Relazione – SQL Injection con analisi tramite Wireshark

In questo esercizio ho voluto simulare un attacco di tipo SQL Injection, osservando nel dettaglio cosa succede "dietro le quinte" usando Wireshark, un software che permette di catturare e analizzare il traffico di rete.

L'obiettivo era capire come un'iniezione SQL venga trasmessa sulla rete, attraverso un pacchetto HTTP, e come analizzarla per imparare a riconoscere i segnali di una potenziale vulnerabilità.

Ho lavorato su una macchina virtuale Kali Linux, e ho utilizzato Wireshark per avviare la cattura del traffico di rete. Il file della cattura è stato salvato come `sql_injection_test.pcapng`.

Mi sono collegata al sito *testphp.vulnweb.com*, un ambiente di test pensato proprio per simulazioni di sicurezza. Dopo aver aperto la pagina di login (come si vede negli screen), ho inserito una classica stringa di iniezione SQL:

```
username: ' or '1'='1
password: qualsiasi_valore
```

In un'altra variante, ho usato:

```
username: ' UNION SELECT 1,2,3,4,5,6,7,8-- -
```

Queste stringhe servono a manipolare la query SQL che viene eseguita dal server per forzare un accesso non autorizzato o ottenere dati.

Analisi con Wireshark

Dopo l'attacco, ho filtrato i protocolli nel file pcap per facilitare l'analisi.

HTTP

Nel filtro HTTP ho trovato diverse richieste POST alla pagina */userinfo.php*. Tra i parametri trasmessi nel pacchetto in chiaro si vedono:

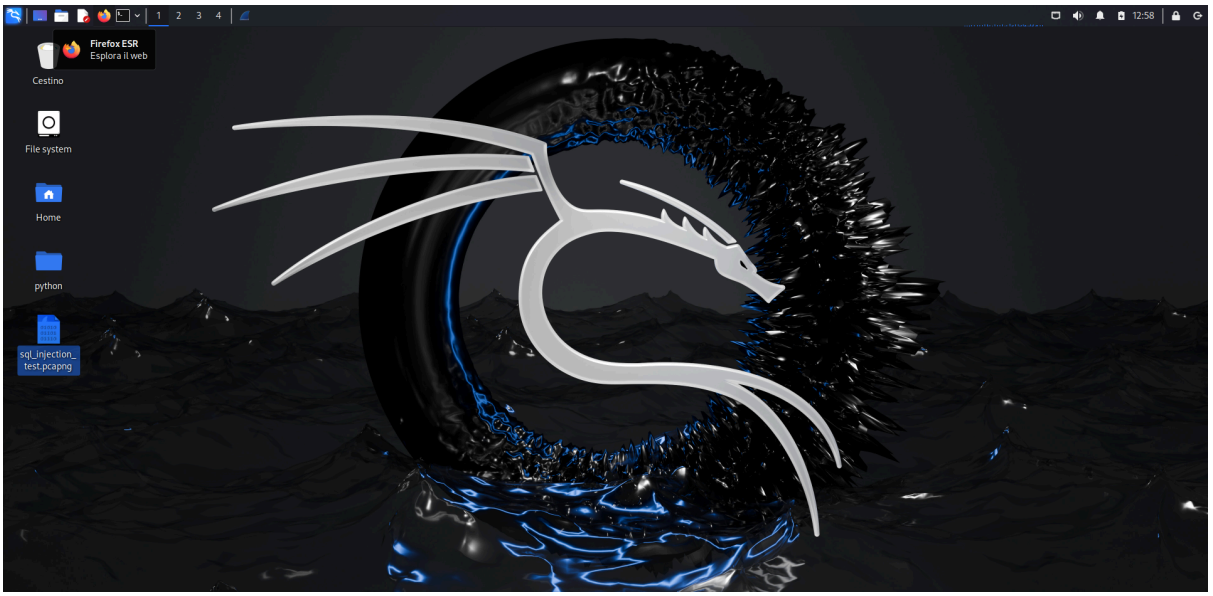
```
uname = ' UNION SELECT 1,2,3,4,5,6,7,8-- -
pass = abc
```

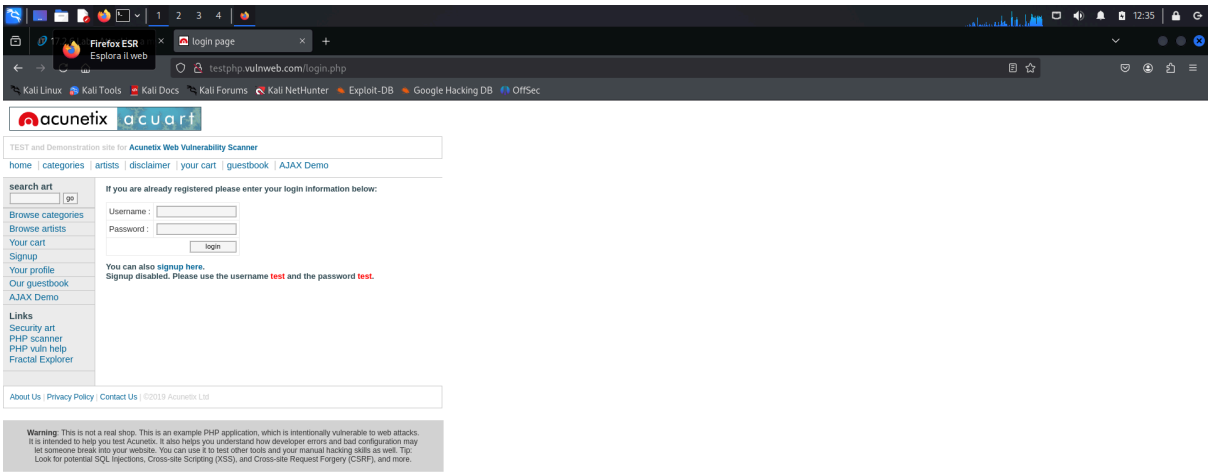
Questo dimostra che l'iniezione SQL è stata inviata con successo, e che l'applicazione non cifra i dati, rendendo visibili tutte le credenziali e i tentativi di attacco. È un esempio chiaro di come una comunicazione non cifrata possa esporre dati sensibili.

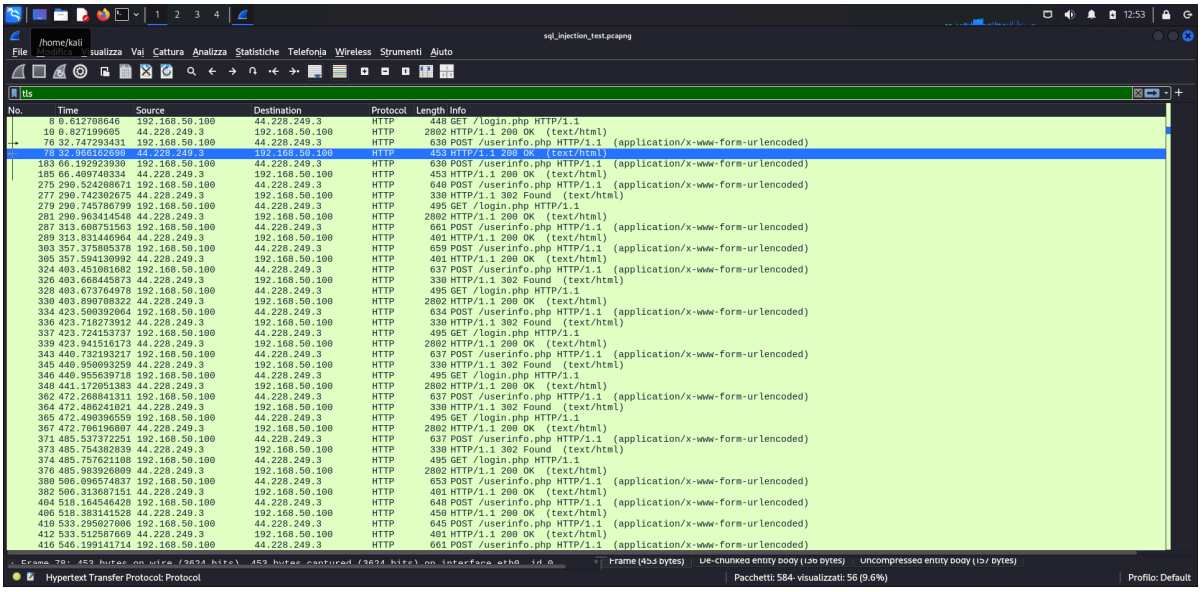
TLS

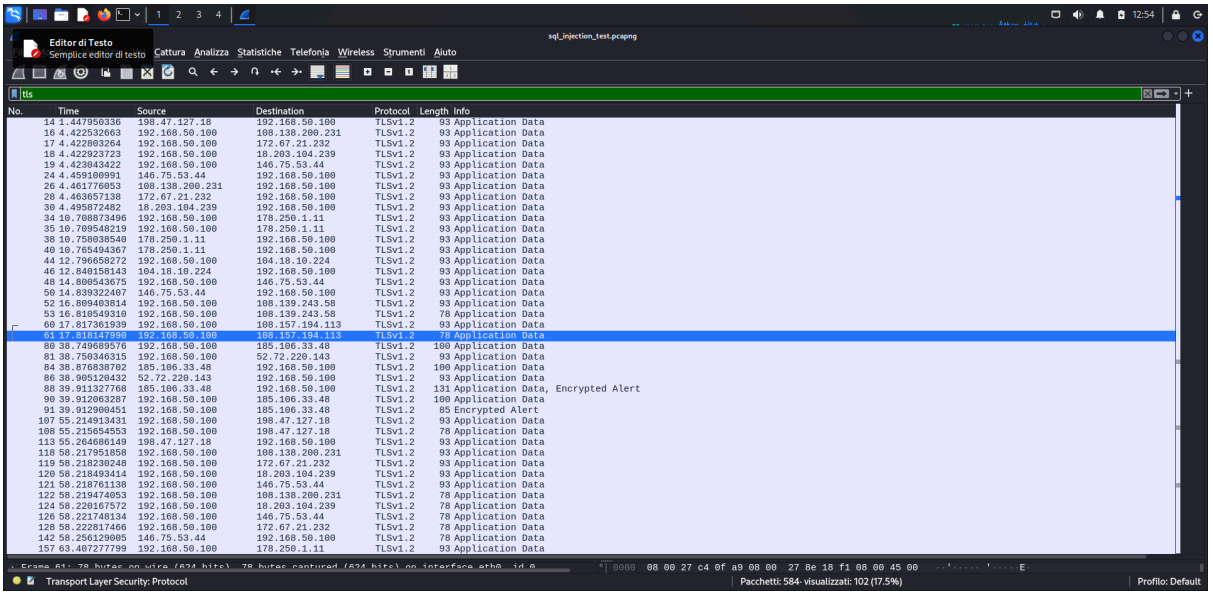
In parallelo ho analizzato il traffico cifrato TLS. Qui, ovviamente, i contenuti non erano leggibili, ma si potevano comunque vedere le connessioni tra il client (192.168.50.100) e diversi IP esterni, indicativi di connessioni cifrate HTTPS.

Screenshot









Conclusioni

Questo esercizio è stato molto utile per capire come funziona un attacco SQL Injection a livello di rete, riconoscere i pacchetti HTTP vulnerabili, in cui le credenziali e i parametri sono visibili, vedere la differenza tra HTTP (in chiaro) e TLS (cifrato) e riflettere sull'importanza della cifratura delle comunicazioni nei siti web, soprattutto quando si usano form di login.

Wireshark permette di isolare e identificare facilmente i pacchetti sospetti con pochi click, il che lo rende uno strumento prezioso sia per la difesa che per l'analisi.