2. Impatti sul business

Tempo totale dell'attacco: 10 minuti
Perdita per minuto: € 1500
Totale impatto economico: € 1500 x 10 minuti = € 15.000

Azioni di prevenzione:

- Utilizzare un Servizio di Protezione DDoS: esistono servizi specifici per proteggere le applicazioni web dagli attacchi DDoS. Questi servizi rilevano e bloccano il traffico malevolo prima che raggiunga i server dell'applicazione.
- Content Delivery Network (CDN): le CDN distribuiscono il traffico attraverso molteplici server in diverse località geografiche, riducendo il carico su un singolo punto e mitigando gli effetti di un attacco DDoS.
- Firewalls e sistemi di rilevamento delle intrusioni (IDS): configurare i firewall per limitare il traffico sospetto e implementare sistemi di rilevamento delle intrusioni per identificare e rispondere rapidamente agli attacchi.
- Rate limiting: implementare limitazioni di velocità (rate limiting) per controllare il numero di richieste che un singolo indirizzo IP può effettuare in un determinato periodo.

3.Response

L'azione prioritaria da intraprendere è l'isolamento della macchina infetta.

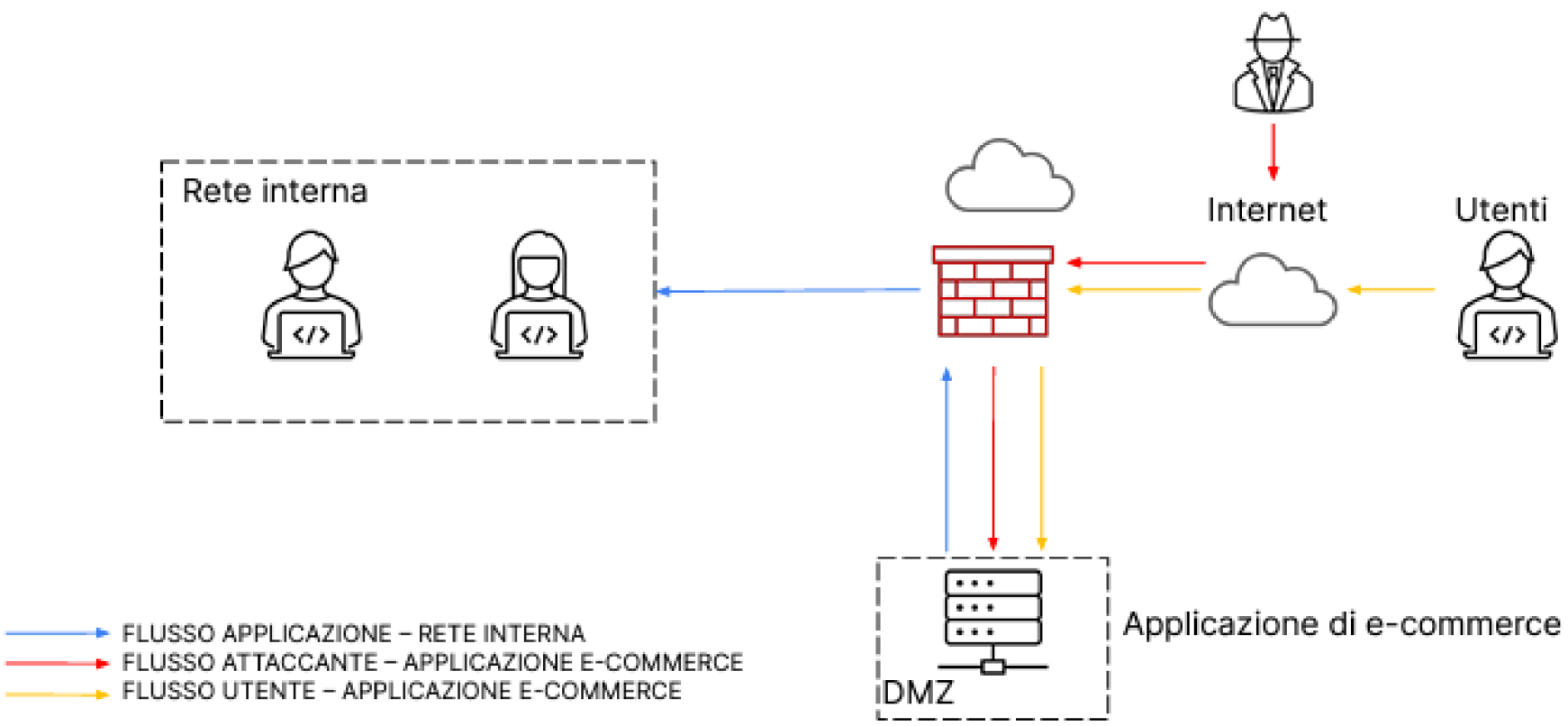
- Scollegare immediatamente la macchina dalla rete: disconnettere fisicamente o logicamente la macchina infetta da qualsiasi rete aziendale, inclusa la rete interna e internet.
- Segmentazione della rete: se l'isolamento fisico non è possibile, utilizzare firewall o switch di rete per segmentare il traffico della macchina infetta, limitandone la comunicazione solo con sistemi controllati.

EPICODE

Esercizio
Traccia e requisiti

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.



Progetto M5 Daniele Atzori

1.Azioni preventive contro SQL Injection (SQLi) e Cross-Site Scripting (XSS)

SQL Injection (SQLi)

- Evitare di concatenare direttamente i dati forniti dall'utente nelle query SQL. Usa i prepared statements che separano la logica della query dai dati forniti dall'utente.
- Usare un ORM come Hibernate, Doctrine o Entity Framework può ridurre il rischio di SQL injection automatizzando l'escape dei dati forniti dall'utente.
- Validare tutti i dati in ingresso per assicurarti che siano nel formato corretto (ad es. e-mail, numeri, stringhe di lunghezza predefinita).
- Sanitizzare gli input per rimuovere o codificare i caratteri pericolosi.
- Assicurarsi che l'utente del database utilizzato dall'applicazione web abbia solo i privilegi necessari. Ad esempio, evitare di utilizzare un account con privilegi di amministratore per operazioni quotidiane.
- Implementare un database firewall che possa rilevare e bloccare query sospette.
- Utilizzare un WAF per filtrare e monitorare il traffico HTTP per attacchi comuni.

Cross-Site Scripting (XSS)

- Sanitizzare tutti i dati in input per rimuovere script o codice HTML non desiderato.
- Utilizzare funzioni di escaping per assicurarti che i dati forniti dall'utente siano trattati come testo normale e non come codice eseguibile.
- Utilizzare le intestazioni HTTP di CSP per limitare le sorgenti da cui possono essere caricate risorse come script e stili.
- Usare framework e librerie che offrono protezioni integrate contro XSS, come Django per Python, Ruby on Rails per Ruby, e Express.js con Helmet per Node.js.
- Non fare affidamento esclusivamente sulla validazione del lato client. Assicurarsi di validare e sanitizzare gli input anche sul server.
- Impostare i cookie con l'attributo `HttpOnly` per evitare che JavaScript lato client acceda ai cookie sensibili.
- Usare l'attributo `Secure` per garantire che i cookie siano trasmessi solo su connessioni HTTPS.

EPICODE

Esercizio
Traccia e requisiti

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.