Enabling Forensics by Proposing Heuristics to Identify Mandatory Log Events

Jason King
Rahul Pandita
Laurie Williams
Motivation

Deputy’s snooping into driving records brings lawsuit

Unauthorised staff snoop into Ryder's medical records

MN Health Clinic Fires 32 for HIPAA Violations

RECORD PEEKING
HOSPITALS STRUGGLE TO CONTROL SNOOPY STAFFERS

NJ hospital suspends 27 over Clooney record snoop

Paging Dr Ross

Snooping by Metro Transit employee leads agency to restrict access to records

UK civil servants routinely snoop on citizens' private financial and health information

Former Financial Institution Employee Sentenced For Unauthorized Computer Access

Six people fired from Cedars-Sinai over patient privacy breaches

Cedars-Sinai officials say that 14 patient medical records were 'inappropriately accessed' between June 18 and June 24.

Alleged privacy violations lead to 16 firings

Hospital staff sacked for breaches of data

Snooping staff still a big concern

Snoop Through Records, Go Directly To Jail
Motivation

• Repudiation Threats
• Who, What, When, Where, Why, & How
• 2011 Veriphyr Survey of Patient Privacy Breaches
  – 52%: inadequate tools for monitoring PHI access
• No single resource for log specifications
• Requirements specifications describe user activity
Objective

...help software engineers strengthen forensic-ability and user accountability by
1) systematically identifying mandatory log events through processing of unconstrained natural language software artifacts; and
2) proposing empirically-derived heuristics to help determine whether an event must be logged
Research Questions

• **RQ1**: How *often* do descriptions of mandatory log events appear in natural language software artifacts?

• **RQ2**: What similarities and differences exist in the *grammar* and *vocabulary* used in different software artifacts?

• **RQ3**: What factors help *decide* whether an action must be logged?
Software & Artifacts

• **iHRIS: Open Source Human Resources Information Solutions v4.2.** more than 675,000 health worker records currently supported
  – Artifact: traditional software requirements specification for Page Builder module

• **iTrust: Open Source Electronic Health Record System v18.**
  educational, test-bed healthcare software developed & maintained at NCSU
  – Artifact: use-case based requirements specification

• **OCS (Open Conference Systems): Open Source Scholarly Conference Management System v2.3.6.** from Public Knowledge Project (PKP)
  – Artifact: “OCS in an Hour” User Manual

Identifying Verb-Object Pairs

• For each statement in the software artifact
• Identify each event in terms of the action (verb) and resource (object) acted upon
  – <verb, object>
• Examples:
Classification

• Individually classify verb-object pairs as *mandatory log event* or *not a mandatory log event*
  
  – A *mandatory log event (MLE)* is an action that *must be logged in order to hold the software user accountable for performing the action*.

• Reconcile differences, document discussion

• Third individual serves as “tiebreaker”
Inter-rater Agreement

Table 2: Confusion matrix for iTrust classifications

<table>
<thead>
<tr>
<th></th>
<th>Author 1</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Log</td>
<td>Not Log</td>
<td>Total</td>
</tr>
<tr>
<td>Author 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log</td>
<td>788</td>
<td>148</td>
<td>936</td>
</tr>
<tr>
<td>Not Log</td>
<td>615</td>
<td>377</td>
<td>992</td>
</tr>
<tr>
<td>Total</td>
<td>1403</td>
<td>525</td>
<td>1928</td>
</tr>
</tbody>
</table>

Cohen’s Kappa $\kappa = 0.22$
Results: Verb-Object Pairs

<table>
<thead>
<tr>
<th>Software</th>
<th>Total Statements</th>
<th>Total Verb-Object Pairs</th>
<th>Total MLE Verb-Object Pairs</th>
<th>Statements with at least One MLE Verb-Object Pair</th>
</tr>
</thead>
<tbody>
<tr>
<td>iHRIS</td>
<td>36</td>
<td>106</td>
<td>96 (91%)</td>
<td>27 (75%)</td>
</tr>
<tr>
<td>iTrust</td>
<td>1301</td>
<td>1928</td>
<td>1217 (63%)</td>
<td>802 (62%)</td>
</tr>
<tr>
<td>OCS</td>
<td>791</td>
<td>1479</td>
<td>747 (51%)</td>
<td>434 (55%)</td>
</tr>
<tr>
<td>TOTAL</td>
<td>2128</td>
<td>3513</td>
<td>2060 (59%)</td>
<td>1263 (59%)</td>
</tr>
</tbody>
</table>

• **RQ1**: How often do descriptions of mandatory log events appear in natural language software artifacts?
  • Over half (59%) of the individual statements studied contain descriptions of MLEs
Results: Differences between Artifacts

- iHRIS: traditional requirements specification
  - “The system shall...”
- iTrust: use-case based requirements specification
  - “A patient chooses to...”
- OCS: user manual
  - “You can also elect to require authors to agree to...”
# Results: Differences between Artifacts

## Table 1: Summary of top 5 verbs (all verbs vs. mandatory log event verbs)

<table>
<thead>
<tr>
<th>Software Artifact</th>
<th>All Verbs</th>
<th>Mandatory Log Event Verbs Only</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Verb</td>
<td>Frequency</td>
</tr>
<tr>
<td>iHRIS traditional requirements</td>
<td>allow</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>edit</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>save</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>display</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>add</td>
<td>5</td>
</tr>
<tr>
<td>iTlock use-case based requirements</td>
<td>is</td>
<td>217</td>
</tr>
<tr>
<td></td>
<td>view</td>
<td>120</td>
</tr>
<tr>
<td></td>
<td>choose</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>select</td>
<td>81</td>
</tr>
<tr>
<td></td>
<td>enter</td>
<td>71</td>
</tr>
<tr>
<td>OCS user guide</td>
<td>is</td>
<td>137</td>
</tr>
<tr>
<td></td>
<td>use</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td>add</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td>select</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>allow</td>
<td>52</td>
</tr>
</tbody>
</table>
Results: Differences between Artifacts

• **RQ2**: What similarities and differences exist in the **grammar** and **vocabulary** used in different software artifacts?
  
  – iHRIS traditional requirements specification has more constrained & consistent grammar/vocabulary
  
  – *States* of a system are commonly described in iTrust and OCS
    
    • “A patient is a registered user”
    
    • “OCS is designed to be a multilingual system”
Results: Heuristics

• **RQ3**: What factors help decide whether an action is a mandatory log event?
Heuristics

- **Heuristic H1**: If the verb involves creating, reading, updating, or deleting resource data in the software system, then the event must be logged.
  - 134 <verb, object> pairs explicitly used create, read, update, delete terminology

- **Heuristic H2**: If the verb can be accurately rephrased in terms of creating, reading, updating, or deleting resource data in the software system, then the event must be logged.
  - Example: “A patient designates a representative”
    “A patient creates a representative for…”
  - 1,243 <verb,object> pairs can be rephrased as CRUD actions
Heuristics

- **Heuristic H3**: If the verb implies the system displaying or printing resource data that is capable of being viewed in the user interface or on paper, then the event must be logged.
  - Examples: view, see, print, display, appear
  - 397 <verb, object> pairs relate to viewing data

- **Heuristic H4**: If the verb expresses the intent to perform an action, such as “choose to”, “select to”, “plan to”, or “wish to”, then the intent event is not a mandatory log event.
  - 351 <verb, object> pairs express intent
Heuristics

• **Heuristic H5**: If the verb expresses the granting or revocation of access privileges in the software system, then the event must be logged.
  – Example: “Doctors are allowed to...”
  – 126 <verb, object> pairs relate to access privileges

• **Heuristic H6**: If the verb is ambiguous, such as “provide” or “order”, context must be considered when determining if the event must be logged.
  – Example: “A doctor *orders* lab procedures...”
  “Lab procedures are *ordered* alphabetically in a list...”
  – 158 <verb, object> pairs contain ambiguous verbs
Heuristics

• **Heuristic H7**: If the verb describes an action that takes place outside the scope of the functionality of the software, then the event is not a mandatory log event.
  – Example: “To enable electronic payment, register for a PayPal business account”
  – 314 <verb, object> pairs were outside the scope

• **Heuristic H8**: If the verb involves the creation or termination of a user session, then the event must be logged.
  – Example: “A patient authenticates...”
  – 100 <verb, object> pairs relate to user sessions
Heuristics

- **Heuristic H9**: If the verb describes a **state or quality** within the system, then the event is not a mandatory log event.
  - Example: “A patient is a registered user”
    “A list of pages is available”
  - 253 <verb, object> pairs describe system states or qualities

- **Heuristic H10**: If the verb describes **possession or composition** of a resource or quality, then the event is not a mandatory log event.
  - Example: “The patient has a known interaction”
    “The row contains the doctor’s comments”
  - 57 <verb, object> pairs describe possession or composition
Heuristics

- **Heuristic H11**: If the verb describes navigation or mechanical interaction with the software interface, then the event is not a mandatory log event.
  - Example: “The doctor remains on the allergy page”
  “The author needs to click on ‘Active Submissions’”
  - 226 <verb, object> pairs relate to navigation or interface actions

- **Heuristic H12**: If the verb describes initialization of the software or configuration of the software, then the event must be logged.
  - Example: “The administrator can install additional locales”
  - 8 <verb, object> pairs relate to initialization/configuration
Heuristics: Summary

- Our 12 heuristics were helpful for ~96% of verb-object pairs
Lessons Learned

• Use Consistent Terminology
  – View, display, present, show, see, provide, read...

• Use Consistent Perspective
  – The user reads, views...
  – The system shows, displays, presents...
  – The data appears...

• Use CRUD Terminology
Future Work

• Include additional types of domains, artifacts
• Tool-assisted approach using the heuristics
• User study:
  – Controlled experiment
  – Relevance, quantity, efficiency of identifying MLEs using heuristics
• Metric for forensic-ability
Summary

• A set of **empirically-derived heuristics** to assist software engineers in determining whether a given user action described in a software artifact must be logged.

• A **set of considerations for requirements engineers** to help clearly and unambiguously document mandatory log events in software artifacts.

• An **oracle of mandatory log event classifications** for three open-source software systems. The oracle is publically available on the project website.
  – http://go.ncsu.edu/NLPLogging