

# CERIAS

the center for education and research in information assurance and security

## Integrating the Common Weakness Enumeration into a Secure Programming Course

Pascal Meunier, M.Sc., Ph.D., CISSP

### CS390S: Secure Programming

- Since Fall 2002
- Review common mistakes
  - Buffer Overflows
  - Format String Vulnerabilities
  - etc...
- Concepts
  - Trust Management
  - Input Validation
  - Meta-Characters and escapes
  - Character encodings
  - etc...
- **Is the coverage representative?**
- **Is the coverage done correctly?**
- **How do employers know what students learned in the class?**

### New Slides Based on View

PURDUE CERIAS Crossing Boundaries	PURDUE CERIAS Web Parameter Tampering, ID 472
<ul style="list-style-type: none"><li>• Moving low-trust data across a boundary to a high-trust area, or for a high trust use, requires input validation<ul style="list-style-type: none"><li>- Type, range, format, validity</li><li>- If even allowed... (c.f. access control later)</li></ul></li><li>• Additional Issues:<ul style="list-style-type: none"><li>- Source authentication<ul style="list-style-type: none"><li>◆ Did this data really come from where it says it did, or from where I think it did?</li></ul></li><li>- Data Integrity<ul style="list-style-type: none"><li>◆ Has this data been tampered with?</li></ul></li><li>- Is the storage location trustworthy?</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Access Control Bypass Through User-Controlled SQL Primary Key, CWE ID 566</li><li>• Scenario<ul style="list-style-type: none"><li>- You list items on a web page (orders, invoices, etc...) with links to view more details about a specific item</li><li>- Bob is not supposed to be able to access someone else's items</li><li>- urls are of the form: list_specific_item?id=98</li><li>- Mallory can change the id and view someone else's items</li></ul></li><li>• It's not because you list only good values as choices that you won't get an incorrect one back...</li></ul>

### Conclusions

- Creating views is work intensive
- Views highlight missing CWE IDs
  - Feedback improves the CWE
  - Course declared CWE compatible
  - Course quality improved
  - More examples and cases
  - Strong linkage of concepts to a systematic empirical collection
- Views could help form the basis of new taxonomies or ontologies

### The Common Weakness Enumeration

- Compatibility declarations, Spring 2007
- Group CVE entries by similarity
- CWE ID given to each
- Entries are linked in a tree (parent/children)
  - **Tree is huge! (see on right)**
  - Organization is often more appropriate for code scanners than teaching
  - Example: No branch matching trust management concepts

### New CWE Views Needed For Teaching

Goal: Re-organize CWE entries in a tree that matches the concept to be taught

### Example: Trust

Legend: **Matched CWE IDs** **Missing CWE IDs**

- **Trust Boundary Problems**
  - **Inconsistent validation mechanisms**
    - Same source handled differently in different code locations
    - At different times
    - In different circumstances
    - From different sources
      - Authentication Bypass by Alternate Path/Channel, ID 288
      - Unprotected Alternate Channel, ID 420
  - **Ill-defined trust boundaries**
  - **Trust Boundary Violation, ID 501**
  - **Misplaced or Absent Trust Boundaries**
    - **Self-reported information**
      - Trusting self-reported IP address, ID 291
      - Self-reported & reverse DNS name, ID 292
      - Using referrer field for authentication, ID 293
    - **Trusting the client**
      - Client-Side Makes Server Security Decisions
      - Server trusting client-side-controlled data
        - "Trusting Cookie Information" is Use of Cookies, ID 565
        - Web Parameter Tampering, ID 472
          - Access Control Bypass Through User-Controlled SQL Primary Key, ID 566
    - **Trusting Events**
      - Trust of system event data, ID 360
      - Unprotected Windows Messaging Channel ('Shatter'), ID 422
    - **Trusting the integrity of shared data writable by others**
      - Misused Authentication: getlogin (not reentrant), ID 558
  - **Cryptographic Trust Assurance**
    - Certificate Issues, ID 295
      - Failure to follow chain of trust in certificate validation, ID 296
      - "Failure to validate host-specific certificate data" ID 297
      - No OpenSSL Certificate Check Performed before this Use, ID 599
      - Failure to validate certificate expiration, ID 298
      - Failure to check for certificate revocation, ID 299
      - Race condition in checking for certificate revocation, ID 370
    - **Use of Encrypted Cookies**
      - Counterexample: Plaintext Storage in Cookie, ID 315 (different perspective on 565, but essentially the same mistake)

