Web Application Security

•••

IntraWorlds s.r.o.

Lucie Hartlová, Antonín Neumann, Ondřej Ešler

2017-11-29

Information Security - definition

Information security is the protection of information and systems from unauthorized access, disclosure, modification, destruction or disruption.

The three objectives if information security are:

- CONFIDENTIALITY
- INTEGRITY
- AVAILABILITY

ISO 27001

International

Organization for

Standardization

Provide requirements for establishing, implementing, maintaining and continually improving an information security management system.

ISMS

Topics:

- Human Resources
- Access control
- Cryptography
- Physical and environmental security
- Communications security

and many others

Development Security

- Coding standards
- Security principles
- Penetration testing

OWASP - Open Web Application Security Project

https://www.owasp.org

XSS (Cross-site Scripting)

- victim is the user and not the application
- escaping input vs. output
- use template engine → never forget espacing
  - Twig, Mustache, Plates, Latte, ...
- e.g. `<script>alert(1);</script>`
  - https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet
- helpfully HTTP header
  - `X-XSS-Protection`
  - `Content-Security-Policy`

SQL injection

- unprotected user input
- WHERE, LIMIT, OFFSET
- Defense
  - use some library with prepare statement and binding values
    - PDO (PHP Data Objects)
    - dibi (www.dibiphp.com)
    - Doctrine 2 (www.doctrine-project.org)
    - NotORM (www.notorm.com)
    - Symfony, Zend framewrok, Nette

- `example.com/...&limit=50;update%20users%20set%20name=%27Anonymous%27;`

Directory Traversal

- through an application
  - Attack - http://localhost:8088?download=../../../etc/passwd
  - Defense - `open_basedir`
    http://php.net/manual/en/ini.core.php#ini.open-basedir

- through a webserver
  - **Attack -** http://localhost:8088/docker-compose.yml
  - Defense - `Require all denied`
    https://httpd.apache.org/docs/current/mod/mod_authz_core.html#require

CSRF (Cross-Site Request Forgery)

- GET, POST
- Attack types
  - User assistance (visit attacker page, click on link)
  - link to resource (<img src=”http://example.com/vote?option=1”>)
  - XSS combination (send AJAX via injected javascript)
- Content-Security-Policy
  - Frame-ancestors (previously X-Frame-Options)
- Defense
  - user → critical apps run in a separate browser
  - app → protect action by password or by token
  - code → don’t use $_REQUEST, use $_POST instead

Other security issues

- Sensitive Data Exposure
  - Weak hashes or ciphers
- Weak authentication and session management
  - Only use inbuilt session management
  - Set "secure" and "HttpOnly" flags for session cookies.
- Security Misconfiguration
  - Ensure allow_url_fopen and allow_url_include are both disabled in php.ini
  - Ensure web servers and application servers are hardened
- Using Components with Known Vulnerabilities
  - Hide Server header
  - Disable Apache directives - ServerSignature, ServerTokens, TraceEnable
  - Disable Apache modules - mod_info, mod_dav*, etc.

Thank you

https://github.com/intraworlds/zcu-security-demo