OWASP Top 10 2007
Le nostre informazioni sono "veramente" al sicuro?

Carlo, Pelliccioni
Security Consultant,
@Mediaservice.net

carlo@mediaservice.net

OWASP-Day
Università La Sapienza
Rome
10th September 2007

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

Agenda

- Cos'è OWASP Top 10 2007?
- OWASP Top 10 2004 vs OWASP Top 10 2007
- OWASP Top 10 2007 (deep inside)

Cos'è OWASP Top 10 2007 ?

OWASP Top 10 2007 è un progetto basato sulla classificazione delle 10 vulnerabilità maggiormente critiche estratte dal MITRE Vulnerability Trends 2006, nell'ambito delle Web Application.

L'obiettivo principale è quello di sensibilizzare il mondo dell'IT (e non solo) sulle conseguenze che queste vulnerabilità possono avere sul proprio business, sulla propria privacy e su quella degli altri.

OWASP Top 10 2004 vs OWASP Top 10 2007

A1 2004 Unvalidated Input

Information from web requests is not validated before being used by a web application. Attackers can use these flaws to attack backend components through a web application.

A2 2004 Broken Access Control

Restrictions on what authenticated users are allowed to do are not properly enforced. Attackers can exploit these flaws to access other users' accounts, view sensitive files, or use unauthorized functions.

A3 2004 Broken Authentication and Session Management

Account credentials and session tokens are not properly protected. Attackers that can compromise passwords, keys, session cookies, or other tokens can defeat authentication restrictions and assume other users' identities.

A4 2004 Cross Site Scripting

The web application can be used as a mechanism to transport an attack to an end user's browser. A successful attack can disclose the end user?s session token, attack the local machine, or spoof content to fool the user.

A5 2004 Buffer Overflow

Web application components in some languages that do not properly validate input can be crashed and, in some cases, used to take control of a process. These components can include CGI, libraries, drivers, and web application server components.

A6 2004 Injection Flaws

Web applications pass parameters when they access external systems or the local operating system. If an attacker can embed malicious commands in these parameters, the external system may execute those commands on behalf of the web application.

A7 2004 Improper Error Handling

Error conditions that occur during normal operation are not handled properly. If an attacker can cause errors to occur that the web application does not handle, they can gain detailed system information, deny service, cause security mechanisms to fail, or crash the server.

A8 2004 Insecure Storage

Web applications frequently use cryptographic functions to protect information and credentials. These functions and the code to integrate them have proven difficult to code properly, frequently resulting in weak protection.

A9 2004 Application Denial of Service

Attackers can consume web application resources to a point where other legitimate users can no longer access or use the application. Attackers can also lock users out of their accounts or even cause the entire application to fail.

A10 2004 Insecure Configuration Management

Having a strong server configuration standard is critical to a secure web application. These servers have many configuration options that affect security and are not secure out of the box.

A1 - Cross Site Scripting (XSS)	XSS flaws occur whenever an application takes user supplied data and sends it to a web browser without first validating or encoding that content. XSS allows attackers to execute script in the victim's browser which can hijack user sessions, deface web sites, possibly introduce worms, etc.
A2 - Injection Flaws	Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.
A3 - Malicious File Execution	Code vulnerable to remote file inclusion (RFI) allows attackers to include hostile code and data, resulting in devastating attacks, such as total server compromise. Malicious file execution attacks affect PHP, XML and any framework which accepts filenames or files from users.
A4 - Insecure Direct Object Reference	A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.
A5 - Cross Site Request Forgery (CSRF)	A CSRF attack forces a logged-on victim's browser to send a pre-authenticated request to a vulnerable web application, which then forces the victim's browser to perform a hostile action to the benefit of the attacker. CSRF can be as powerful as the web application that it attacks.
A6 - Information Leakage and Improper Error Handling	Applications can unintentionally leak information about their configuration, internal workings, or violate privacy through a variety of application problems. Attackers use this weakness to steal sensitive data, or conduct more serious attacks.
A7 - Broken Authentication and Session Management	Account credentials and session tokens are often not properly protected. Attackers compromise passwords, keys, or authentication tokens to assume other users' identities.
A8 - Insecure Cryptographic Storage	Web applications rarely use cryptographic functions properly to protect data and credentials. Attackers use weakly protected data to conduct identity theft and other crimes, such as credit card fraud.
A9 - Insecure Communications	Applications frequently fail to encrypt network traffic when it is necessary to protect sensitive communications.
A10 - Failure to Restrict URL Access	Frequently, an application only protects sensitive functionality by preventing the display of links or URLs to unauthorized users. Attackers can use this weakness to access and perform unauthorized operations by accessing those URLs directly.

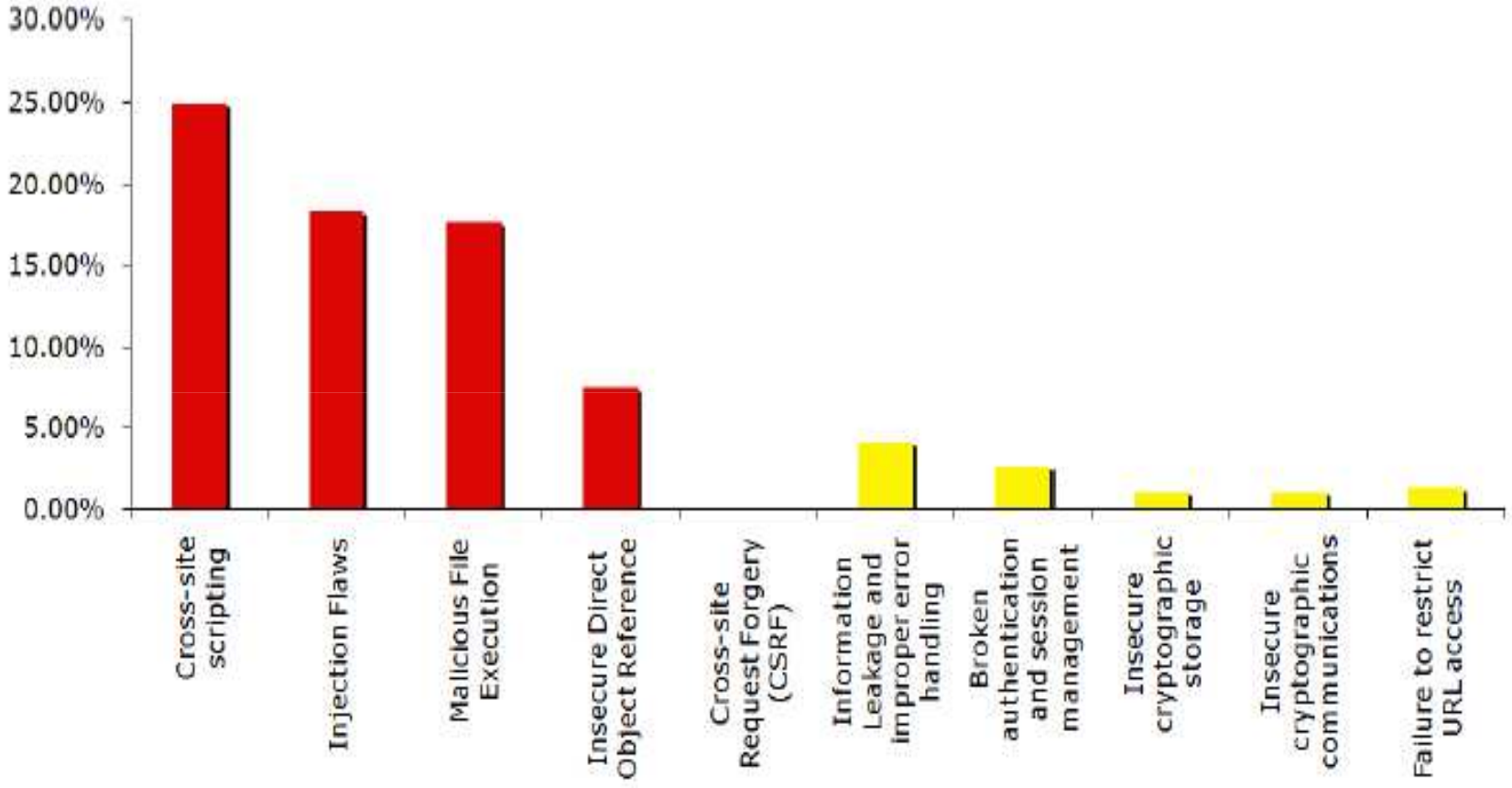
OWASP Top 10 2004 vs OWASP Top 10 2007

- Cosa è stato aggiunto?
  - Malicious File Execution
  - Cross Site Request Forgery (CSRF)
  - Insecure Communications

OWASP Top 10 2004 vs OWASP Top 10 2007

- Cosa è stato rimosso?
  - Unvalidated Input
  - Buffer Overflows
  - Denial of Service
  - Insecure Configuration Management

OWASP Top 10 2007 (deep inside)



OWASP Top 10 2007 (deep inside) – (1/10)

- Cross Site Scripting (XSS)

Una vulnerabilità Cross Site Scripting (XSS) si verifica quando un'applicazione riceve dati forniti dall'utente senza effettuare alcun tipo di validazione del contenuto; nel restituire tali dati all'utente essi vengono interpretati dal browser consentendo l'esecuzione di codice malevolo lato client.

OWASP Top 10 2007 (deep inside) – (2/10)

- Injection Flaws

Le Injection Flaws sono vulnerabilità relative all'invio da parte di un agente di minaccia di input non validato e inviato ad un interprete come comandi di sistema o query SQL.

OWASP Top 10 2007 (deep inside) – (3/10)

- Malicious File Execution

Applicazioni vulnerabili a Malicious File Execution consentono l'upload di file contenente codice malevolo lato server in grado di eseguire comandi di sistema e di conseguenza compromettere a livello globale il sistema.

OWASP Top 10 2007 (deep inside) – (4/10)

- Insecure Direct Object Reference

Un'Insecure Direct Object Reference si verifica quando nell'applicazione sono presenti elementi liberamente manipolabili (file, directory, record database) che consentono di risalire ad ulteriori informazioni che possono portare ad un'escalation di privilegi o all'acquisizione di dati critici.

OWASP Top 10 2007 (deep inside) – (5/10)

- Cross Site Request Forgery (CSRF)

  Un attacco di tipo Cross Site Request Forgery è mirato a forzare il browser di una vittima ad eseguire operazioni da lui non espressamente richieste sfruttando vulnerabilità applicative come una non corretta implementazione delle sessioni ed un passaggio dei parametri e dei rispettivi valori di una richiesta attraverso l'uso del metodo GET.

OWASP Top 10 2007 (deep inside) – (6/10)

- Information Leakage and Improper Error Handling

  Information Leakage e Improper Error Handling si verificano spesso in presenza di web application non correttamente sviluppate che consentono l'acquisizione d'informazioni utilizzabili in un secondo momento da un agente di minaccia per effettuare un attacco mirato all'applicazione o al sistema server.

OWASP Top 10 2007 (deep inside) – (7/10)

- Broken Authentication and Session Management

Broken Authentication e Sessionion Management sono vulnerabilità riconducibili ad un'errata implementazione della gestione delle credenziali di accesso e/o delle chiavi di sessioni che rendono possibili accessi non autorizzati all'applicazione tramite l'impersonificazione di altre identità vittima.

OWASP Top 10 2007 (deep inside) – (8/10)

- Insecure Cryptographic Storage

L'Insecure Cryptographic Storage è una vulnerabilità causata dalla non corretta conservazione dei dati e dall'errata implementazione delle funzioni applicative create allo scopo di proteggere tali informazioni.

OWASP Top 10 2007 (deep inside) – (9/10)

- Insecure Communications

Le Insecure Communications consentono ad un agente di minaccia di intercettare informazioni come credenziali di accesso e informazioni riservate in transito su canali privi di cifratura (plaintext).

OWASP Top 10 2007 (deep inside) – (10/10)

- Failure to Restrict URL Access

Failure to Restrict URL Access si verifica quando per proteggere aree di una web application non visibili ad utenti non autorizzati si fa affidamento esclusivamente ad un approccio di tipo "security by obscurity" poiché si tende a pensare che ciò che non è direttamente visibile non possa essere in alcun modo raggiunto e sfruttato da un agente di minaccia.

Domande?