Welcome to our Session:

Designing Secure Applications to address the "OWASP Top 10" Critical Web Application Risks

- Host: Matt Kunzman
- Information Security Lead Analyst
- Clearwater Analytics
- Slides: https://github.com/littlecodemonkey/presentations

What is the OWASP Foundation

OWASP Foundation

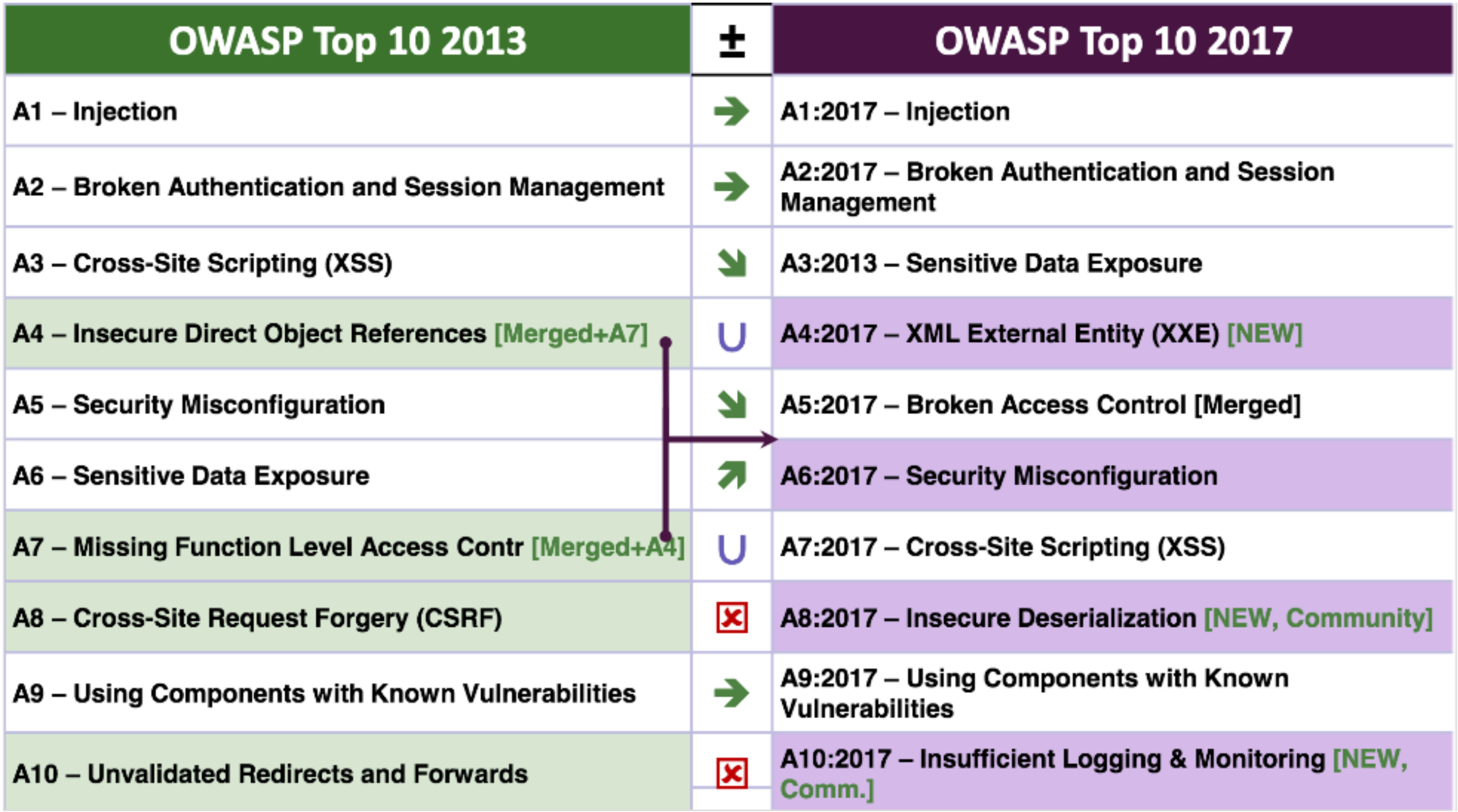
- International organization
- Not for profit
- Established April 21, 2004
- https://www.owasp.org

- Dedicated to enable organizations to create and operate trusted applications.

What is the OWASP Top 10

- A **security awareness** document
- A broad industry consensus about what the most critical web application security flaws are.

OWASP Top 10 2017 RC2



OWASP Top 10 2013	±	OWASP Top 10 2017
A1 – Injection	→	A1:2017 – Injection
A2 – Broken Authentication and Session Management	→	A2:2017 – Broken Authentication and Session Management
A3 – Cross-Site Scripting (XSS)	↘	A3:2013 – Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	∪	A4:2017 – XML External Entity (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017 – Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017 – Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	∪	A7:2017 – Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017 – Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017 – Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017 – Insufficient Logging & Monitoring [NEW, Comm.]

SOURCE: https://github.com/OWASP/Top10/tree/master/2017
COMMIT: 13a119d (Pushed Oct 20, 2017)

#10 Unvalidated Redirects and Forwards (Drops off list in 2017)

- Passing url or part of url in user paramerter
    - http://example.com/example.php?url=http://malicious.example.com
    - http://example.com/example.php?url=accounts
        - Directing to or including accounts.php

#9 Using Components with Known Vulnerabilities

- Known vulnerabilities may or may not have patches
- Usually there’s a CVE and a patch

- Examples
  - Equifax: Apache Struts
  - TJX Companies, Inc.: Weak data encryption during wireless transfers
  - WannaCry: SMBv1 (EternalBlue and DoublePulsar)
  - NotPetya: SMBv1 (EternalRomance) (Same MS March patch)
  - Mumsnet: Heartbleed

#8 Cross Site Request Forgery (CSRF) (Dropped off list in 2017)

- Issuing a request from one site to another
- BAD FIXES
  - Checking the origin header is not reliable
    - Not always present and can be spoofed
- FIXES
  - Anti-CSRF Tokens
  - Same-Site Cookies (Just add SameSite to cookie – Default Strict)
    - **Set**-Cookie: sess=<sessionID>; path=/
    - **Set**-Cookie: sess=<sessionID>; path=/; SameSite
    - **Set**-Cookie: sess=<sessionID>; path=/; SameSite=Strict
    - **Set**-Cookie: sess=<sessionID>; path=/; SameSite=Lax

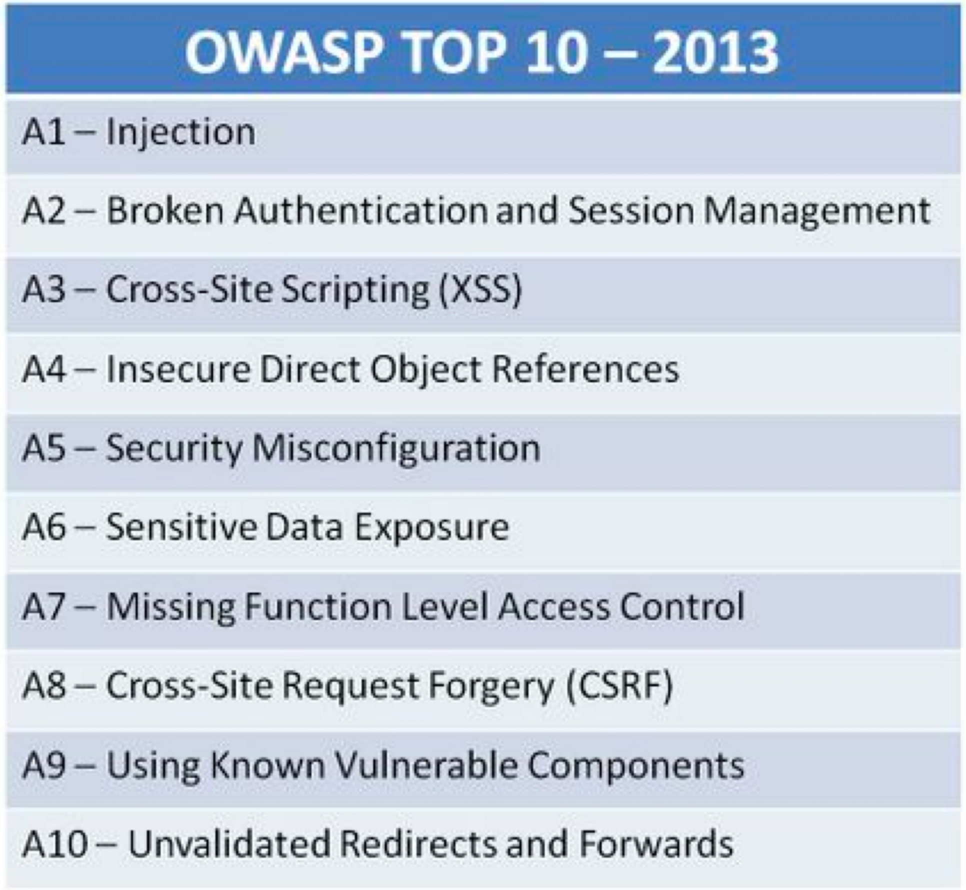
#7 Missing Function Level Access Control

- (Merged with #4 in 2017)
  - Attacker, who is an authorized system user, simply changes the URL or a parameter to a privileged function.

#6 Sensitive Data Exposure

- OWASP Scenerios. (https://www.owasp.org/index.php/Top_10_2013-A6-Sensitive_Data_Exposure)
- **Scenario #1:** An application encrypts credit card numbers in a database using automatic database encryption. However, this means it also decrypts this data automatically when retrieved, allowing an SQL injection flaw to retrieve credit card numbers in clear text. The system should have encrypted the credit card numbers using a public key, and only allowed back-end applications to decrypt them with the private key.
- **Scenario #2:** A site simply doesn’t use SSL for all authenticated pages. Attacker simply monitors network traffic (like an open wireless network), and steals the user’s session cookie. Attacker then replays this cookie and hijacks the user’s session, accessing the user’s private data.
- **Scenario #3:** The password database uses unsalted hashes to store everyone’s passwords. A file upload flaw allows an attacker to retrieve the password file. All of the unsalted hashes can be exposed with a rainbow table of precalculated hashes.

OWASP 2013 – RECAP



OWASP TOP 10 – 2013
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

#5 Security Misconfiguration

- Default Users/Password
- Unused features enabled (services, pages, ports, accounts, privileges)
- Unpatched flaws
- Open ports
- Unprotected Files/Directories
- Error handling revealing stack traces or other info
- Etc.

#4 Insecure Direct Object References

- (Merged with #7 in 2017)

- Attacker changes parameter to refer to a system object the user isn’t authorized for.

- http://example.com/app/accountInfo?acct=notmyacct

#3 Cross Site Scripting (XSS)

- Attacker sends text-based attack scripts that exploit the interpreter in the browser.
    - http://example.com/page.asp?pageid=<script>alert(‘XSS%20attack’)</script>

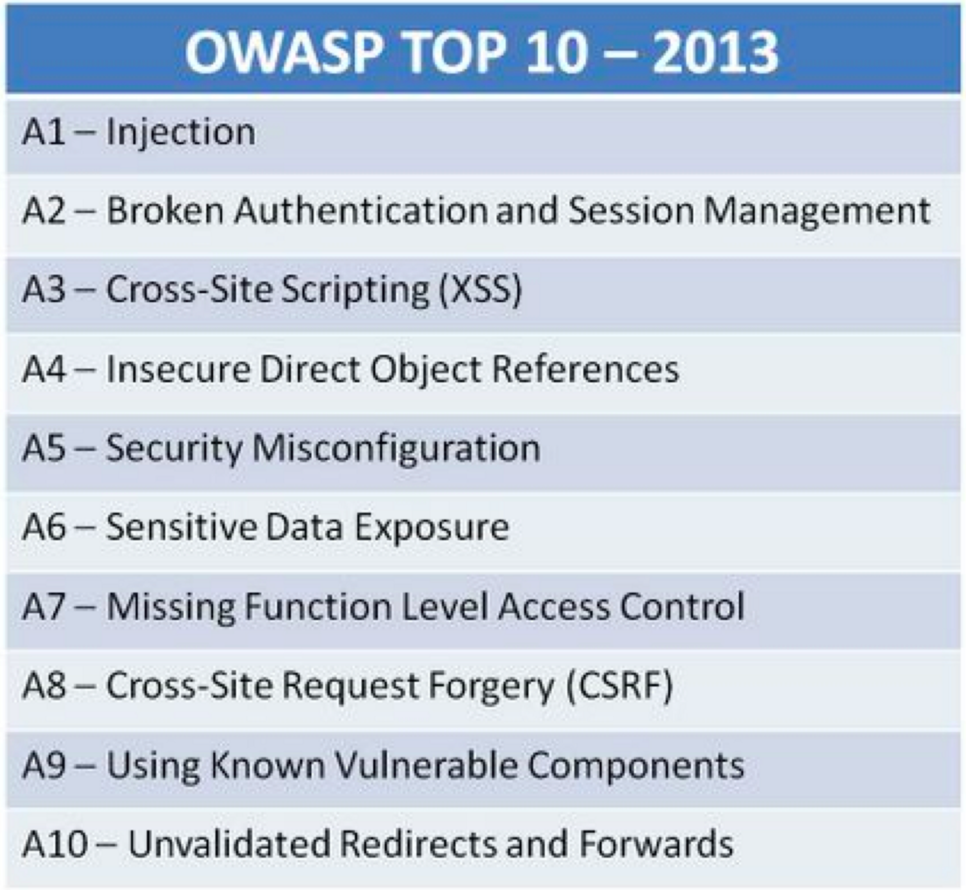
#2 Broken Authentication and Session Management

- Attacker uses leaks or flaws in the authentication or session management functions to impersonate users

#1 Injection

- Attacker sends simple text-based attacks that exploit the syntax of the targeted interpreter.

OWASP 2013 – RECAP



OWASP TOP 10 – 2013
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

Sources

- OWASP Wiki
  - https://www.owasp.org
- OWASP Wikipedia
  - https://en.wikipedia.org/wiki/OWASP
- Github (OWASP 2017 master branch)
  - https://github.com/OWASP/Top10/tree/master/2017
- CSRF is dead!
  - https://scotthelme.co.uk/csrf-is-dead/
- The Web Application Hacker’s Handbook (Second Edition)
  - ISBN 978-1-118-02647-2

- Thank you!

Tintri clearwater® ANALYTICS