Log Your CRUD
Design Principles for Software Logging Mechanisms

Jason King
PhD Student

Laurie Williams
Adviser

---

Stolen Hopkins patient info used in $600K credit card fraud

Ex-UCLA Healthcare Employee Pleads Guilty to Four Counts of Illegally Peeking at Patient Records

RECORD PEEKING
HOSPITALS STRUGGLE TO CONTROL SNOOPY STAFFERS

Allina fires 32 for peeking at records of patients after high-profile overdose

NJ hospital suspends 27 over Clooney record snoop

Paging Dr Ross

Hospital staff sacked for breaches of data
2011 Veriphyr Survey of Patient Privacy Breaches

- Top 3 most commonly reported breaches
  - Snooping into medical records of employees
  - Snooping into medical records of friends/family
  - Loss or theft of physical records

- 52% of respondents: organization does not have adequate tools for monitoring inappropriate access to patient data

Logging Mechanisms

- Mitigate repudiation attacks
- Recreate traces of user activity after a security/privacy breach
- Identify unauthorized access of sensitive data
- Forensic analysis: who, what, when, where, how?
Previous Work

- "Modifying Without a Trace" [IHI’12]
  - General events
    - "view data"
    - "create data"
  - Specific events
    - "view demographics data"
    - "create immunization data"
- Evaluating logging of specific events gives a much better picture

- "Cataloging and Comparing..." [HealthTech’13]
  - Compiled catalog of data transactions, security events, and log entry content
    - 10 healthcare sources
    - 6 non-healthcare sources
  - Must consider 13 out of 16 to identify 100% of catalog
  - Should not rely on a single source document
Objective

• to observe the current state of logging mechanisms by performing an exploratory case study in which we systematically evaluate logging mechanisms by supplementing the expected results of existing functional black-box test cases to include log output

Research Questions

• RQ1: What observations can we make to understand why the four studied EHR logging mechanisms do not capture some specific user actions?

• RQ2: What observations can we make about the general security of the four studied EHR logging mechanisms?

• RQ3: What principles of logging mechanism design, implementation, and testing may be proposed based on observations of the four studied EHR systems?
Supplementing Existing Black-box Test Cases

• From NIST Approved 2014 Edition Test Procedures for EHR systems
  – Randomly select 10 test criteria
  – Extract 34 individual test cases from the 10 criteria

• Individual test cases
  – Identify actions taken by the tester in the system
  – Generate expected logging output
  – Toss 4 test cases that had no expected log output

• Example:
  “The Tester shall enter the provided demographic test data.”
Supplementing Existing Black-box Test Cases

• Expected log entry content:

  Based on ASTM International E2147-01 Standard Specification for Audit and Disclosure Logs for Use in Health Information Systems

  – Date and time
  – Patient Identification
  – User Identification
  – Type of action
  – Identification of the patient data accessed

Test Cases Summary

• 30 test cases

<table>
<thead>
<tr>
<th>Test Identification</th>
<th>Test Description</th>
<th>Expected Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTR176.314(a)(3) 1</td>
<td>TE176.314(a)(3) – 1.01: The tester shall select the test data provided in TE176.314(a)(3) – 1.02: Using the Vendor-Identified EHR function(s) and three test patients, the tester shall enter the provided demographic test data selected in TE176.314(a)(3) – 1.01</td>
<td>TE176.314(a)(3) – 1.03: Using the Inspection Test Guide, the Tester shall verify that the patient demographic data entered in TE176.314(a)(3) – 1.02 are entered correctly and without omission, and in conformance with the standards for race, ethnicity and preferred language. LOG: The tester shall verify that the act of entering demographic data is recorded by the logging mechanism.</td>
</tr>
</tbody>
</table>
Electronic Health Record Systems Studied

- **OpenEMR v4.1.2**
  - Used by an estimated 15,000 physicians
  - “Certified EHR” in the USA
- **OSCAR v12.1**
  - Used by an estimated 2,000 clinical providers in Canada
- **Tolven eCHR v2.1.3**
  - Used internationally
  - “Certified EHR” in USA
- **WorldVistA v2**
  - Version of VistA, developed by US Department of Veterans Affairs
  - “Certified EHR” in USA

Evaluation Methodology

- For a default installation of each EHR system
  - Perform each of 30 test cases
    - Use logging interface used to achieve “Certified” status
    - FAIL if expected log output is incorrect, not logged, or missing a required field
    - NA if functionality cannot be located
    - PASS if expected log output is correct with all required data fields
Results

<table>
<thead>
<tr>
<th>EHR System</th>
<th>Pass</th>
<th>Fail</th>
<th>NA</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenEMR</td>
<td>17 (62.69%)</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>OSCAR</td>
<td>8 (38.1%)</td>
<td>13</td>
<td>9</td>
</tr>
<tr>
<td>Tolven eCHR</td>
<td>4 (21.1%)</td>
<td>15</td>
<td>11</td>
</tr>
<tr>
<td>WorldVistA</td>
<td>0 (0.00%)</td>
<td>23</td>
<td>7</td>
</tr>
</tbody>
</table>

Research Question 1

- What observations can we make to understand why the four studied EHR logging mechanisms do not capture some specific user actions?
RQ1: OpenEMR

- Logs SQL queries
- Does not log SELECT queries by default
  - 7 “view” test cases fail
### RQ1: Tolven eCHR

- "updates" are recorded as "additions"
  - 4 test cases fail
- Stores additional query parameters in database
  - Must have authorized access to database
  - Not viewable in a black-box evaluation

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>User</th>
<th>Patient</th>
<th>URL</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
<tr>
<td>04/07/2014</td>
<td>22:06:11</td>
<td>admin</td>
<td>Walton, John Jackie</td>
<td>user/admin/account/113005/list/echr/patient:280001-280005/m...-more</td>
<td>POST (add)</td>
</tr>
</tbody>
</table>

---

**Science of Security Lablet**

**Security Metrics-Driven Evaluation, Design, Development, & Deployment**

**Computer Science**

**NC State University**
RQ1: WorldVistA

- Only data accesses seem to generate log entries
  - No entries indicated “create” “modify” or “delete”
- No human-readable, clear descriptions of event that happened
- 0 passing test cases
Research Question 2

- What observations can we make about the general security of the four studied EHR logging mechanisms?

Security of Logging Mechanisms

- Administrative users may simultaneously be physicians or other users
  - Saltzer & Schroeder’s *separation of privilege* and *least privilege*
  - OpenEMR administrative users have direct read/write access to log entry database table
Security of Logging Mechanisms

• CWE532: Information Exposure through Log Files
  – OpenEMR’s logged SQL queries can reveal protected information
  – Sensitivity of log content should be considered when granting/revoking access to log entries

  INSERT INTO lists (date, pid, type, title)
  VALUES (NOW(), '1', 'allergy', 'penicillin')

Security of Logging Mechanisms

• CWE778: Insufficient Logging
  – All 4 EHR systems do not adequately log critical events, such as viewing, by default
Security of Logging Mechanisms

• CWE779: Logging Excessive Data
  – Enabling SELECT logging generates MANY entries
    • View a patient summary record generates 80 entries within 2 seconds

• Tolven eCHR and WordVistA
  – Do not log authentication attempts
• All 4 EHR systems
  – Do not log when logs are accessed
  – Security events seem overlooked
Principles

• Log by Default
• Specify Logging Requirements
• Capture Adequate Context
• Support Human-readable Reporting
• Succinctly Represent User Behavior
• Enforce Immutability
• Perform Systematic Black-box Testing

Future Work

• Extract specific events to log from requirements specifications
• Automate black-box testing of logging mechanisms
• RQ: What criteria should be considered when constructing an evaluation framework for evaluating the ability of logging mechanisms to hold users accountable and promote meaningful forensic analysis?
• RQ: What metrics can be used to represent the degree to which logging mechanisms promote user accountability?
Summary

• 61 out of 90 (67.8%) of applicable test cases fail
• 6 tests fail in all four EHR systems
  – 4 tests related to viewing protected information
• Our design principles can help guide software engineers when developing logging mechanisms for user accountability, but they are not an exhaustive list of *everything* that should be considered