

CSC309 Tutorial – Security

March 25, 2010

Login security

- Encrypt sensitive user data
- Store only hashed passwords
- Use unique session IDs for logged-in users
- Remember, a malicious user can send whatever cookies he/she wants
- Make sure cookies eventually expire
- Store the IP address a login cookie is given to and check for a match when the cookie is used

Configuration data

- ALWAYS disable debug information on live servers (even during testing)
- Don't show the version of backend software
- The more the outside world knows about your server, the easier it is to exploit

File Transfer

- If you let users upload files, be sure the file size and location is properly controlled
- If you let users download files, check the path the files are being downloaded from

Processing user data

- Never trust the user
- Don't dump user data directly into pages
- Make sure all requests are properly formed
- Never allow JavaScript to be run from an external site

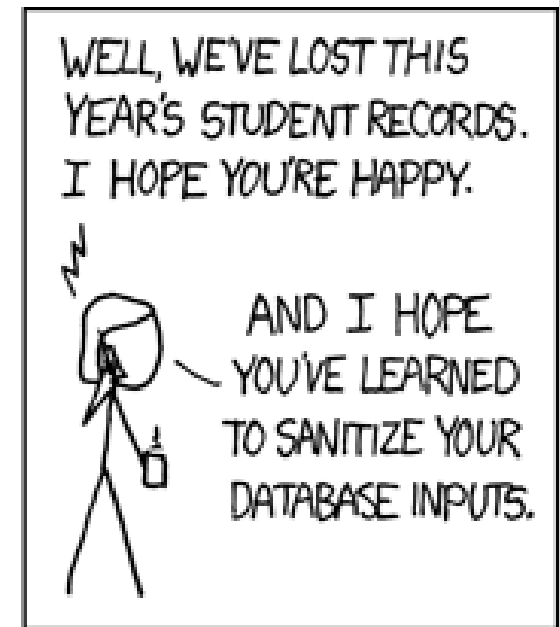
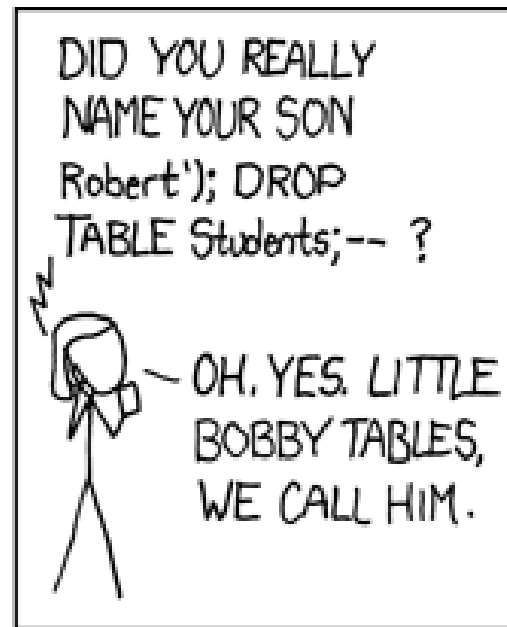
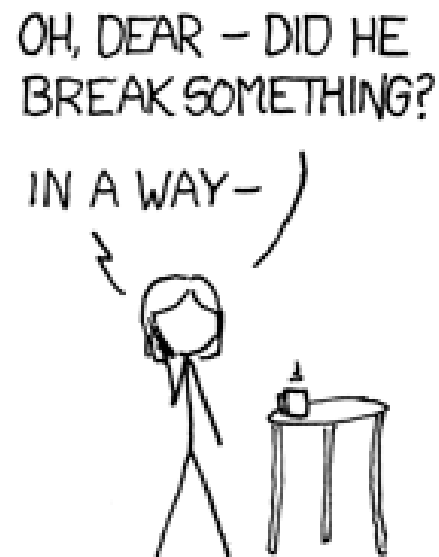
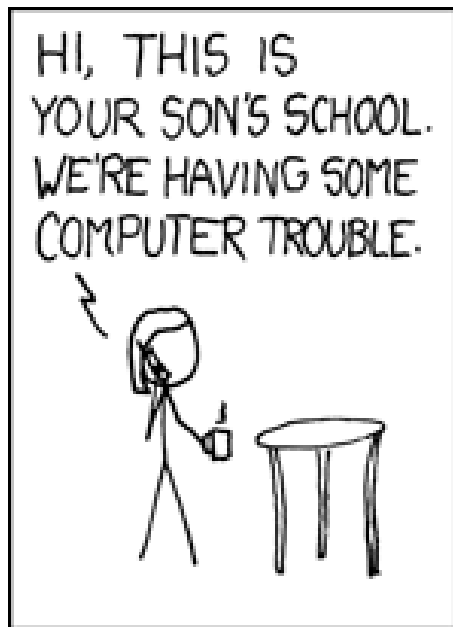
Database concerns

- Validate everything that ever goes in your database
- Also be sure to validate user data when doing select queries
- Your database library should provide a function to escape user data
- If you are using an ORM, this should be done automatically (but check!)

SQL Injection Example

- Consider the query constructed as
“SELECT * from users WHERE username='%s' AND password='%s'” % (username, password)
- We pass username=“user'--” and password=“”
- The query turns into “SELECT * from users WHERE username='user'--' AND password=“”
- -- is a comment character, so the password check is ignored

You can do much nastier things...



XSRF – Cross-site request forgery

- Unauthorized requests to another website from a different site
- Makes use of a users login at the other site without permission
- Can (almost) be protected by checking the request comes from the proper site
- For real protection, force the request to contain a secret key generated from your site

XSS – Cross-site scripting

- Malicious user injecting JavaScript into another site's page
- Could be used to force unwanted changes to application behaviour
- Can also be used to steal user cookies
- To protect against this, make sure any data inserted from the page is escaped to remove any HTML tags

XSS – Cross-site scripting