OWASP Atlanta
State of the Union

Tony UV – Chapter Lead

A look at what the Atlanta chapter stands to gain from this collaborative consortium of professionals.

OWASP
04.02.09

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

INTRO

Introduction – Who am I?

Tony UV (UcedaVelez), GSEC, CISM, CISA

- OWASP Chapter Lead 2009
- Founder, VerSprite
- Former Sr. Director @ Equifax
- SunTrust ETRM, SecureWorks, Tandberg, Morgan Stanley
- Code Review, Security Architecture, Threat Modeling, Pen Testing, Security Risk Management
- Favorite Drink: Kamikazee
- Favorite [Security] Quote: 'Security is a Process' (Schneier)
- Personal Objective for OWASP-ATL:
  - Become the most prolific security chapter in the Western Hemisphere

Introduction - Why am I here ?

- Passion for security…
- Constructive & Destructive tendencies…
- Enjoy collaborative group think on emerging security tools and research…
- Wife thinks I need a hobby other than playing Chutes & Ladders with the kids…
- OWASP needed a kick in the pants
- Evangelizing security strategy

Introduction – Who are you?

You could be a ….

- Student
- Professor
- Pen Tester
- Developer
- Architect
- Security Engineer
- Risk Analyst
- ISO
- CISO
- CIO, CTO
- Social Butterfly

Introduction - Why should you be here?

- Passion for security…
- Constructive & Destructive tendencies…
- Enjoy collaborative group think on emerging security tools and research…
- Socializing is healthy…
- OWASP needs a kick in the pants
- Increase knowledge on new collaborative tools, methodologies, and papers surrounding WebAppSec
- You don’t get enough security at home
- An growing desire to learn and contribute.

OWASP GLOBAL PERSPECTIVE

- 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, 82 local chapters in the world
- Millions of hits on www.owasp.org

OWASP?

- Provide free resources to the community
  - Publications, Articles, Standards, e.g.
    - OWASP Top 10
    - OWASP Guide
    - Testing Guide
  - Testing and Training Software, e.g.
    - WebGoat
    - WebScarab
    - .NET Projects
  - Local Chapters, Mailing Lists & Conferences
- Dual license model:
  - Open Source Licenses
  - Commercial License for Members

Release Quality Projects | Beta Status Projects | Alpha Status Projects | Inactive Projects

- Release quality projects are generally the level of quality of professional tools or documents.
- Projects are listed below.

Tools

PROTECT:

OWASP AntiSamy Java Project
an API for validating rich HTML/CSS input from users without exposure to cross-site scripting and phishing attacks (Assessment Criteria v1.0)

OWASP AntiSamy .NET Project
an API for validating rich HTML/CSS input from users without exposure to cross-site scripting and phishing attacks. (Assessment Criteria v1.0)

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. (Assessment Criteria v1.0)

DETECT:

OWASP Live CD Project
this CD collects some of the best open source security projects in a single environment. Web developers, testers and security professionals can boot from this Live CD and have access to a full security testing suite. (Assessment Criteria v1.0)

OWASP WebScarab Project
a tool for performing all types of security testing on web applications and web services (Assessment Criteria v1.0)

LIFE CYCLE:

OWASP WebGoat Project
an online training environment for hands-on learning about application security (Assessment Criteria v1.0)

Documentation

PROTECT:

OWASP Development Guide
a massive document covering all aspects of web application and web service security (Assessment Criteria v1.0)

OWASP Ruby on Rails Security Guide V2
this Project is the one and only source of information about Rails security topics. (Assessment Criteria v1.0)

DETECT:

OWASP Code Review Guide
a project to capture best practices for reviewing code. (Assessment Criteria v1.0)

OWASP Testing Guide
a project focused on application security testing procedures and checklists (Assessment Criteria v1.0)

OWASP Top Ten Project
an awareness document that describes the top ten web application security vulnerabilities (Assessment Criteria v1.0)

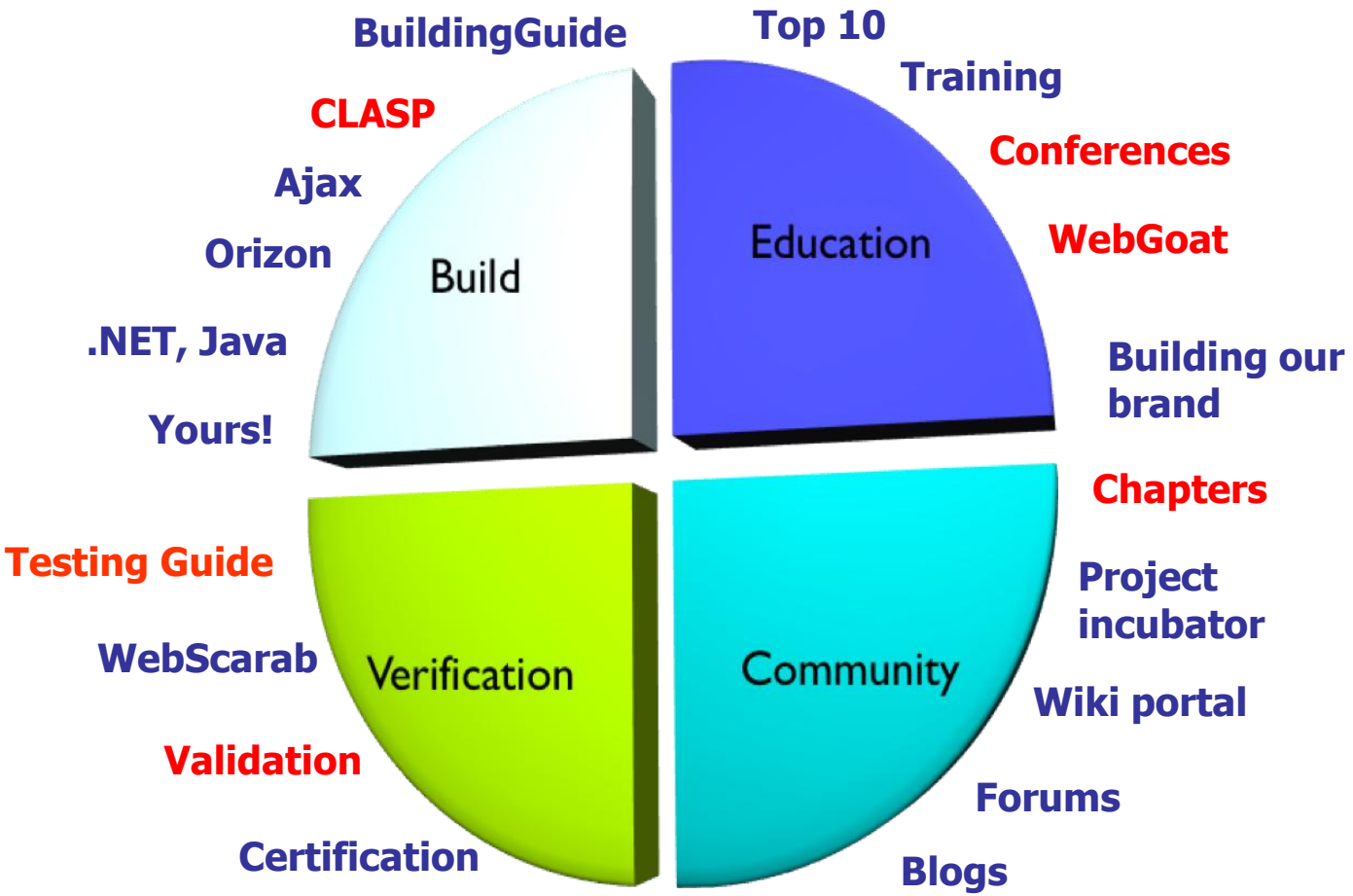
LIFE CYCLE:

OWASP AppSec FAQ Project
FAQ covering many application security topics (Assessment Criteria v1.0)

OWASP Legal Project
a project focused on providing contract language for acquiring secure software (Assessment Criteria v1.0)

OWASP Source Code Review for OWASP-Projects
a workflow for OWASP projects to incorporate static analysis into the Software Development Life Cycle (SDLC). (Assessment Criteria v1.0)

OWASP
The Open Web Application Security Project



Agenda:

- CLASP
- OWASP Testing Guide
- WebGoat
- OWASP Pantera
- OWASP in the ATL
- Special Announcement

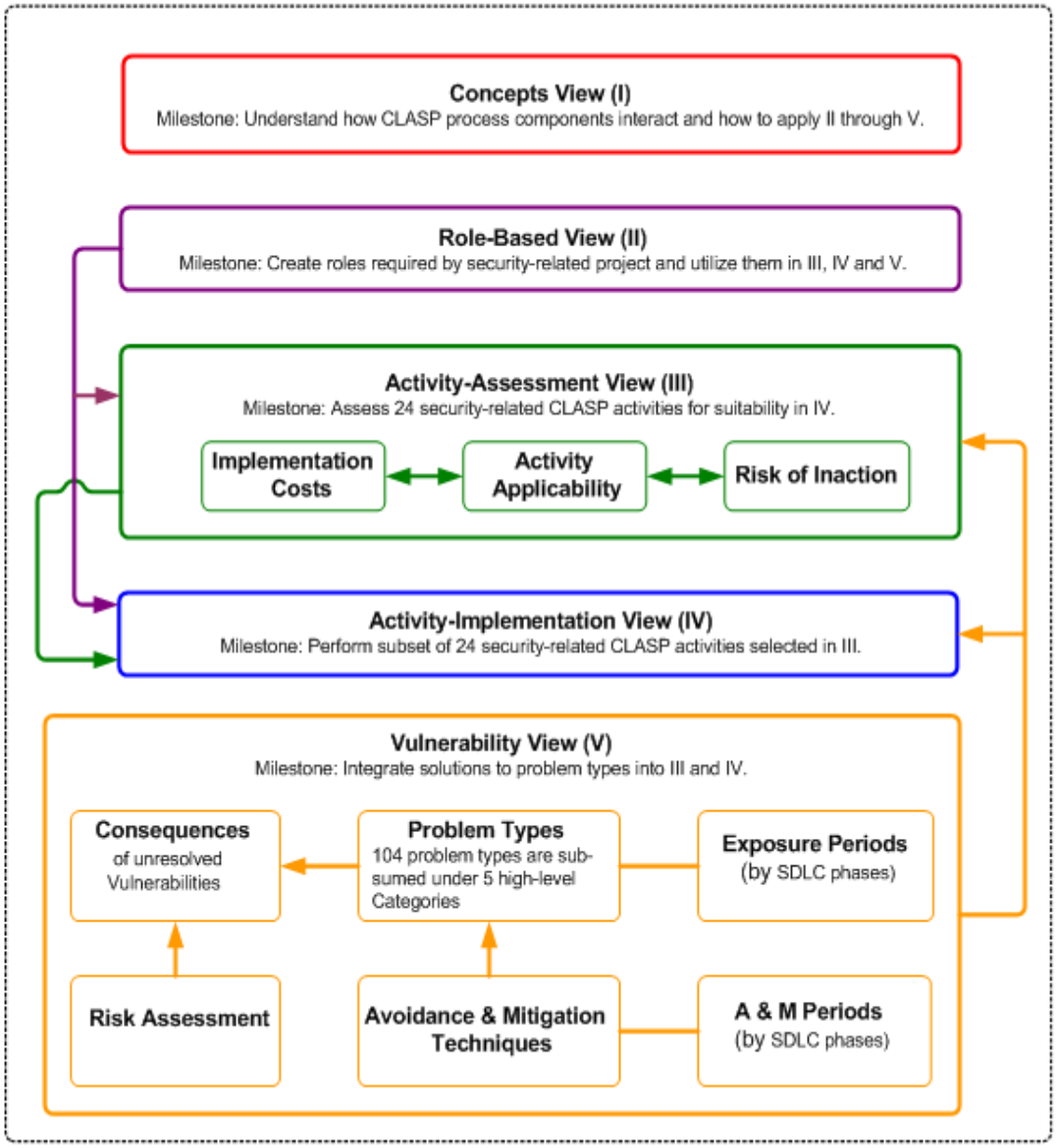
Comic Relief

OWASP CLASP

What Is CLASP and How Do I Catch It?

- Not an STD spreading across OWASP events
- **C**omprehensive, **L**ightweight **A**pplication **S**ecurity **P**rocess
  - Addresses 7 key Ingredients
    1. Security Concepts,
    2. Application Roles,
    3. Activity Assessment,
    4. Activity Implementation
    5. Vulnerabilities,
    6. Use Cases,
    7. Resources
- Integrates into existing enterprise processes:
  - Software development
  - Software assurance group
  - Risk assessment team
- Takes a prescriptive approach, documenting activities that organizations should be doing.Describe the OWASP methodology

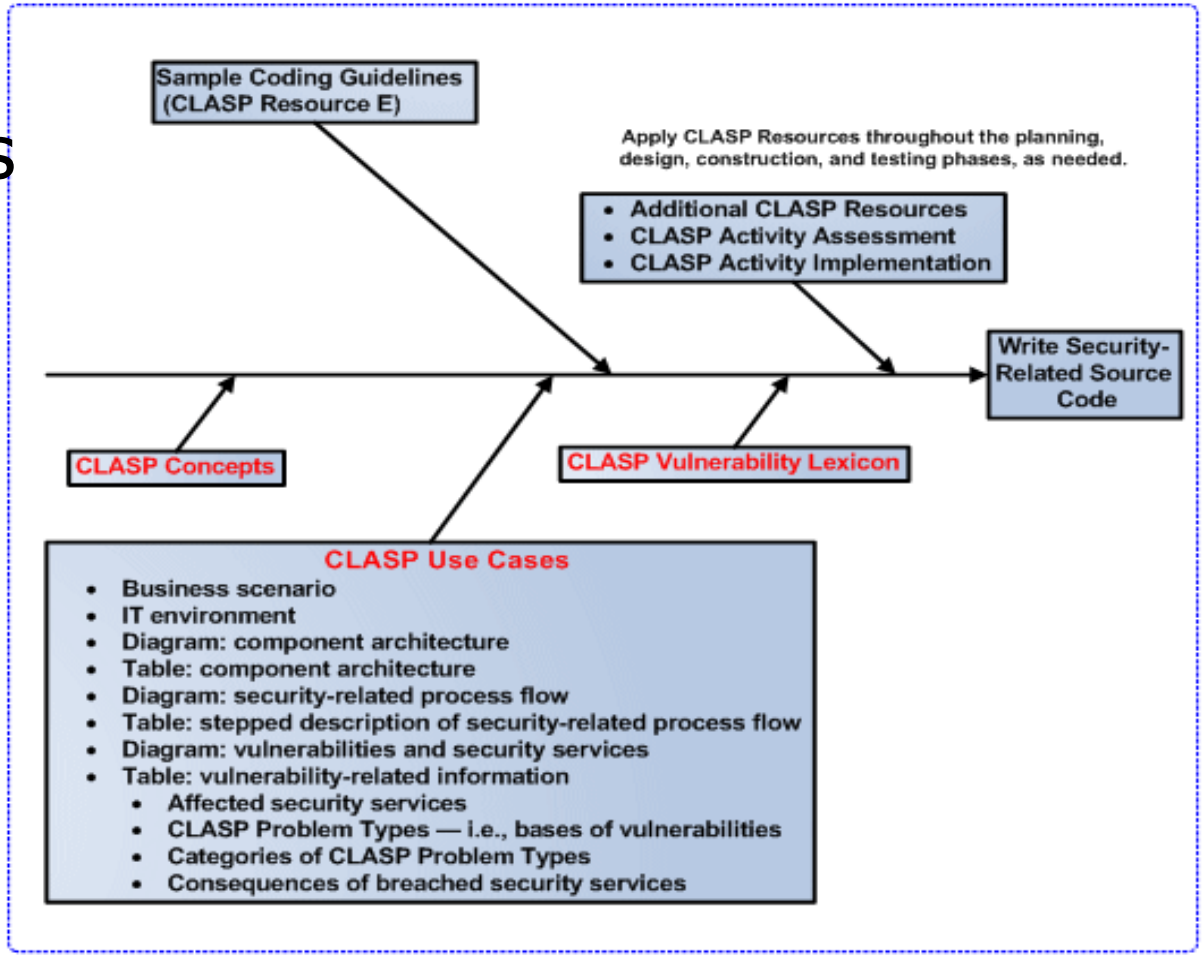
5 Levels of VIEWS & Resources



CLASP Resources	Location
Basic Principles in Application Security (all Views)	Resource A
Example of Basic Principle: Input Validation (all Views)	Resource B
Example of Basic-Principle Violation: Penetrate-and-Patch Model (all Views)	Resource C
Core Security Services (all Views; especially III)	Resource D
Sample Coding Guideline Worksheets (Views II, III & IV) Note: Each worksheet can be pasted into a MS Word document.	Resource E
System Assessment Worksheets (Views III & IV) Note: Each worksheet can be pasted into a MS Word document.	Resource F
Sample Road Map: Legacy Projects (View III)	Resource G1
Sample Road Map: New-Start Projects (View III)	Resource G2
Creating the Process Engineering Plan (View III)	Resource H
Forming the Process Engineering Team (View III)	Resource I
Glossary of Security Terms (all Views)	Resource J

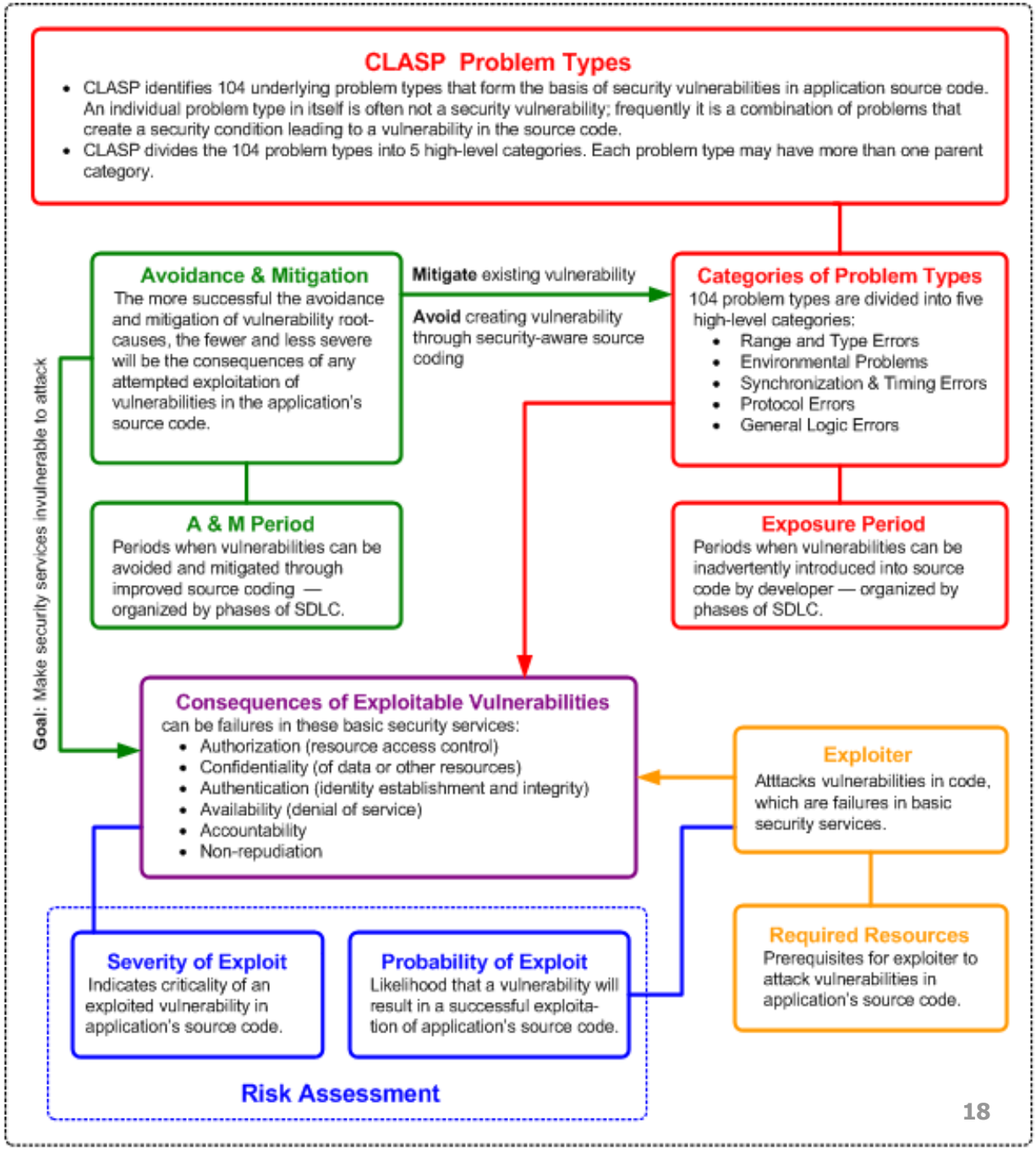
CLASP Use Cases

- Apply secure coding guidelines to use cases in web app
- Correlate use cases to vulnerabilities
- Apply security tools against identified use cases



CLASP Lexicon

- Comprehensive (~220 defintions) taxonomy of vulnerability definitions
- Highly flexible taxonomy enables ease of use
- Can be enforced using today's existing suite of static analysis tools



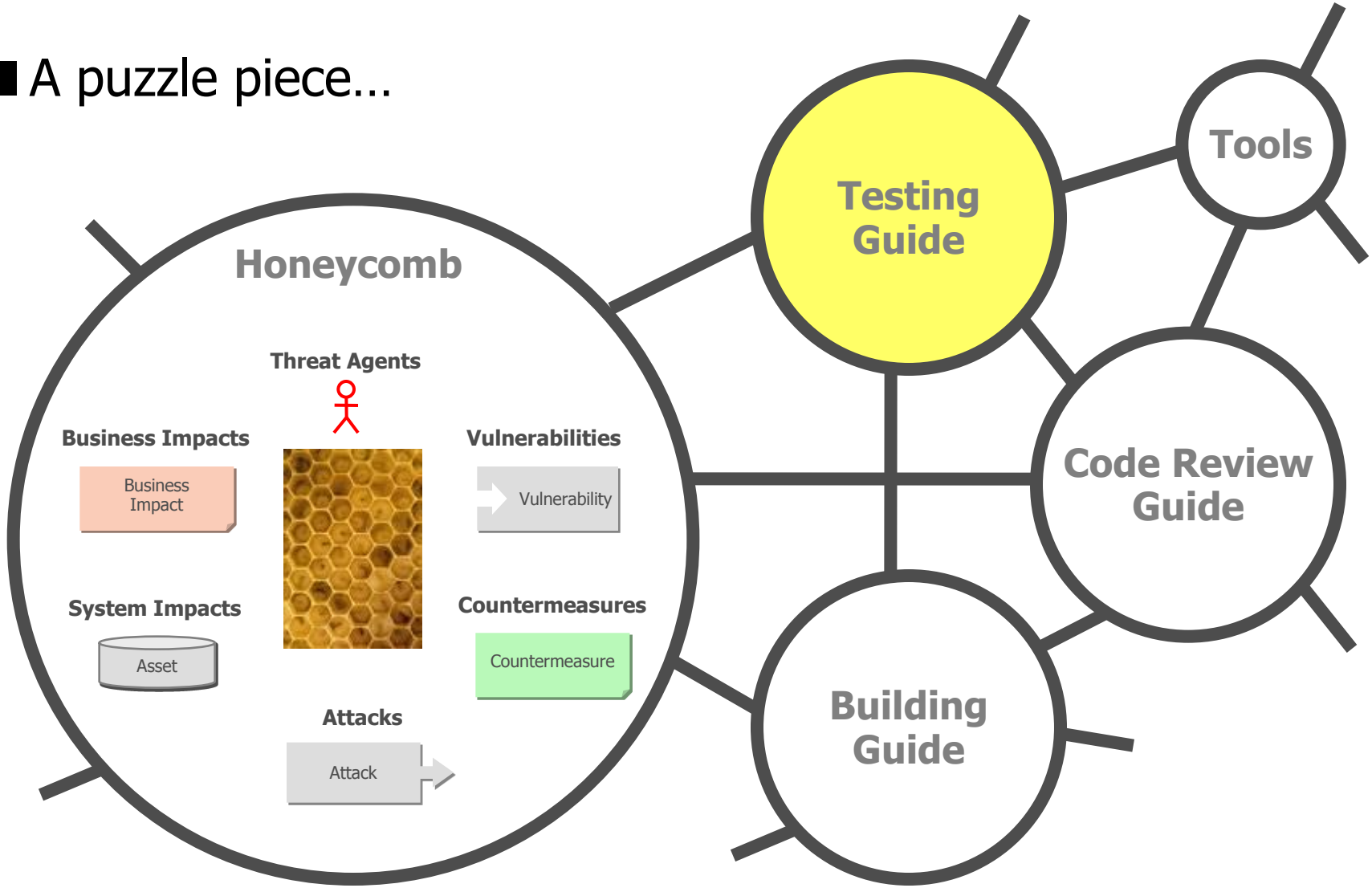
CLASP Summarized

- Stakeholders
  - Read & understand "Concepts View"
  - Read & understand "Role-Based View"
- Project manager
  - Reads and understands "Activity-Assessment View"
  - Determines applicable and feasible "Security Activities" to implement
  - Ties stakeholder roles to "Security Activities"
  - Facilitates "Roles" to learn and execute "Security Activities"
  - Measures progress and holds "Roles" accountable (Metrics)
- Roles (PM, Architect, Designer, Implementer, ...)
  - Execute "Security Activities" leveraging automated tools and CLASP & Organization knowledge base (Vulnerability Lexicon and other Resources)

OWASP TESTING GUIDE

What Is the OWASP Testing Guide?

- A puzzle piece…



OWASP Testing Guide v3: Goals

- "OWASP Testing Guide", Version 3.0 Create a complete new project focused on Web Application Penetration Testing
- Published 11.2008
- Create a complete new project focused on Web Application Penetration Testing
- Create a reference for application testing
- Describe the OWASP Testing methodology

Testing Guide v3: Index

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?

- 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

The OWASP Testing Framework

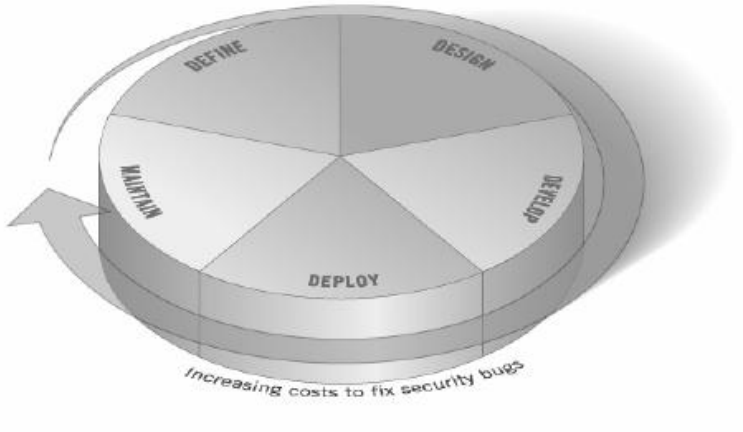
- The problem of insecure software: companies next challenge
- Why OWASP?
  - "It's impossible to underestimate the importance of having this guide available in a completely free and open way"– *Jeff Williams (OWASP Chair)*
- Principles of Testing: comparing the state of something against a set of criteria defined and complete.
  - We want security testing not be a black art
- Testing Techniques:
  - Manual Inspections & Reviews
  - Threat Modeling
  - Code Review
  - Penetration Testing

The OWASP Testing Framework

Phase 1: Before Development Begins

Before application development has started:

- Test to ensure that there is an adequate SDLC where security is inherent.
- Test to ensure that the appropriate policy and standards are in place for the development team.
- Develop Measurement and Metrics Criteria

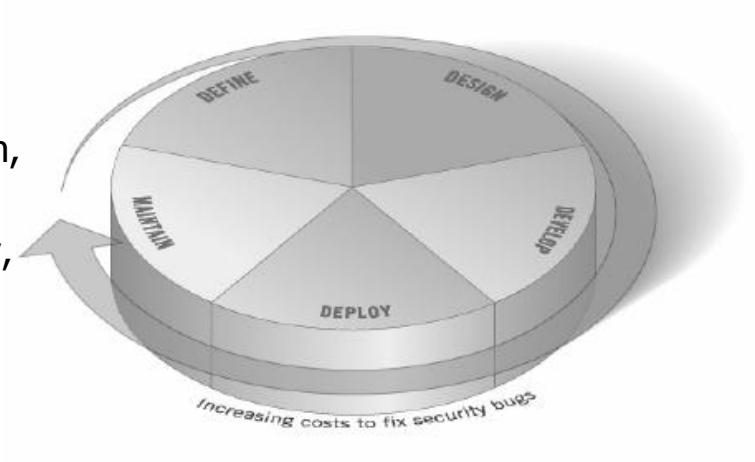


The OWASP Testing Framework

Phase 2: During Definition and Design

Before application development has started:

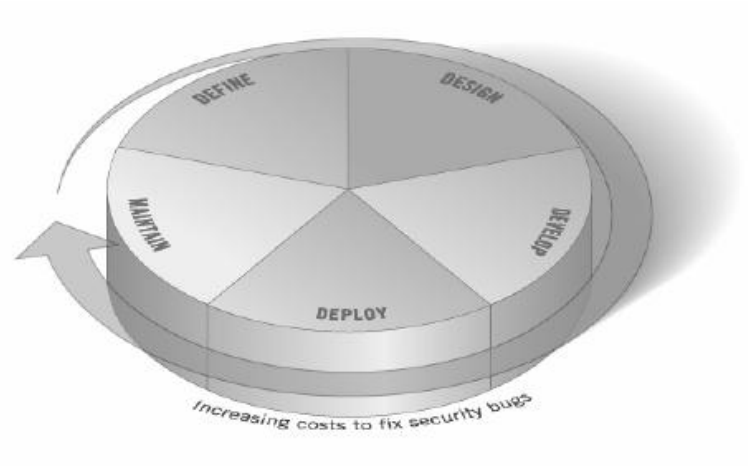
- Security Requirements Review:
  - User Management (password reset etc.), Authentication, Authorization, Data Confidentiality, Integrity, Accountability, Session Management,Transport Security, Privacy
- Design an Architecture Review
- Create and Review UML Models
  - How the application works
- Create and Review Threat Models
  - Develop realistic threat scenarios



The OWASP Testing Framework

Phase 3: During Development

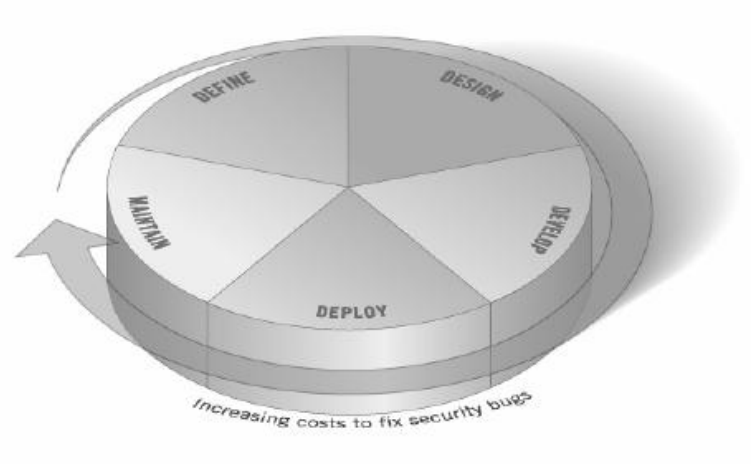
- Code Walkthroughs:
  - high-level walkthrough of the code where the developers can explain the logic and flow.
- Code Reviews:
  - Static code reviews validate the code against a set of checklists:
    - CIA Triad
    - OWASP Top10, OWASP Code Review
    - Sox, ISO 17799, etc…



The OWASP Testing Framework

Phase 4: During Deployment

- Application Penetration Testing
  - Focus of this guide
- Configuration Management Testing
  - The application penetration test should include the checking of how the infrastructure was deployed and secured.



Phase 5: Maintenance and Operations

- Conduct operational management reviews
- Conduct periodic health checks
- Ensure change verification

Web Application Penetration Testing

- What is a Web Application Penetration Testing?
  - The process involves an active analysis of the application for any weaknesses, technical flaws or vulnerabilities
- What is a vulnerability?
  - A weakness on a asset that makes a threat possible
- Our approach in writing this guide
  - Open
  - Collaborative
- Defined testing methodology
  - Consistent
  - Repeatable
  - Under quality

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

Testing Model

- We have split the set of tests in 8 sub-categories (for a total amount of 48 controls):
  - Information Gathering
  - Business logic testing
  - Authentication Testing
  - Session Management Testing
  - Data Validation Testing
  - Denial of Service Testing
  - Web Services Testing
  - AJAX Testing

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:
  - Raw HTTP Connections (netcat)
  - The good ol' tools: nmap, amap, ...
  - Web Spiders
  - Search engines ("Google Dorking")
  - SSL fingerprinting
  - File extensions handling
  - Backups and unreferenced files

Information Gathering (cont.)

- **Application Fingerprint**

  Knowing the version and type of a running web server allows testers to determine known vulnerabilities and the appropriate exploits to use along the tests. Netcat is the tool of choice for this very well known technique

```
$ nc 216.48.3.18 80
HEAD / HTTP/1.0
HTTP/1.1 200 OK
Date: Mon, 16 Jun 2003 02:53:29 GMT
Server: Apache/1.3.3 (Unix)  (Red Hat/Linux)
Last-Modified: Wed, 07 Oct 1998 11:18:14 GMT
ETag: "1813-49b-361b4df6"
Accept-Ranges: bytes
Content-Length: 1179
Connection: close
Content-Type: text/html
```

...But what if the “Server:” header is obfuscated ?

Information Gathering (cont.)

Other hints can be found by sending the server a malformed request, for instance a “GET / HTTP/3.0”

```
HTTP/1.1 400 Bad Request
Date: Sun, 15 Jun 2003 17:12: 37 GMT
Server: obfuscated :P
Connection: close
Transfer: chunked
Content-Type: text/HTML; charset=iso-8859-1
```

Apache 1.3.23

```
HTTP/1.1 505 HTTP Version Not Supported
Server: obfuscated :P
Date: Mon, 16 Jun 2003 06:04: 04 GMT
Content-length: 140
Content-type: text/HTML
Connection: close
```

Netscape Enterprise 4.1

```
HTTP/1.1 200 OK
Server: obfuscated :P
Content-Location: http://target.com/Default.htm
Date: Fri, 01 Jan 1999 20:14: 02 GMT
Content-Type: text/HTML
Accept-Ranges: bytes
Last-Modified: Fri, 01 Jan 1999 20:14: 02 GMT
ETag: W/e0d362a4c335be1: ae1
Content-Length: 133
```

IIS 5.0

...But what if the application simply returns a generic error page ?

Information Gathering (cont.)

The good news is that each server has a favorite way to order headers !

Here are the results for some common web servers when responding to a “HEAD / HTTP/1.0” command:

Apache 1.3.23	**IIS 5.0**	**Netscape Enterprise 4.1**	**SunONE 6.1**
Date	Server	Server	Server
Server	Content-Location	Date	Date
Last-Modified	Date	Content-Type	Content-Length
ETag	Content-Type	Last-Modified	Content-Type
Accept-Ranges	Accept-Ranges	Content-Length	Last-Modified
Content-Length	Last-Modified	Accept-Ranges	
Connection:	ETag	Connection	
Content-Type	Content-Length		

Business logic testing

In this phase, we look for flaws in the application business logic rather than in the technical implementation. Areas of testing include:

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

One of the most common results in this step of the analysis are flaws in the order of actions that a user has to follow: an attacker could perform them in a different order to get some sort of advantage

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

Business logic testing: example

FlawedPhone, a mobile phone operator, has launched a webmail+SMS service:

- New customers, when buying a SIM card, can open a free, permanent webmail account with the flawedphone.com domain
- The webmail account is preserved even if the customer “transfers” the SIM card to another telecom operator
- However, as long as the SIM card is registered to FlawedPhone, each time an email is received an SMS message is sent to the customer
- The SMS application checks that the target phone number is a legitimate customer from its own copy of the FlawedPhone customers list

Nice, but what about the list synchronization ?!

Business logic testing

FlawedPhone was soon targeted by a fraud attack

- The attacker bought a new FlawedPhone SIM card
- The attacker immediately requested to transfer the SIM card to another mobile carrier, which credits 0.05 € for each received SMS message
- When the SIM card was “transferred” to the new provider, the attacker then started sending thousands of emails to her FlawedPhone email account
- The attacker had a 6-8 hours window before the email+SMS application had its list updated and stopped delivering messages
- By that time, the attacker had ~50-100 € in the card, and proceeded to sell it on eBay

All FlawedPhone systems worked as expected, and there were no bugs in the application code. Still, the logic was flawed.

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:

- Default or Guessable Accounts
- Brute-force
- Bypassing Authentication
- Directory Traversal / File Include
- Vulnerable “Remember Password” and Password Reset
- Logout and Browser Cache Management

Session management testing

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:

- Analysis of the session management scheme
- Cookie and session token manipulation
- Exposed session variables
- Cross Site Request Forgery
- HTTP Exploiting

Example: Cross Site Request Forgery

Test if it is possible to force a user to submit an undesirable command to the application he/she is currently logged into

- Also known as "Session Riding" or "Sea Surf"
- Exploits trust between the site and the user (different from XSS which exploits trust between user and site)
- A quite old type of attack, whose impact has always been underestimated
- It relies on the fact that browsers automatically send information used to identify a specific session
- Applications that allow a user to perform some action without requiring some unpredictable parameter are likely to be vulnerable
- ...That means **a lot** of applications!
- All it takes is to trigger the victim to follow a link (e.g.: by visiting an attacker-controlled site) while he/she is logged into the application

Example: Cross Site Request Forgery (cont.)

- trade.com is an online trading company
- trade.com uses an “über-paranoid triple-factor”™ authentication scheme, but does not want to bother users with confirmations, since traders need to act fast!
- A simple website and some social engineering will do the job

```
<html>
<title>I am a very evil HTML page... visit me ! :)</title>
<body>
..
<img src=”https://trade.com/transfer?eu=90000&to=1234” width=”0” height=”0”>
...
</body>
</html>
```

The link triggers a fund transfer

The image is not visible

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**

  Test that the application filters JavaScript code that might be executed by the victim in order to steal his/her cookier

- **HTTP Methods and XST**

  Test that the remote web server does not allow the TRACE HTTP method

- **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/OS injection
  - Buffer overflows

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**

- Step 1: 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)

Testing Report: impact

- 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.

Testing Report: decide what to fix

- Step 5: 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

- Step 6: 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 pwd reset				
	OWASP-AT-006	Logout and Browser Cache Management Testing				
Session Management	OWASP-SM-001	Session Management Schema				
	OWASP-	Session Token				

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

What's next

- You should adopt this guide in your organization
- Continuously reprioritize
- OWASP Testing Guide next steps:
  - Continuously improve the Testing Guide: it's a live document!
  - Contribute to the new version
  - Improve the client side testing

OWASP WEBGOAT

WebGoat

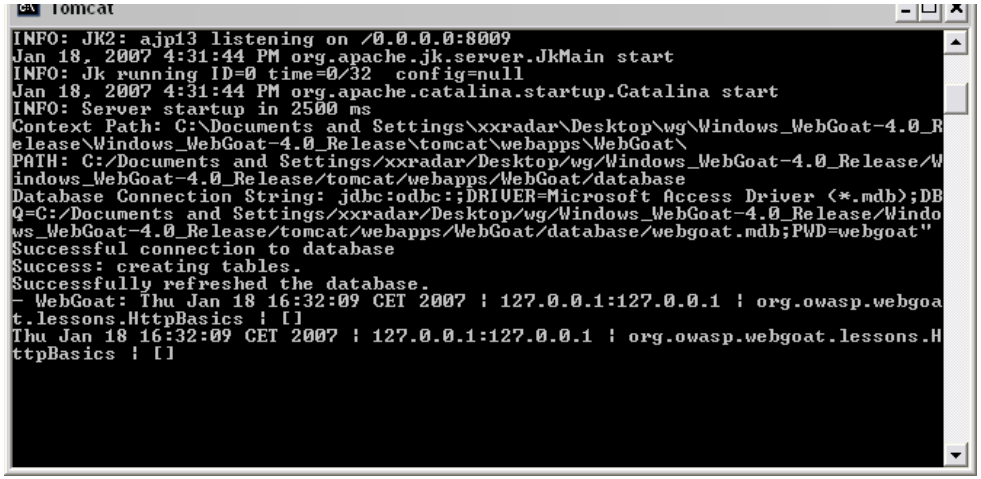
- **WebGoat** is a deliberately insecure J2EE web application maintained by OWASP
- Designed to teach web application security
- … but also useful to test security products
  - IPS, Firewalls, Web Application Firewalls …
    - … against OWASP top 10 promise
    - … against XML and AJAX security threats
- Who already played around with WebGoat ?

Installing WebGoat

- Download available via OWASP project pages]
- Windows and Unix/Linux versions
- Today we are using
  - *WebGoat 5.2 Developer Release*
  - *WebGoat 5.2 Standard*
  - *WebGoat 5.3 to be release very soon…*
- Just unzip the archive and click *webgoat.bat*
  - *Some pitfalls*
    - *Make sure other web servers are stopped*
    - *Skype for some reason dares to use port 80*
    - *Verify with "netstat –an" port 80 is not used*

Connecting the first time

- http://webgoat_server/WebGoat/attack
- login with usn:guest and pwd:guest

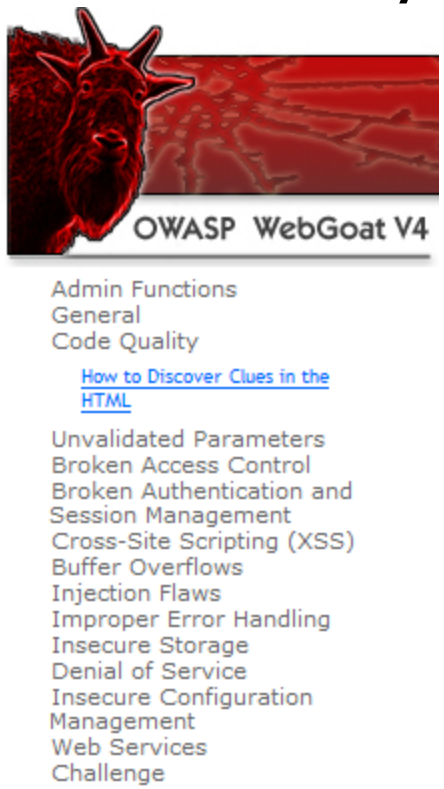


Configuration tuning

- ...Windows_WebGoat-5.2_Release\tomcat\conf\server.xml
  - Port numbers of the web server
- ...Windows_WebGoat-5.2_Release\tomcat\conf\tomcat-users.xml
  - Tomcat usernames, passwords and role

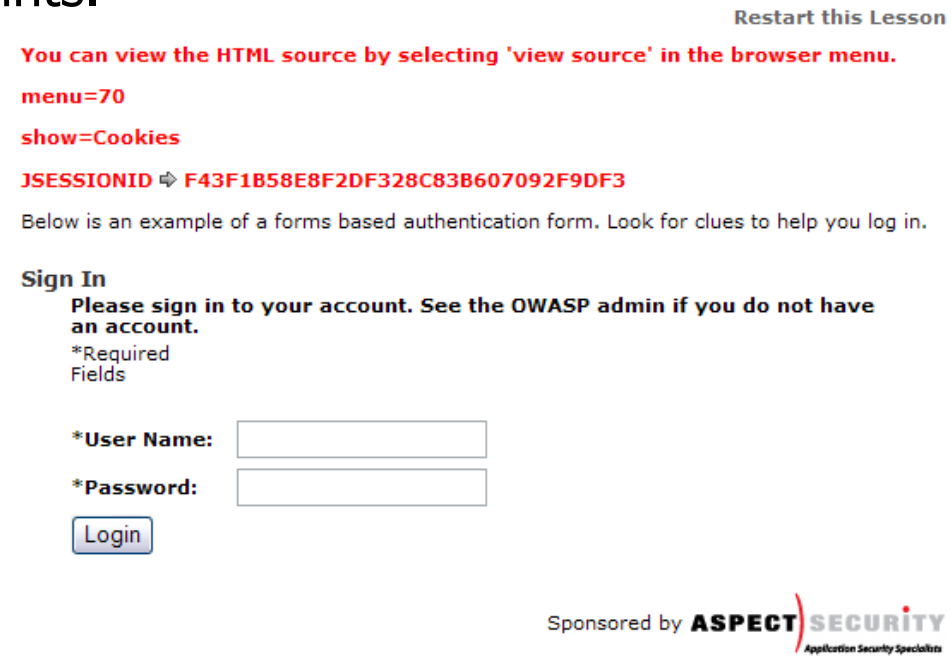
WebGoat V5.2

- A set of lessons and exercises to learn about basic and advanced web application security issues.
  - Intro & WebGoat Instructions
  - Multi-Level Login Lesson
  - Session Fixation Lesson
  - Insecure Login Session
  - Lesson Solution Videos
  - Bug Report Feature
  - Many upgrades & fixes



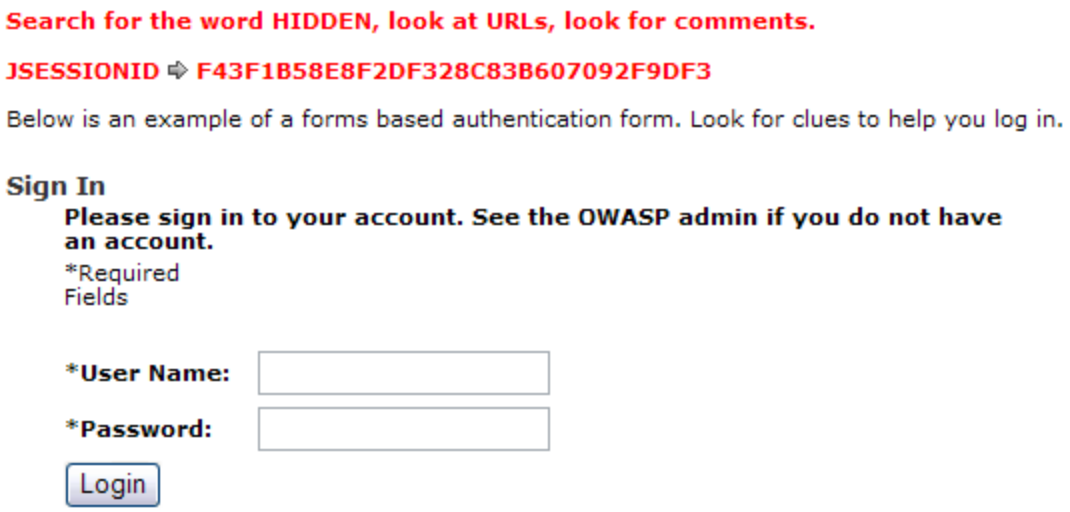
WebGoat is a training tool

- Tools to assist
  - Hints
    - Starting tips up to the solutions of the problem
    - Scroll through the hints.
  - Show Cookies
  - Show Java
  - Show Params
  - Report Card



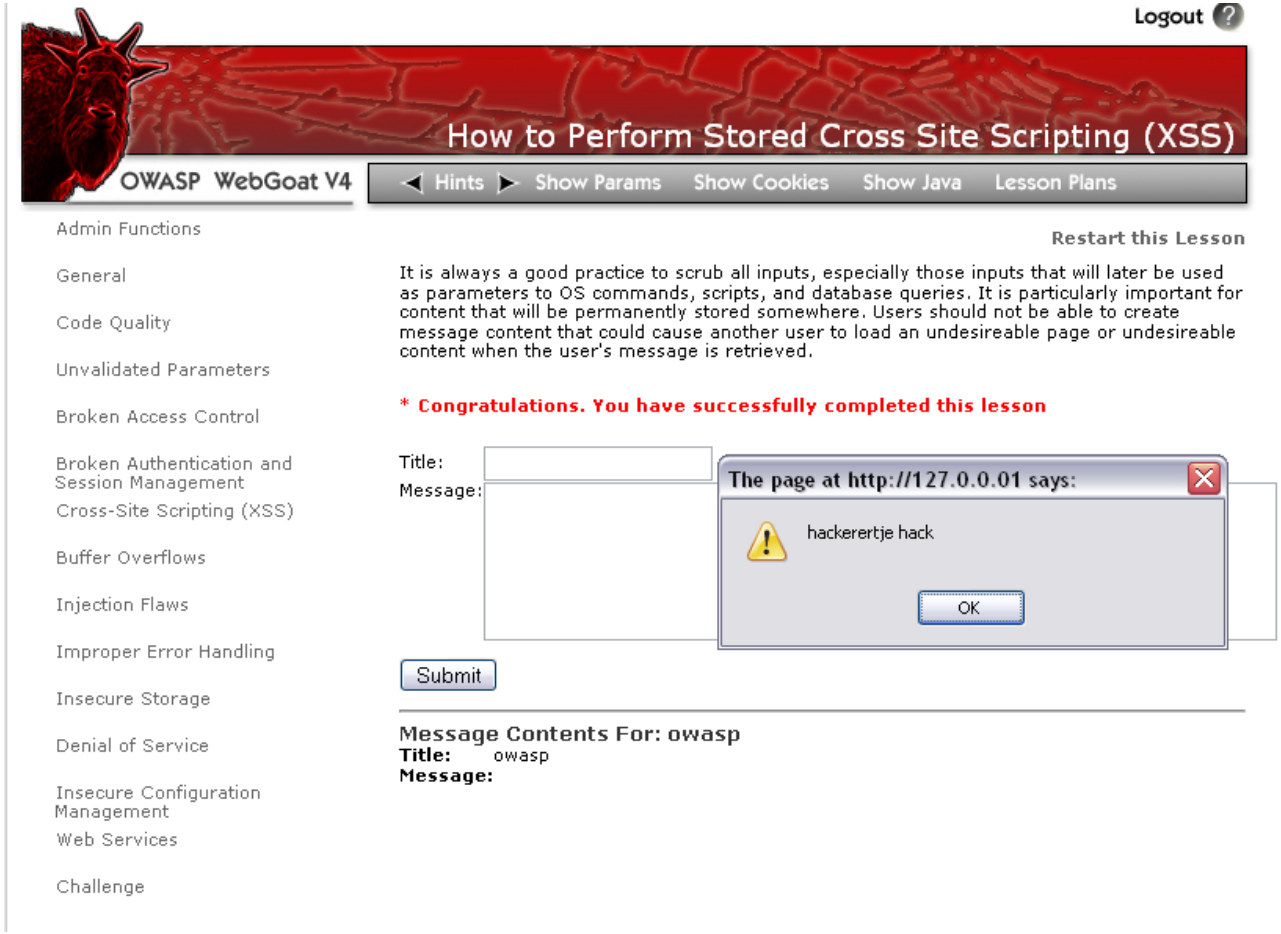
Example 1

- Code Quality
  - Look in the source code
  - Use WebScarab !!!
    - Fragments module



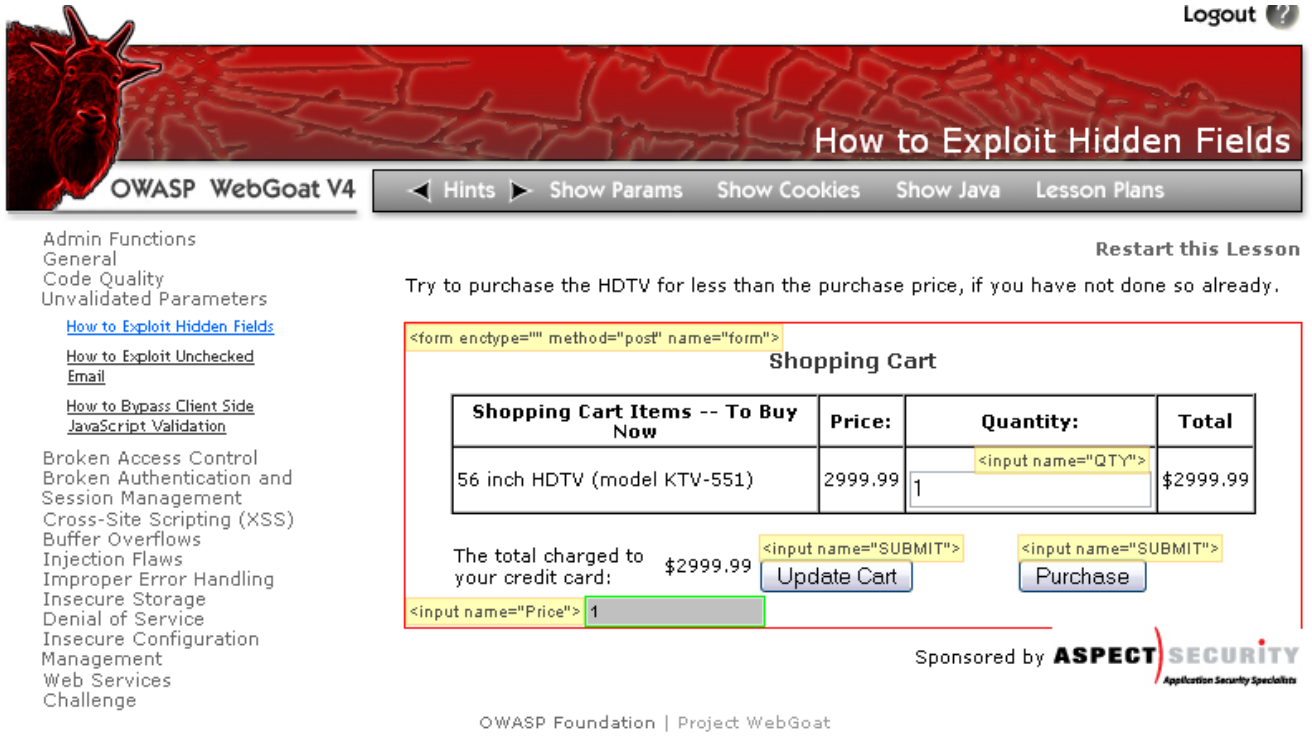
Example 2

- Stored XSS



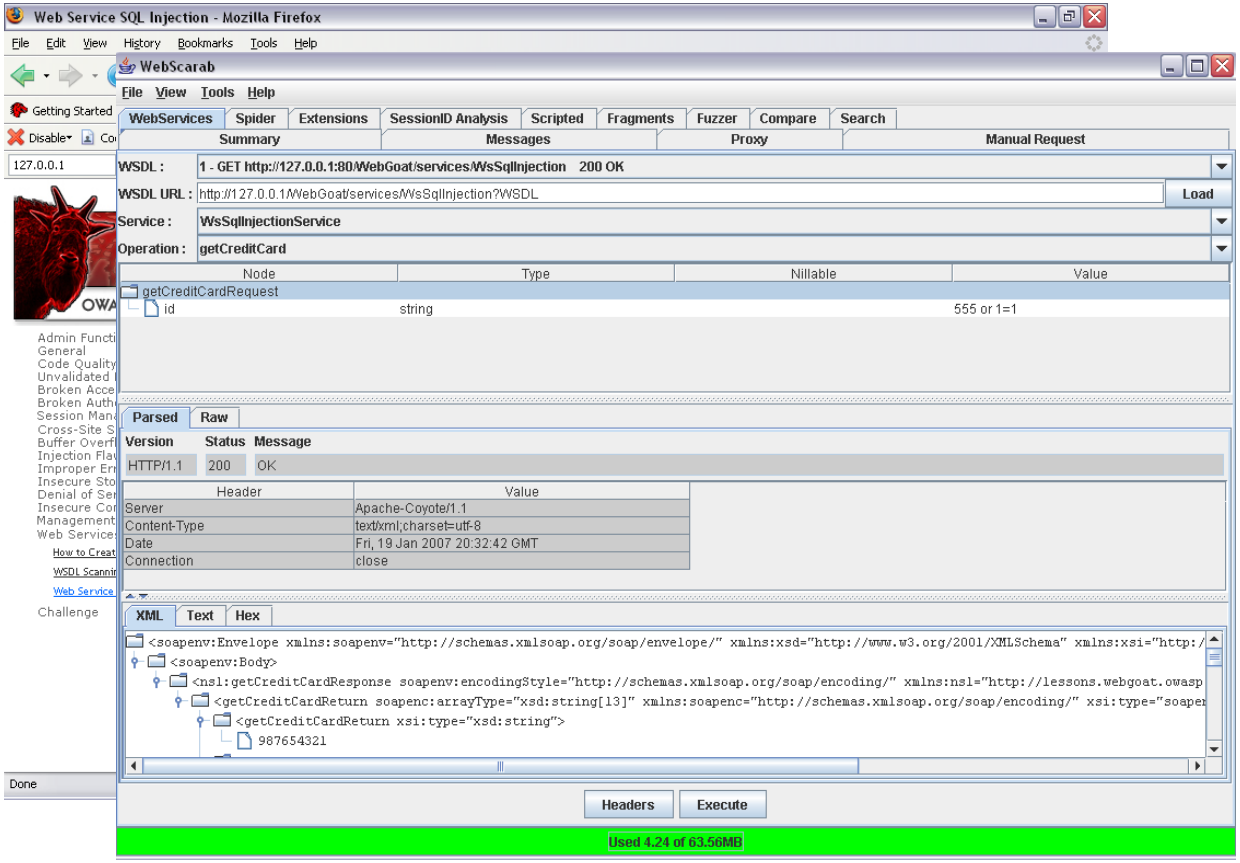
Example 3

- Exploiting Hidden Fields
  - Web Developer plug-in Firefox



Example 4

- Exploiting Web Services with SQL Injection
  - WebScarab

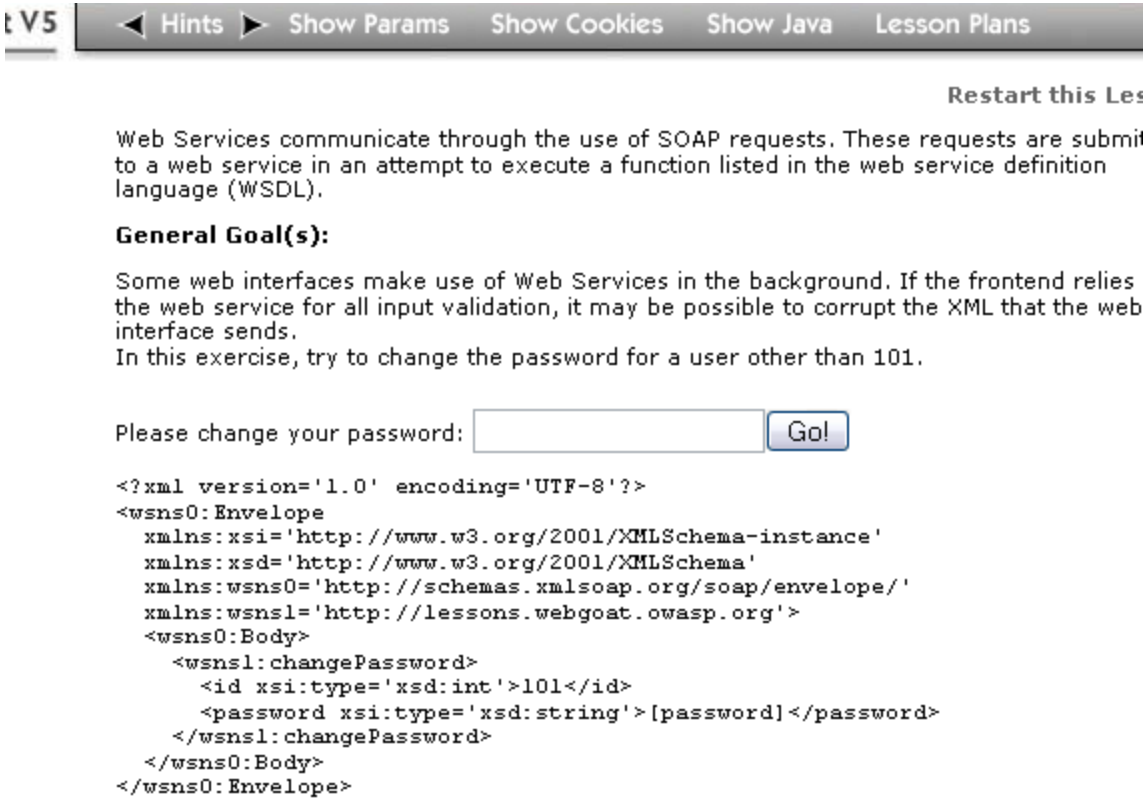


WebGoat V5 (rc1)

- What new ?
  - More XSS
    - Forced Browsing
    - How to Perform CSRF
  - More on SQL Injection
    - Blind SQL Injection
    - XPATH Injection
  - Web Services
    - SAX parser injection
  - AJAX security lessons
  - … and much more

Example 5

- Web Service SAX injection



OWASP IN THE ATL

Atlanta Chapter - What do we have to offer?

- Quarterly Meetings
- Local Mailing List
- Presentations & Groups
- Open forum for discussion
- Meet fellow InfoSec professionals
- Create (Web)AppSec awareness in Belgium
- Local projects
- Beer Socials

Atlanta Chapter - OWASP Membership

- Using OWASP material?
- Join us and become member!
  - Individual Supporter
  - Organizational Supporter
  - Atlanta OWASP Leadership Board
- Support OWASP to continue to provide unbiased:
  - Tools
  - Documentation
  - Conferences
  - Mailing Lists

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

OWASP Local Chapter Meetings 2009

- Next Meeting:
  - Saturday, April 25th, 2009
    - Filter Evasion Workshop
    - Rob Regan, Presenter
    - Location: GA Tech (Most likely Kraus Bldg)
- Meeting Program Formats
  - Short OWASP intro
  - Presentation on introduction topic
  - Panel, workshop, round-table, presentation
  - Sponsor acknowledgement
  - Break for post meeting social
- Topics:
  - Call for input!
  - tonyuv@versprite.com

Atlanta Chapter- Sponsorship

- Local sponsors:
  - Fortify, GA Tech (GTISC)

- Call for additional sponsors
  - Chapter meeting places & catering
  - Support for local projects

- OWASP cannot recommend the use of products, services, or recommend specific companies
  - However, we can acknowledge our sponsors and their contribution to the industry and OWASP

Atlanta Chapter - Comm

- Keep up to date!
  - OWASP Atlanta Chapter Page (http://www.owasp.org/index.php/Atlanta_Georgia)
- Subscribe to BE Chapter mailing list
  - https://lists.owasp.org/mailman/listinfo/owasp-atlanta
- Post your (Web)AppSec questions/ comments
- Contribute to discussions!
  - Join our own IRC channel on EfNet
    - #owasp-atlanta
    - Basicop
    - manEfaces
    - Src

Atlanta Chapter - House Rules

- Free & open to everyone
- Language
  - English preferred
  - Native language: no problem!
- No vendor pitches or $ales presentations
- Respect for different opinions
- No flaming (including M$ bashing)

- 1 CISSP CPE for each hour of OWASP chapter meeting
- Sign Sheet & I'll e-mail scan: you claim CPE credits

Case Study CFP

- OWASP Atlanta Case Study
  - Leverage relationship between OWASP members & local Atlanta based organizations
  - Real world applications of OWASP tools & methodologies
  - Company Incentive: Free FTEs
  - Member Incentive: Do things in your profession/ field of study other than theoretical analysis and compliance reports
  - Proposed topics include:
    - Static Analysis Case Study
    - Threat Modeling Case Study
    - Pen Testing Case Study
  - For more information email: tonyuv@versprite.com
  - Results to be shared amongst local chapter community and other security groups in the ATL
  - Results to be shared globally at other OWASP conferences

SPECIAL EVENT ANNOUNCEMENT

Block your agendas for May 11-14

the Biggest European AppSec event of the year

2 fantastic key notes

Ross Anderson

Professor in Security Engineering

University of Cambridge

Bruce Schneier
Chief Security Technology Officer
BT

3 tracks stuffed with
high quality topics and great speakers

Day 1 - May 13, 2009			
	Track 1: Room 1	Track 2: Room 2	Track 3: Room 3
08:00-08:50	Registration and Coffee		
08:50-09:00	Welcome to OWASP AppSec 2009 Conference *Sebastien Deleersnyder, OWASP Foundation*		
09:00-09:45	Keynote *Ross Anderson, Professor in Security Engineering, University of Cambridge*		
09:45-10:30	OWASP State of the Union *Dinis Cruz & Sebastien Deleersnyder, OWASP Foundation*		
10:30-10:45	Break - Expo - CTF		
10:45-11:25	Wild Wild Wild (www) Security Planet *Mano Paul, SecuRisk Solutions*	Secure Applications for PCI DSS *Tim Holman, QCC Information Security Ltd*	Mirage: building an application model made easy (OWASP Orizon v 1.2) *Paolo Perego, Spike Reply*
11:30-12:10	OWASP Application Security Verification Standard (ASVS) Project *Dave Wichers, Aspect Security*	Securing the .EDU: Application Security for Academia and Education Institutions *Marcus Prendergast, Educational Testing Service*	The Truth about Web Application Firewalls: What the vendors do not want you to know *Wendel Guglielmetti Henrique, Trustwave & Sandro Gauci, EnableSecurity*
12:10-13:30	Lunch - Expo - CTF		
13:30-14:10	The Software Assurance Maturity Model (SAMM) *Pravir Chandra, Cognosticus*	Web Application Harvesting *Esteban Ribičić, tbd*	Refereed Paper Track *Speaker, Organisation*
14:15-14:45	Application Penetration Testing - Client's Perspective *Timo Sivonen, UBS*	Advanced SQL injection exploitation to operating system full control *Bernardo Damele Assumpcao Guimaraes, lead developer of sqlmap*	Refereed Paper Track *Speaker, Organisation*
14:50-15:30	O2 - Advanced Source Code Analysis Toolkit *Dinis Cruz, Ounce Labs*	Tracking the effectiveness of an SDL program: lessons from the gym *Cassio Goldschmidt, Symantec Corporation*	Refereed Paper Track *Speaker, Organisation*
15:30-15:45	Break - Expo - CTF		
15:45-16:25	Exploiting Web 2.0 – Next Generation Vulnerabilities *Shreeraj Shah, Blueinfy*	OWASP Live CD: An open environment for Web Application Security *Matt Tesauro, Texas Education Agency*	Refereed Paper Track *Speaker, Organisation*
16:30-17:30	Panel Discussion *Moderator: tbd , Panelists: tbd*		Refereed Paper Track *Speaker, Organisation*

Day 2 - May 14, 2009			
	Track 1: Room 1	Track 2: Room 2	Track 3: Room 3
08:00-09:00	Registration and Coffee		
09:00-09:45	Keynote *Bruce Schneier, Chief Security Technology Officer, BT*		
09:45-10:30	OWASP Projects *Dave Wichers, OWASP Foundation*		
10:30-10:45	Break - Expo - CTF		
10:45-11:25	Threat Modeling *John Steven, Cigital*	OWASP Source Code Flaws Top 10 Project *Paolo Perego, Spike Reply*	Flash Parameter Injection *Adi Sharabani, IBM*
11:30-12:10	OWASP Enterprise Security API (ESAPI) Project *Dave Wichers, Aspect Security*	w3af, A framework to 0wn the web *Andrés Riancho, tbd*	Brain's hardwiring and its impact on software development and secure software *Alexandru Bolboaca & Maria Diaconu, Mosaic Works*
12:10-13:30	Lunch - Expo - CTF		
13:30-14:10	OWASP "Google Hacking" Project *Christian Heinrich, tbd*	Deploying Secure Web Applications with OWASP Resources *Kuai Hinojosa, New York University*	The Bank in the Browser - Defending web infrastructures from banking malware *Giorgio Fedon, Minded Security*
14:15-14:45	HTTP Parameter Pollution *Luca Carettoni, Independent Researcher & Stefano Di Paola, MindedSecurity*	Leveraging agile to gain better secuity *Erlend Oftedal, Bekk Consulting*	Advanced Code Review Techniques - How to Find Needles in the Haystack Efficiently *Siddharth Anbalahan, Plynt & Jaideep Jha, Plynt*
14:50-15:30	Business Logic Attacks: Bots and Bats *Amichai Shulman, Imperva*	Real Time Defenses against Application Worms and Malicious Attackers *Michael Coates, Aspect Security*	OWASP ROI: Optimize Security Spending using OWASP *Matt Tesauro, Texas Education Agency*
15:30-15:45	Break - Expo - CTF		
15:45-16:25	Factoring malware and organized crime in to Web application security *Gunter Ollmann, IBM*	Can an accessible web application be secure? Assessment issues for security testers, developers and auditors *Colin Watson, Watson Hall Ltd*	I thought you were my friend Evil Markup, browser issues and other obscurities *Mario Heiderich, Business-IN*
16:30-17:30	Panel discussion *Moderator: tbd, Panelists: tbd*		The New Web-Based Man-in-the-Middle Attack *Adi Sharabani, IBM*

Eight Tutorials

Hands on application security with the OWASP Live CD
by Matt Tesauro, Texas Education Agency

Web Services Security
by Dave Wichers, Aspect Security

Advanced Testing
by Michael Coates, Aspect Security

Web Application Security for Managers and Executives – The Road Less Travelled
by Mano Paul, SecuRisk Solutions

Introduction to ModSecurity, the Apache Security Module
by Christian Folini, Netnea

Web 2.0 Hacking – Attacks & Countermeasures
by Shreeraj Shah, Blueinfy

Threat Modeling
by John Steven, Cigital

In-depth Assessment Techniques: Design, Code, and Runtime
by Pravir Chandra, Cognosticus

Krakow @ Day

Krakow @ Night

www.owasp.org/index.php/AppSecEU09

Registrations are

Thank You