S6L5

Traccia: Nell’esercizio di oggi, viene richiesto di exploitare le vulnerabilità
● XSS stored.
● SQL injection (blind). Presenti sull’applicazione DVWA in esecuzione sulla macchina di laboratorio Metasploitable, dove va preconfigurato il livello di sicurezza=LOW. Scopo dell’esercizio:
● Recuperare i cookie di sessione delle vittime del XSS stored ed inviarli ad un server sotto il controllo dell’attaccante.
● Recuperare le password degli utenti presenti sul DB (sfruttando la SQLi). Agli studenti verranno richieste le evidenze degli attacchi andati a buon fine.

SVOLGIMENTO

La valutazione delle vulnerabilità dei siti web è un processo intricato che implica l'analisi di diversi aspetti, tra cui la configurazione del server, il codice sorgente dell'applicazione web e la gestione delle informazioni sensibili. Di seguito, vengono forniti alcuni indicatori comuni che possono segnalare la presenza di vulnerabilità in un sito web.

Errore di configurazione del server: Una configurazione errata può compromettere informazioni sensibili e renderle accessibili a potenziali attacchi noti. Ad esempio, la visualizzazione dettagliata degli errori potrebbe rivelare informazioni importanti a un utente malintenzionato.

Test di penetrazione: Un penetration test, condotto da esperti di sicurezza, simula un attacco per identificare e correggere le vulnerabilità. Tuttavia, anche se il sito web supera i test, potrebbe essere ancora vulnerabile ad attacchi simili.

Scanner di sicurezza automatico: Gli scanner automatici possono rilevare vulnerabilità comuni come SQL injection e cross-site scripting. Tuttavia, possono generare falsi positivi o ignorare vulnerabilità più complesse.

Vecchia versione del software: L'utilizzo di versioni obsolete del software aumenta il rischio di sfruttare exploit noti. Mantenere aggiornato il software è cruciale per ridurre questo rischio.

Codice sorgente non sicuro: La revisione del codice può rivelare vulnerabilità come la mancata convalida dell'input o la gestione impropria delle sessioni, che possono essere sfruttate dagli aggressori.

Mancanza di protezione contro attacchi comuni: Una difesa inadeguata contro attacchi comuni come SQL injection e cross-site scripting indica una vulnerabilità della sicurezza.

Violazione dei dati: Eventuali violazioni passate potrebbero indicare la presenza di vulnerabilità non ancora corrette. Monitorare attività non autorizzate può aiutare a identificare e risolvere i problemi.

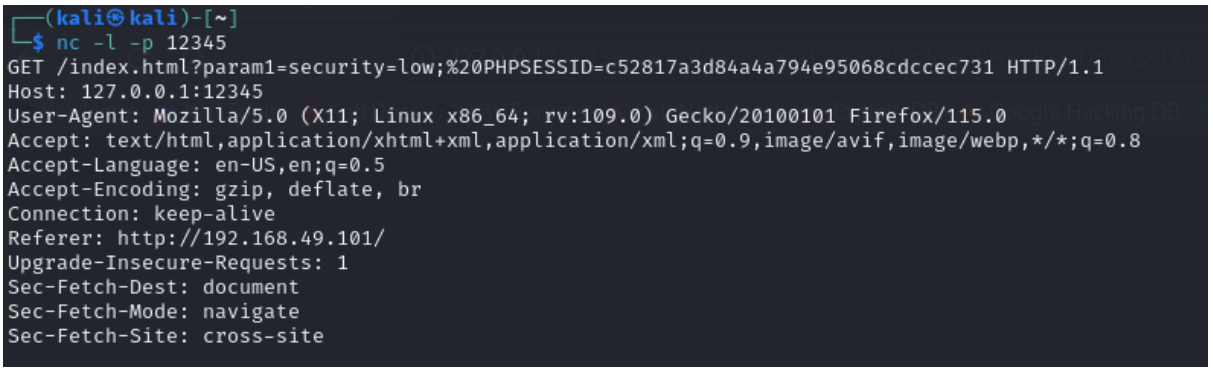
Nessun SSL/TLS: La mancanza di crittografia SSL/TLS rende il sito vulnerabile agli attacchi man-in-the-middle, consentendo agli aggressori di intercettare comunicazioni sensibili.

Mancanza di autenticazione e privilegi: Implementazioni deboli possono consentire a utenti non autorizzati di accedere a risorse sensibili.

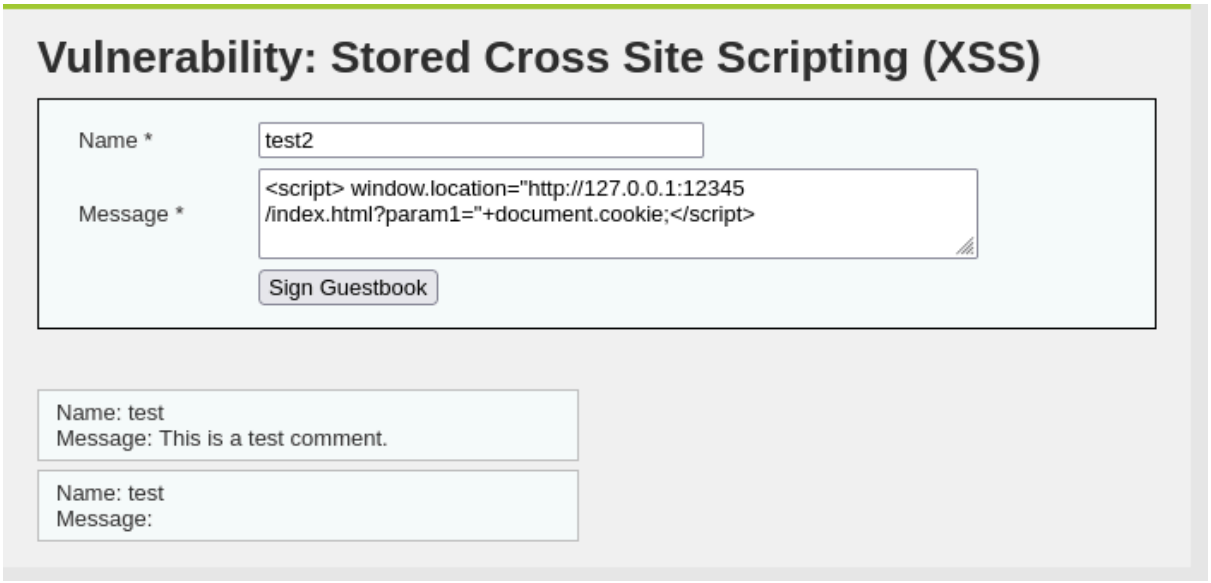
Monitoraggio di attività sospette: Monitorare l'attività del sito per individuare modelli o comportamenti insoliti può aiutare a rilevare una possibile violazione della sicurezza.

XSS stored

Un attacco XSS su un web server in ascolto sulla porta 12345. Viene utilizzato un comando netcat per aprire una connessione in ascolto e catturare una richiesta HTTP inviata al server. Lo script inserito nel testo XSS è progettato per rubare i cookie della macchina vittima. Il report successivo analizza dettagliatamente la richiesta HTTP catturata, indicando il metodo, il percorso, la versione di HTTP, l'host, l'agente utente, i tipi di contenuti accettati, la lingua, la codifica, la connessione, il referer e altri dettagli. In sintesi, il report fornisce informazioni dettagliate sulla richiesta inviata, evidenziando il successo dell'attacco XSS.



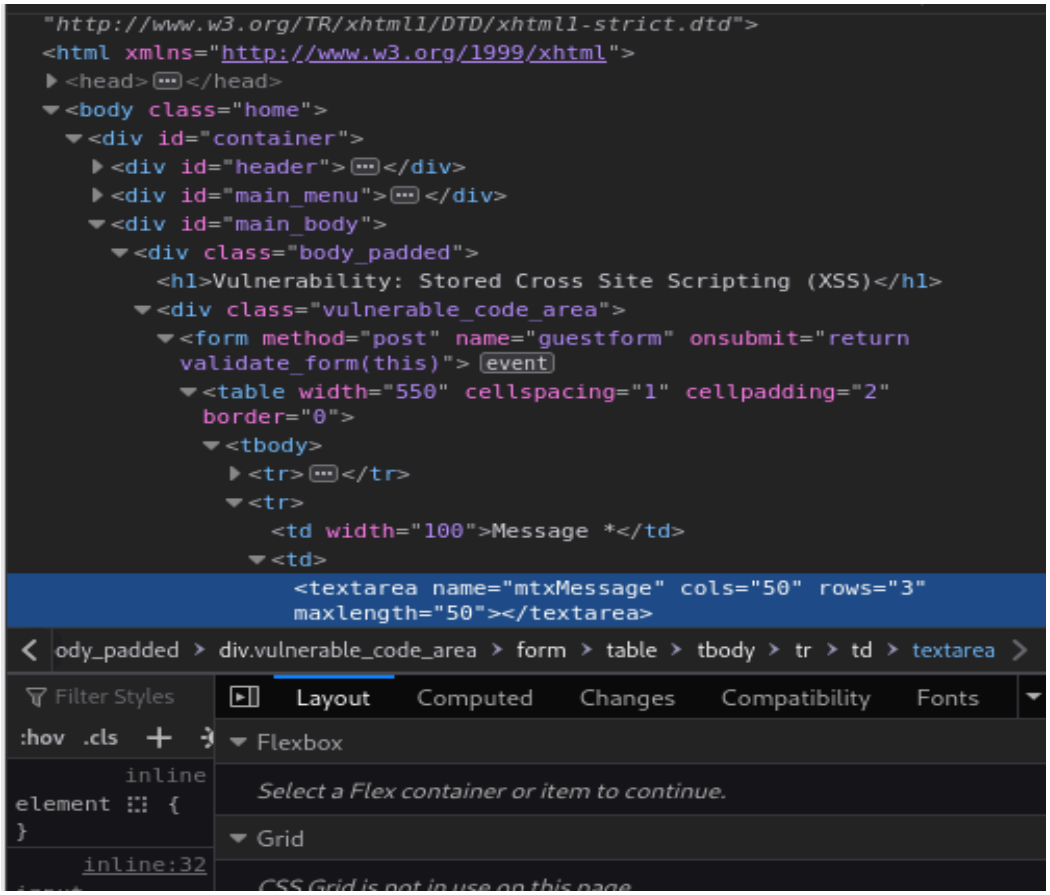
l’attacco:



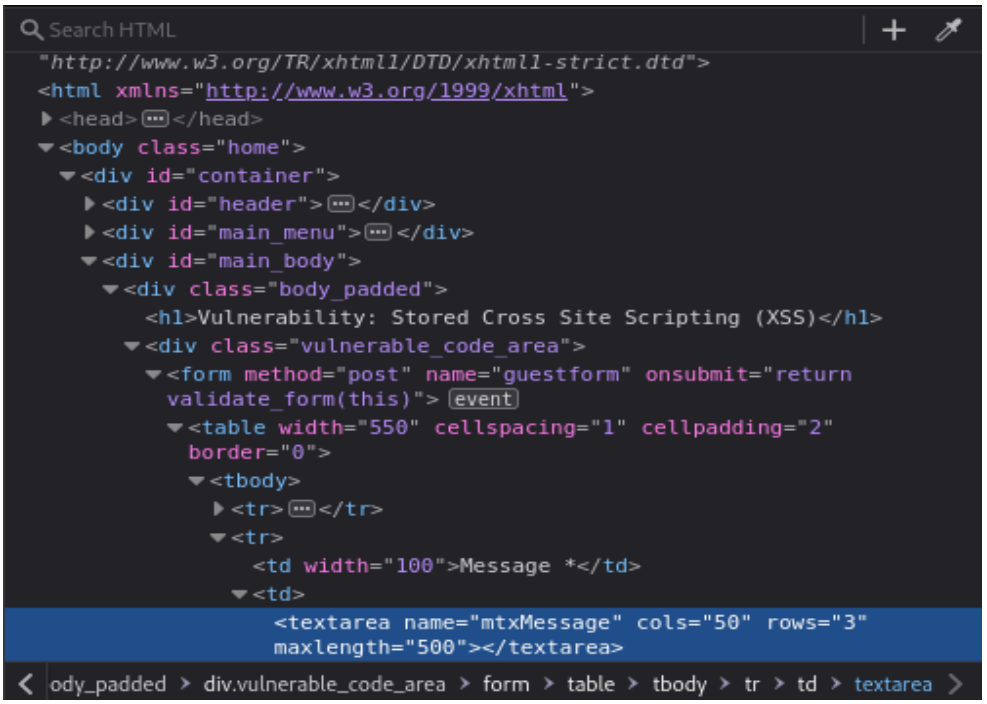
inserendo lo script come nella figura sopra, possiamo vedere che tutto il traffico della rete verrà inviata non al client (come di consueto) bensì alla nostra macchina Kali (l’attaccante)

l’analisi per l’attacco:

andando ad analizzare il codice sorgente della pagina che stiamo andando ad attaccare notiamo che i commenti non sono “sanitizzati”
quindi inserendo un codice malevolo questo verrà eseguito,
ma notiamo anche che i caratteri massimi inseribili in un commento sono 50.



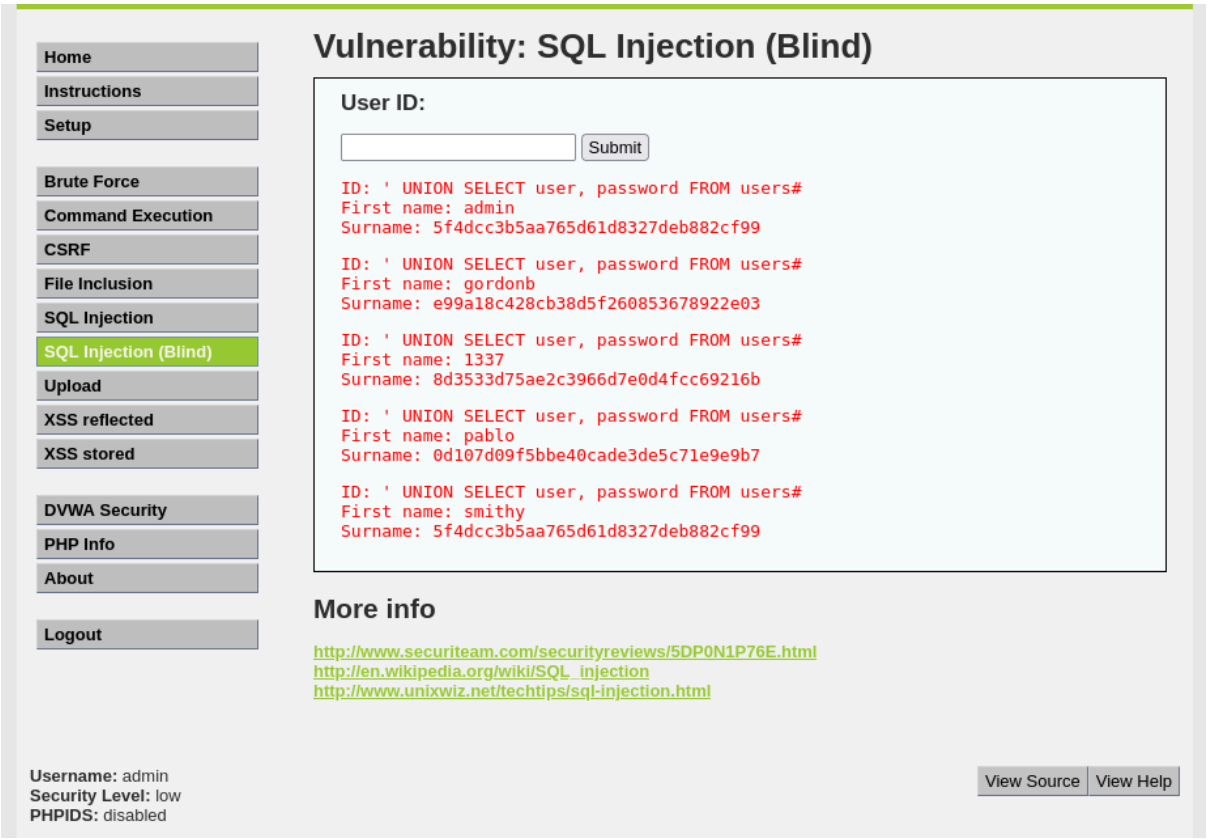
andando ad aggiungere uno “0” (zero), portandolo così a 500 avremo abbastanza caratteri per inviare il nostro codice, in realtà ne bastano molti meno (la lunghezza del nostro script)
dopo la modifica si presenterà in questa maniera:



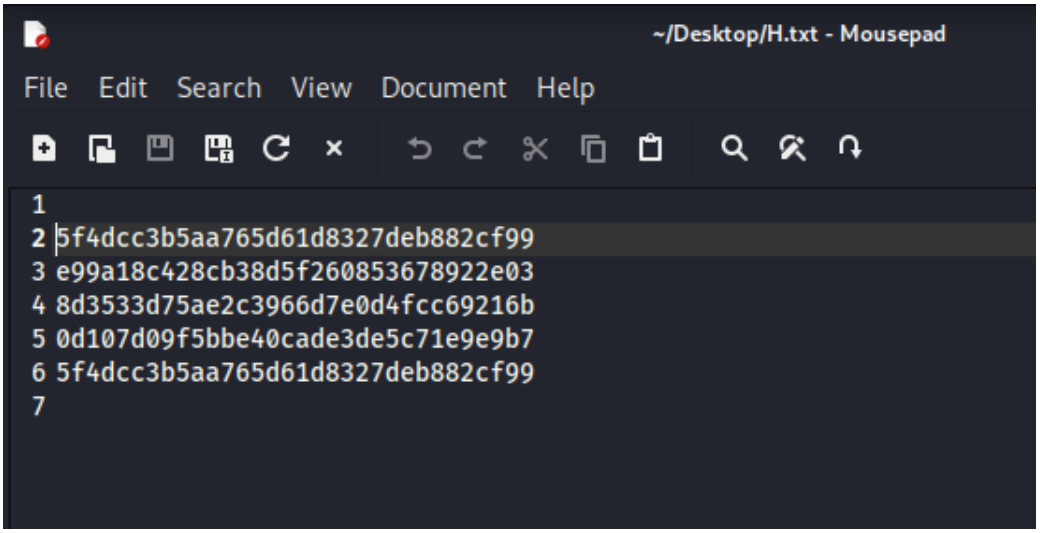
una volta lanciato il commento gli utenti che accedono sulla pagina “corrotta” non riusciranno più a visualizzare la pagina web, ed ignari di tutto ci avranno inviato tutte le informazioni e i cookie di sessione.

SQL injection (blind)

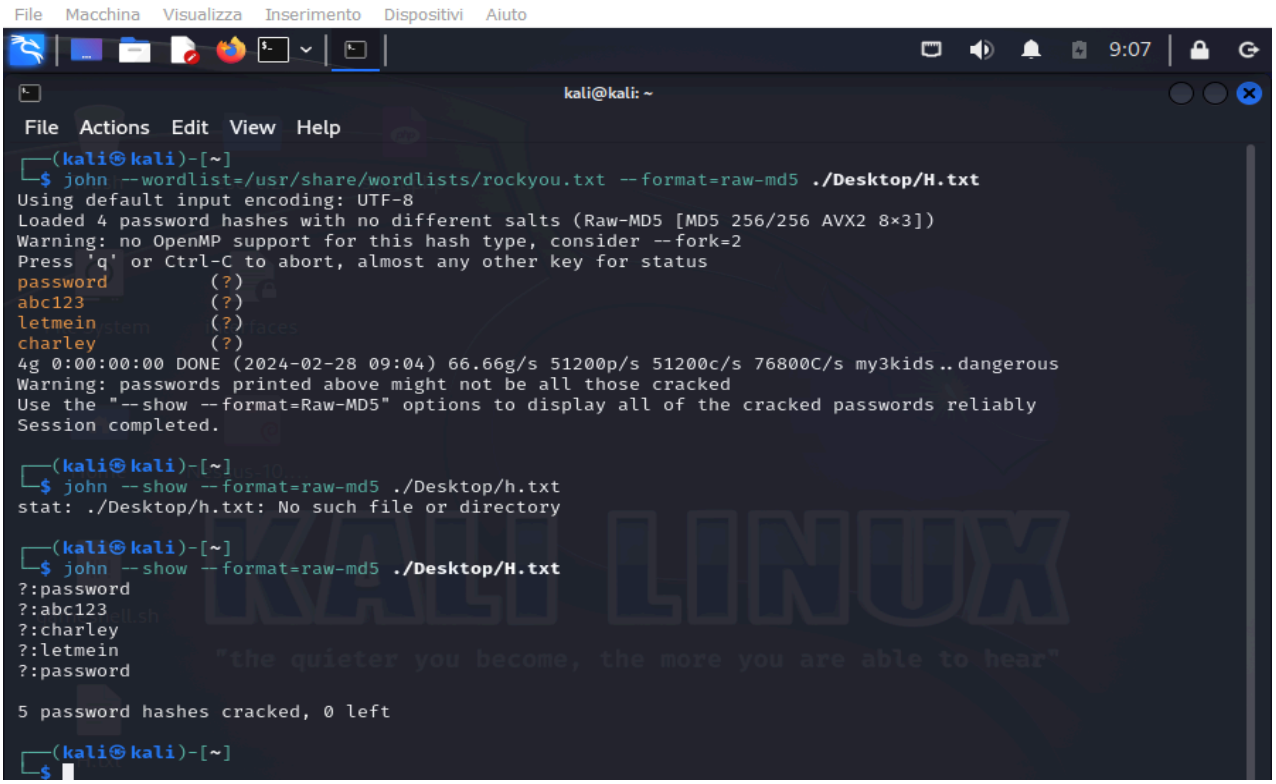
Per cominciare l’attacco, dopo aver loggato su DVWA impostando livello di sicurezza low e inseriamo nel campo di testo User ID la seguente stringa di testo : “' UNION SELECT user, password FROM users#”
Questa stringa che iniettiamo andrà a restituirci vari dati a noi utili come in questo caso le password cifrate in MD5 e sotto forma di hash.



Ora che le abbiamo ottenute, le inseriamo una per riga dentro un file di testo .txt chiamandolo per esempio H.txt per il prossimo passaggio.



Avendo preparato alla fine del passaggio precedente il file H.txt (come sopra) con gli hash delle password all'interno, ora possiamo finalizzare l'obiettivo dell'esercizio di oggi, ossia ricavare le password decifrate degli utenti.
Per fare ciò inseriremo il comando "john --wordlists=/usr/share/wordlists/rockyou.txt --format=raw-md5 ./Desktop/H.txt"
Il comando tutto il necessario al fine di decifrare gli hash ed ottenere le password in chiaro, come dimostra lo screenshot sottostante.



Grazie per l'attenzione

Marco Fasani