#332 Neural Machine Translation Inspired Cross-Architecture Binary Code Similarity Detection

Your submissions  
(All) Search

PC conflicts
None

Rejected

Submission  12 Aug 2017 12:14:06am EDT  329e863b

▼ Abstract
The problem of cross-architecture code similarity detection is to detect whether two binary functions compiled for different architecture are semantically equivalent or similar. It has many security applications, such as bug search, vulnerability discovery, clone detection, malware detection, and software plagiarism detection, etc. Existing code similarity detection approaches, however, fall short in that they 1) require source code, 2) handle only a single architecture (typically x86), 3) rely on dynamic analysis, which is difficult for embedded devices, or 4) use formal methods, which is very slow when analyzing large code bases.

▼ Authors (blind)
Qiang Zeng (Temple University)  
<qzeng@temple.edu>
Lannan Luo (University of South Carolina)  
<lluo@cse.sc.edu>

Contact
Lannan Luo <lzl144@ist.psu.edu>

▼ Topics

https://ndss18.hotcrp.com/paper/332
processing various languages. In the case of detecting similarity of binaries, which are compiled for different architectures and hence can be expressed in different assembly languages, we are inspired by Neural Machine Translation (a neural network-based approach recently proposed by NLP researchers to performing machine translation), and propose a novel neural network-based cross-lingual deep learning approach, which converts a binary function into an embedding, i.e., a high dimensional numerical vector, such that the code similarity detection can be done efficiently by measuring the distance between the embeddings of functions. Such embeddings capture not only semantics of functions, but also semantic relationships across various architectures. This property allows the embeddings to define semantic similarity metrics across binary functions, making them perfect features for cross-architecture code similarity detection. We implement a prototype, and evaluate it with different experiments. We conduct a systematic comparison against the state-of-the-art approaches. The experimental results demonstrate that our approach can successfully solve the cross-architecture code similarity detection task, and outperforms the state-of-the-art approaches by large margins with respect to both accuracy, efficiency and scalability. Our real-world vulnerability case studies and cryptographic function detection tasks demonstrate that our approach can identify more vulnerable firmware images than the
prior state-of-the-arts and detect
cryptographic function with high accuracy.
Our research gives a successful
demonstration of applying approaches in
NLP to large-scale binary analysis.

To edit this submission, sign in using your email and password.

<table>
<thead>
<tr>
<th>OveMer</th>
<th>RelRan</th>
<th>RevExp</th>
<th>WriQua</th>
</tr>
</thead>
<tbody>
<tr>
<td>Review #332A</td>
<td>3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Review #332B</td>
<td>1</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Review #332C</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Review #332D</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

1 Comment: Response (L. Luo)