Statistical Reconstruction of Class Hierarchies in Binaries

Omer Katz  Noam Rinetzky  Eran Yahav

Problem: Existing techniques fail to reconstruct an actual class hierarchy resulting in gaps and partial hierarchies
Goal: Identify and output the most likely class hierarchy for a stripped binary
Solution: Reconstruct hierarchy based on behavioral similarity between types using tracelets

Evaluation Scenario:
Q: which types inherit from type t?

• Useful for Control Flow Integrity
• Relevant to virtual function calls
• Generate policy from hierarchy
• Additional types → false positives
• Missing types → false negatives

Input

\[
\begin{align*}
 t_2 & \quad t_3 \\
 t_4 & \quad t_1
\end{align*}
\]

set of types

Type representation

Collect a set of tracelets for each type
• Statically analyze the binary
• Extract usage sequences for types
• Sequence of operations applied to instances of that type
  • Filed accesses, function calls, etc...

\[
T_r(t) = \{W(16) \rightarrow R(0) \rightarrow C(0) \rightarrow R(8), \ldots \}
\]

Output

Minimum-weight directed spanning tree

Convert to a weighted graph

\[
\begin{align*}
 W(t_1 \rightarrow t_2) &= D_{KL}(M_t1 || M_t2) \\
 t_2 & \quad t_3 \\
 t_4 & \quad t_1
\end{align*}
\]

Simplify graph

• Using structural cues used by existing techniques
• Split to smaller subgraphs
  • Shared virtual table entries
• Eliminate impossible edges
  • Call to parent constructor
  • Virtual table size
  • etc...

Evaluate

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>size (Kb)</th>
<th>num of types</th>
<th>Without SLMs</th>
<th>Missing</th>
<th>Added</th>
<th>With SLMs</th>
<th>Missing</th>
<th>Added</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyzer</td>
<td>419</td>
<td>24</td>
<td>0.21</td>
<td>6.79</td>
<td>0.25</td>
<td>1.38</td>
<td>0.25</td>
<td>1.38</td>
</tr>
<tr>
<td>CGGridListCtrlEx</td>
<td>151</td>
<td>28</td>
<td>0.0</td>
<td>0.46</td>
<td>0.07</td>
<td>0.07</td>
<td>0.07</td>
<td>0.07</td>
</tr>
<tr>
<td>echoparams</td>
<td>58</td>
<td>4</td>
<td>0.0</td>
<td>2.25</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>gperf</td>
<td>84</td>
<td>10</td>
<td>0.0</td>
<td>3.8</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>libtemplate</td>
<td>1233</td>
<td>36</td>
<td>0.25</td>
<td>0.33</td>
<td>0.25</td>
<td>0.11</td>
<td>0.25</td>
<td>0.11</td>
</tr>
<tr>
<td>ShowTraf</td>
<td>137</td>
<td>25</td>
<td>0.04</td>
<td>0.4</td>
<td>0.04</td>
<td>0.08</td>
<td>0.04</td>
<td>0.08</td>
</tr>
<tr>
<td>Smoothing</td>
<td>453</td>
<td>31</td>
<td>0.19</td>
<td>7.9</td>
<td>0.23</td>
<td>1.1</td>
<td>0.23</td>
<td>1.1</td>
</tr>
<tr>
<td>tld_unittest</td>
<td>101</td>
<td>2</td>
<td>0.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.5</td>
<td>0.0</td>
<td>0.5</td>
</tr>
<tr>
<td>tinyserver</td>
<td>46</td>
<td>4</td>
<td>0.0</td>
<td>2.25</td>
<td>0.0</td>
<td>0.25</td>
<td>0.0</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Highlights:
• echoparams: reconstruct exact hierarchy
• Smoothing: from 7.9 false positives to 1.1
• Analyzer: from 6.79 false positives to 1.38

• Tradeoff between false negatives and false positives
  • Can use more than a single hierarchy to generate policy
  • Reduces missing types but increases added types

The research leading to the results presented in this paper is partially supported by the European Union’s Seventh Framework Programme (FP7) under grant agreement no. 615688 (PRIME) and the Israel Science Foundation grant no.1319/16.