Computer Security
(or: The Worst Internet Security Blunders, and What You Can Learn From Them)

David Wagner
i206, April 12, 2012
MAIL FROM: <daw@cs.berkeley.edu>
RCPT TO: <hearst@ischool.berkeley.edu>
DATA
Subject: A good one

Hey, Marti, have you heard the one about why all good computer programmers like Shakespeare?
Because 2B OR NOT 2B = FF. Ha ha ha!

-- David

Demo
Important Ethics Note

• We will be discussing attacks in this class.

  This is not an invitation to undertake these attacks on your own.

• Attacking systems without the consent of all affected parties is unethical, contrary to UCB policy, and a possible violation of state and federal law. *Don’t do it!*
Discussion

• What are the practical consequences of this vulnerability?

• What was the blunder?

• What lessons can we learn?
Web Security

• Next, let’s look at security on the web.
How the Web Works

GET /index.html HTTP/1.0

<HTML><HEAD>...</HEAD><BODY><P>Welcome! ...
How the Web Works

GET /addcomment?msg=Hi%20mom! HTTP/1.0

<HTML> ...
Demo
Firesheep
Firesheep
Firesheep
Secure Connection Failed

An error occurred during a connection to linode.mrgall.com.

SSL received a record that exceeded the maximum permissible length.

(Error code: ssl_error_rx_record_too_long)

- The page you are trying to view cannot be shown because the authenticity of the received data could not be verified.
- Please contact the website owners to inform them of this problem. Alternatively, use the command found in the help menu to report this broken site.

Try Again
What’s the solution?
More demo
How the Web Works: Databases

GET /userpage?user=dave

<html>...I love prog...</html>

SELECT body, time
FROM squigs
WHERE username='dave'

I love programming|2010-...
Debugging is hard, though|...
def get_squigs(user):
    conn = ...
    s = "SELECT body,time FROM squigs
          WHERE username='%s'" % user
    return conn.execute(s).fetchall()
Logging in

GET /do_login?user=dave&pass=1234

SELECT password
FROM accounts
WHERE username='dave'

Database

1234

<HTML>...welcome....</HTML>

Browser
Posting a new squig

GET /do_squig?...squig=hi

<HTML>...

INSERT INTO squigs
VALUES ('dave', 'hi', ...)
What the web server code looks like

def post_squig(user, squig):
    conn = ...
    s = "INSERT INTO squigs VALUES ('%s', '%s', ...)
    %(user, squig)
    conn.execute(s)
<table>
<thead>
<tr>
<th>Squig</th>
<th>SQL</th>
</tr>
</thead>
<tbody>
<tr>
<td>hi</td>
<td>VALUES ('dave', 'hi', ...)</td>
</tr>
<tr>
<td>I’m good</td>
<td>VALUES ('dave', 'I'm good', ...)</td>
</tr>
</tbody>
</table>
Hacking payphones (1960’s)
Examples

<table>
<thead>
<tr>
<th>Squig</th>
<th>SQL</th>
</tr>
</thead>
<tbody>
<tr>
<td>hi</td>
<td>... VALUES ( 'dave', 'hi', ...)</td>
</tr>
<tr>
<td>I’m good</td>
<td>... VALUES ( 'dave', 'I'm good', ...)</td>
</tr>
<tr>
<td>I’</td>
<td></td>
</tr>
</tbody>
</table>

I’ || (SELECT...) || ’m

... VALUES ( 'dave', 'I' || (SELECT...) || 'm', ...)
Demo
SQL Injection Hack Infects 1 Million Web Pages

SANS warns of uptick in 'Lilupophilupop' attack, but Cisco said total number of infected Web pages likely lower.

By Kelly Jackson Higgins, Dark Reading
January 05, 2012 01:55 PM

Another SQL injection campaign is literally going viral, with some 1 million URLs possibly infected.

The SANS Internet Storm Center over the weekend counted some 1,070,000 URLs injected with the so-called lilupophilupop.com malware. That's up from 80 pages it had found in early December, according to SANS ISC handler Mark Hofman.
Hi, this is your son's school. We're having some computer trouble.

Oh, dear - did he break something?

In a way -

Did you really name your son Robert'); DROP TABLE Students;-- ?

Oh, yes. Little Bobby Tables, we call him.

Well, we've lost this year's student records. I hope you're happy.

And I hope you've learned to sanitize your database inputs.
Discussion

• What are the practical consequences of this vulnerability?

• What was the blunder?

• What lessons can we learn?
Solution

• Use library functions that are designed for security
• Avoid mixing untrusted data with trusted control stuff
Top Internet Security Blunders

• Unencrypted email => spam, phishing
• Unencrypted web => man-in-the-middle attacks
• Mixing data and control => Vulnerable web sites
Some defenses

• Don’t trust input from untrusted sources
• Use input validation: Check that inputs have the expected form
  – Check against a whitelist of known-good characters, not a blacklist of known-bad stuff
• Authenticate users
• Ensure all access attempts are checked to see whether they are authorized
• Encrypt data sent over the network
To learn more...

• Check out CS 161 (Computer Security)
• http://security.stackexchange.com/
Bonus Slides
Lessons

• Need mutual authentication (both parties authenticate each other)
ACCIDENT ON MOTORWAY
CAUTION! ZOMBIES! AHEAD!!!

RUN FOR COLD CLIMATES

THE END IS NEAR! !!!!!!!!
RAPTORS AHEAD CAUTION
TRAPPED IN SIGN FACTORY
SEND HELP!
Lessons

• Don’t rely upon security through obscurity