AUTOMATED REVERSE ENGINEERING OF UML SEQUENCE DIAGRAMS FOR DYNAMIC WEB APPLICATIONS

Manar Alalfi, James Cordy, Thomas Dean
School of Computing, Queen’s University
Kingston, Canada
Part of a Framework

- Role-based access control security testing using model checking [C3S2E09]
**CONTEXT**

- **PHP2XMI**
  - Used to recover role permissions at the level of page access (this work)
WEB APPLICATION CHALLENGES

- Identification of interaction elements
  - Web applications often not built based on OO concepts

- Identification of control structures (loops and conditions)
  - Multiple entry points
  - Implicit control flow that is difficult to detect until run time

- Recognition of similar execution trace patterns
  - Using both static and run time information

- Representation of the complete set of behavioral changes from state to state
  - Several components may be affected
  - Databases, sessions, cookies

- Multilingual documents
MOTIVATION

- PHP2XMI is a key part of our role-based access control (RBAC) security analysis framework
  - Used to recover role permission behavior at the page access level

- Why UML Sequence Diagrams?
  - Standard for behavior representation
  - Facilitates the process of understanding system interaction behavior using standard UML tools
  - Helps in inferring user role permissions

- Why UML 2.0 in XMI 2.0 format?
  - International standard for UML model representation and interchange
  - Easy interchange with UML-based modeling tools such as Rational Software Architect
  - Supports model integration and model to model transformation
GOALS & CONTRIBUTIONS

- A new reverse engineering tool for dynamic web applications
  - PHP-based dynamic web applications
    - UML 2.1 sequence diagrams
- Adaptable behavioral model recovery technique
  - Extendable to other scripting languages in plug-and-play fashion
  - Handles mixed-language documents
  - Facilitates testing for web application security vulnerabilities
- Dynamic trace filtration
  - Reduces redundant information that can complicate program understanding
- Supports Model-Driven-Engineering
  - Result can be directly imported and visualized using standard UML toolsets such as Rational Software Architect
Three stages
- Instrument, exercise, extract model
PHP2XMI - APPROACH

- Three stages
  - Instrument, exercise, extract model
Three stages
- Instrument, exercise, extract model
PHP2XMI - APPROACH

- Three stages
  - Instrument, exercise, extract model
PHP2XMI - APPROACH

- Three stages
  - Instrument, exercise, extract model
  - Source transformation based
STAGE 1 - INSTRUMENT

- Parsing and insertion of dynamic instrumentation code
  - Source transformation using TXL
  - Adaptable to any scripting language, mixed languages
**Stage 1 - Instrument**

- Parsing and insertion of dynamic instrumentation code

```plaintext
rule instrumentHttpVar
    replace  [Expr]
        E    [Expr]
    construct NewE  [Expr]
        E    [Conv_func_GET]
            [Conv_func_POST]
            [Conv_func_COOKIE]
            [Conv_func_SESSION]
    where      not
        NewE  [= E]
    by
        NewE
end rule

rule Conv_func_GET
    replace  [Expr]
        E    [ReferenceVariable]
    deconstruct E
        '$HTTP_GET_VARS
            '[ Param  [Expr] ']
    by
        HttpVar_track(Param,
            '$HTTP_GET_VARS'[Param], 'GET')
end rule
```
Stage 1 - Instrument

- Parsing and insertion of dynamic instrumentation code

```php
<?php

ob_start();
    include_once ('sensfunc.php');

define ('IN_PHPBB', true);
$phpbb_root_path = '.';
include ($phpbb_root_path . 'extension.inc');
include ($phpbb_root_path . 'common', $phpEx);
$userdata = session_pagestart ($user_ip, PAGE_INDEX);
init_userprefs($userdata);
$viewcat = (! empty ($HTTP_GET_VARS [POST_CAT_URL])) ?
    HttpVar_track (POST_CAT_URL, $HTTP_GET_VARS [POST_CAT_URL], GET) : -1;
if (isset ($HTTP_GET_VARS ['mark']) || isset ($HTTP_POST_VARS ['mark']))
{
    $mark_read = (isset ($HTTP_POST_VARS ['mark'])) ?
        HttpVar_track ('mark', $HTTP_POST_VARS ['mark'], POST) :
        HttpVar_track ('mark', $HTTP_GET_VARS ['mark'], GET);
}
else
{
    $mark_read = '';
}
if ($mark_read == 'forums')
{
    if ($userdata ['session_logged_in'])
    {
        setcookie ($board_config ['cookie_name'] . '_f_all', time (), 0,
            $board_config ['cookie_path'], $board_config['cookie_domain'],
            $board_config['cookie_secure']);
        cookie_track ($board_config ['cookie_name'] . '_f_all', time (), 0,
            $board_config ['cookie_path'], $board_config['cookie_domain'],
            $board_config['cookie_secure']);
    }
}
...

$template -> pparse ('body');
include ($phpbb_root_path . 'includes/page_tail'. $phpEx);
ob_flush ();
?>
```
Dynamic test case generation and scripting, using WATIR open source tool

- **Web Application Testing In Ruby**
- Scriptable library to drive browser by clicking links, pressing buttons, filling in forms
STAGE 2 - EXERCISE

- Automatic trace collection from instrumentation
  - No human interaction
Automatic dynamic filtering
- Avoids unmanageably large trace database
- Automatically minimizes unique paths, loops
- New page, old page with new parameters, old page with same parameters but new path
### Stage 2 - Exercise

- Automatic dynamic filtering

<table>
<thead>
<tr>
<th>Page ID</th>
<th>Page Name</th>
<th>Page parameters</th>
<th>Prev. Page ID</th>
<th>Page Type</th>
<th>Page Acc_TS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><a href="http://phpBB2/index.php">http://phpBB2/index.php</a></td>
<td></td>
<td>0</td>
<td>PHP</td>
<td>1211134854</td>
</tr>
<tr>
<td>2</td>
<td><a href="http://phpBB2/viewforum.php">http://phpBB2/viewforum.php</a></td>
<td>?f=1&amp;sid=668ea9c6f9d3530aa85152da6fc3d7c6</td>
<td>1</td>
<td>PHP</td>
<td>1211134870</td>
</tr>
<tr>
<td>4</td>
<td><a href="http://phpBB2/posting.php">http://phpBB2/posting.php</a></td>
<td>?mode=reply&amp;t=5</td>
<td>3</td>
<td>PHP</td>
<td>1211134902</td>
</tr>
<tr>
<td>6</td>
<td><a href="http://phpBB2/login.php">http://phpBB2/login.php</a></td>
<td></td>
<td>5</td>
<td>PHP</td>
<td>1211134918</td>
</tr>
<tr>
<td>7</td>
<td><a href="http://phpBB2/posting.php">http://phpBB2/posting.php</a></td>
<td>?mode=reply&amp;t=5&amp;sid=668ea9c6f9d3530aa85152da6fc3d7c6</td>
<td>6</td>
<td>PHP</td>
<td>1211134918</td>
</tr>
<tr>
<td>8</td>
<td><a href="http://phpBB2/posting.php">http://phpBB2/posting.php</a></td>
<td>?mode=topicview&amp;t=5</td>
<td>6</td>
<td>PHP</td>
<td>1211134919</td>
</tr>
<tr>
<td>9</td>
<td><a href="http://phpBB2/posting.php">http://phpBB2/posting.php</a></td>
<td></td>
<td>8</td>
<td>PHP</td>
<td>1211134957</td>
</tr>
<tr>
<td>13</td>
<td><a href="http://phpBB2/login.php">http://phpBB2/login.php</a></td>
<td></td>
<td>12</td>
<td>PHP</td>
<td>1211135018</td>
</tr>
</tbody>
</table>
## Stage 2 - Exercise

<table>
<thead>
<tr>
<th>Page ID</th>
<th>Page Name</th>
<th>Page parameters</th>
<th>Prev. Page ID</th>
<th>Page Type</th>
<th>Page Acc_TS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><a href="http://phpBB2/index.php">http://phpBB2/index.php</a></td>
<td></td>
<td>0</td>
<td>PHP</td>
<td>1211134854</td>
</tr>
<tr>
<td>2</td>
<td><a href="http://phpBB2/viewforum.php">http://phpBB2/viewforum.php</a></td>
<td>?f=1&amp;sid=668ea9c6f9d3530aa85152da6fc3d7c6</td>
<td>1</td>
<td>PHP</td>
<td>1211134870</td>
</tr>
<tr>
<td>4</td>
<td><a href="http://phpBB2/posting.php">http://phpBB2/posting.php</a></td>
<td>?mode=reply&amp;t=5</td>
<td>3</td>
<td>PHP</td>
<td>1211134902</td>
</tr>
<tr>
<td>6</td>
<td><a href="http://phpBB2/login.php">http://phpBB2/login.php</a></td>
<td></td>
<td>5</td>
<td>PHP</td>
<td>1211134918</td>
</tr>
<tr>
<td>7</td>
<td><a href="http://phpBB2/posting.php">http://phpBB2/posting.php</a></td>
<td>?mode=reply&amp;t=5&amp;sid=668ea9c6f9d3530aa85152da6fc3d7c6</td>
<td>6</td>
<td>PHP</td>
<td>1211134918</td>
</tr>
<tr>
<td>8</td>
<td><a href="http://phpBB2/posting.php">http://phpBB2/posting.php</a></td>
<td>?mode=topicreview&amp;t=5</td>
<td>7</td>
<td>PHP</td>
<td>1211134919</td>
</tr>
<tr>
<td>9</td>
<td><a href="http://phpBB2/posting.php">http://phpBB2/posting.php</a></td>
<td></td>
<td>8</td>
<td>PHP</td>
<td>1211134957</td>
</tr>
<tr>
<td>13</td>
<td><a href="http://phpBB2/login.php">http://phpBB2/login.php</a></td>
<td></td>
<td>12</td>
<td>PHP</td>
<td>1211135018</td>
</tr>
</tbody>
</table>

### Detailed Table:

<table>
<thead>
<tr>
<th>Page ID</th>
<th>Page Name</th>
<th>Page Param.</th>
<th>PageAccess</th>
<th>Prev</th>
<th>HttpVar Name</th>
<th>HttpVar Value</th>
<th>HttpVar Type</th>
<th>HttpVar Time_stamp</th>
<th>CookiesName</th>
<th>Cookies value</th>
<th>Cookies ExpireTime</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><a href="http://phpBB2/index.php">http://phpBB2/index.php</a></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
<td>phpbb2mysql_data</td>
<td>a:2:{s:11:“autologinId”;s:0:“”;s:5:“userid”};i:1}</td>
<td>1242570855</td>
</tr>
<tr>
<td>2</td>
<td><a href="http://phpBB2/viewforum.php">http://phpBB2/viewforum.php</a></td>
<td></td>
<td></td>
<td></td>
<td>f</td>
<td>1</td>
<td>GET</td>
<td>1211134871</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
</tr>
<tr>
<td>3</td>
<td><a href="http://phpBB2/viewtopic.php">http://phpBB2/viewtopic.php</a></td>
<td></td>
<td></td>
<td></td>
<td>t</td>
<td>5</td>
<td>GET</td>
<td>1211134895</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
</tr>
<tr>
<td>4</td>
<td><a href="http://phpBB2/posting.php">http://phpBB2/posting.php</a></td>
<td>?mode=reply&amp;t=5</td>
<td></td>
<td></td>
<td>mode</td>
<td>reply</td>
<td>GET</td>
<td>1211134903</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
</tr>
<tr>
<td>5</td>
<td><a href="http://phpBB2/login.php">http://phpBB2/login.php</a></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
</tr>
<tr>
<td>6</td>
<td><a href="http://phpBB2/login.php">http://phpBB2/login.php</a></td>
<td></td>
<td></td>
<td></td>
<td>username</td>
<td>admin</td>
<td>POST</td>
<td>1211134918</td>
<td>phpbb2mysql_data</td>
<td>a:2:{s:11:“autologinId”;s:0:“”;s:5:“userid”};i:1}</td>
<td>1242570818</td>
</tr>
</tbody>
</table>
Database analysis and model generation
- Transformation from trace DB to UML 2.0 sequence model in XMI 2.1
- Visualized and refined using any UML 2.0 toolset
STAGE 3 – EXTRACT MODEL

- Database analysis and model generation

Diagram:
- Browser Session
  - index
  - viewforum
  - login
  - :posting
  - viewtopic

1: index()
1.1: viewforum(?f=1, sid=668ea9c6f9d3530aa85152da6fc3d7c6)
1.1.1: viewtopic(?t=5)
1.1.1.1: posting(?mode=reply, t=5)
1.1.1.1.1: login(?redirect=posting.php, mode=reply, t=5)
1.1.1.1.1.1: login()
CURRENT STATUS

- Phase 1 (Behavior)
  - PHP2XMI
  - Completed, tuning (this work)
CURRENT STATUS

- **Phase 2 (Structure)**
  - Extracts ER class model
  - Completed, tuning [WCRE 08]

Diagram:
- **Static Analysis**
  - PHP TXL Grammar
  - TXL Transformation Rules
  - Static Analysis
- **Instrumentation**
  - PHP Documents
  - TXL Transformation Rules
  - Static Analysis
- **Static Information**
  - Static Information
  - Identification of Users, Roles, Resources
  - Static Information
  - Class Diagram
  - Sequence Diagram
- **Dynamic Information**
  - Execution Traces
  - Dynamic Information
  - MySQL TXL Grammar
  - TXL Transformation Rules
  - MySQL TXL Grammar
  - Database schema
- **Mutant Operators**
  - Instrumented PHP documents
  - Mutant Operators
  - Instrumented PHP documents
  - Random User Input
  - Mutant Instrumented PHP documents
- **Instrumented PHP documents**
  - Instrumented PHP documents
  - MUTANT
  - Instrumented PHP documents
- **Model Checker**
  - Correct
  - Counter Example
  - Access control properties
  - XMI SecureUML Model
  - XMI to Formal Model transformation
CURRENT STATUS

- Phase 3 (Unify)
  - Unified security model
  - Completed, tuning

Flowchart:
- Identificaion of Users, Roles, Resources
- XMI SecureUML Model
- XMI to Formal Model transformation
  - Access control properties
  - Correct
  - Counter Example

Processes:
- Static Analysis
- Instrumentation
- Static Information
- Mutant Operators
- Instrumented PHP documents
- Random User Input
- Mutant Instrumented PHP documents
- Execution Traces
- Dynamic Information
- Class Diagram
- Sequence Diagram
- PHP TXL Grammar
- PHP Documents
- TXL Transformation Rules
- MySql TXL Grammar
- TXL Transformation Rules
- Database schema
CURRENT STATUS

- Phase 4 (Model check)
  - Model checking of unified model for access control
  - In progress
RELATED WORK IN BEHAVIORAL MODELS

- Many techniques for standard OO systems
  - CPP2XMI, Briand et al., …
  - Do not deal with web applications, mixed languages
  - Mostly use XMI 1.0 (old standard) - not compatible with UML tools

- Other web application techniques recover behavioral models
  - But in custom formats, not UML standard sequence diagrams
    - Ricca and Tonella (ReWeb, TestWeb(2001))
      - UML object model + custom behavioral model format
      - Aimed at generating test cases
    - Antoniol et al. (WANDA)
      - Execution traces are not filtered, scalability problems
      - UML sequence diagram + other models, in custom model format
      - Overhead in processing multilingual documents
    - Di Lucca and Di Penta (filtered WANDA)
      - Statically identify and cluster equivalent web application client pages
      - Does not preserve all possible paths the application allows users
**CONCLUSION**

- PHP2XMI: a new reverse engineering tool for dynamic behavior of web applications
  - Scalable, adaptable model recovery
  - Largely language independent
  - Automated instrumentation
  - Dynamic trace filtration
  - Uses UML / XMI 2.1 standards
  - Supports web application validation and security testing using standard tools
CURRENT AND FUTURE WORK

- **Coverage**
  - Enumerate all possible executions for each role
  - Added coverage instrumentation to assure accuracy and completeness of the generated sequence diagram
    - By unique page, PHP function, SQL query/update, HTTP var

- **Level of detail**
  - Raise the sequence diagram to the entity level from the page level
  - Integrate sequence diagrams from different sessions into a sequence diagram for the entire web application

- **Generalization**
  - Testing of other web application vulnerabilities
  - Other scripting languages
  - Extension to web services
Questions?

Automated Reverse Engineering of UML Sequence Diagrams for Dynamic Web Applications

Manar Alalfi, James Cordy, Thomas Dean