# 9 LINEE GUIDA PER L'ADOZIONE DI UN CICLO DI SVILUPPO SOFTWARE SICURO

Negli ultimi anni, un numero considerevole di applicazioni e sistemi hanno dovuto affrontare gravi minacce alla sicurezza a causa di un significativo incremento nell'impiego di nuove tecnologie disponibili e nel contempo della mancanza di conoscenza e di tecniche di indagine che riguardano la sicurezza informatica. In passato, le problematiche in materia di sicurezza riguardavano essenzialmente i livelli dell'infrastruttura di rete. Attualmente, a causa del crescente utilizzo delle reti e della dominanza del concetto di Internet, come il cloud computing, Software as a Service (SaaS), gli aggressori stanno scoprendo sempre più la presenza di gravi vulnerabilità nel livello applicativo del software.
Il concetto di sicurezza a livello di applicazione è quindi emerso come una attività essenziale da integrare nel processo di sviluppo del software.
La sicurezza dell'informazione richiede una particolare attenzione a causa di un gran numero di vulnerabilità individuate nelle applicazioni/sistemi dichiarate come sicure. Sono ben note la complessità e la difficolta nel realizzare un'applicazione priva di difettosità e/o vulnerabilità, tuttavia, le organizzazioni che producono hardware e software non possono astenersi nel migliorare i propri processi di sviluppo e adattarli agli attuali scenari. Oltre alle numerose pubblicazioni di ricercatori accademici e industrie del software che evidenziano l'importanza di integrare pratiche di sicurezza nel System Development Life Cycle (SDLC), esiste un paradosso nell'effettiva implementazione. La maggior parte dei centri di sviluppo non attua le raccomandazioni di cui sopra a causa della resistenza ai nuovi processi e al mancato adeguamento di mentalità da parte degli attori che operano nell'ambito del SDLC. E' anche solita una certa resistenza da parte d'ingegneri e sviluppatori nell'accettare che il software/hardware da loro realizzato possa essere soggetto a difetti di sicurezza. Anche i team di sviluppo oramai comprendono l'importanza di un nuovo paradigma di sicurezza per il SDLC, anche se, purtroppo, ciò non è sufficiente.
Per raggiungere i livelli di sicurezza adeguati, è necessaria una conoscenza approfondita e dettagliata delle procedure e delle tecniche di sicurezza da adottare: una Security Policy completa è il giusto riferimento per guidare lo sviluppo della sicurezza e tutti gli attori coinvolti come gli ingegneri hardware, sviluppatori, architetti applicativi, ingegneri software, collaudatori e project leader devono considerarla come una regola imprescindibile.
Questa deve stabilire le opportune indicazioni per ciascuna fase di sviluppo: requisiti, progettazione/architettura, implementazione, collaudo e manutenzione, e deve definire le responsabilità per tutti i ruoli coinvolti nel processo di sviluppo. Deve inoltre, stabilire le regole per la definizione dei requisiti di fase abilitando i principi di sicurezza, come la sicurezza delle informazioni, integrità, privacy, riservatezza, disponibilità delle informazioni, continuità, in base all'ambiente e alle minacce pubbliche che possono in qualche modo coinvolgere il sistema.
Al fine di dare copertura agli aspetti di sicurezza è necessario riunire i team di business, di sviluppo e di sicurezza per comprendere le principali vulnerabilità e le conseguenze sul business causate dal rischio dovuto alla presenza di difetti di sicurezza nella versione finale del prodotto. Poiché il SDLC è un processo di "feed forward" come tale, eventuali errori introdotti in questa fase, saranno poi diffusi nelle fasi successive. Per questo motivo è importante analizzare i rischi per la sicurezza sin dalle primissime fasi del ciclo di sviluppo del software.
L'analisi dei requisiti, rappresenta il primo passo nell'SDLC. Attraverso questo, vengono identificati e definiti gli obiettivi delle specifiche di sicurezza, i metodi necessari per implementarle e l'importanza che queste ricoprono. I requisiti di sicurezza definiscono i requisiti funzionali e non funzionali che devono essere soddisfatti per ottenere le caratteristiche di sicurezza di un sistema IT. Tali requisiti possono essere formulati a diversi livelli di astrazione; al più alto livello, riflettono fondamentalmente solo gli obiettivi di