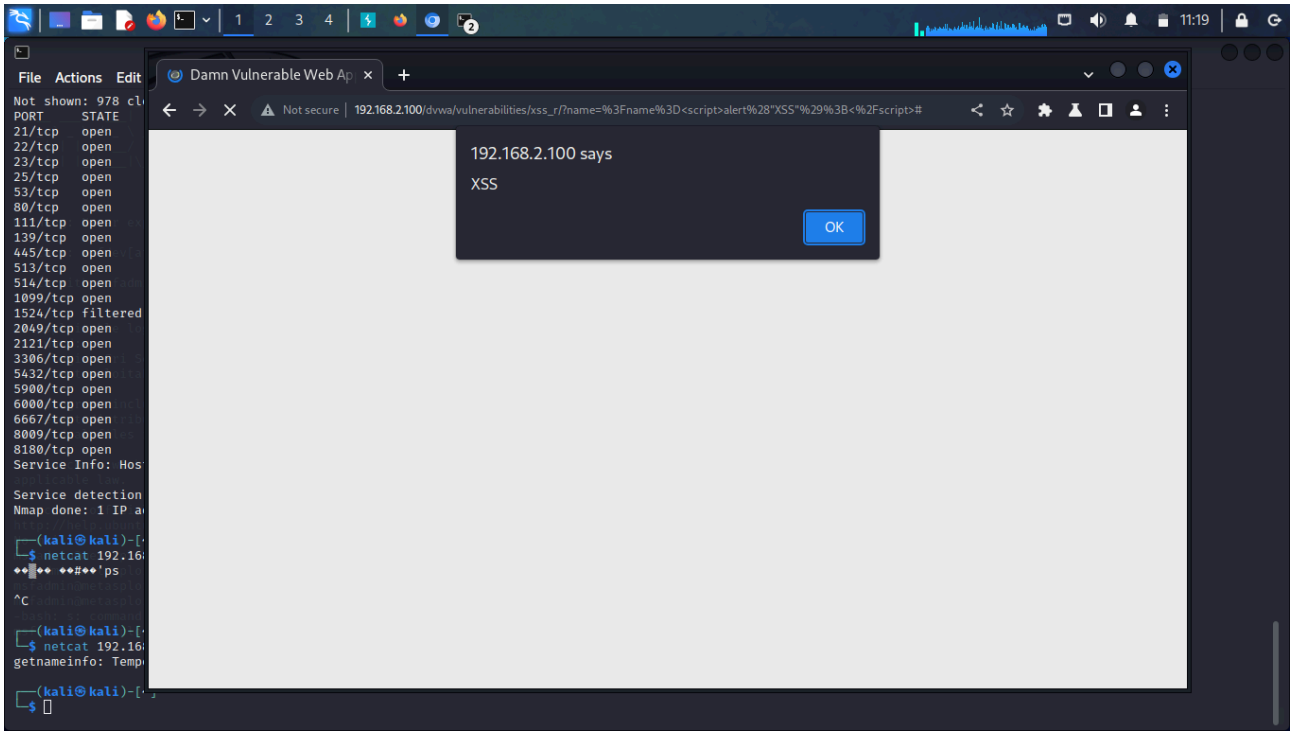
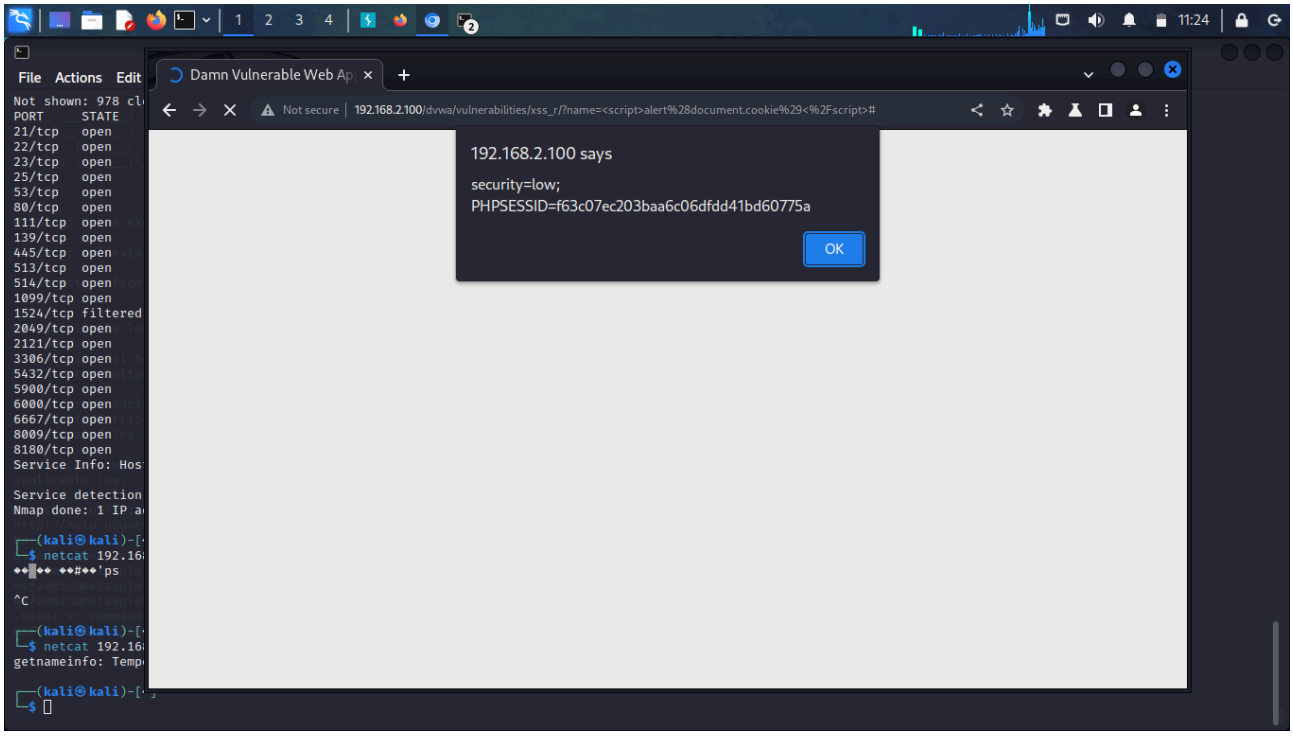
XSS Reflected e SQL Injection

Per quanto riguarda l'XSS la vulnerabilità risulta in quanto non c'è nessuna sanificazione dell'input da parte della web app. Questo ci può permettere di alterare il linguaggio html/js della pagina senza che una potenziale vittima se ne accorga.



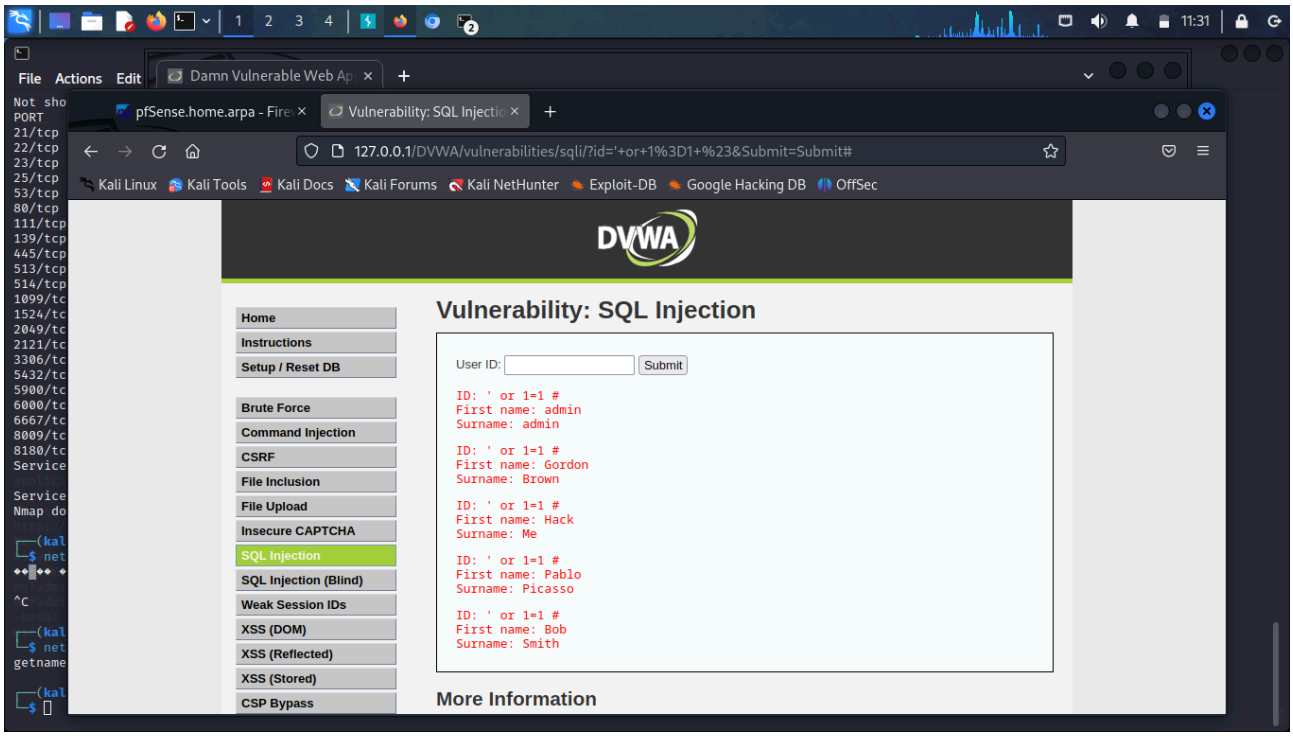
Qui possiamo vedere che con "?name=<script>alert("XSS");</script>" il form prende tale quale l'input utente e crea una finestra di avviso sul browser.

Semplicemente scrivendo <script>alert(document.cookie)</script> sul form possiamo rubare il cookie di sessione di una potenziale vittima



Lo stesso discorso vale per l'sql injection in quanto nella casella l'input dell'utente non viene in nessun modo controllato. Questo ci permettere di alterare a nostro piacimento le query sul DB.

usando il classico '%' or 1=1 e le sue varianti possiamo già tirare fuori tutti i nomi e cognomi dal database, in quanto la condizione 1=1 è sempre vera.



Selezionando con una union una select (con lo stesso numero di statements della prima) user and password from user, possiamo tirarci fuori tranquillamente le hash delle password degli utenti.