SQL Injection and Security

PDBM 15.3.5

Dr. Chris Mayfield

Department of Computer Science
James Madison University

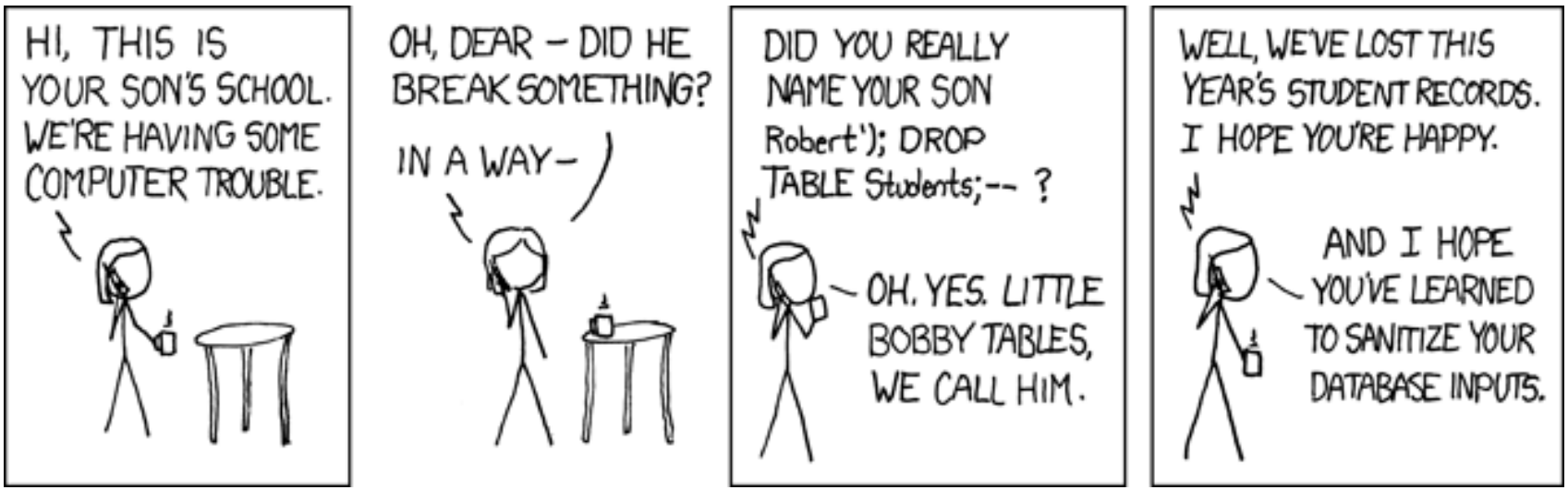
Mar 24, 2022

Database security 101

- Access control, users/groups
- Views (for limiting access)
- Encryption (e.g., passwords)
- Denial of service attacks
- Fault tolerance (hot standby)
- Privacy of user's information
- Audit trail (using triggers?)

https://commons.wikimedia.org/wiki/File:Hacker_Inside_Logo.svg

Exploits of a Mom



https://xkcd.com/327/

Traffic cameras?

https://danielmiessler.com/blog/a-fantasy-explanation-of-standard-vs-blind-sql-injection/

Other examples

NEVER CONCATENATE USER INPUT!

```
String sql = "SELECT * FROM users\n"
           + "WHERE name = '" + userName + "';"
```

Hello, my name is: `' OR '1'='1`

```
SELECT * FROM users
WHERE name = '' OR '1'='1';
```

Or, my password is: `' OR 1=1;--`

```
SELECT * FROM users
WHERE name = '' OR 1=1;--';
```

Little Bobby Tables: `'; DROP TABLE users;--`

```
SELECT * FROM users
WHERE name = ''; DROP TABLE users;--';
```

SQL injection attacks

- Adding or modifying data
  - Denial of service
  - Privilege escalation
- Bypassing authentication
  - Evading detection
  - Executing remote commands
- Extracting data
  - Identifying injectable parameters
  - Inferring sensitive information

https://en.wikipedia.org/wiki/SQL_injection
https://www.unixwiz.net/techtips/sql-injection.html

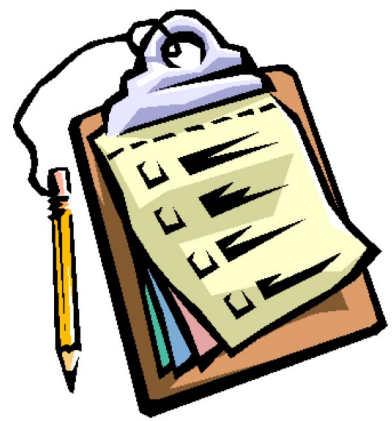
How to prevent attacks

Your application should:

- **Validate all user input**
- Use parameter substitution (i.e., `PreparedStatement`)
- Use stored procedures (SQL functions, views)

Your user account should:

- Have minimal privileges
  - Create application-specific user accounts
  - Never use admin account for applications!



Your db server should:

- Be separate from your web/app servers
- Install security patches when released

REMEMBER

Don't make *any* assumptions about user input.
NEVER concatenate user input with code like this!

Java:

```
String username = request.getParameter("username");
   String sql = "SELECT ..." + username + "...";
```

PHP:

```
$username = $_POST["username"];
$sql = "SELECT ..." . $username . "...";
```

Python:

```
username = request.args.get("username", "")
   sql = "SELECT ..." + username + "..."
```