DevSecOps, la sicurezza come elemento fondante del processo di sviluppo del software



È ormai assodato che la pandemia abbia accelerato molti cambiamenti nella vita lavorativa di tutti noi, così come nei servizi e nelle tecnologie. Cambiamenti come i colloqui di lavoro virtuali, l'utilizzo di applicazioni che permettono di ordinare facilmente il pasto senza dover affrontare le lunghe code, la gestione delle finanze personali tramite telefono, rivelano che ad ogni settore viene richiesto di fornire ai propri dipendenti e clienti esperienze digitali altamente performanti e servizi sempre raggiungibili.

Nell'ambito della sicurezza, questa nuova normalità fa si che qualsiasi interruzione o violazione abbia il potere di compromettere la fidelizzazione dei clienti o di rendere inaccessibili operazioni indispensabili alla quotidianità.

E' in questo contesto che la sicurezza del software diventa imprescindibile e, vista la massiva trasformazione digitale, per realizzare un software sicuro in modo scalabile è necessario integrare la sicurezza fin dall'origine, già dal processo di sviluppo. E' questa nuova realtà a far sì che il Development Security Operations (Sviluppo, Sicurezza e Operazioni, abbreviato in DevSecOps), considerato fino ad oggi un obiettivo futuro, diventi un tassello fondamentale per i team di sicurezza, indispensabile durante tutto il ciclo di vita di un'applicazione.

Nell'approccio DevSecOps concetti come Security by Design e Security as Code sono oggi processi determinanti.

Come realizzare un programma di AppSec efficiente?

Nel contesto della sicurezza delle applicazioni (AppSec), molte organizzazioni fanno ancora affidamento su una postura reattiva eseguendo test incoerenti e isolati, effettuati di solito troppo tardi in quello che è definito “ciclo di vita dello sviluppo del software” (SDLC). Ciò si traduce in un enorme collo di bottiglia per gli sviluppatori visto che, come dimostrato dallo studio sulle minacce informatiche di Forbes del 2021, sono necessari in media 48 giorni per chiudere una vulnerabilità critica, causando così un progressivo aumento del rischio legato al software potenzialmente non controllato.

Gli sviluppatori software sanno che la sicurezza è importante ma spesso non hanno tempo da dedicarle e questo si traduce nell’utilizzo di pratiche AppSec limitate. L’impossibilità di integrare *developer tools* e fonti di scansione nelle *pipelines* esistenti o nelle *developer tool chains*, così come la mancanza di efficaci cicli di feedback, possono causare il fallimento di molte iniziative AppSec.

Tuttavia, nonostante la resistenza messa in campo dagli sviluppatori, molte organizzazioni, per far fronte ai rischi legati alle vulnerabilità, hanno investito in soluzioni che testano la sicurezza delle applicazioni (AST, Application Security Testing).

La maggior parte delle aziende oggi utilizza una moltitudine di strumenti AST per le diverse fasi previste nel ciclo di vita dello sviluppo delle applicazioni: Static Application Security Test (SAST) e Software composition analysis (SCA) sono solitamente utilizzati nella fase di realizzazione e sviluppo, mentre i Dynamic Application Security Test (DAST) analizzano la presenza di vulnerabilità nelle applicazioni in esecuzione.

Va considerato che ciascuna di queste categorie di soluzioni

AST ha differenti capacità di rilevamento, inoltre le tipologie di applicazioni e i linguaggi di programmazione supportati possono variare a seconda del fornitore. Questo significa che ogni strumento identifica difetti specifici del software, quindi, se non usati in sinergia saranno in grado di scoprire un numero limitato di vulnerabilità.

Un programma AppSec completo necessita dell'utilizzo di diversi strumenti previsti dalla categoria AST, implementandoli in maniera appropriata nelle diverse fasi dell'SDLC.

Al costante aumento degli investimenti negli strumenti AST da parte delle organizzazioni, si contrappone però la sottovalutazione delle pratiche di *security hygiene*, la cui applicazione tutt'oggi rimane difforme all'interno dei team di sviluppo. Nel rapporto ESG del 2022 "Walking the Line: GitOps and Shift Left Security", il **35% degli intervistati ha riferito di aver rilasciato codice con vulnerabilità note e il 45% ammette di aver rilasciato software senza alcun test o controllo di sicurezza**. Questo modo di fare condiziona negativamente i risultati degli investimenti nella sicurezza delle applicazioni sostenuti dalle aziende, ed è qui che DevSecOps offre un approccio interessante a tutte le parti interessate unendo flussi di lavoro, di sicurezza e sviluppo per facilitare la collaborazione, l'efficienza e la responsabilità.

Come implementare un programma DevSecOps?

Sebbene non esista un unico modo di implementare un programma DevSecOps, esistono alcuni principi guida nelle soluzioni AppSec indicativi della scalabilità e dell'efficacia. Questi includono:

- Protezione del codice con la stessa velocità con cui lo si scrive: adottare soluzioni in grado di funzionare con strumenti e processi di CI level;

- Esecuzione dei test giusti al momento giusto: standardizzazione dei flussi di lavoro e delle policy di test tra i team di sicurezza e sviluppo, con soglie definite per l’attivazione automatica dei test;
- Filtraggio del rumore AppSec per concentrarsi su ciò che conta di più: visualizzare e semplificare tutti i dati AppSec nell’SDLC per fare una selezione rapida delle informazioni e adottare rapide decisioni di remediation.

Condividi sui Social Network:
https://www.ictsecuritymagazine.com/notizie/devsecops-la-sicurezza-come-elemento-fondante-del-processo-di-sviluppo-del-software/