Evaluation of Fuzzy Hashing for Ethereum Smart Contract Analysis

Bachelor's Thesis in Software and Information Engineering

Author: Raphael Nußbaumer - 01526647 - raphaeln@outlook.com

Advisor: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Gernot Salzer

1 Abstract

This thesis explores fuzzy hashing methods to compute similarities between EVM byte-codes. For this purpose a set of python utilities was implemented and an data-set for evaluation generated.

2 Goal

The goal was to compare different similarity measures and pre-processing steps and evaluate how they respond to changes in the codes. This should be accomplish with a set of reusable python utilities.

3 Data sets

3.1 Solc Versions and Options

To evaluate the similarity measures I selected a set of 12 solidity smart contracts and compiled them with different solc version and compiler options. Necessary changes where made to the source code to ensure compatibility with the various solc versions.

The following solc version where used:

1. `0.5.16`
2. `0.6.12`
3. `0.7.6`
4. `0.8.4`

To evaluate the effect of optimization I applied the following options:

1. `{ enabled: false, runs: 200 }`
2. `{ enabled: true, runs: 0 }`
3. `{ enabled: true, runs: 200 }`
4. `{ enabled: true, runs: 999999 }`

Further the contracts where compiled with ABI encoding v1 and v2.

4 Pre-Processing

4.1 Segmentation

4.2 Skeletonization

4.3 Op-Code Filtering

5 Hashing Methods

5.1 ssdeep - Context Triggered Piecewise Hashes (CTPH)

Wenn der Algorithmus schon in einer Quelle exakt beschrieben ist, so wie Sie es verwendet haben, geben Sie eine genaue Referenz, inklusive Seite (oder Abbildungsnummer im Falle eines Algorithmus) an. In Ihrer Arbeit sollten Sie aber zumindest mit ein paar Sätzen beschreiben, wie die Methode vorgeht und worin sie sich von anderen Ansätzen unterscheidet. Kann aber ganz kurz sein.

5.2 Op-Code Frequency

5.3 LZJD - Binary-Hashing

This is how a cite[1] looks like.

6 Test Framework

7 Results

endpunkte (was da ist), hypotesen

8 Remarks

Section um alles zu notieren, was Ihnen an Absonderlichkeiten oder Schwierigkeiten untergekommen ist, inklusive Lessons learned (also was Sie beachten würden, wenn Sie nochmals beginnen würden); wenn es da viel zu berichten gibt, können Sie es natürlich weglassen. Der Sinn einer eigenen Section ist, dass Sie hier Ihre Eindrücke informell ohne tiefere Begründungen wiedergeben können, während die Aussagen in den anderen Abschnitten begründet sein sollten.

9 Conclusion

next steps

goals for followup papers

References

[1] Andrew H Sung et al. “Static analyzer of vicious executables (save)”. In: *20th Annual Computer Security Applications Conference*. IEEE. 2004, pp. 326–334.