Who is OWASP?

Software powers the world, but insecure software threatens safety, trust, and economic growth. The Open Web Application Security Project (OWASP) is dedicated to making application security visible by empowering individuals and organizations to make informed decisions about true application security risks.
The value of volunteerism

- 94% think volunteering adds to the skills of their workforce
- 58% say voluntary work can be more valuable than experience gained in paid employment
- 25% offer paid time off to employee volunteers
- 15% allow sabbaticals for volunteering projects

Employer supported volunteering can help a company’s:
- Reputation and credibility
- Recruitment and staff retention
- Staff morale and work performance
- Training and development
- Change management
- Government and regulatory relations.

*Reed Executive*
16

Years of community service
276 Active Chapters
Active Projects
129+

Government & Industry Citations!
67

Paid Corporate Memberships
2406
Individual Members
2017 Board/Officers

- Chairperson: Matt Konda, Dallas, TX
- Vice Chairperson: Johanna Curiel, Curacao
- Secretary: Tom Brennan, New Jersey, USA
- Treasurer: Andrew van der Stock, Melbourne, Australia
- Board Member: Tobias Gondrom, Hong Kong
- Board Member: Michael Coates, San Francisco, CA, USA
- Board Member: Josh Sokol, Austin, TX
8 Employees
Employees

• Kate Hartmann, Operations Director, NJ - USA
• Matt Tesauro, Senior Project Coordinator, TX - USA
• Tiffany Long, Community Manager, CA - USA
• Kelly Santalucia, Membership and Business Liaison, NJ - USA
• Claudia Casanovas, Project Coordinator, NJ - USA
• Alison Shrader, Accounting, MD – USA
• Laura Grau, Event Manager, CA – USA
• Dawn Aitken, Administrative Assistant, NJ - USA
• Hugo Costa, Graphic Design, (Contractor), Portugal
• ** help wanted!!

https://www.owasp.org/index.php/About_The_Open_Web_Application_Security_Project
OWASP Projects
A QUICK DEVELOPER'S GUIDE
TO OWASP PROJECTS

Learn how to secure your web applications against the most common web vulnerabilities

OWASP
2015

I'm new to application security...where should I start?
- Watch the APPSEC tutorial series to get you started
- OWASP TOP TEN: the classic guidelines
- OWASP Cheat Sheets to get into the stuff without getting annoyed

I want to 'see' vulnerabilities and learn how they happen...
- We have some cool 'vulnerable applications' to learn how you should not code them:
  - Security Shepherd: Great app for understanding vulnerable web apps including lessons
  - WebGoat: OWASP classic JAVA vulnerable site with lessons, all solutions can be found in Youtube videos
  - OWASP Bricks: A PHP vulnerable site with lessons

I want to use pen testing tools to 'hack' my apps and test for vulnerabilities
- OWASP ZAP: an attack proxy, crème de la crème tool for hacking your site
- OWTF: A complete pen testing framework which includes test cases and it's aligned with the latest security standards
- Xenotix Exploit: Indulge into XSS with this tool

Is there a checklist to make sure I don't forget anything?
- OWASP ASVS is 'the list' you can apply to your development process. The OWASP Application Security Verification Standard (ASVS) Project provides a basis for testing web application technical security control
- The Secure Coding Practices Quick Reference Guide is a technology agnostic set of general software security coding practices. In a comprehensive checklist format, that can be integrated into the development lifecycle. At only 17 pages long, it is easy to read and digest.

What about a Developer's Guidelines?
The OWASP Developer Guide is the original OWASP project. It was first published in 2002, when Ajax was only a mote in Microsoft's eye with the now e-mail notification in Outlook Web Access (and only if you used Internet Explorer). Since then, the web has come a long way.

How can I check for vulnerable libraries in my application?
- Keeping up to date with the latest vulnerabilities is not easy, let alone finding them in your dependency libraries. What about a tool that helps you check this automatically?
- Dependency-Check is a utility that identifies project dependencies and checks if there are any known, publicly disclosed, vulnerabilities. Currently Java, .NET, and Python dependencies are supported. This tool can be part of a solution to the OWASP Top 10 2013

OK. Is time to secure my site!
- If you are looking for specific code libraries to protect your application against some nasty vulnerabilities and attacks, here are some great ones:
  - Appsensor: Intrusion detection for your site
  - OWASP HTML Sanitizer is written in Java which lets you include HTML authored by third-parties in your web application while protecting against XSS
  - CSRF Guard: Protect your site against CSRF attacks

I want to analyse my code deeper...
- OWASP has also Guidelines and Static Analysis tools like:
  - Code Review Guidelines: How to check and review your code for common vulnerabilities
  - O2 Platform: Strong Static Analysis tool which can also be a very powerful prototyping and fast-development tool for .NET.

Check more projects
Visit OWASP projects wiki page to learn more about application security:
https://www.owasp.org/index.php/Category:OWASP_Projects#Project_Inventory
Flagship: Strategic value to OWASP, major review process to evaluate candidate projects

As of March 2017:

<table>
<thead>
<tr>
<th>Tools</th>
<th>Code</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>OWASP Zed Attack Proxy (ZAP)</td>
<td>OWASP ModSecurity Core Rule Set (CRS) Project</td>
<td>OWASP Application Security Verification Standard (ASVS) Project</td>
</tr>
<tr>
<td>OWASP Web Testing Environment Project</td>
<td>OWASP CSRFGuard Project</td>
<td>OWASP Software Assurance Maturity Model (SAMM)</td>
</tr>
<tr>
<td>OWASP OWTF</td>
<td>OWASP AppSensor Project</td>
<td>OWASP Top Ten Project</td>
</tr>
<tr>
<td>OWASP Dependency Check</td>
<td>OWASP AppSensor Project</td>
<td>OWASP Testing Project</td>
</tr>
<tr>
<td>OWASP Security Shepherd</td>
<td>OWASP AppSensor Project</td>
<td>OWASP Top Ten Project</td>
</tr>
</tbody>
</table>

https://www.owasp.org/index.php/Category:OWASP_Project#tab=Project_Inventory
T10

OWASP Top 10 Application Security Risks – 2013

A1 – Injection
Injection flaws, such as SQL, OS, and LDAP injection occur when untrusted data is sent to an interpreter as part of a command or query. The attacker’s hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.

A2 – Broken Authentication and Session Management
Application functions related to authentication and session management are often not implemented correctly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to assume other users’ identities.

A3 – Cross-Site Scripting (XSS)
XSS flaws occur whenever an application takes untrusted data and sends it to a web browser without proper validation or escaping, XSS allows attackers to execute scripts in the victim’s browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.

A4 – Insecure Direct Object References
A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, or database key. Without an access control check or other protection, attackers can manipulate these references to access unauthorized data.

A5 – Security Misconfiguration
Good security requires having a secure configuration defined and deployed for the application, frameworks, application server, web server, database server, and platform. Secure settings should be defined, implemented, and maintained, as defaults are often insecure. Additionally, software should be kept up to date.

A6 – Sensitive Data Exposure
Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes. Sensitive data deserves extra protection such as encryption at rest or in transit, as well as special precautions when exchanged with the browser.

A7 – Missing Function Level Access Control
Most web applications verify function level access rights before making that functionality visible in the UI. However, applications need to perform the same access control checks on the server when each function is accessed. If requests are not verified, attackers will be able to forge requests in order to access functionality without proper authorization.

A8 – Cross-Site Request Forgery (CSRF)
A CSRF attack forces a logged-on victim’s browser to send a forged HTTP request, including the victim’s session cookie and any other automatically included authentication information, to a vulnerable web application. This allows the attacker to force the victim’s browser to generate requests the vulnerable application thinks are legitimate requests from the victim.

A9 – Using Components with Known Vulnerabilities
Components, such as libraries, frameworks, and other software modules, almost always run with full privileges. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications using components with known vulnerabilities may undermine application defenses and enable a range of possible attacks and impacts.

A10 – Unvalidated Redirects and Forwards
Web applications frequently redirect and forward users to other pages and websites, and use untrusted data to determine the destination pages. Without proper validation, attackers can redirect victims to phishing or malware sites, or use forwards to access unauthorized pages.
Lab: Projects that have produced a deliverable of value. Expectation of producing releases that are ready for mainstream usage.

Some examples (as of March 2017):

- OWASP WebGoat Project
- OWASP Cornucopia
- OWASP Mobile Security Project
- OWASP Enterprise Security API (ESAPI)
Mobile App Security Requirements and Verification

The OWASP Mobile Application Security Verification Standard (MASVS) is a standard for mobile app security. It can be used by mobile software architects and developers seeking to develop secure mobile applications, as well as security testers to ensure completeness and consistency of test results. The latest release is MASVS v0.9.2.

Mobile App Security Testing Guide

A comprehensive guide for iOS and Android mobile security testers with the following content:

1. Mobile platform internals
2. Testing in the secure development lifecycle
3. Basic white-box and black-box security testing
4. Mobile reverse engineering and tampering
5. Assessing software protections
6. Detailed white-box and black-box test cases that map to the requirements in the MASVS.

The MSTG is a work-in-progress. Currently, we hope to be "feature-complete" in Q2 2017. You can contribute and comment in the GitHub Repository. A book version of the current master branch is available on GitHub.
Incubator: Projects that are still maturing.  
As of March 2017:

**Code [Reviewed January 2017]**
- OWASP Java Encoder Project
- OWASP Java HTML Sanitizer Project
- OWASP Node.js Goat Project
- OWASP Micrometer Framework Project
- OWASP W3C WP7 PHP Project
- OWASP Secure Headers Project *Review Needed
- OWASP Vcure Project *Review Needed
- OWASP DeepViolet TLS/SSL Scanner
- OWASP Off the record 4 Java Project

**Research**
- OWASP WASC Distributed Web Honeypots Project *Review Needed

**Tools [Reviewed last: January 2017]**
- OWASP Benchmark
- OWASP Wordpress Vulnerability Scanner *Review Needed
- OWASP Threat Dragon
- OWASP Faux Bank Project *Review Needed
- OWASP Dismantle *Review Needed
- WAPF Web Application Protection *Review Needed
- OWASP Multilevel 2 Project *Review Needed
- OWASP WebSite Project *Review Needed
- OWASP Pytacker Project
- OWASP Rainbow Maker Project *Review Needed
- OWASP ZSC Tool Project
- OWASP DefectDojo Project
- OWASP _Web Malware Scanner Project
- OWASP Basic Expression Lexicon Variation Algorithms (Belva) Project
- OWASP VBScan
- OWASP Appsec Pipeline
- OWASP Juice Shop Project
- OWASP Bug Logging Tool

**Documentation [Review: May 2015 - Health Check February 2016]**
- OWASP Snake and Ladders Project
- OWASP Automated Threats to Web Applications
- OWASP Vulnerable Web Applications Directory Project
- OWASP NET Project *Review Needed
- OWASP WASC Web Hacking Incidents Database Project *Review Needed
- OWASP Incident Response Project
- OWASP KALP Mobile Project *Review Needed
- OWASP_Secure_Configuration_Guide *Review Needed
- OWASP_Knowledge_Base_Authentication_Performance_Metrics_Project
- OWASP RFFP Criteria *Review Needed
- OWASP Web Mapper Project
- OWASP 10 Fuhr Entwickler *Review Needed
- WASC_OWASP/Web Application Firewall Evaluation Criteria Project
- OWASP Mobile Security Testing Guide
- OWASP Ransomware Guide Project
Participating in a project ...

**Get involved!**

Join the mailing list, get in touch, contact the project leader, test the software / documentation, report bugs, propose features, etc.

OWASP Projects are community driven and most projects are open for anyone motivated to join!
Starting a project ...

Review the OWASP Project Inventory for existing projects

Follow the 2016 OWASP Project Process - Workflow ...

https://www.owasp.org/index.php/Category:OWASP_Project#tab=Starting_a_New_Project
Thank you!