Cyber Security Report

W13D4

06/10/2025

Autore :
Pace Massimiliano **email** : *efmpas@gmail.com*

Indice :

INTRODUZIONE

° Obiettivo dell’Esercizio

Il presente report documenta l’attività pratica di penetration testing condotta sull’applicazione web vulnerabile DVWA (Damn Vulnerable Web Application) nell’ambito dell’esercizio W13D4 - Pratica del corso Epicode. L’obiettivo principale è stato quello di applicare le conoscenze teoriche acquisite sulle vulnerabilità web più comuni, in particolare Cross-Site Scripting (XSS) e SQL Injection, in un ambiente controllato e sicuro.

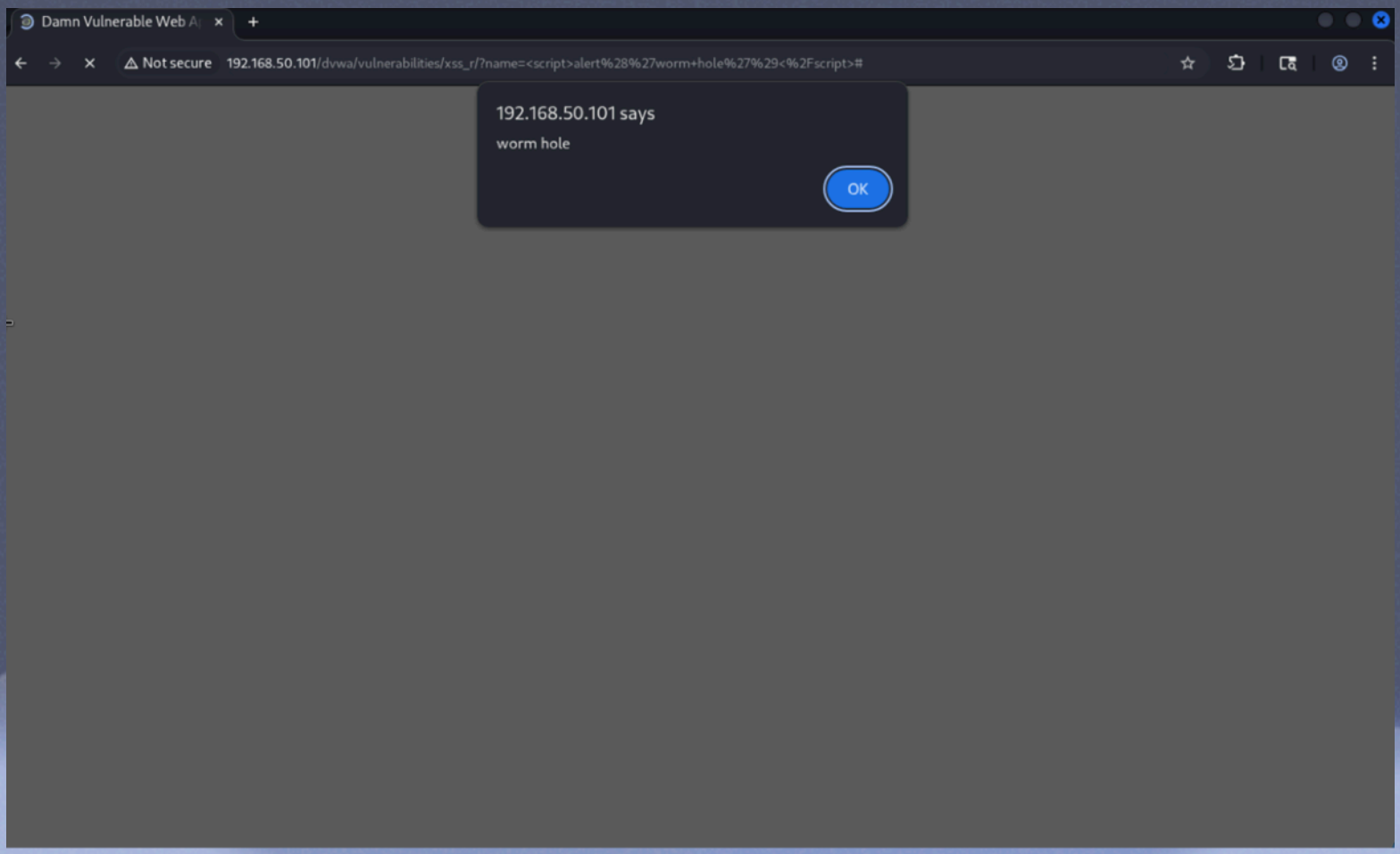
SPIEGAZIONE ESERCIZIO

° Contesto di Sicurezza

Nel panorama attuale della cybersecurity, le vulnerabilità web rappresentano uno dei principali vettori di attacco utilizzati dai cybercriminali. Secondo le statistiche del settore, XSS e SQL Injection figurano costantemente tra le prime dieci vulnerabilità più critiche.

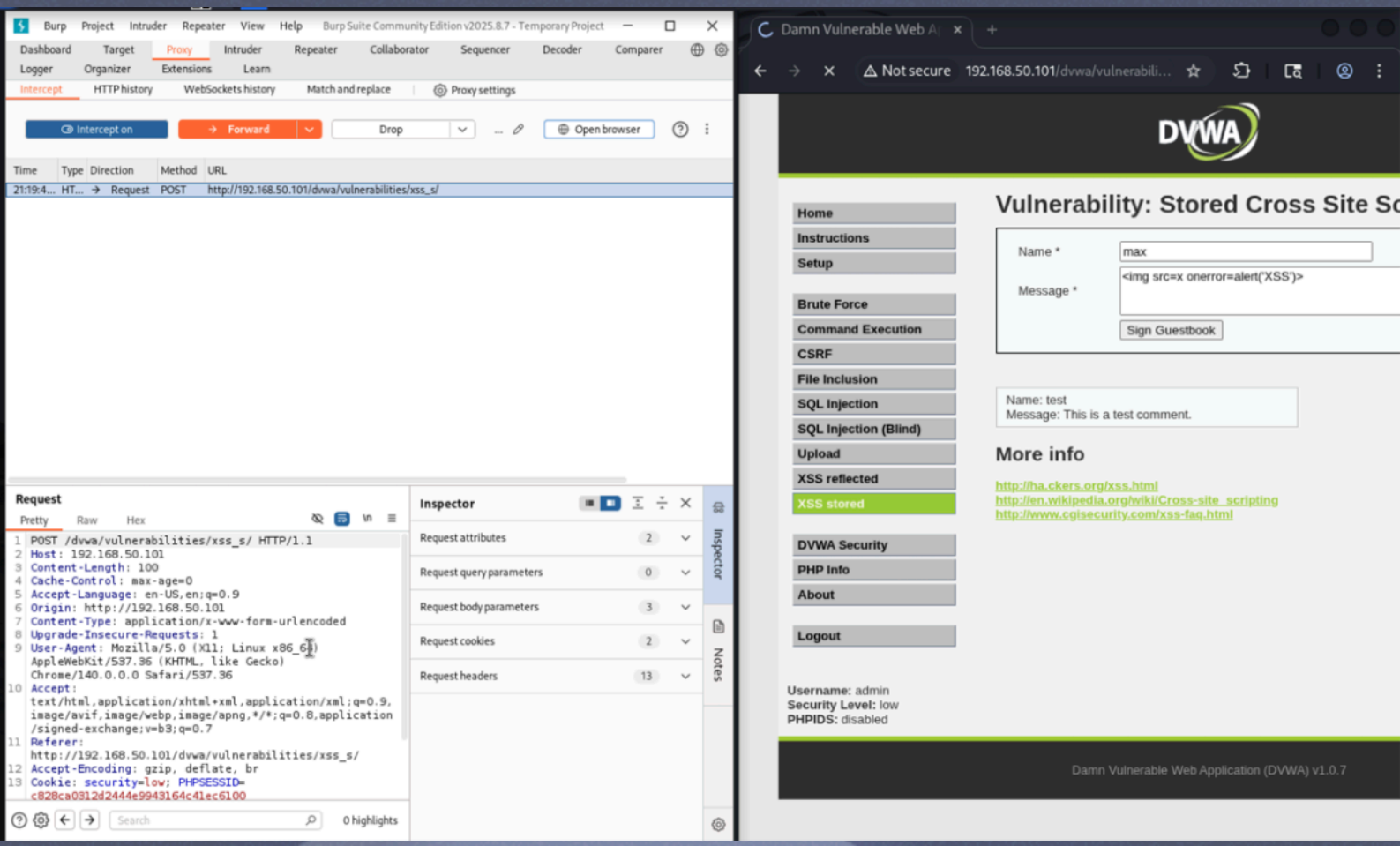
XSS REFLECTION

° **Step 1** : Impostiamo un livello basso di sicurezza nella DVWA
° **Step 2** : Apriamo Burpsuite per poter controllare il traffico
° **Step 3** : Selezioniamo su DVWA XSS Reflection



° **Step 4** : Codice usato
<script>worm hole('XSS')</script>

° **Step 5** : Recuperiamo il nostro Cookie

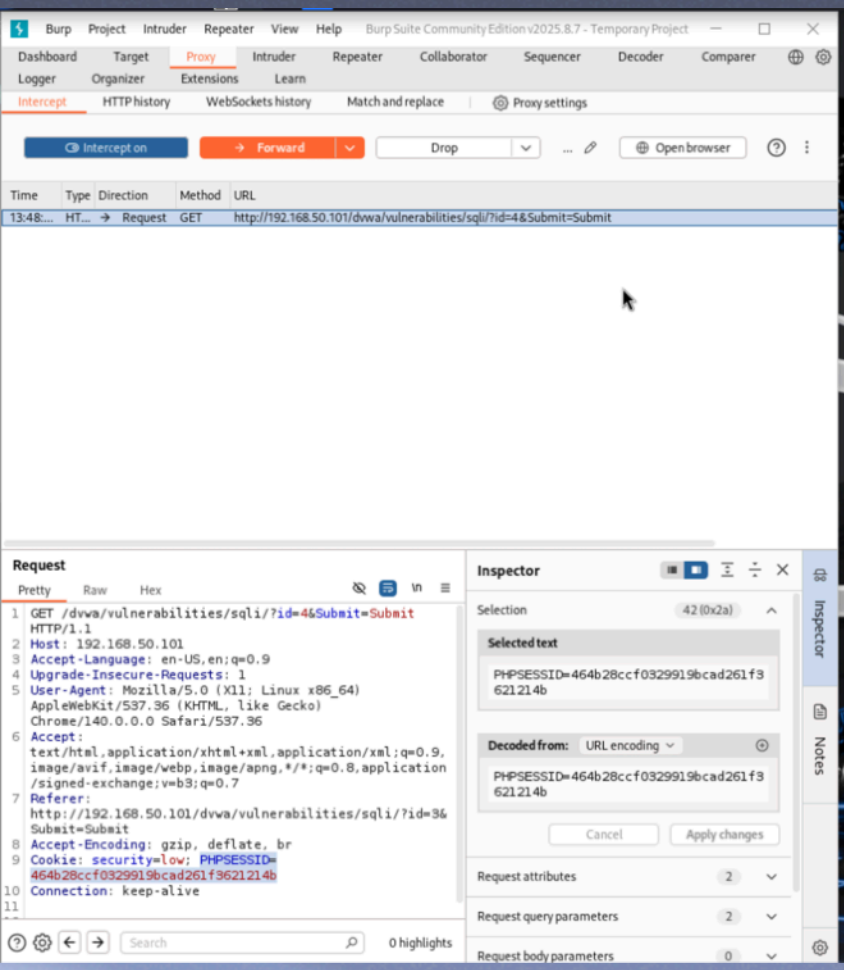


° **Step 6** : Avendo accesso al nostro cookie ora possiamo accedere al sito senza inserire le credenziali , verremo reindirizzati subito all' Home page.

SQL INJECTION

° **Step 1** : Per realizzare questo esercizio mi sono affidato totalmente a Sqlmap

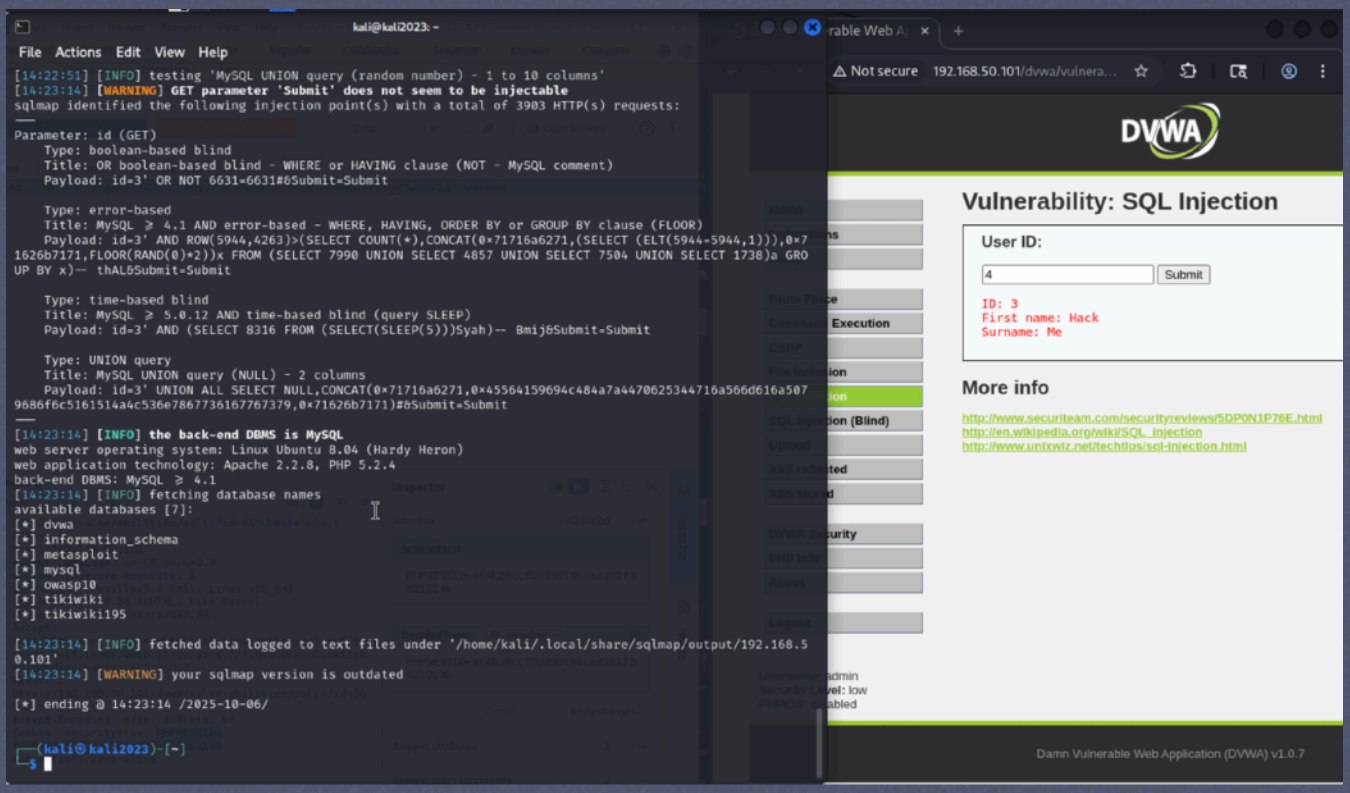
° **Step 2** : Con i dati recuperati in precedenza cookie e url andiamo ad aprire il terminale su Kali



° **Step 3** : Utilizzeremo il codice **sqlmap -u ${u}—cookie=${c}** dove U sta per il nostro url e C per il nostro cookie

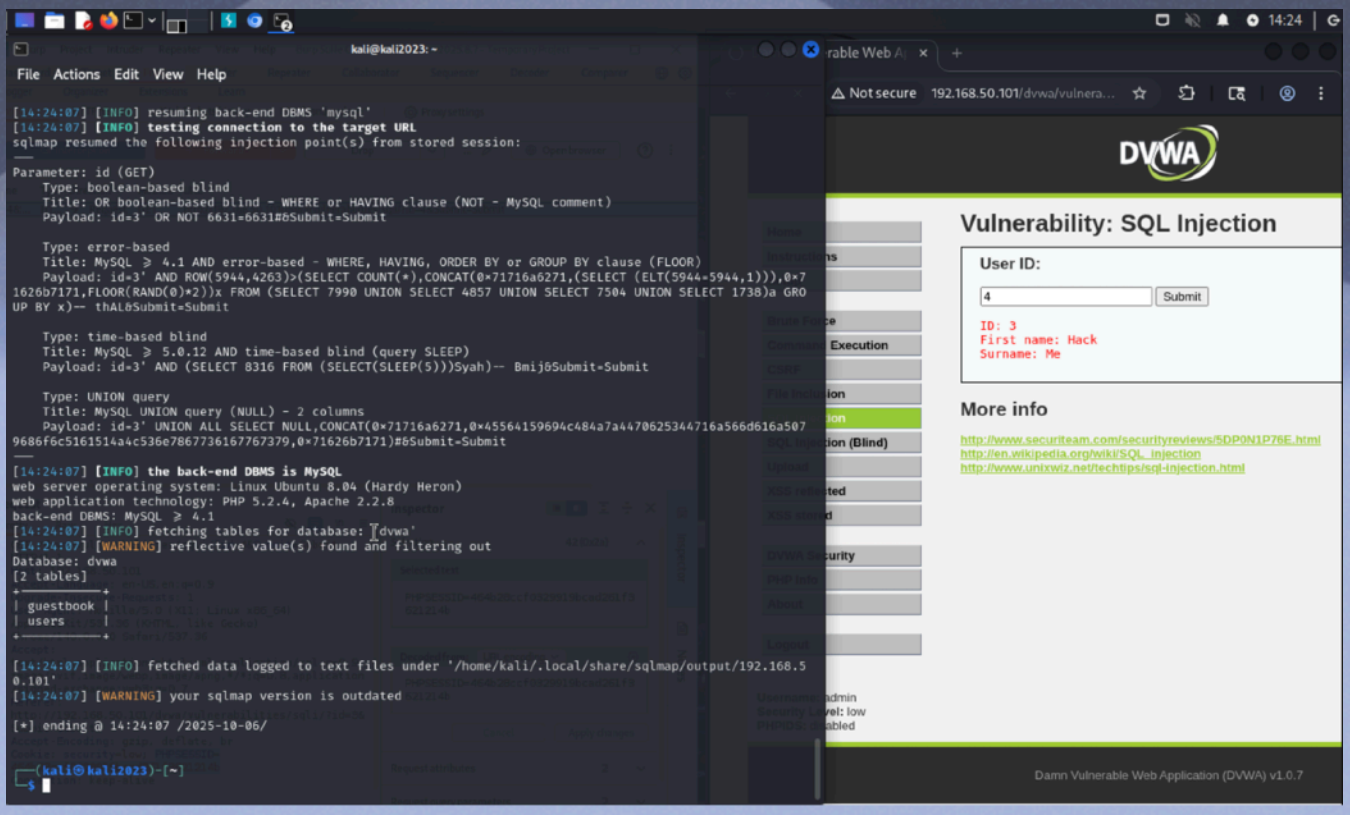
° **Step 4** : cerchiamo la lista dbs(database)

sqlmap -u ${u} --cookie=${c} —dbs # lista db



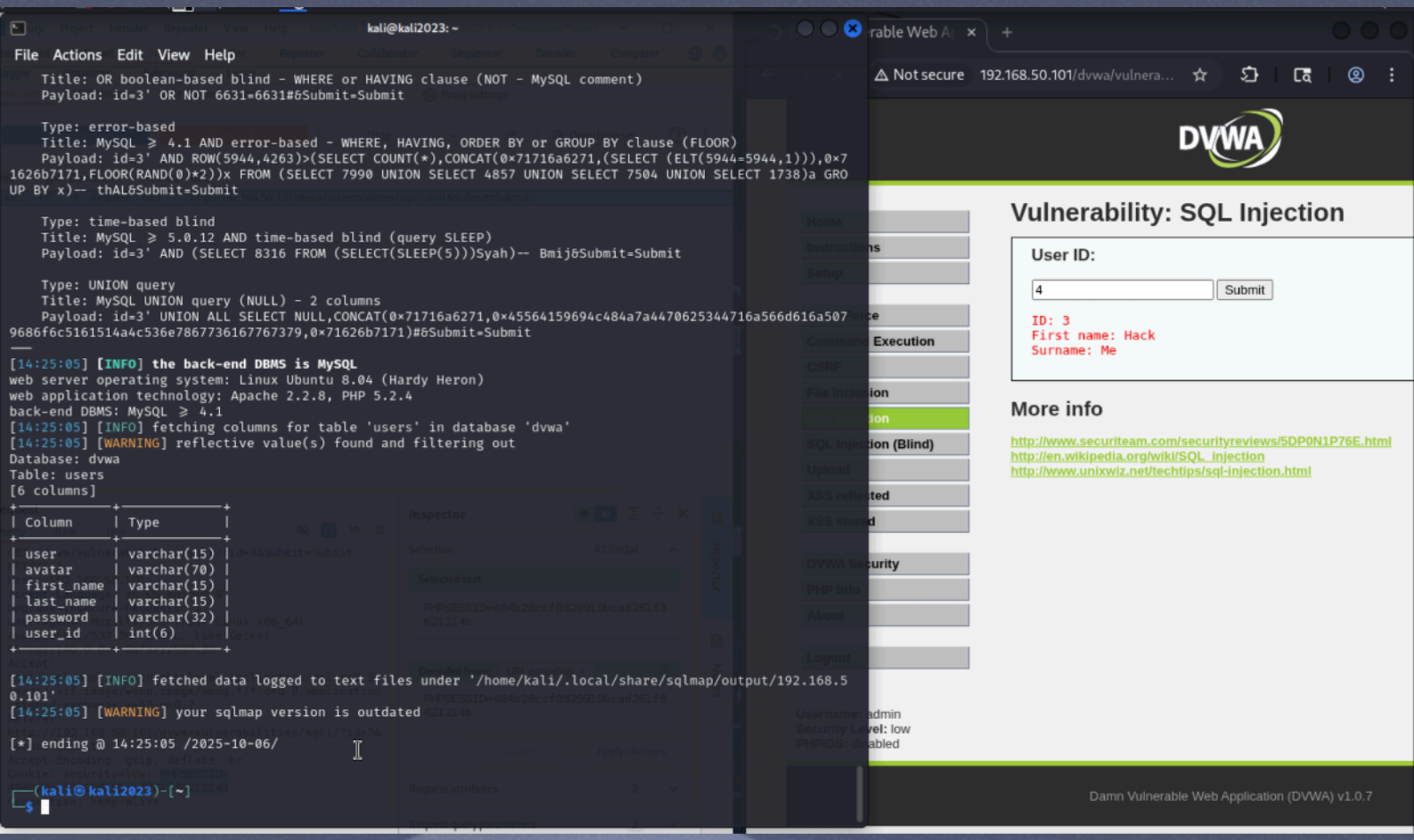
° **Step 5** : cerchiamo le tables (tabelle)

sqlmap -u ${u} --cookie=${c} -D dvwa --tables



° **Step 6** : Cerchiamo le colonne utenti

sqlmap -u ${u} --cookie=${c} -D dvwa -T users — columns



Conclusioni

L'analisi e lo sfruttamento della vulnerabilità SQL Injection sull'applicazione DVWA hanno dimostrato in modo inequivocabile la criticità di questa tipologia di attacco e l'impatto devastante che può avere sulla confidenzialità, integrità e disponibilità dei dati gestiti da un'applicazione web

Riflessioni

L'utilizzo di strumenti come SQLMap ha dimostrato come il processo di individuazione e sfruttamento di una SQL Injection possa essere quasi completamente automatizzato. Questo riduce drasticamente il livello di competenza tecnica richiesto a un potenziale attaccante e aumenta la velocità e l'efficacia dell'attacco, rendendo la vulnerabilità ancora più pericolosa.
L'esercizio sottolinea in modo critico la necessità di adottare un approccio di "defense-in-depth" per mitigare il rischio di SQL Injection. La sola sanitizzazione dell'input non è sufficiente. È imperativo l'utilizzo di Prepared Statements (con query parametrizzate), che separano nettamente il codice SQL dai dati forniti dall'utente, eliminando alla radice la possibilità di iniezione. Questa è l'unica contromisura realmente efficace e dovrebbe essere considerata uno standard obbligatorio in qualsiasi sviluppo di applicazioni web moderne, come dimostrato dal livello "Impossible" di DVWA, che risulta immune a questo tipo di attacco.