WEB NEL MIRINO DEL CYBER CRIME

LE 10 VULNERABILITÀ PIÙ CRITICHE IN AMBITO WEB

Marco Bozzetti, OAI founder

In questo periodo, mentre è iniziata l'attività per la realizzazione del prossimo Rapporto OAI 2013, incominciano a essere pubblicati a livello mondiale vari rapporti sugli attacchi e sulle vulnerabilità più critici e/o più diffusi. Tra questi di particolare interesse 'pratico' è il rapporto Owasp (open web application security project, organizzazione mondiale no-profit, www.owasp.org) sui 10 rischi più critici per le applicazioni web, applicazioni che ormai sono quelle più diffuse in ogni sistema informativo. Con un'indagine su più di 500.000 vulnerabilità, sono state individuate le più critiche e le più diffuse, seguendo una metodologia di analisi basata sullo schema di valutazione illustrato nella figura 1.

Per il 2013 le top 10 sono le seguenti, volutamente nella dizione inglese:
- 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

A parte qualche piccola differenza nelle definizioni e nel raggruppamento di talune vulnerabilità, tale classifica è sostanzialmente la stessa dell'anno precedente. Le 10 vulnerabilità sono da tempo ben conosciute e alcune, come A1 e A3, sono state trattate in questa stessa rubrica.

Viste da vicino

Al vertice della classifica permane il così detto 'injection flaw', che consiste nell'invio a un interprete, come parte di un comando o di una 'query' a un data base, di dati differenti da quelli previsti, che, se non controllati, possono indurre l'interprete a eseguire comandi non voluti o l'accesso ai dati senza autorizzazione; tipici e diffusi casi di injection sono relativi all'SQL, all'LDAP e ai sistemi operativi. Si deve tener conto che ogni volta che una applicazione web utilizza un interprete di qualsiasi tipo, pensiamo ad esempio ai Javascript, vi è il rischio di un attacco di iniezione.
Al secondo posto la rottura dell'autenticazione e della gestione delle sessioni. La non corretta implementazione e/o configurazione dei sistemi di autenticazione e di gestione delle sessioni può comportare per l'attaccante il furto di identità digitale, di password, di modifica e di sfruttamento dei diritti d'accesso e così via.
Al terzo posto XSS, cross-site scripting, una vulnerabilità simile all'injection A1: dipende infatti dalla mancanza di controlli sui dati in input a form del browser. Consente all'attaccante di creare ed eseguire script nel browser della vittima, così da dirottare sessioni, bloccare siti web o reindirizzare l'utente su siti criminali.

La vulnerabilità A4, insecure direct object references, avviene quando il programmatore espone il diretto riferimento a un oggetto quale un file, un directory, una chiave di un database senza alcuna protezione per il suo accesso: questo permette l'accesso a dati cui non si avrebbe diritto. La configurazione non corretta (o in certi casi totalmente mancante) di sistemi rappresenta la vulnerabilità A5: il più comune problema è il non aggiornamento del software installato e la non tempestiva installazione di patch: e questo vale anche per le librerie usate dai programmi, le cui vulnerabilità sconfinano e si sovrappongono con A9, più avanti considerata. Grave ma non infrequente, la non attivazione delle opzioni di sicurezza o addirittura il non conoscerle/considerarle. Tali errori/mancanze possono aprire ampi spazi per gli attaccanti, e consentire accessi non autorizzati ai dati.

Threat Agents	Attack Vectors	Weakness Prevalence	Weakness Detectability	Technical Impacts	Business Impacts
App Specific	Easy	Widespread	Easy	Severe	App / Business Specific
	Average	Common	Average	Moderate	
	Difficult	Uncommon	Difficult	Minor	

Fig. 1 - Schema sulla valutazione delle vulnerabilità Owasp (fonte: Owasp)

Un'altra significativa vulnerabilità, la A6, riguarda la non o la debole protezione nei siti web di informazioni critiche, quali le identità digitali, le credenziali di autenticazione, i dati sensibili in termini di privacy, e così via. Le conseguenze sono tipicamente l'utilizzo criminale di tali informazioni per frodi.
La mancanza o il debole controllo degli accessi a livello funzionale costituisce la vulnerabilità A7. La maggior parte delle applicazioni web verifica i diritti d'accesso dell'utente che richiede una determinata funzionalità, prima che questa gli sia mostrata. L'applicazione dovrebbe effettuare questo controllo anche a livello server, altrimenti gli attaccanti possono falsificare le richieste e accedere a funzioni senza averne i diritti.
Un attacco Csrf, cross-site request forgery, forza un ingresso (log in) del browser della vittima per inviare una falsificata richiesta http, che include informazioni di autenticazione (anche automatizzate, quali i cookie di sessione della vittima) a una applicazione non sicura. In pratica l'attaccante si maschera da utente con gli appropriati diritti per accedere a un'applicazione che non è in grado di accorgersi del mascheramento.
L'uso di componenti software vulnerabili, dai moduli alle librerie, rappresenta la penultima (A9) vulnerabilità delle top ten, che apre un ampio ventaglio di possibili attacchi. Anche in questo caso il problema è il non tempestivo aggiornamento dei componenti, soprattutto se contenuti in ampie librerie che non vengono quasi mai controllate. Il problema è ingigantito dalla moderna programmazione software, che si basa sull'assemblaggio di moduli forniti da terzi e che ben difficilmente sono controllati e controllabili in termini di sicurezza.
L'ultima vulnerabilità tra le 10 principali individuate da Owasp, la A10 unvalidated redirects and forwards, consiste nel re-indirizzamento dell'utente ad altre pagine e ad altri siti in Internet: tali indirizzamenti, se manipolati, possono portare a siti e pagine web criminali. Spesso tale re-indirizzamento non è controllato e non è protetto dall'applicazione che lo usa.

Difendersi aggiornando e controllando

Analizzando le 10 vulnerabilità più critiche per Oswap emerge come esse abbiano un comune 'fil rouge': la mancanza di tempestivi aggiornamenti e la mancanza di efficaci controlli nel software. Il primo aspetto dipende da una cattiva gestione dei sistemi, il secondo da uno sviluppo software non sicuro e quindi poco professionale. Come dire, i soliti ben noti problemi ...

Marco Bozzetti
marco.bozzetti@malaboadvisoring.it