



REPORT

Web Application Exploit SQL Injection

Obiettivo

L'obiettivo di questo report approfondisce il contenuto del documento che illustra un attacco SQL Injection sulla macchina target Metasploitable2 per estrarre password da un database vulnerabile.

Per raggiungere questo scopo, è stata utilizzata la web application vulnerabile DVWA (Damn Vulnerable Web Application) come ambiente di test.

Cos' è SQL Injection

L'attacco SQL Injection è una tecnica che consiste nell'inserire codice malevolo in una query per estrarre informazioni dal database. L'obiettivo è recuperare dati sensibili, come credenziali di accesso, che normalmente non sarebbero visibili agli utenti.

Scenario d' attacco

L'obiettivo è estrarre la password dell'utente Pablo Picasso in formato hash e successivamente riportarla in chiaro.

Step 1

Per prima cosa ci si collega alla piattaforma DVWA di Metasploitable2.

Successivamente si effettua l' accesso tramite le credenziali

Step 2

Una volta effettuato il login, si accede alla sezione DVWA Security.

Impostare il livello di sicurezza su Low per protezioni avanzate.

The screenshot shows the DVWA Security interface. On the left is a sidebar menu with various exploit categories: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload, XSS reflected, and XSS stored. Below these is a green bar labeled 'DVWA Security'. Underneath the sidebar, the current session information is displayed: Username: admin, Security Level: low, and PHPIDS: disabled. The main content area is titled 'DVWA Security' and contains a section for 'Script Security' where the security level is currently set to 'low'. A red box highlights the 'low' dropdown menu. Below this is a 'PHPIDS' section which is currently 'disabled'. At the bottom of the page, a footer bar indicates 'Damn Vulnerable Web Application (DVWA) v1.0.7'.

Esecuzione dell' attacco SQL Injection

A questo punto, ci si sposta nella sezione SQL Injection della DVWA.

Step 3

Nel campo USER ID, si inserirà la seguente query malevola:

1' UNION SELECT user, password FROM users#

Una volta lanciata la query, restituirà come risultato, gli utenti registrati all' interno del database con i loro nomi e gli hash delle password, incluse le informazioni sull' utente target Pablo Picasso.

The screenshot shows the DVWA application interface. On the left, a sidebar lists various security vulnerabilities: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection (selected), SQL Injection (Blind), Upload, XSS reflected, XSS stored, DVWA Security, PHP Info, About, and Logout. The main content area is titled "Vulnerability: SQL Injection". It features a "User ID:" input field and a "Submit" button. Below the input field, several user records are displayed in red text, each starting with "ID: 1' UNION SELECT user, password FROM users#". The last two records are highlighted with a blue border: "First name: pablo Surname: 0d107d09f5bbe40cade3de5c71e9e9b7" and "First name: smithy Surname: 5f4dcc3b5aa765d61d8327deb882cf99". At the bottom of the page, there is a "More info" section with three links: <http://www.securiteam.com/securityreviews/5DP0N1P76E.html>, http://en.wikipedia.org/wiki/SQL_injection, and <http://www.unixwiz.net/tctips/sql-injection.html>. The footer of the page reads "Damn Vulnerable Web Application (DVWA) v1.0.7".

Esecuzione dell' attacco SQL Injection

GLi hash delle password recuperate sono in formato MD5. Per avere le password in chiaro si utilizzerà il software John the Ripper.

Step 4

Attraverso questo software è possibile scoprire la password in chiaro partendo da un hash.

Si procede con l' inserimento dell' hash in un file di testo, esempio *Pablo.txt*.

Successivamente si utilizzerà il software John the Ripper tramite il seguente comando:

```
john --format=RAW-MD5 Pablo.txt
```

```
(kali㉿kali)-[~/Desktop]
$ john --format=RAW-MD5 Pablo
Created directory: /home/kali/.john
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 SSE2 4x3])
Warning: no OpenMP support for this hash type, consider --fork=4
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
letmein      (?)
1g 0:00:00:00 DONE 2/3 (2024-11-18 04:20) 10.00g/s 1920p/s 1920c/s 1920C/s 123456.. knight
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.
```

Come si può notare dall' immagine, dopo aver eseguito il comando, il risultato restituito è la password in chiaro **letmein**.

Considerazioni su misura di sicurezza maggiore

Se DVWA fosse configurato con un livello di sicurezza più alto (medium), il server filtrerebbe o sintetizzerebbe l'input dell'utente, bloccando il carattere ' (virgoletta).

Così facendo, la query ***1' UNION SELECT user, password FROM users#*** non funzionerebbe poiché il filtro impedirebbe l'esecuzione della parte malevola.

Strategia di Bypass:

Per oltrepassare questo filtraggio, si deve modificare la query. Ad esempio, eliminando il carattere ' (virgoletta) o sostituendolo con alternative valide, come: ***1 UNION SELECT user, password FROM users --***. Questa variante sfrutta un diverso metodo di commento (--).

Conclusioni

L'attacco dimostra quanto sia cruciale proteggere le applicazioni da input non sanitizzati.

Strumenti come *John the Ripper* amplificano l'impatto dell'attacco, trasformando gli hash delle password in testo leggibile.

Implementando misure di sicurezza come l'uso di query parametrizzate e una corretta configurazione del server è essenziale per prevenire SQL Injection.