

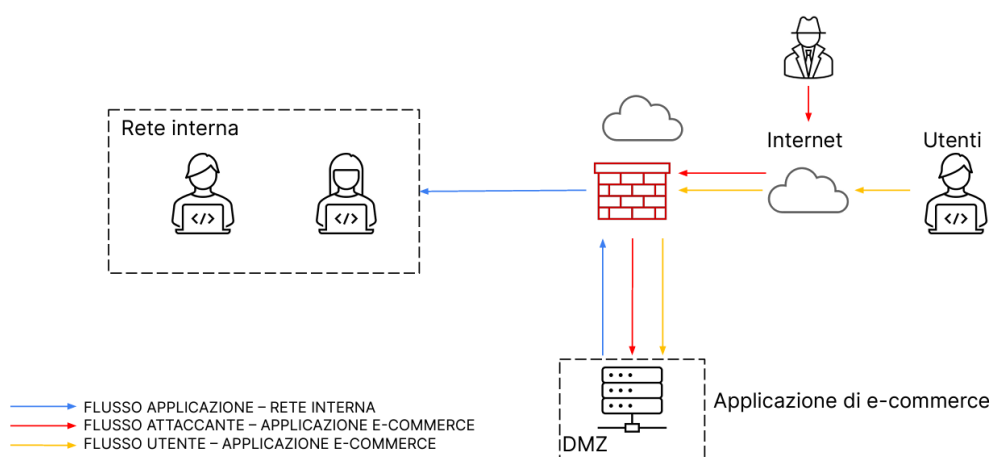
PROGETTO W20D4

Secondo la traccia mostrata in figura, lo scopo principale è quello di mettere in sicurezza l'applicazione web da diverse tipologie di attacchi.

Traccia:

Con riferimento alla figura in slide 2, rispondere ai seguenti quesiti.

1. **Azioni preventive:** quali azioni preventive si potrebbero implementare per difendere l'applicazione Web da attacchi di tipo SQLi oppure XSS da parte di un utente malintenzionato?
Modificate la figura in modo da evidenziare le implementazioni
2. **Impatti sul business:** l'applicazione Web subisce un attacco di tipo DDoS dall'esterno che rende l'applicazione non raggiungibile per **10 minuti**.
Calcolare l'impatto sul business dovuto alla non raggiungibilità del servizio, considerando che in media **ogni minuto gli utenti spendono 1.500 €** sulla piattaforma di e-commerce. **Fare eventuali valutazioni di azioni preventive che si possono applicare in questa problematica**
3. **Response:** l'applicazione Web viene infettata da un malware.
La vostra priorità è che il malware non si propaghi sulla vostra rete, mentre non siete interessati a rimuovere l'accesso da parte dell'attaccante alla macchina infettata.
Modificate la figura in slide 2 con la soluzione proposta.
4. **Soluzione completa:** unire i disegni dell'azione preventiva e della response (unire soluzione 1 e 3)
5. **Modifica «più aggressiva» dell'infrastruttura (se necessario/facoltativo magari integrando la soluzione al punto 2)**



1. Azioni preventive

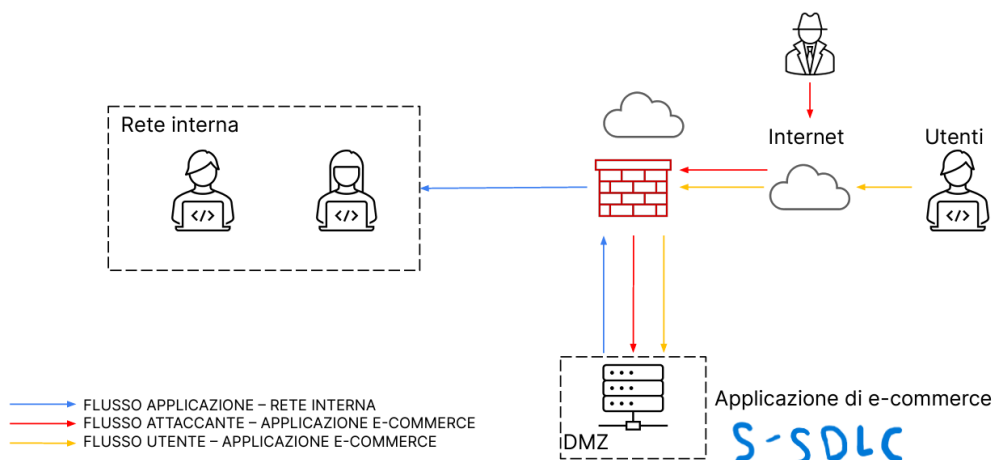
Abbiamo diverse opzioni per la messa in sicurezza. Una delle più importanti sicuramente è definire un team di sviluppo consapevole delle direttive **S-SDLC**. L'SQLi ed il Cross site Scripting sono difatti applicabili in caso di vulnerabilità insite nel codice.

SQLi - esempio di metodi:

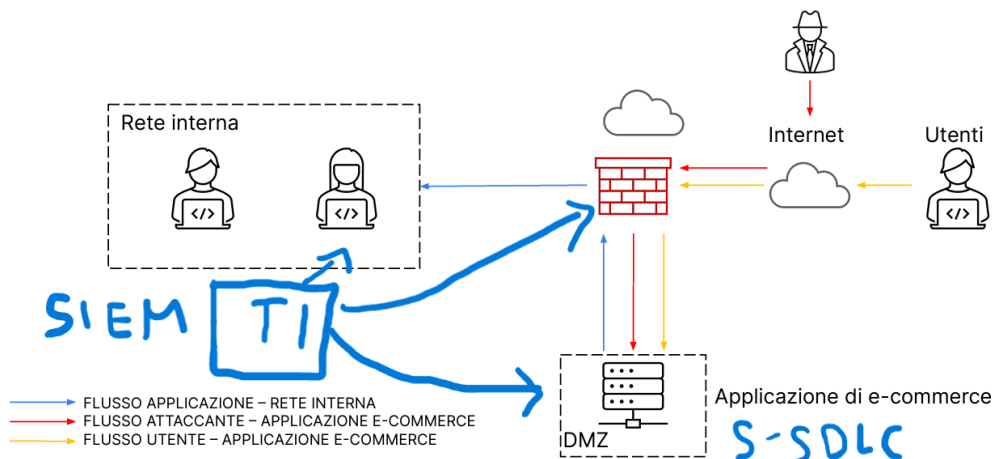
- Occorre prestare attenzione a virgolette e parentesi che potrebbero essere integrate con elementi di controllo e sfruttate.
- Occorre oscurare i risultati di errore dalla pagina del sito.
- Utilizzare l'estensione MySQLi abbinate alle librerie PHP.

XSS - esempio metodi:

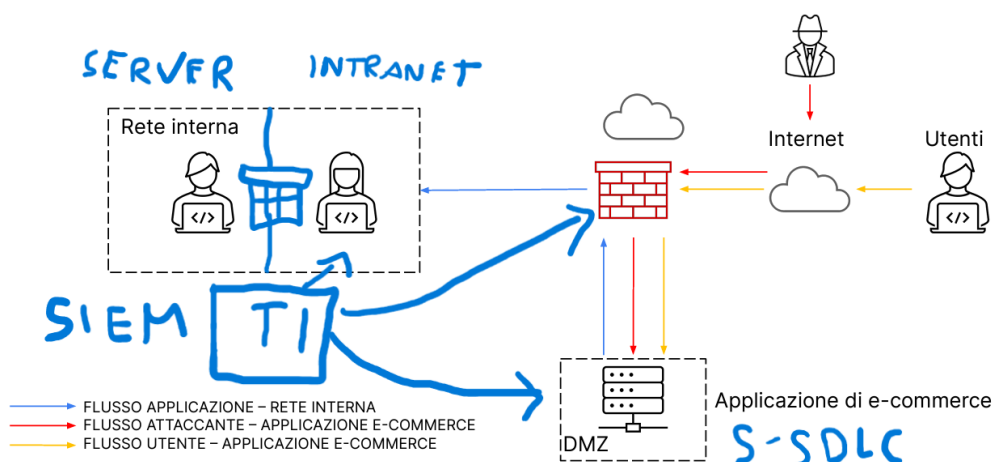
- Aggiornamento/scansione periodico/a del codice.
- Policy zero-trust e sanificazione dell'input utente.
- Controllo della dichiarazione delle risorse dinamiche tramite intestazione CSP(Content Security Policy).
- Flag httpOnly, per proteggere i cookie da Javascript lato client.



Un altro modo per alzare il livello di sicurezza è la messa in funzione di un sistema TI(Threat Intelligence) i cui punti interesse ricadono a livello di rete, firewall e endpoint della rete interna. Altrimenti ci viene molto comodo un raccoglitore dei log di sistema, del firewall delle applicazioni, etc (SIEM – Security Information and Event Management).



Se vogliamo rendere un'intrusione ancora più complicata per l'attaccante possiamo segmentare la rete aziendale in aree di importanza, protette da ulteriori firewall o proxy, (multi-tier DMZ), oppure controllare ad esempio se il firewall supporta l'IPS/IDS (Intrusion Prevention System, Intrusion Detection System).



2.Impatti Business

Nel caso l'azienda subisca un attacco **DDos** di 10 minuti, andiamo a ipotizzare l'impatto subito in base ai dati messi a disposizione. Nel caso l'evento si presenti 4 volte ogni 5 anni ad esempio, la perdita annuale sarà.

$$SLE = 15000 \$$$

$$ARO = 4/5$$

$$ALE = 15000 \times 0,8 = 12000 \$$$

In questo caso non sono necessarie delle misure troppo stringenti perché i DDos si verificano poche volte durante gli anni. Potrebbe essere sufficiente l'utilizzo di un **WAF**(Firewall Application Firewall) per contenere, filtrare e limitare un eventuale traffico malevolo. Inoltre, sarebbe opportuno preparare un piano di emergenza in caso di attacco DDos, in modo che il personale IT sappia come comportarsi. Volendo è possibile considerare anche il monitoraggio dei tempi di attività del sito, ad esempio con Uptime Robot, che è gratuito.

Al contrario, se l'evento si verifica più volte all'anno, ad esempio 10 volte all'anno, la situazione si fa decisamente più problematica.

SLE = 15000 \$

ARO = 10

ALE = 15000 x 10 = 150000\$

I costi sono più ingenti e sono richiesti dunque più mezzi di risoluzione del problema. Un WAF, seguito da un piano di emergenza e Uptime Robot, può non darci tutta la protezione di cui necessitiamo. Potremmo affidarci ad un servizio di hosting gestito, che fornisca un ulteriore strato di difesa, aiutando anche a riportare il sito web in funzione nel minor tempo possibile. Il **CDN**(Content Distributed Network) inoltre è un'altra soluzione efficace, che divide la rete in nodi aggiuntivi geograficamente distribuiti; include proxy server, datacenter etc.

3.Response

Se vogliamo proteggere la rete senza espellere immediatamente l'attaccante, il modo migliore di agire è quello di ricorrere alla tecnica di **isolamento**. Spostiamo l'end point in una rete di quarantena e gli lasciamo la connessione a internet.

