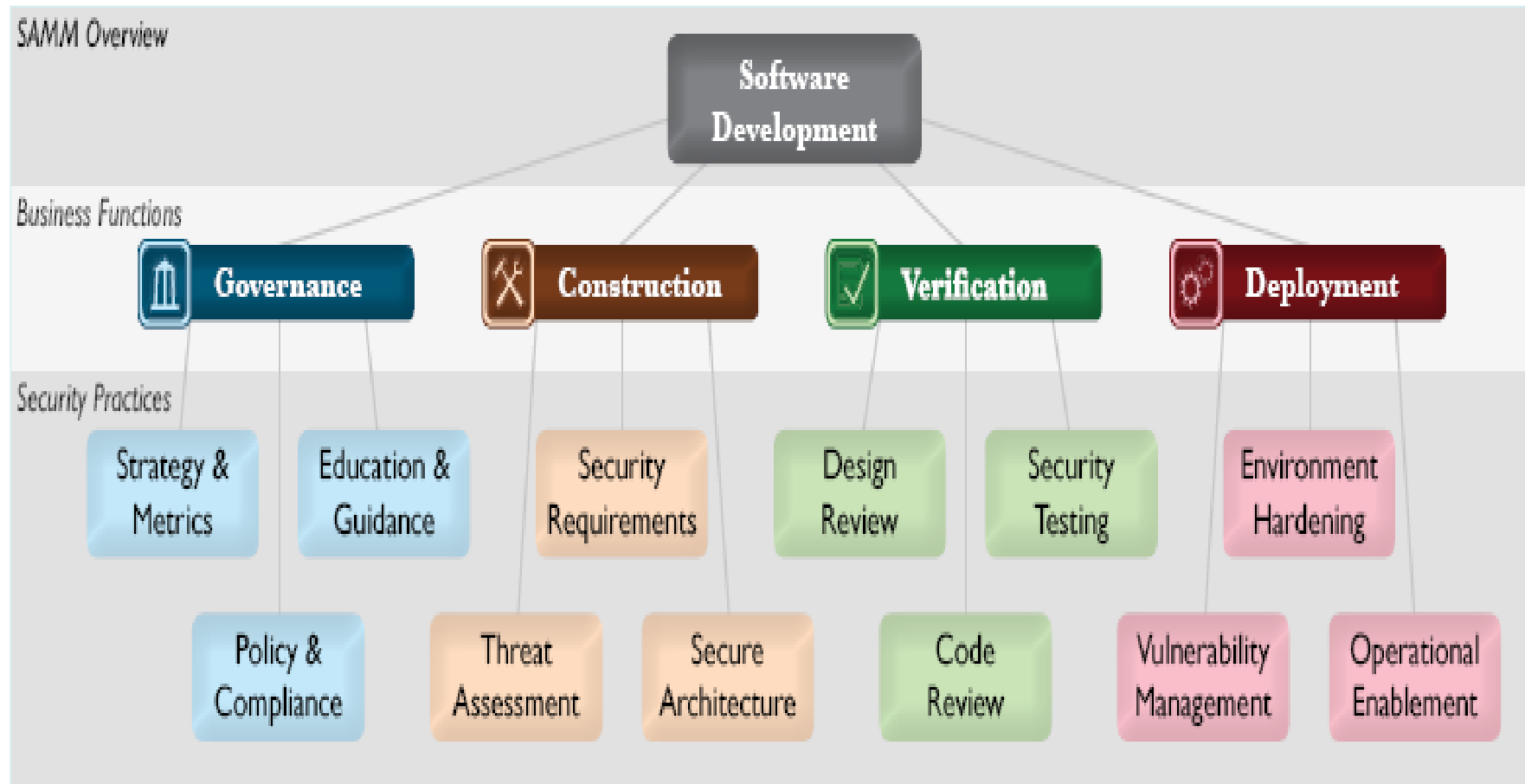
OWASP SAMM

- The Software Assurance Maturity Model (SAMM) project is committed to building a usable framework to help organizations formulate and implement a **strategy for application security** that's tailored to the specific business risks facing the organization.
- The goal is to create well-defined and measurable objectives that can be used by small, medium and large sized organizations in any line of business that involves software development.
- SAMM when:
 - Assess existing software assurance practice
 - Build a strategic roadmap for the organization
 - Implement or perform security activities

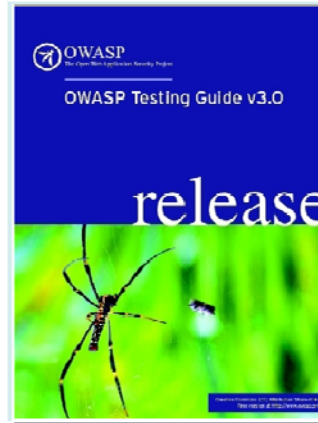


Project of Pravir Chandra
V1 released 25th March 09



OWASP SAMM: overview

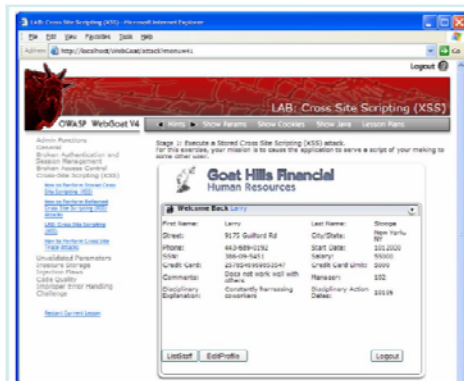


Principali progetti OWASP



BOOKS

- Owasp top10
- Building guide
- Code review guide
- Testing guide 
- Back end security 



TOOLS

- WebGoat
- WebScarab
- SQLMap – SQL Ninja (independent projects) 
- SWF Intruder 
- Orizon 
- Code Crawler 

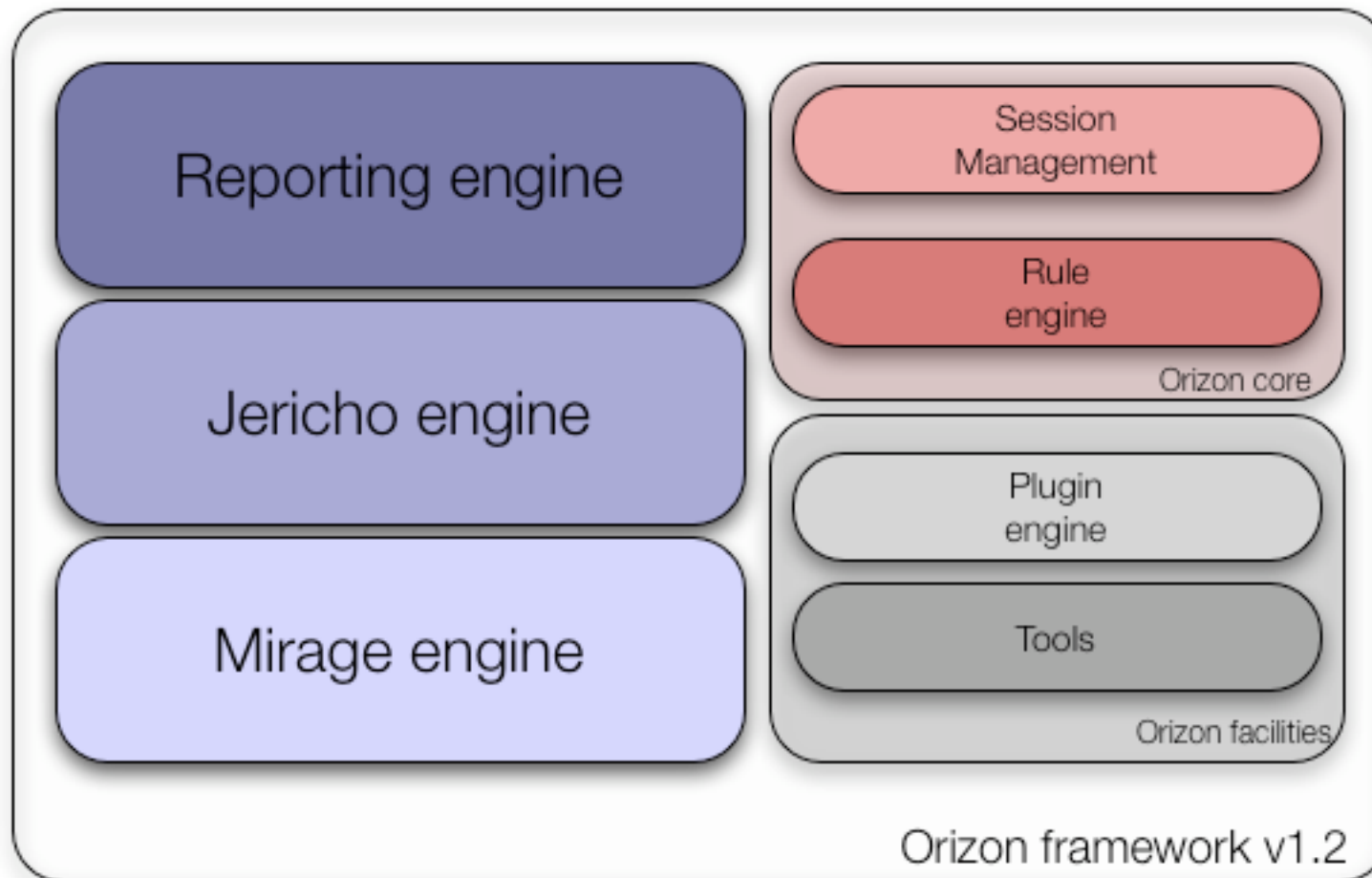
- OWASP Italy nasce nel Gennaio 2005
- Raccoglie centinaia di persone appassionati alla Web Application Security
- Obiettivi
 - Organizzazione conferenze
 - Scrittura articoli
 - Sviluppo tool
 - Sviluppo documentazione e linee guida
- La ricerca come base per l'industria
 - Mai come nell'application security si ha un'esigenza di ricerca per lo sviluppo di attività di innovazione

- Day I
 - Marzo 2008 – Università La Sapienza - Roma
- Day II
 - Settembre 2008 – Università La Sapienza - Roma
- Day III
 - Febbraio 2009 – Università di Bari
- Day IV
 - Ottobre/Novembre 2009 – Milano

<http://www.owasp.org/Index.php/Italy>
Mailing list!

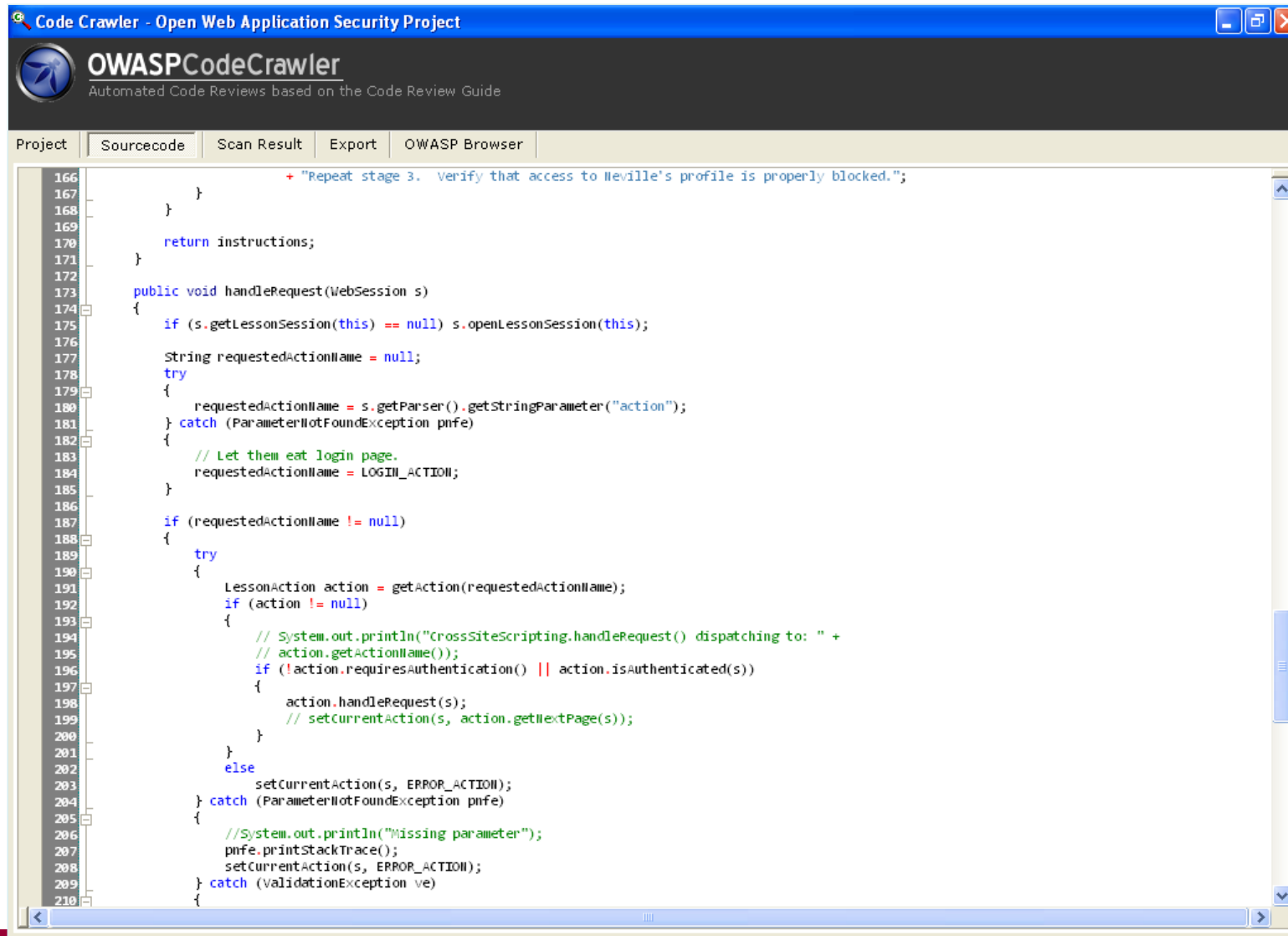
OWASP-Italy tools: Orizon

by Paolo Perego



OWASP Code Crawler

by Alessio Marziali



The screenshot shows the OWASP Code Crawler application window. The title bar reads "Code Crawler - Open Web Application Security Project". The application has a dark header with the "OWASPCodeCrawler" logo and the tagline "Automated Code Reviews based on the Code Review Guide". Below the header is a tabbed interface with tabs for "Project", "Sourcecode", "Scan Result", "Export", and "OWASP Browser". The "Sourcecode" tab is active, displaying a Java code snippet. The code is a method named `handleRequest` that takes a `WebSession s` as a parameter. It includes comments and logic for handling requests, including session management and action dispatching. A red comment at the top of the code block says: "+ Repeat stage 3. Verify that access to Neville's profile is properly blocked." The code is line-numbered from 166 to 210. The interface also includes a vertical scrollbar on the right and a horizontal scrollbar at the bottom.

```
166      + "Repeat stage 3. Verify that access to Neville's profile is properly blocked.";
167    }
168  }
169
170  return instructions;
171 }
172
173 public void handleRequest(WebSession s)
174 {
175     if (s.getLessonSession(this) == null) s.openLessonSession(this);
176
177     String requestedActionName = null;
178     try
179     {
180         requestedActionName = s.getParser().getStringParameter("action");
181     } catch (ParameterNotFoundException pnfe)
182     {
183         // Let them eat login page.
184         requestedActionName = LOGIN_ACTION;
185     }
186
187     if (requestedActionName != null)
188     {
189         try
190         {
191             LessonAction action = getAction(requestedActionName);
192             if (action != null)
193             {
194                 // System.out.println("CrossSiteScripting.handleRequest() dispatching to: " +
195                 // action.getActionName());
196                 if (!action.requiresAuthentication() || action.isAuthenticated(s))
197                 {
198                     action.handleRequest(s);
199                     // setCurrentAction(s, action.getNextPage(s));
200                 }
201             }
202             else
203             {
204                 setCurrentAction(s, ERROR_ACTION);
205             }
206         } catch (ParameterNotFoundException pnfe)
207         {
208             //System.out.println("Missing parameter");
209             pnfe.printStackTrace();
210             setCurrentAction(s, ERROR_ACTION);
211         } catch (ValidationException ve)
212         {
213         }
214     }
215 }
```


OWASP Code Crawler

by Alessio Marziali

Code Crawler - Open Web Application Security Project

OWASPCodeCrawler
Automated Code Reviews based on the Code Review Guide

Project | Sourcecode | Scan Result | Export | OWASP Browser

Threat Description

Locating where a database may be involved in the code is an important aspect of the code review. Looking at the database code will help determinate if the application is vulnerable to SQL Injection. One aspect of this is to verify that the code uses either SqlParameter, OleDbParameter or OdbcParameter (System.Data.SqlClient). These are type and treats parameter as the literal value and not the executable code in the database.

Threat Analysis

Found in D:\Software-WAS\WebGoat\WebGoat-5.2\JavaSource\org\owasp\webgoat\lessons\SQLInjection\SQLInjection.java

Threat Level ✖

Threat Family

Keywords Found

Threat	Description
Java.io	This command are generally used to read data into ones application. T
Java.io	This command are generally used to read data into ones application. T
Java.io	This command are generally used to read data into ones application. T
delete	Locating where a database may be involved in the code is an importar
update	Locating where a database may be involved in the code is an importar
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
update	Locating where a database may be involved in the code is an importar
delete	Locating where a database may be involved in the code is an importar
Public	Public and Sealed relate to the design at class level. Classes which are
Insert	Please provide a description for this item
select	Sorry. There is no description for this item
Public	Public and Sealed relate to the design at class level. Classes which are
Public	Public and Sealed relate to the design at class level. Classes which are
Bypass	Developers say the darnedest thing in their source code. Look for the
delete	Locating where a database may be involved in the code is an importar

OWASP Guidelines

This text has to change (awaiting owasp code review project hints)

OWASP SWF Intruder

by Stefano Di Paola

The screenshot shows the OWASP SWF Intruder web application running in a Mozilla Firefox browser. The browser's address bar displays the URL: `http://swf.mindedsecurity.loc/getVars.html?swfurl=http%3A%2F%2Fswf.minc`. The application interface includes a navigation bar with tabs: View, Config, History, Help, and About. Below this, there are input fields for 'Flash Movie' (containing `http://swf.mindedsecurity.loc/testSwf/test.swf`) and 'Query Object', each with a corresponding 'Load' or 'Query' button. The main content area is divided into three panels: 'Undefined Variables' (listing `[C]_url`, `_global.Test`, `_root.test`, `_root.obj`, `_root.sd`, and `_root.Stage`), 'SWF Instantiated Variables' (listing `_root.varTarget.onLoad`, `_root.de`, `_root.app`, `_root.$version`, `_root.image_mc`, `_root.varTarget`, `_root.my_txt`, and `_global.Test`), and 'Js/SWF Errors:'. A fourth panel, 'root.varTarget', shows a tree view of the object's properties, including `object_content` (with sub-properties `type_string`, `$version`, and `LNK 9,0,48,0`), `type_string` (with sub-property `resolvefor`), `type_string` (with sub-property `varToSend`), `$version=LNK 9,0,48,0&`, and `type_function` (with sub-property `resolve`). A search bar is present next to the 'root.varTarget' panel. The status bar at the bottom indicates 'In attesa di swf.mindedsecurity.loc...'.

SQLMap

By Bernardo Damele e Daniele Bellucci

```
inquis@leboyer:~/sqlmap$ python sqlmap.py -u "http://192.168.1.47/sqlmap/mysql/get.  
sqlmap/0.6-rc5 coded by inquis <bernardo.damele@gmail.com>  
and belch <daniele.bellucci@gmail.com>  
  
[*] starting at: 16:59:37  
  
remote DBMS:    active fingerprint: MySQL >= 5.0.2 and < 5.1  
                comment injection fingerprint: MySQL 5.0.45  
                html error message fingerprint: MySQL  
  
[*] shutting down at: 16:59:39
```

Screenshot: Database management system back-end extensive fingerprint



SQL Ninja

by Alberto Revelli

- 🌐 Sqlninja è sviluppato in PERL da Alberto Revelli (aka Icesurfer).
Tool che sfrutta SQL Injection per MS SQL Server.
- 🌐 Non individua SQL Injection, ma si focalizza nel creare una shell interattiva sul DB remoto e sfruttare questa per avere una “base” nella rete target.
 - Fingerprint del SQL Server
 - Bruteforce della password dell'utente 'sa'
 - Privilege escalation to 'sa'
 - Creazione di custom xp_cmdshell
 - Upload di file eseguibili
 - DNS tunneled pseudoshell, when no ports are available for a bindshell
 - E molto altro...

- Objectives:
 - Guide that could allow developers, administrators and testers to comprehend any parts of the security process about back-end components that directly communicate with the web applications as well as databases, Idaps, payment gateway...
- Composed of three sections:
 - security development
 - security hardening
 - security testing

- La diffusione di Malware risulta in continuo aumento. Nel solo anno 2008 su Internet si sono contati circa 15 milioni di malware.
- Banking Malware: sempre più sofisticati. Si aggiornano in base al paese e alle configurazioni del server su cui si installano.
- Obiettivi:
 - Descrivere i comuni problemi di sicurezza nel design per la protezione di siti di banking
 - Fornire best-practice che dovrebbero essere considerate per realizzare soluzioni antimalware

The OWASP Testing Guide v3

Testing Guide history

- January 2004
 - "The OWASP Testing Guide", Version 1.0
- July 14, 2004
 - "OWASP Web Application Penetration Checklist", Version 1.1
- December 25, 2006
 - "OWASP Testing Guide", Version 2.0
- December 16, 2008
 - "OWASP Testing Guide", Version 3.0 – Released at the OWASP Summit 08

Project Complexity



- Review all the documentation on testing:
 - July 14, 2004
 - "OWASP Web Application Penetration Checklist", Version 1.1
- Create a complete new project focused on Web Application Penetration Testing
- Create a reference for application testing
- Describe the OWASP methodology

Nov 2006:

- ▶ Brainstorming for index and templates
- ▶ Write articles using our Wiki model
- ▶ Review articles

Dec 2006:

- ▶ Review all the Guide
- ▶ Write the Guide in doc format

Jan 2007:

- ▶ OWASP Testing Guide Release Candidate 1: 272 pages, 46 tests
- ▶ Feedback and review

Feb 2007:

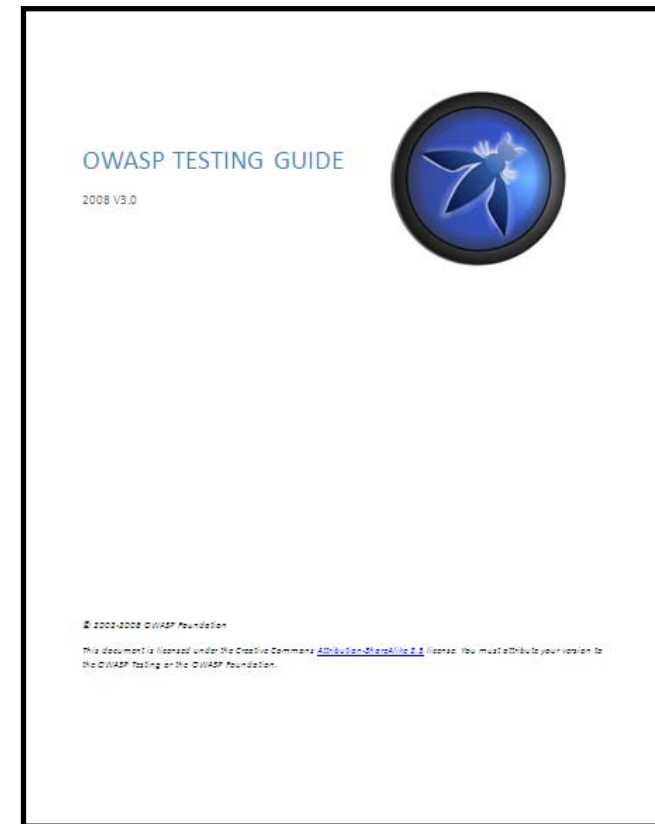
- ▶ OWASP Testing Guide v2 officially released

OWASP Testing Guide v3: roadmap

- 26th April 2008: start the new project
- OWASP Leaders brainstorming
- Call for participation: 21 authors (-18!)
- Index brainstorming
- Discuss the article content
- 20th May 2008: New draft Index
- 1st June 2008: Let's start writing!
- 27th August 2008: started the reviewing phase: 4 Reviewers (-16!)
- October 2008: Review all the Guide
- December 2008: published the new version of the OWASP Testing Guide: http://www.owasp.org/index.php/OWASP_Testing_Project (347pages +80!)



1. Frontispiece
 2. Introduction
 3. The OWASP Testing Framework
 4. Web Application Penetration Testing
 5. Writing Reports: value the real risk
- Appendix A: Testing Tools
- Appendix B: Suggested Reading
- Appendix C: Fuzz Vectors
- Appendix D: Encoded Injection



What's new in v3?

- V2 → 8 sub-categories (for a total amount of 48 controls)
- V3 → 10 sub-categories (for a total amount of 66 controls)
- 36 new articles!

- Information Gathering
- Business Logic Testing
- Authentication Testing
- Session Management Testing
- Data Validation Testing
- Denial of Service Testing
- Web Services Testing
- Ajax Testing

- Information Gathering
- **Config. Management Testing**
- Business Logic Testing
- Authentication Testing
- **Authorization Testing**
- Session Management Testing
- Data Validation Testing
- Denial of Service Testing
- Web Services Testing
- Ajax Testing
- **Encoded Appendix**



- What is a Web Application Penetration Testing?
 - The process involves an active analysis of the application for any weaknesses, technical flaws or vulnerabilities
 - It's a Black Box process (we don't know the source code of the application)
 - Methodology + tools (OWASP WebScarab)
- Our approach in writing this guide
 - Open
 - Collaborative
- Defined testing methodology
 - Consistent
 - Repeatable
 - Under quality

Testing paragraph template

- **Brief Summary**

Describe in "natural language" what we want to test. The target of this section is non-technical people (e.g.: client executive)

- **Description of the Issue**

Short Description of the Issue: Topic and Explanation

- **Black Box testing and example**

- How to test for vulnerabilities:

- Result Expected:

...

- **Gray Box testing and example**

- How to test for vulnerabilities:

- Result Expected:

...

- **References**

- Whitepapers

- Tools

Example

Black Box vs. Gray Box

Black Box

- ✓ The penetration tester does not have any information about the structure of the application, its components and internals

Gray Box

- ✓ The penetration tester has partial information about the application internals. E.g.: platform vendor, sessionID generation algorithm

White box testing, defined as complete knowledge of the application internals, is beyond the scope of the Testing Guide and is covered by the OWASP Code Review Project

Introduction to the methodology

In the next slides we will look at a few examples of tests/attacks and at some real-world cases

Information Gathering

- The first phase in security assessment is of course focused on collecting all the information about a target application.
- Using public tools it is possible to force the application to leak information by sending messages that reveal the versions and technologies used by the application
- Available techniques include:
 - Testing: Spiders, robots, and Crawlers (OWASP-IG-001)
 - Search engine discovery/Reconnaissance (OWASP-IG-002)
 - Identify application entry points (OWASP-IG-003)
 - Web Application Fingerprint (OWASP-IG-004)
 - Application Discovery (OWASP-IG-005)
 - Analysis of Error Codes (OWASP-IG-006)

Example

- SSL/TLS Testing (OWASP-CM-001)
- DB Listener Testing (OWASP-CM-002)
- Infrastructure Configuration Management Testing (OWASP-CM-003)
- Application Configuration Management Testing (OWASP-CM-004)
- Testing for File Extensions Handling (OWASP-CM-005)
- Old, Backup and Unreferenced Files (OWASP-CM-006)
- Infrastructure and Application Admin Interfaces (OWASP-CM-007)
- Testing for HTTP Methods and XST (OWASP-CM-008)

SSL Testing

Example

Session management is a critical part of a security test, as every application has to deal with the fact that HTTP is by its nature a stateless protocol. Session Management broadly covers all controls on a user from authentication to leaving the application

Tests include the following areas:

- Testing for session management scheme
- Testing for cookie attributes
- Session Fixation
- Exposed session variables
- Cross Site Request Forgery

Session Management Testing

Mario Rossi



--Authentication process--



Web
Application

[3] Insert username/password via HTTPS

Credential verify: if ok →
client authenticated
→ Cookie generation

Cookie=TWfYaW8123

Token di
autenticazione

WebScarab - conversation 6

Previous Next 6 - POST https://www.

Parsed Raw

Parsed Raw

```
HTTP/1.0 200 OK
Set-Cookie: Authentication=TWfYaW8123, Expires=
Cache-Control: no-cache
Pragma: no-cache
Content-Type: text/html; charset=utf-8
X-Content-Encoding: gzip
Server:
Date: GMT
X-Cache: MISS from proxy
Proxy-Connection: close

<html>
<head>
```

1st Authentication:

User = Mario Rossi; password=12aB45cD:

Cookie=TWFyaW8123

2nd Authentication :

User = Mario Rossi; password=12aB45cD:

Cookie=TWFyaW8125

3rd Authentication :

User = Mario Rossi; password=12aB45cD:

Cookie=TWFyaW8127

Cookie Guessable: Cookie=TWFyaW8129

Session Management Testing

Mario Rossi



--Following request--



Web Application

Cookie=TWFyaW8179

Authentication Token

[5] Request "movimenti"

Cookie=TWFyaW8179

[6] Send Mario Verdi data

Cookie verify:
TWFyaW8177

Identify user Mario Verdi
Send Mario Verdi data

Mia-Banca - Mozilla Firefox

File Modifica Visualizza Vai Segnalibri Strumenti ?

https://www.mia-banca.it/mov/movimenti.jsp

Mia-Banca

Movimenti: Mario Verdi

| MIBTEL ▲ +5,03% | SPMIB ▲ +0,06% | DOW JONES ▲ +0,08% | NASDAQ ▲ +0,43%

Conto Corrente: 54321 Divisa: EUR

Saldo disponibile: 410.150,11

Data operazione	Data valuta	Importo	Causale
15/10/2004	14/10/2004	-331,05	pagamento pos pagobancomat 54321 del 7/10
17/10/2004	17/10/2004	-500,00	prelevamento da distributore automatico numero 274 carta 54321
23/10/2004	21/10/2004	5.000,00	vostri emolumenti bonifico da VERDI s.r.l.
17/10/2004	16/10/2004	-974,00	pagamento pos pagobancomat 54321 del 13/10 D&G Store
20/10/2004	21/10/2004	-650,00	sottoscrizione titoli 54321678
14/10/2004	12/10/2004	-77,00	pagamento pos pagobancomat 54321 del 12/10 agip petrol
25/10/2004	24/10/2004	-75,00	pagamento pos pagobancomat 54321 del 24/10 distrib.erg mattes ini
31/10/2004	28/10/2004	-1.240,00	disposizione di addebito generica bonifico a Marco VERDI per affitto

Conto Saldo Movimenti Disposizioni Disp. sul Conto Accreditato Bonifici Giroconti Informazioni Attiva Lista

Cookie Manipulation

burp proxy v1.1

intercept options history alerts

Request to http://

forward drop

GET http://
url=Q2Fzbz03JkFkdj0wJkRic3Q9IDM5MzI4MzAxOTU1OSZTaXpIPTE0M
Accept: image/gif, image/x-bitmap, image/jpeg, image/pjpeg, applica
application/vnd.ms-excel, application/msword, application/x-shockwave
Accept-Language: it
Cookie: codeOneShot=51566; msisdnOneShot=3; ;59;
sessionId=A2ASvpbNirh79Vt0u2gwwChq1aXuffq0JC941TRFsQoqCL
80011700211082015782428;
IOLADVACT=ACP0-00-0-00; IOLADVPRF=WCP0000; IOLADVLC=CL
JSESSIONID=A2ASvpbNirh79Vt0u2gwwChq1aXuffq0JC941TRFsQoq
800117002
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 4.0)
Host:
Proxy-Connection: Keep-Alive
Proxy-Authorization: Basic f

burp proxy v1.1

intercept options history alerts

Request to http://

forward drop

GET http://
url=Q2Fzbz03JkFkdj0wJkRic3Q9IDM5MzI4MzAxOTU1OSZTaXpIPTE0MzUw HTTP/1.0
Accept: image/gif, image/x-bitmap, image/jpeg, image/pjpeg, application/vnd.ms-powerpoint,
application/vnd.ms-excel, application/msword, application/x-shockwave-flash, *
Accept-Language: it
Cookie: codeOneShot=51566; msisdnOneShot=3; ;59;
sessionId=A2ASvpbNirh79Vt0u2gwwChq1aXuffq0JC941TRFsQoqCLmF1DVI-855668859I-1062677649I
80011700211082015782428; ; IOLADVID=B155250362;
IOLADVACT=ACP0-00-0-00; IOLADVPRF=WCP0000; IOLADVLC=CLP0000;
JSESSIONID=A2ASvpbNirh79Vt0u2gwwChq1aXuffq0JC941TRFsQoqCLmF1DVI-855668859I-1062677649I
800117002
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 4.0)
Host:
Proxy-Connection: Keep-Alive
Proxy-Authorization: Basic

If we modify the Cookie...

[7] Call the server to bill the user

Charge Sender 3xxxxxxxx99 !!

- **CSRF: When**
 - The application permits to send requests in a not authorized manner or send duplicate requests
 - The application uses implicit authentication (session cookie)
- **CSRF: How**

The attack:

 - A Web site contains an TAG inside the HTML code that runs an action on the target site (TAG has no restriction to origin level)

Testing for CSRF

- 1st Step: find the vulnerable function
 - Create a new user (admin)
 - Fund transfer (users)
- 2nd Step: Force the user to perform that action
 - Malicious Email
 - Malicious site
- 3rd Step: the user will authenticate on the application
 - Browser will be forced to execute an HTTP request
- Result
 - The authorized action will be executed

Testing for Cross Site Request Forgery

First Step

- Online banking. We analyze the transfer fund mechanism
- We notice that after inserting the receiver coordinate and the money amount we will generate the following HTTP GET
- <https://exampleBank.com/transfer?eu=1000&to=1234>

Second Step

- User must be authenticated on the application and forced to go to a malicious site or read an email
- IMG tag will execute the request without user interaction

Testing for Cross Site Request Forgery

Third Step

```
<html>
<title> This is a new interesting site..visiting me </title>
<body>
..

...
</body>
</html>
```

Forth Step

- User browser will execute the action
- There is no way for the application to understand that the action is not forced → log file
- It works!

Authentication testing

Testing the authentication scheme means understanding how the application checks for users' identity and using that information to circumvent that mechanism and access the application without having the proper credentials

Tests include the following areas:

- Credentials transport over an encrypted channel (OWASP-AT-001)
- Testing for user enumeration (OWASP-AT-002)
- Default or guessable (dictionary) user account (OWASP-AT-003)
- Testing For Brute Force (OWASP-AT-004)
- Testing for Bypassing authentication schema (OWASP-AT-005)
- Testing for Vulnerable remember password and pwd reset (OWASP-AT-006)
- Testing for Logout and Browser Cache Management (OWASP-AT-007)
- Testing for Captcha (OWASP-AT-008)
- Testing for Multiple factors Authentication (OWASP-AT-009)
- Testing for Race Conditions (OWASP-AT-010)

- ## User



UserDN: 100

Testing for Broken Authentication (2)

Here is the Authentication POST:

```
POST https://192.168.1.1:1443/AuthenticationServlet HTTP/1.1
Host: 192.168.1.1:1443
...
Referer: https://192.168.1.1:1443/logonDN.jsp
Cookie: IV_JCT=AncDfj8439Fdfjci454;
Content-Type: application/x-www-form-urlencoded
Content-length: 44

login=true&action=MenuCommand&userDN=100
```

userDN is a value contained in the Digital Certificate . Why the Application does not take this information from the digital certificate received (once verified the CA signature and the certificate integrity)?

Authorization Testing

Authorization is the concept of allowing access to resources only to those permitted to use them. Testing for Authorization means understanding how the authorization process works, and using that information to circumvent the authorization mechanism.

Tests include the following areas:

- Testing for path traversal (OWASP-AZ-001)
- Testing for bypassing authorization schema (OWASP-AZ-002)
- Testing for Privilege Escalation (OWASP-AZ-003)

Example

Testing for privilege escalation

Server Response after the user authentication

```
HTTP/1.1 200 OK
Server: Netscape-Enterprise/6.0
Date: Wed, 1 Apr 2006 13:51:20 GMT
Set-Cookie: USER=aW78ryrGuTWs4MnOd32Fs51yDqp; path=/; domain=.dom.it
Set-Cookie: SESSION=k+KmKpHXTgDi1J5fT7Zz; path=/; domain=.dom.it
Cache-Control: no-cache
Pragma: No-cache
Content-length: 247
Content-Type: text/html
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Connection: close
```

Authorization parameter

```
<form name="autoriz" method="POST" action="visual.jsp">
<input type="hidden" name="profile" value="SistInf1">
  <body onload="document.forms.autoriz.submit()">
</td>
```

...

- What if the user modifies the value SistInf1 to SistInf3?

Business logic may include:

- Business rules that express business policy (such as channels, location, logistics, prices, and products); and
- Workflows based on the ordered tasks of passing documents or data from one participant (a person or a software system) to another.

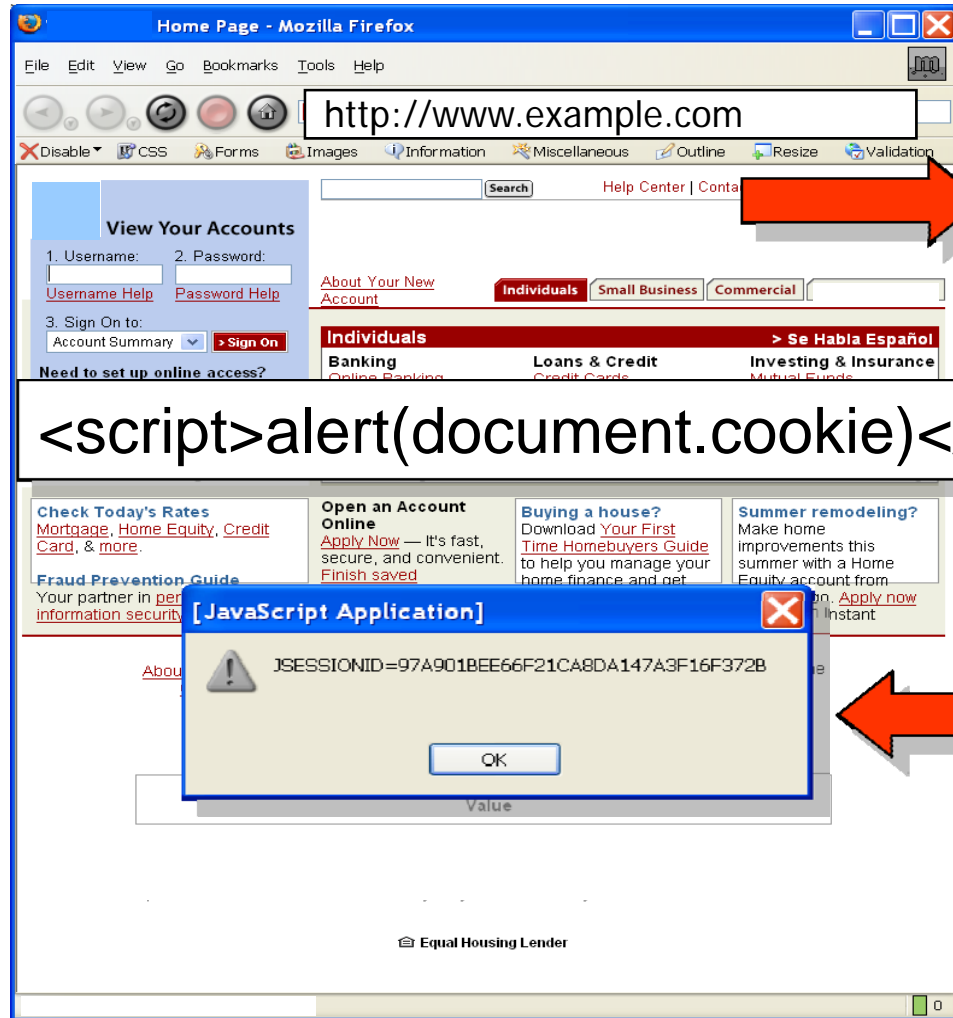
This step is the most difficult to perform with automated tools, as it requires the penetration tester to perfectly understand the business logic that is (or should be) implemented by the application

Data validation testing

In this phase we test that all input is properly sanitized before being processed by the application, in order to avoid several classes of attacks

- **Cross site scripting (Reflected, Stored, DOM, Flashing)**
Test that the application filters JavaScript code that might be executed by the victim in order to steal his/her cookie
- **SQL Injection**
Test that the application properly filters SQL code embedded in the user input
- **Other attacks based of faulty input validation...**
 - LDAP/XML/SMTP/Command injection
 - Buffer overflows

Reflected Cross Site Scripting



Search field print in output the word searched.

```
<script>alert(document.cookie)</script>
```

Site sends the script to the user that see his session cookie.

Stored XSS

Title:

Message:

Could not find message 0

Message List

Title:

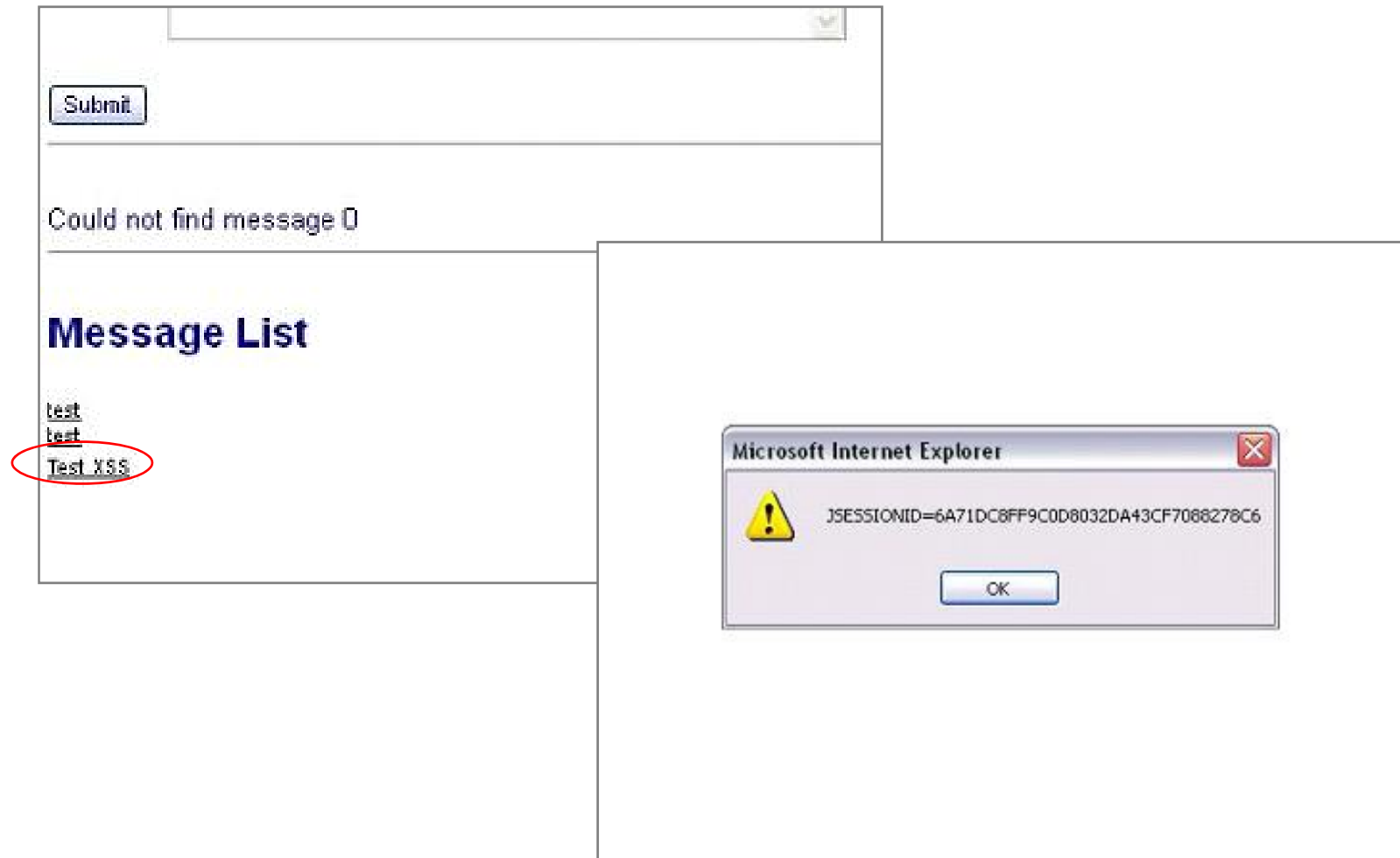
Message:
<Title>Welcome</Title>
Hi <script>alert(document.cookie)</script>

Welcome to our system

Could not find message 0

Message List

Stored XSS (2)



Testing for Command Injection

```
POST http://127.0.0.1:80/WebGoat/attack HTTP/1.1
```

```
Host: 127.0.0.1
```

```
...
```

```
HelpFile=BasicAuthentication.help | dir:
```

```
ExecResults for 'cmd.exe /c type
```

```
"D:\Prog\WebGoat\tomcat\webapps\WebGoat\lesson_plans\"Basic  
Authentication.html | dir c:'
```

```
Output...
```

```
Il volume nell'unit? C ? WinXP
```

```
Numero di serie del volume: 1871-8F02
```

```
Directory di C:\
```

```
27/12/2007 03.51 0 AUTOEXEC.BAT
```

```
27/12/2007 03.51 0 CONFIG.SYS
```

```
18/06/2008 09.54 cygwin
```

```
18/06/2008 11.43 Dev-Cpp
```

```
...
```

Denial of Service Testing

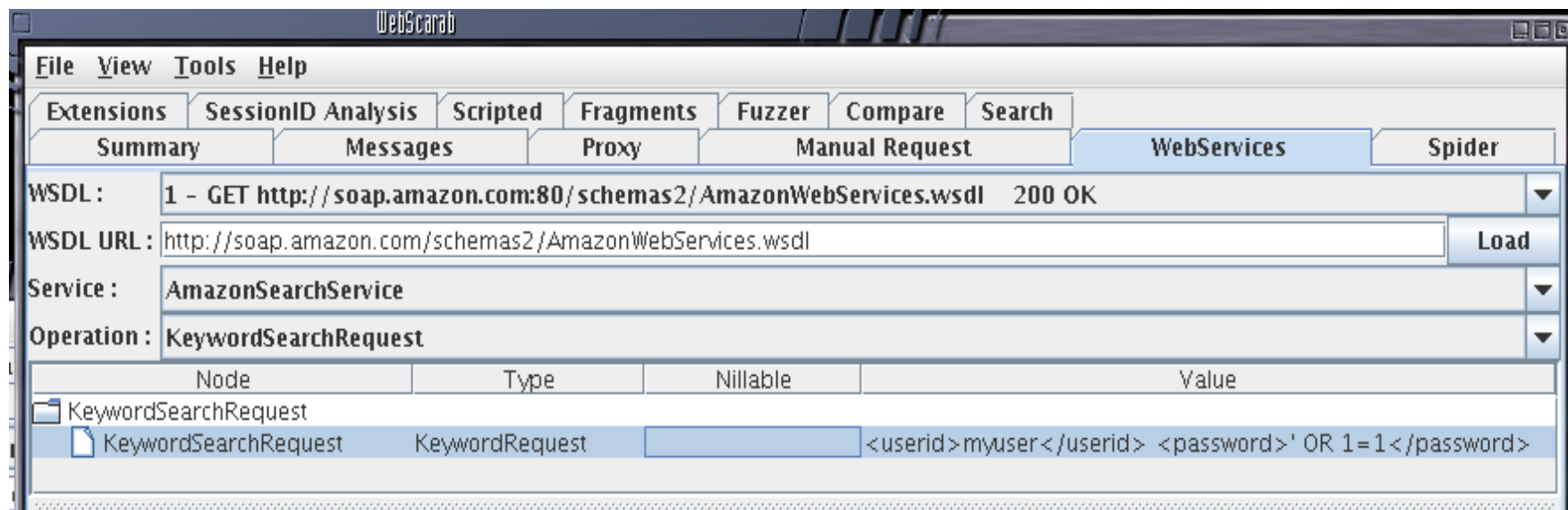
DoS are types of vulnerabilities within applications that can allow a malicious user to make certain functionality or sometimes the entire website unavailable. These problems are caused by bugs in the application, often resulting from malicious or unexpected user input

- Testing for SQL Wildcard Attacks (OWASP-DS-001)
- Locking Customer Accounts (OWASP-DS-002)
- Buffer Overflows (OWASP-DS-003)
- User Specified Object Allocation (OWASP-DS-004)
- User Input as a Loop Counter (OWASP-DS-005)
- Writing User Provided Data to Disk (OWASP-DS-006)
- Failure to Release Resources (OWASP-DS-007)
- Storing too Much Data in Session (OWASP-DS-008)

Usually not performed in performed on production environments

Web Services Testing

- The vulnerabilities are similar to other “classical” vulnerabilities such as SQL injection, information disclosure and leakage etc but web services also have unique XML/parser related vulnerabilities.
- WebScarab (available for free at www.owasp.org) provides a plug-in specifically targeted to Web Services. It can be used to craft SOAP messages that contains malicious elements in order to test how the remote system validates input



- **XML Structural Testing**

In this example, we see a snippet of XML code that violates the hierarchical structure of this language. A Web Service must be able to handle this kind of exceptions in a secure way

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note id="666">
  <to>OWASP
  <from>EOIN</from>
  <heading>I am Malformed </to>
</heading>
<body>Example of XML Structural Test</body>
</note>
```

- **XML Large payload**

Another possible attack consists in sending to a Web Service a very large payload in an XML message. Such a message might deplete the resource of a DOM parser

```
<Envelope>
  <Header>
    <wsse:Security>
      <Hehehe>I am a Large String (1MB)</Hehehe>
      <Hehehe>I am a Large String (1MB)</Hehehe>
      <Hehehe>I am a Large String (1MB)</Hehehe>...
    <Signature>...</Signature>
    </wsse:Security>
  </Header>
  <Body>
    <BuyCopy><ISBN>0098666891726</ISBN></BuyCopy>
  </Body></Envelope>
```


Testing Report: model

- The OWASP Risk Rating Methodology
 - Estimate the severity of all of these risks to your business
 - This is not universal risk rating system: vulnerability that is critical to one organization may not be very important to another
- Simple approach to be tailored for every case
 - standard risk model: **Risk = Likelihood * Impact**
- Identifying a risk

You'll need to gather information about:

 - the vulnerability involved
 - the threat agent involved
 - the attack they're using
 - the impact of a successful exploit on your business.

Testing Report: likelihood

- **Step 2: factors for estimating likelihood**

Generally, identifying whether the likelihood is low, medium, or high is sufficient.

Threat Agent Factors:

- Skill level (0-9)
- Motive (0-9)
- Opportunity (0-9)
- Size (0-9)

Vulnerability Factors:

- Ease of discovery (0-9)
- Ease of exploit (0-9)
- Awareness (0-9)
- Intrusion detection (0-9)

- Step 3: factors for estimating impact

Technical impact:

- Loss of confidentiality (0-9)
- Loss of integrity (0-9)
- Loss of availability (0-9)
- Loss of accountability (0-9)

Business impact:

- Financial damage (0-9)
- Reputation damage (0-9)
- Non-compliance (0-9)
- Privacy violation (0-9)

Testing Report: value the risk

- Step 4: determining the severity of the risk

Threat agent factors				Vulnerability factors			
Skill level	Motive	Opportunity	Size	Ease of discovery	Ease of exploit	Awareness	Intrusion detection
5	2	7	1	3	6	9	2
Overall likelihood=4.375 (MEDIUM)							

Technical Impact				Business Impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability	Financial damage	Reputation damage	Non-compliance	Privacy violation
9	7	5	8	1	2	1	5
Overall technical impact=7.25 (HIGH)				Overall business impact=2.25 (LOW)			

- In the example above, the likelihood is MEDIUM, and the technical impact is HIGH, so from technical the overall severity is HIGH. **But business impact is actually LOW**, so the overall severity is best described as **LOW** as well.

- Deciding What To Fix

As a general rule, you should fix the most severe risks first.

Some fix seems to be not justifiable based upon the cost of fixing the issue but may be reputation damage from the fraud that could cost the organization much more than implement a security control

- Customizing Your Risk Rating Model

- Adding factors
- Customizing options
- Weighting factors

Writing Report

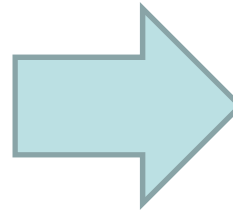
- I. Executive Summary
- II. Technical Management Overview
- III Assessment Findings
- IV Toolbox

Category	Ref. Number	Name	Affected Item	Finding	Comment/Solution	Risk
Authentication Testing	OWASP-AT-003	Bypassing authentication schema				
	OWASP-AT-004	Directory traversal/file include				
	OWASP-AT-005	Vulnerable remember password and <u>pwd</u> reset				
	OWASP-AT-006	Logout and Browser Cache Management Testing				
Session Management	OWASP-SM-001	Session Management Schema				
	OWASP-	Session Token				

Risk Rating Example

- SSL v2
- Error pages
- Access to information not authorized
- Reflected XSS
- Stored XSS
- Session Fixation
- Cross Site Request Forgery

$R=(P,I)$



How the Guide will help the security industry

Pen-testers

- ✓ A structured approach to the testing activities
- ✓ A checklist to be followed
- ✓ A learning and training tool

Clients

- ✓ A tool to understand web vulnerabilities and their impact
- ✓ A way to check the quality of the penetration tests they buy

More in general, the Guide aims to provide a pen-testing standard that creates a 'common ground' between the pen-testing industry and its client.

This will raise the overall quality and understanding of this kind of activity and therefore the general level of security in our infrastructures

Status and Future Steps

- Discuss how to integrate the Develop, Code Review, Testing and ASDR Guide
- Improve Client Side Security
- You should adopt this guide in your organization

Building
Guide

Code Review
Guide

Testing Guide

Application Security Desk Reference (ASDR)

Thanks!

V3 Authors

• Anurag Agarwal	• Kevin Horvath	• Matteo Meucci
• Daniele Bellucci	• Gianrico Ingrosso	• Marco Morana
• Arian Coronel	• Roberto Suggi Liverani	• Antonio Parata
• Stefano Di Paola	• Alex Kuza	• Cecil Su
• Giorgio Fedon	• Pavol Luptak	• Harish Skanda Suredy
• Alan Goodman	• Ferruh Mavituna	• Mark Roxberry
• Christian Heinrich	• Marco Mella	• Andrew Van der Stock

V3 Reviewers

• Marco Cova	• Kevin Fuller	• Matteo Meucci
• Nam Nguyen		

Thank you!



Matteo Meucci
matteo.meucci@owasp.org
matteo.meucci@mindedsecurity.com

References

OWASP Italy:

www.owasp.org/index.php/Italy

OWASP Italy mailing list:

<http://lists.owasp.org/mailman/listinfo/owasp-italy>

OWASP Guidelines:

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

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

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

<http://www.opensamm.org>