7 BEST PRACTICES PER LO SVILUPPO IN SICUREZZA

Molti dei problemi di sicurezza del software sono da attribuire alla scarsa conoscenza, da parte degli sviluppatori, delle principali vulnerabilità e dei possibili attacchi che potrebbero sfruttarle.
Il presente capitolo fornisce una vista delle principali vulnerabilità e delle relative contromisure, contestualizzate per ogni specifica area di sviluppo (C/C++, Java, PL/SQL, etc), anche in termini di tecniche da utilizzare per riconoscerle e difendersi opportunamente.

7.1 C/C++

Il linguaggio di programmazione procedurale denominato C fu sviluppato da Dennis Ritchie tra il 1969 e il 1973 presso i Bell Labs, con lo scopo di implementare parti di sistema operativo Unix. Da allora è diventato uno dei linguaggi di programmazione più diffusi e utilizzati, grazie alla sua grande potenza e flessibilità. Il linguaggio C, infatti, consente al programmatore di accedere alla memoria della macchina in maniera diretta, in modo da indirizzare e sfruttare qualsiasi risorsa, software e hardware.

Dal C deriva il linguaggio di programmazione C++ (o CPP acronimo di “C plus plus”) , orientato agli oggetti, con tipizzazione statica. È stato sviluppato (in origine col nome di "C con classi") da Bjarne Stroustrup, sempre presso ai Bell Labs nel 1983 nell’ottica della modernizzazione del linguaggio C.

Poiché i linguaggi C e C++ hanno caratteristiche molto simili, ai fini della sicurezza del codice le vulnerabilità e le relative contromisure sono da considerarsi valide per entrambi i linguaggi.

7.1.1 Cross-site scripting (XSS)

Come riconoscerla

Il Cross-site scripting (XSS) è una vulnerabilità che affligge siti web dinamici che operano un controllo insufficiente dell'input. Un XSS permette ad un attaccante di inserire o eseguire codice script lato client, al fine di attuare i seguenti exploit:

- raccolta, manipolazione e reindirizzamento di informazioni riservate;
- visualizzazione e modifica di dati presenti sui server;
- alterazione del comportamento dinamico delle pagine web.

Rientrano nelle problematiche di tipo XSS:

- **Stored XSS**. Gli attacchi di tipo “stored XSS” sono quelli in cui lo script iniettato viene memorizzato in modo permanente sui server di destinazione, come ad esempio in un database, in un forum di messaggi, in un registro dei visitatori, in un campo commentato, etc. Da quel momento in poi, ogni qualvolta verrà rischiesta la pagina che include lo script memorizzato, quest’ultimo verrà rispristinato ed eseguito.
- **Reflected XSS**. Gli attacchi XSS riflessi, noti anche come attacchi non persistenti, si verificano quando uno script dannoso viene restituito da un'applicazione Web al browser della vittima. Sono più diffusi, proprio per la facilità di propagazione: non è neessario individuare alcun meccanismo per memorizzare permanentemente gli script malevoli. Sono i più evitabili e spesso i danni che apportano sono di entità inferiore, rispetto agli stored XSS.

Come difendersi

Convalidare tutti gli input, indipendentemente dalla fonte: la convalidazione dovrebbe essere basata su una white list (una lista di valori ammessi), per cui verrebbero accettati solo i dati che corrispondono, e verrebbero rifiutati tutti gli altri.
Occorre controllare, oltre che i valori siano fra quelli ammessi o che rientrino in un determinato intervallo di validità, se corrispondano alle attese anche il tipo, la dimensione e il formato dei dati in input.