

CSE331: Introduction to Networks and Security

Lecture 36

Fall 2006

Announcements

- Project 4 is due today at midnight.
- Homework 3 is today (now).
- Final Exam:
 - Friday Dec. 15th 12:00--2:00
 - CHEM B13



Plan for today

- Briefly: Two more MAC policies
- Briefly: Web security
 - Cross-site scripting (XSS)
- Course wrap up/overview
- Course evaluations

Two Other MAC Policies

- "Chinese Wall" policy: [Brewer & Nash '89]
 - Object labels are classified into "conflict classes"
 - If subject accesses one object with label L1 in a conflict class, all access to objects labeled with other labels in the conflict class are denied.
 - Policy changes dynamically
- "Separation of Duties":
 - Division of responsibilities among subjects
 - Example: Bank auditor cannot issue checks.

Web Security

- What security concerns are there on the web?
 - Links can lie
 - may not take you where you think they do (phishing)
 - Cookies
 - can reveal private information, questions of their security
 - Spyware/Malware
 - mobile code / trojan horses / "bot nets"
 - Eavesdropping / keylogger
 - Embedded code / scripts / flash / ActiveX / ... executable content
 - Profile stealing
 - Trusting remotes sites with your confidential information
 - Spam
 - today, spam costs about \$0.0001 / e-mail to send

Scripts & Mobile Code

- Client side: embedded in HTML sent to the client
 - Java Applets, JavaScript, ActiveX, Flash
- Server Side: receive & process arguments from forms filled in by client
 - CGI "Common Gateway Interface"
 - Allows server to call code written in any language, commonly C or Perl
 - Code typically stored in /cgi-bin directory
 - PHP "PHP Hypertext Preprocessor"
 - Embed dynamically generated content into HTML pages

Example PHP

```
<html>
  <head>
    <title>PHP Test</title>
  </head>
  <body>
    <?php echo '<p>Hello World</p>';
    ?>
  </body>
</html>
```

Cross Site Scripting (XSS)

- Consider the following scenario:
 - You click on a URL:
<http://www.cis.upenn.edu/~stevez/foo.html>
 - What happens? Server responds:

Not Found

The requested URL /~stevez/foo.html was not found on this server.

Apache/1.3.33 Server at www.cis.upenn.edu Port 80

- What's the problem?

XSS continued

- Suppose that the malicious URL contained HTML tags for an embedded script:

[http://www.cis.upenn.edu/~stevez/<script>alert\('hello'\)</script>](http://www.cis.upenn.edu/~stevez/<script>alert('hello')</script>)

- If the server generates the error page naively, it might accidentally include the script in the page displayed to the client!
 - (Fortunately, CETS here at Penn gets this right...)

XSS

- These techniques can be used to steal cookies, redirect users to bogus web pages, grab data entered by user.
- Other tricks:
 - Attackers can encode malicious part of the URL to make it harder to detect (e.g. use Unicode)
 - Not all attacks need the "<" and ">" symbols
- What can be done?
 - Validate URLs at the server side
 - Rewrite "problematic" inputs to HTML entity codes:
 - < becomes <
 - > becomes >
 - ...



Main Take-away Ideas (1)

- Security is about Tradeoffs
 - Balance risk vs. expense
- *Principles of Secure System Design:*
- Security is a process
- Least privileges
- Complete Mediation
- System Design
 - Economy of mechanism
 - Open standards
 - Failsafe Defaults

Main Take-away Ideas (2)

- Cryptography is important...
 - Can be used for more than just hiding information
 - Authentication and integrity
- ... but not the only facet of security
 - Buggy software is often more relevant
 - Social engineering is effective
 - Cryptography applied inappropriately is useless
- So: use it where necessary, and use it correctly
 - See Schneier's book *Applied Cryptography*



Main Take-away Ideas (3)

- Concepts of security:
 - Confidentiality
 - Integrity
 - Availability
- General Mechanisms
 - Authentication
 - Challenge / Response
 - Authorization
 - Reference monitors
 - Access control matrices
 - Audit
 - Logs



Main Take-away Ideas (4)

- Cryptography & Protocol Design
 - Shared vs. Public key cryptography
- Cryptographic protocols can be used for:
 - Authentication, privacy, confidentiality
- Challenge—Response is the fundamental method of authentication
- Nonces, Time stamps, Sequence numbers prevent replay attacks

Main Take-away Ideas (5)

- Malicious Code
 - Viruses & Worms
 - Defense in depth: patching, firewalls, proper configuration, auditing
- Buffer overflows are the #1 vulnerability
 - Choose safe languages:
 - Java, C#, Scheme, ML
 - Be aware of format string and input errors, take care when writing programs and scripts.
 - Software audit and design is important.
 - If you must use C or C++, use StackGuard, ProPolice, or another buffer-overflow preventative measure.

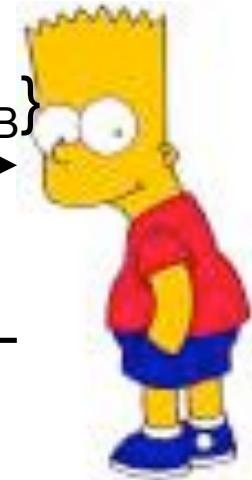
Further study

- Advanced cryptography & cryptographic protocols
 - Elliptic curves
 - Protocol analysis - logic and model checkers
 - Secret sharing, voting
- Systems security
 - Fault tolerance: replication, consensus algorithms
 - "Analysis of an Electronic Voting System" by Kohno et al.
- Additional sources of information
 - IEEE Symposium on Security & Privacy ("Oakland conference")
 - ACM Conference on Computer and Communications Security
 - Computer Security Foundations Workshop
 - CRYPTO, EUROCRYPT

Thanks!



$K_{AB}\{\text{"Let's close this session, Bart"}, n_A, n_B\}$



$K_{AB}\{\text{"Bye, Alice"}, n_A, n_B\}$

