



Saverio Rizza



Svolge attività di formazione e di verifica sulla sicurezza dei sistemi informativi. Si occupa di sicurezza applicativa nella pubblica amministrazione e nel settore delle telecomunicazioni (www.codeinspection.it)



Claudio De Rossi



Consulente in ambito sicurezza informatica, si occupa di ricerca e sviluppo, penetration test e security assessment (www.studiocdr.com)

La sicurezza di un'applicazione web è importante in misura alla criticità del business trattato. Risulta evidente a tutti che a seconda delle operazioni da eseguire e dei dati che l'applicazione tratta il requisito della sicurezza possa acquisire o meno una rilevanza fondamentale! In linea generale è importante considerare le applicazioni web come componenti "sensibili" dell'infrastruttura informatica perché interagiscono con una serie di servizi esterni come server di posta, gateway e database. Una compromissione del software web potrebbe permettere ad un attaccante di aggirare le limitazioni dei firewall perimetrali sulla rete ed interagire direttamente con quei servizi, dando vita ad un ampio ventaglio di possibili attacchi nella rete interna.

Generalmente è sempre bene prestare particolare attenzione agli attacchi che possono venire perpetrati da agenti automatizzati come worms e tool di analisi, che rappresentano una minaccia concreta per tutti i portali web data la facilità d'esecuzione e la frequenza con cui si verificano. Fra questi attacchi troviamo l'inserimento di vettori d'attacco negli url o nei form, il "tampering" dei cookie e gli attacchi volti ad eludere i meccanismi di autenticazione ed autorizzazione degli utenti. Ma prima di considerare gli attacchi ai quali potremmo esporci in rete sarebbe bene che uno sviluppatore approfondisca l'analisi delle vulnerabilità presenti sul software: ogni classe di vulnerabilità può prestarsi a differenti tipologie di attacco.

Il consiglio che mi sembra importante dare è quello di maturare la consapevolezza della necessità d'integrare lo sviluppo applicativo con l'analisi dei requisiti e l'implementazione delle funzionalità di sicurezza. Più pragmaticamente, reputo importante porre particolare attenzione al filtraggio dell'input (sia esso proveniente da un client o da una base dati), alla stampa a video di dati provenienti da servizi esterni ed alla realizzazione di politiche di verifica dell'identificazione e dell'autenticazione efficaci. Last but not least, un campanello d'allarme deve scattare ogni qual volta il nostro codice diventa esageratamente complesso o quando si presenta la necessità di risolvere un problema per mezzo di "workaround": l'imperativo è seguire le best practice!

Oggigiorno le applicazioni web sono "complesse strutture di integrazione e gestione di dati provenienti da sistemi eterogenei spesso dislocati geograficamente". In quest'ottica è facile comprendere quanto sia fondamentale garantire a tali sistemi alti livelli di sicurezza. Le applicazioni Web accedono ai dati aziendali, gestiscono informazioni, elaborano modelli e logiche business, espongono servizi e presentano i risultati all'utente; tutte queste funzioni vengono svolte all'interno di architetture complesse dove un piccolo errore potrebbe propagarsi e creare notevoli disagi. Un'errata o insufficiente implementazione di procedure di sicurezza può causare perdite di "dati" (il vero patrimonio di un'azienda), intrusioni da parte di criminali informatici, furto di informazioni, interruzioni di servizio e, come conseguenza, consistenti perdite in termini di tempo (= denaro)

Attaccare un sistema informatico significa "fargli compiere operazioni per le quali non è stato progettato". Questo può tradursi nell'utilizzo di tecniche complesse e variegate che non sempre possono essere ricondotte a una definizione precisa. Semplificando si potrebbe dire che i principali attacchi sono quelli portati verso: autenticazione (controllo della gestione di credenziali, brute force, utilizzo della cache), autorizzazione (path traversal, privilege escalation), session management (csrf, gestione dei cookie), business logic (Xss, sql injection), web service (test di "wsdl", analisi di http request/response), infrastrutture AJAX e architettura dell'applicazione (lato sistemistico). Un ottimo punto di partenza e approfondimento è www.owasp.org, un'organizzazione internazionale (senza fini di lucro) impegnata nell'approfondimento di tematiche inerenti la sicurezza delle applicazioni Web.

Innanzitutto il ciclo di vita dello sviluppo di un software (SDLC) deve diventare un'abitudine senza la quale non deve esistere programmazione (almeno in ambito aziendale). Poi studio, pratica, determinazione, controllo sul codice, umiltà (sì, l'umiltà di non ritenersi infallibili)... ho già detto lo studio? Conoscere le metodologie di attacco aiuta enormemente a sviluppare con un occhio alla sicurezza. Pensare e creare codice in modo semplice, senza farsi trascinare da mode o strumenti preconfezionati di cui ignoriamo il funzionamento. Tecnicamente è molto importante il controllo dei dati (input/output), del flusso di dati (utilizzare strutture logiche solide e funzionali), e dell'infrastruttura per la quale sviluppiamo. In ultimo non bisogna "reinventare l'acqua calda", a patto di conoscere come viene scaldata!