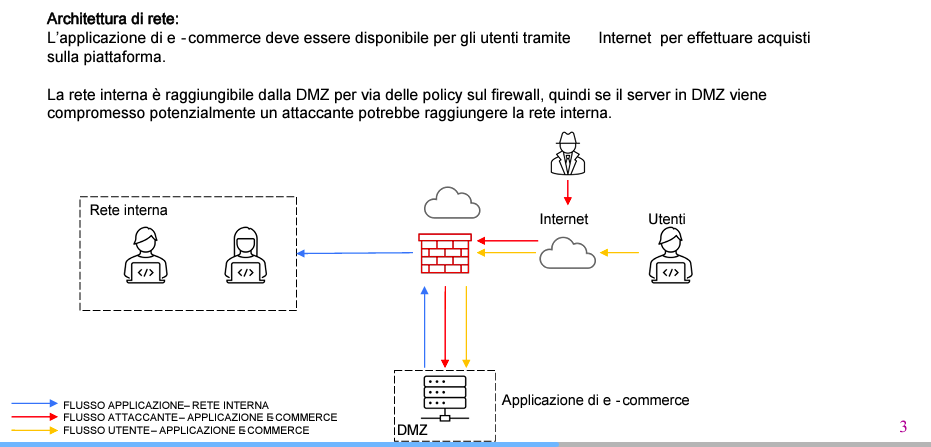
Immagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamente



Le vulnerabilità che vengono sfruttate negli attacchi SQL Injection (SQLi) e Cross-Site Scripting (XSS) sono diverse e hanno origine da errori comuni di programmazione o configurazione delle applicazioni web. Di seguito sono riportate le principali vulnerabilità associate a ciascun tipo di attacco:

**Vulnerabilità sfruttate negli attacchi SQL Injection (SQLi)**:

1. **Input Non Validato o Non Sanitizzato**: Quando un'applicazione web non valida o non sanitizza correttamente i dati di input forniti dagli utenti, permette agli attaccanti di inserire istruzioni SQL dannose attraverso i campi di input.
2. **Mancanza di Preparazione delle Query (Prepared Statements)**: Se un'applicazione web costruisce dinamicamente query SQL concatenando direttamente i dati di input forniti dagli utenti, si apre la possibilità di iniezione di codice SQL dannoso.
3. **Mancanza di Controllo degli Accessi**: Se un'applicazione web non implementa correttamente i controlli di accesso e privilegio, un attaccante potrebbe sfruttare un attacco SQLi per ottenere accesso non autorizzato ai dati sensibili nel database.

**Vulnerabilità sfruttate negli attacchi Cross-Site Scripting (XSS)**:

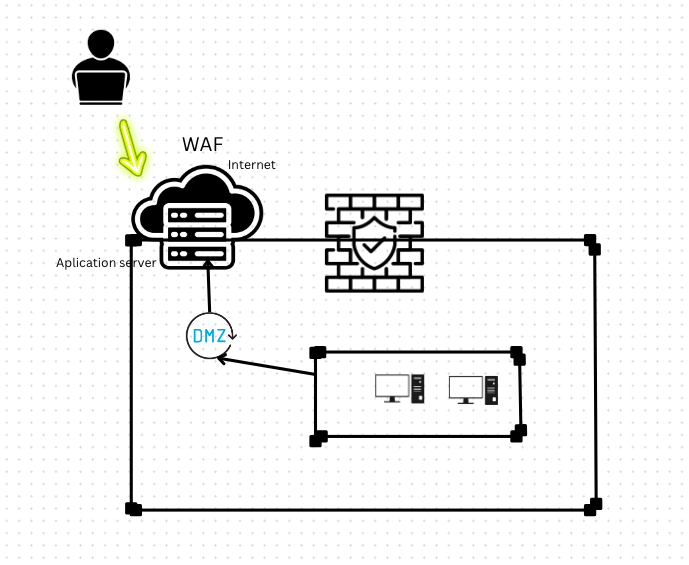
1. **Mancaza di Escape dei Dati di Output**: Se un'applicazione web visualizza dati forniti dagli utenti senza prima escapare i caratteri speciali, gli attaccanti possono inserire script JavaScript dannosi che verranno eseguiti quando gli utenti visitano la pagina.
2. **Mancaza di Validazione dei Dati di Input**: Se un'applicazione web non valida o non sanitizza correttamente i dati di input forniti dagli utenti, gli attaccanti possono inserire script XSS nelle richieste inviate all'applicazione.
3. **Mancanza di Headers di Sicurezza (HTTP Headers)**: Se un'applicazione web non utilizza correttamente headers di sicurezza come Content Security Policy (CSP), gli attaccanti possono sfruttare vulnerabilità XSS per eseguire script dannosi.
4. **Azioni preventive :**

* Possiamo considerare come un azione preventiva la validazione e sanitizzazione dei dati in ingresso però questa azione prenderebbe tempo quindi la soluzione piu rapida ed efficace per la rete sarebbe introdure il WAF ( Web Application Firewall ) .

Il WAF (Web Application Firewall) è uno strumento di sicurezza progettato per proteggere le applicazioni web da una vasta gamma di minacce e attacchi informatici. Ecco da cosa protegge principalmente un WAF:

1. **Attacchi di Iniezione (Injection Attacks)**: Un WAF può proteggere da attacchi di tipo SQL Injection (SQLi), Cross-Site Scripting (XSS), e altri tipi di iniezioni di codice dannoso nelle richieste HTTP. Esso analizza le richieste in arrivo e può bloccare quelle sospette contenenti codice dannoso.
2. **Furti di Sessioni e Cookie Hijacking**: Un WAF può rilevare e bloccare tentativi di furti di sessioni o cookie hijacking, che sono spesso utilizzati per ottenere accesso non autorizzato ai sistemi o per impersonare gli utenti legittimi.
3. **Attacchi di Forza Bruta e DDoS (Distributed Denial of Service)**: Un WAF può mitigare attacchi di forza bruta contro endpoint sensibili come pagine di login e può filtrare il traffico DDoS, aiutando a mantenere le applicazioni web operative anche durante picchi di traffico anomalo.
4. **Manipolazione e Alterazione dei Dati**: Un WAF può rilevare tentativi di manipolare o alterare i dati inviati da client non autorizzati, aiutando a garantire l'integrità dei dati dell'applicazione.
5. **Vulnerabilità delle Applicazioni**: Un WAF può proteggere da vulnerabilità note e sconosciute nelle applicazioni web, fornendo un'ulteriore strato di difesa contro exploit noti come zero-day e vulnerabilità emergenti.
6. **Scansioni e Monitoraggio del Traffico**: Un WAF può eseguire scansioni approfondite del traffico in arrivo e in uscita per identificare comportamenti sospetti o anomalie, consentendo una risposta rapida e un'analisi forense in caso di incidente di sicurezza.

Sotto troviamo costruzione di rete migliore per la compagnia , cosi che i dati sono protetti da gli attacchi sopranominati .



1. **Impatti sul business :**

* Per calcolare l'impatto sul business dovuto alla non raggiungibilità del servizio per 10 minuti a causa di un attacco DDoS, possiamo utilizzare il seguente calcolo:

**Tempo di inattività**: 10 minuti

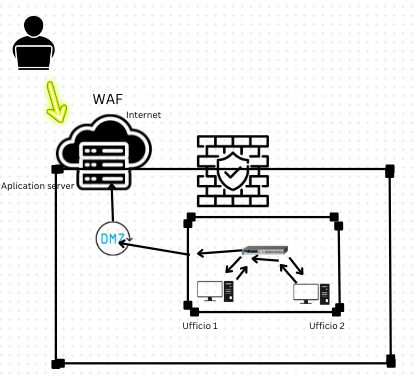
**Valore medio degli acquisti degli utenti al minuto**: 1.500 €

**Numero di transazioni perse**: Tempo di inattività (minuti) x Valore medio delle transazioni al minuto = 10 minuti x 1.500 €/minuto = 15.000 €

Quindi, l'impatto finanziario dovuto alla non raggiungibilità del servizio per 10 minuti sarebbe di 15.000 €.

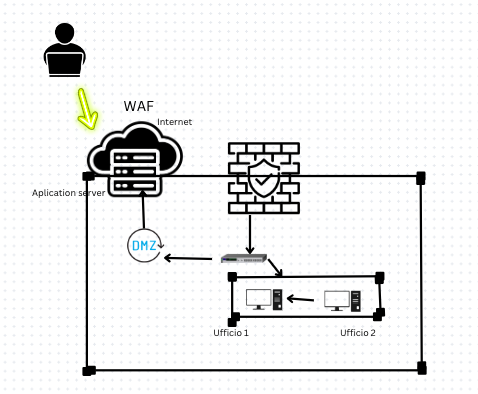
* Come abbiamo detto anche prima un WAF può mitigare attacchi di forza bruta contro endpoint sensibili come pagine di login e può filtrare il traffico DDoS, aiutando a mantenere le applicazioni web operative anche durante picchi di traffico anomalo.

Quindi sarebbe una buona soluzione anche i attacchi DDos . Essendo che il WAF estato già implementato, andiamo ad implementare una segmentazione di rete . La quale sarebbe utile a prevenire questi tipi di attacchi , la segmentazione della rete si può avere anche aggiunge un Switch ed facendo un VLAN tramite configurazioni dello switch questo ci permetterebbe di isolare differente settori oppure in necessità .



1. In questo caso la rete interna si protegge l’entrata tramite il firewall ed l’uscita con il dmz .

Essendo che tutto il traffico passa dallo switch , possiamo bloccare il traffico in entrata .



1. Immagine che contiene diagramma, disegno, schizzo, testo

   Descrizione generata automaticamente

5 - In questo punto l’unica cosa che possiamo aggiunge e un punto ips/ids , nel quale modifichiamo il ips solo entratta ed il ids solo in uscita .

Immagine che contiene diagramma, testo, disegno, schizzo

Descrizione generata automaticamente