Secure Web Apps Team Meeting

Attendees:  Listed at end of document

Meeting Date:  February 9, 2006

Meeting Topic:

Handouts:  OWASP #4, 5, 6, and 7

A4 Cross Site Scripting Flaws

- Definition of Local Authentication vs. Server Authentication:
- Discussion of Roberto’s Forced Browsing examples
- Any type of variable that comes from user or comes from a place where you don’t control needs to be validated.
- State more strongly: If you are supplying user supplied input; Should be displayed by a function that either escapes or converts input into appropriate HTML. (Any variable output – function somewhere that regular expression into escaping HTML);
- Some libraries Ockerbloom is going to get examples of what this function is called in different apps (CGI for example).

Bring XSS example using Paros to test.

A5 Buffer Overflows
Paragraph “For your custom app code”: If you are writing your own C or C++ code or those languages that does their own memory management.

A6 Injection Flaws (Shell Commands as well as SQL)

MySQL – harder – only processes one statement at a time when you pass it a query. Oracle back end – string can contain some malicious code that could be damaging.

PhP New library that interfaces
Multiple commands in one query.

Separate Shell Injection vs. SQL Injection --- Make into two separate sections.

In Perl, when you open a file – would be the place to look for a shell injection. Ockerbloom to add wording/examples here.

6.3 Examples:
• DELETE FROM CRITICALTABLE WHERE USER='$VAR'
  BADGUY' OR "="
This would delete the information in the critical table.

• Coldfusion: CFQueryParam Tag. Need more detail – ask Chris (Law) – puts a "
in the query statement and looks to the CFQueryParam

• PERL Example: DELETE FROM CRITICALTABLE WHERE USER='?'
  BADGUY' OR "="

Avoid single quotes or escape variables in syntax; use bind variables

“Validate all information” --- be more specific and add more text
Use pattern matching to verify user input is an expected value.

If not what expected, throw out and Error messages should be generic. Link to next
section “Error Handling”. Turn off/control debug messages.

Database level: Write procedures to do certain functions; Use stored procedures to limit
giving access to db files; Limit access to web account that is accessing the database.
Limit to READ only where possible – at user level as well as database level.

Reuse previously tested code wherever possible.

**A7 Error Handling**

For example, if a hacker enters an invalid command, the web server sends an error message back to the
end-user. This message should be generic, but often times presents excessive information such as “User
Name Correct, Password Incorrect.” That could help the attacker focus their illicit activities on the
password cracking activities.

Highlight paragraph on “Fail Open Security Check” --- assume no access until proven
otherwise.

Return one error message to the user and another error message to the server. Don’t give
detailed error message to the user and Log the detailed error message.

Can give user diagnostic information but not programmer diagnostic/debug information.
Give Data Validation Errors.
Especially ID/PW errors – don’t describe the password complexity; ID ok but PW invalid, etc.

Intrusion Detection paragraph:
Review logs;
Look for anomalies;
If you are running sensitive data: Log access to system and review error logs.
## Secure Web Application Coding Team Members

<table>
<thead>
<tr>
<th>Name</th>
<th>Email</th>
<th>Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raymond Bokenkamp</td>
<td><a href="mailto:rpbokenk@mail.med.upenn.edu">rpbokenk@mail.med.upenn.edu</a></td>
<td>215-746-0103</td>
</tr>
<tr>
<td>Chris Blickley</td>
<td><a href="mailto:blickley@hr.upenn.edu">blickley@hr.upenn.edu</a></td>
<td>215-898-6162</td>
</tr>
<tr>
<td>Doug Brunk</td>
<td><a href="mailto:bruk@mail.med.upenn.edu">bruk@mail.med.upenn.edu</a></td>
<td>215-573-0253</td>
</tr>
<tr>
<td>√ Eric Chen</td>
<td><a href="mailto:emchen@isc.upenn.edu">emchen@isc.upenn.edu</a></td>
<td>215-573-7550</td>
</tr>
<tr>
<td>Jim Choate</td>
<td><a href="mailto:choate@isc.upenn.edu">choate@isc.upenn.edu</a></td>
<td>215-898-4709</td>
</tr>
<tr>
<td>Chris Filippone</td>
<td><a href="mailto:cff@dental.upenn.edu">cff@dental.upenn.edu</a></td>
<td>215-898-8957</td>
</tr>
<tr>
<td>√ Christopher Herdt</td>
<td><a href="mailto:cherdt@law.upenn.edu">cherdt@law.upenn.edu</a></td>
<td>215-898-9140</td>
</tr>
<tr>
<td>√ Dan Hill</td>
<td><a href="mailto:dwhill@pobox.upenn.edu">dwhill@pobox.upenn.edu</a></td>
<td>215-662-6332</td>
</tr>
<tr>
<td>Bil Kasenchar</td>
<td><a href="mailto:bmk@pobox.upenn.edu">bmk@pobox.upenn.edu</a></td>
<td>215-898-0001</td>
</tr>
<tr>
<td>√ Ian Kelley</td>
<td><a href="mailto:ikelley@pobox.upenn.edu">ikelley@pobox.upenn.edu</a></td>
<td>215-573-5243</td>
</tr>
<tr>
<td>√ Susan Kennedy</td>
<td><a href="mailto:susanken@pobox.upenn.edu">susanken@pobox.upenn.edu</a></td>
<td>215-573-4495</td>
</tr>
<tr>
<td>√ Stephen Kratowicz</td>
<td><a href="mailto:kratowis@uphs.upenn.edu">kratowis@uphs.upenn.edu</a></td>
<td>215-349-8441</td>
</tr>
<tr>
<td>√ John Lupton</td>
<td><a href="mailto:lupton@isc.upenn.edu">lupton@isc.upenn.edu</a></td>
<td>215-573-3811</td>
</tr>
<tr>
<td>√ Roberto Mansfield</td>
<td><a href="mailto:robertom@sas.upenn.edu">robertom@sas.upenn.edu</a></td>
<td>215-573-4712</td>
</tr>
<tr>
<td>√ John Ockerbloom</td>
<td><a href="mailto:ockerblo@pobox.upenn.edu">ockerblo@pobox.upenn.edu</a></td>
<td>215-573-0758</td>
</tr>
<tr>
<td>√ Dominic Pasqualino</td>
<td><a href="mailto:dominicp@pobox.upenn.edu">dominicp@pobox.upenn.edu</a></td>
<td>215-898-1933</td>
</tr>
<tr>
<td>Steven Rosato</td>
<td><a href="mailto:stevenjr@pobox.upenn.edu">stevenjr@pobox.upenn.edu</a></td>
<td>215-898-4294</td>
</tr>
<tr>
<td>Terrence Ryan</td>
<td><a href="mailto:tpryan@wharton.upenn.edu">tpryan@wharton.upenn.edu</a></td>
<td>215-898-6705</td>
</tr>
<tr>
<td>Matt Snyder</td>
<td><a href="mailto:mwsnyder@law.upenn.edu">mwsnyder@law.upenn.edu</a></td>
<td>215-573-5498</td>
</tr>
</tbody>
</table>
✓ indicates in attendance on 1/26/2006