Report Sfruttamento delle Vulnerabilità XSS e SQL Injection sulla DVWA

Obiettivo dell'Esercizio: configurare un ambiente virtuale per sfruttare due vulnerabilità comuni nelle applicazioni web:

- XSS (Cross-Site Scripting) Reflected
- SQL Injection (non-blind)

Passaggi Svolti

1. Configurazione del Laboratorio
   - Sono stati configurati due ambienti virtuali:
     - DVWA (vittima): Un'applicazione vulnerabile ospitata su una macchina virtuale.
     - Kali Linux (attaccante): Sistema operativo usato per condurre gli attacchi.
   - È stata verificata la comunicazione tra le due macchine tramite il comando ping.
2. Accesso alla DVWA
   - Tramite il browser su Kali Linux, si è acceduto all'interfaccia web della DVWA.
   - Si è navigato fino alla pagina di configurazione e il livello di sicurezza è stato impostato su LOW per massimizzare la vulnerabilità.
3. Sfruttamento delle Vulnerabilità
   - XSS Reflected:
     - È stato testato uno script malevolo ***'<script>alert('Attenzione, il pc sta per esplodere!')</script>***' in un campo vulnerabile, ottenendo l'esecuzione del codice sul browser.
   - SQL Injection (non-blind):
     - È stato usato un payload come ***' OR '1'='1*** per accedere al database senza credenziali valide e manipolare i dati.
     - Sono stati usati altri payload come ***' UNION SELECT table_schema,table_name FROM information_schema.tables#*** per cercare ulteriori informazioni su tabelle, colonne etc..

Attività con SQLMap

Durante l'esercizio, è stato utilizzato SQLMap, uno strumento di automazione per l'individuazione e lo sfruttamento di vulnerabilità SQL Injection. I passaggi seguiti sono stati:

1. Configurazione dei Cookie
   - È stato usato un cookie di sessione, impostato con:
     - ***cookie="security=low; PHPSESSID=b02abea11eab9b4ed4276a0e1f67434d***
   - URL target:
     - ***url=http://192.168.50.20/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit***

2. Comandi SQLMap

Questo comando ha permesso di identificare tutte le tabelle presenti nel database dvwa:

sqlmap --cookie="${cookie}" -u "${url}" -D dvwa –tables

Questo comando ha permesso di identificare tutte le colonne con user e password presenti nel database dvwa:

sqlmap --cookie="${cookie}" -u "${url}" -D dvwa -T users --columns -C user,password –dump

Questo comando ha permesso di identificare tutto ciò che è presente nel database dvwa:

sqlmap --cookie="${cookie}" -u "${url}" -D dvwa –dump

Conclusioni

- XSS Reflected: Dimostrato come un campo non sanificato possa essere sfruttato per eseguire script arbitrari.
- SQL Injection: Evidenziato come una query SQL mal formata possa aggirare i meccanismi di autenticazione.

L'esercizio ha permesso di comprendere le tecniche di attacco e l'importanza di implementare contromisure come:

- Validazione degli input
- Preparazione delle query parametrizzate

Con SQLMap, è stata automatizzata l'estrazione delle informazioni sensibili, dimostrando l'importanza della protezione contro attacchi di questo tipo.

È stata confermata la possibilità di accedere a dati riservati, mostrando la vulnerabilità del sistema.

Grazie a questo approccio, ho acquisito competenze pratiche su:

- Uso di strumenti avanzati come SQLMap per test di penetrazione.
- L'importanza di proteggere i cookie di sessione e prevenire attacchi SQL Injection tramite tecniche come:
    - Prepared Statements
    - Sanificazione degli input
    - Limitazione delle informazioni fornite dai messaggi di errore.