OWASP
Open Web Application
Security Project

# OWASP Top 10 - 2017

**David Caissy**
OWASP Los Angeles Chapter

July 2017

# About Me

**David Caissy**

- Web App Penetration Tester
- Former Java Application Architect
- IT Security Trainer:
  - Developers
  - Penetration Testers

# Agenda

- OWASP Top 10 – 2013
  - Overview
  - Critics
- OWASP Top 10 – 2017
  - Changes
  - Critics

# OWASP Top 10

- OWASP flagship project
- Started by Jeff Williams
- Project Leader: Dave Wichers
- Major Releases in 2004, 2007, 2010 and 2013

# OWASP Top 10

The ten most critical web application security risks

**NOT**

The ten most critical web application security ~~vulnerabilities~~

# OWASP Top 10 – 2013 Overview

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Unvalidated Redirects and Forwards

# OWASP Top 10 – 2013 Critics

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Unvalidated Redirects and Forwards

XSS and CSRF are injection vulnerabilities!

Javascript (client side) different than SQL (server side)?

# OWASP Top 10 – 2013 Critics

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Unvalidated Redirects and Forwards

Should XSS and CSRF be merge together?

# OWASP Top 10 – 2013 Critics

A1 – Injection

A2 – Broken Authentication and Session Management

→ A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Unvalidated Redirects and Forwards

Reflected XSS
Stored XSS
DOM-based XSS

# OWASP Top 10 – 2013 Critics

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Unvalidated Redirects and Forwards

Same thing!!!

# OWASP Top 10 – 2013 Critics

- A1 – Injection
- A2 – Broken Authentication and Session Management
- A3 – Cross-Site Scripting (XSS)
- A4 – Insecure Direct Object References
- A5 – Security Misconfiguration
- A6 – Sensitive Data Exposure
- A7 – Missing Function Level Access Control
- A8 – Cross-Site Request Forgery (CSRF)
- A9 – Using Known Vulnerable Components
- A10 – Unvalidated Redirects and Forwards

Should it be ranked higher?

How are they ranked anyways?

# OWASP Top 10 – 2013 Critics

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

→ A10 – Unvalidated Redirects and Forwards

Don’t see much of these anymore…

# OWASP Top 10 – 2013 Critics

A1 – Injection

→ A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

→ A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

→ A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

→ A10 – Unvalidated Redirects and Forwards

Better wording please?

"unvalidated" is not even in the dictionary!!!

# Did the Top 10 - 2013 catch everything?

# OWASP Top 10 – 2013 Critics

- Missing:
  - Server-Side Request Forgery (SSRF)
  - Lack of security logging
  - Missing detection mechanisms
  - Least privilege
- Probability vs impacts

# OWASP Top 10 – 2017

- March 2017: Release Candidate 1
- Public comment period ended June 30th, 2017
  - Oups, just got extended to August 25th, 2017
- Was August 2017, now late November 2017…

# Changes 2013 → 2017

| OWASP Top 10 – 2013 (Previous) | OWASP Top 10 – 2017 (New) |
|---|---|
| A1 – Injection | A1 – Injection |
| A2 – Broken Authentication and Session Management | A2 – Broken Authentication and Session Management |
| A3 – Cross-Site Scripting (XSS) | A3 – Cross-Site Scripting (XSS) |
| A4 – Insecure Direct Object References - Merged with A7 | A4 – Broken Access Control (Original category in 2003/2004) |
| A5 – Security Misconfiguration | A5 – Security Misconfiguration |
| A6 – Sensitive Data Exposure | A6 – Sensitive Data Exposure |
| A7 – Missing Function Level Access Control - Merged with A4 | A7 – Insufficient Attack Protection (NEW) |
| A8 – Cross-Site Request Forgery (CSRF) | A8 – Cross-Site Request Forgery (CSRF) |
| A9 – Using Components with Known Vulnerabilities | A9 – Using Components with Known Vulnerabilities |
| A10 – Unvalidated Redirects and Forwards - Dropped | A10 – Underprotected APIs (NEW) |

# A7 - Insufficient Attack Protection

- Detecting, responding to and blocking attacks
  - –Makes applications dramatically harder to exploit
  - –Missing in many systems…
  - –Patching librairies takes weeks/months/years

# A7 - Insufficient Attack Protection

**Attack Scenarios:**

1. Scanned by an automated tool
   - Should be detected quickly
2. Manual probing by a skilled attacker
   - Tracking malicious intent
3. Attacker starts exploiting a new vulnerability
   - How quickly can you release a patch?

# A7 - Insufficient Attack Protection

**How to prevent this?**

1. Early attack detection
2. Effective response
3. Quick patch release

# A10 – Underprotected APIs

- Rich clients (browser, mobile, servers) that connect to backend APIs
- Web Services (SOAP/XML, REST/JSON), GWT, AJAX, WebSockets, RPC, …
- Designed to be used by programs (not humans)
- Vulnerable to common attacks
- Limited support from scanners

# A10 – Underprotected APIs

**How to prevent this?**

1. Secured communication channel
2. Strong authentication scheme
3. Proper access control
4. Validate data format
5. Protect against injection attacks

# Other Additions

- Server-Side Request Forgery (SSRF) under A8

*Yes.*

*That’s it.*

# OWASP Top 10 – 2017 Critics/Comments

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

→ A4 – Broken Access Control

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Insufficient Attack Protection

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Underprotected APIs

People seem happy about merging A4 and A7 back together

# OWASP Top 10 – 2017 Critics/Comments

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Broken Access Control

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

→ A7 – Insufficient Attack Protection

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Underprotected APIs

Explicit reference to WAFs

# OWASP Top 10 – 2017 Critics/Comments

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Broken Access Control

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Insufficient Attack Protection

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

→ A10 – Underprotected APIs

Added to help organizations focus on this major emerging exposure

Already covered by the other 9…

# OWASP Top 10 – 2017 Critics/Comments

- Who's choosing what makes it to the Top 10?
- Based on 8 datasets from 7 firms?
- Focus on risks or awareness?
- Needs more emphasis on security logs (personal opinion)

# Conclusion

**Personally:**

- –Happy with the new Top 10
- –Better than the previous version
- –Reflects what I see in my penetration tests

OWASP

Open Web Application Security Project

# Thank you!

Don’t hesitate if you have any question!

dave@notools.net