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

Per prevenire attacchi di tipo **SQL Injection (SQLi)** e **Cross-Site Scripting (XSS)** in un'applicazione Web, è fondamentale adottare una combinazione di misure tecniche, pratiche di codifica sicura e configurazioni adeguate, queste possono essere:

Contro SQL Injection (SQLi):

1) con PHP
2) Configurare l'utente del database con il minor numero di privilegi possibile (es. solo lettura e scrittura su tabelle specifiche).
3) Implementare un Web Application Firewall (WAF) o un Database Firewall per rilevare e bloccare attacchi noti.
4) Se proprio necessario costruire query dinamiche, eseguire un'escape sicura degli input per prevenire interpretazioni errate.

Contro Cross-Site Scripting (XSS)

1) **Utilizzo di HTTP Headers di Sicurezza**: **Content Security Policy (CSP)**: Definire una CSP per limitare le origini degli script eseguibili.
   **X-XSS-Protection**: Configurare questo header per abilitare il filtro anti-XSS nei browser.
2) **Sanificazione dell'Input**:
   Rimuovere o codificare caratteri pericolosi come <, >, " e ' negli input che possono finire nel codice HTML o JavaScript.
3) **Librerie di Template Sicure**:
   Usare librerie di template come Handlebars o React, che gestiscono automaticamente l'escape dell'output.

Altre Misure generali possono essere:
Monitoraggio dei log, aggiornamenti frequenti e autenticazione e controllo degli accessi

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**

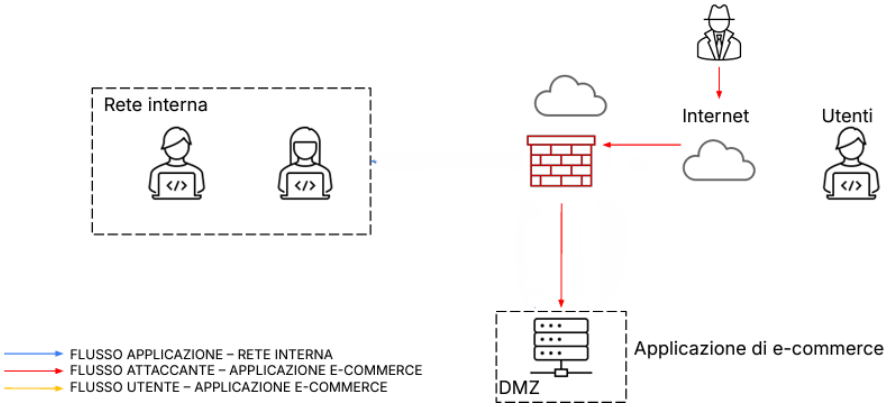
3. **Response**: l'applicazione Web viene infettata da un malware.
   La vostra priorità è che il malware non si propaghi sulla vostre rete, mentre non siete interessati a rimuovere l'accesso da parte dell'attaccante alla macchina infettata.
   Modificate la figura in slide 2 con la soluzione proposta.

Se non siamo interessati a rimuovere l'accesso dell'attaccante possiamo semplicemente ISOLARE l applicazione web infetta ,togliendo il flusso utente e il flusso applicazione della rete interna

---

**Architettura di rete:**
L'applicazione di e-commerce deve essere disponibile per gli utenti tramite Internet per effettuare acquisti sulla piattaforma.

La rete interna è raggiungibile dalla DMZ per via delle policy sul firewall, quindi se il server in DMZ viene compromesso potenzialmente un attaccante potrebbe raggiungere la rete interna.




4

4. **Soluzione completa**: unire i disegni dell'azione preventiva e della response (unire soluzione 1 e 3)

Se il sistema è già infetto prima lo si isola e poi  si applicano tutte le misure come aggiornare i sistemi,modificare il php,implementare un WAF,sanificare gli output e usare librerie sicure