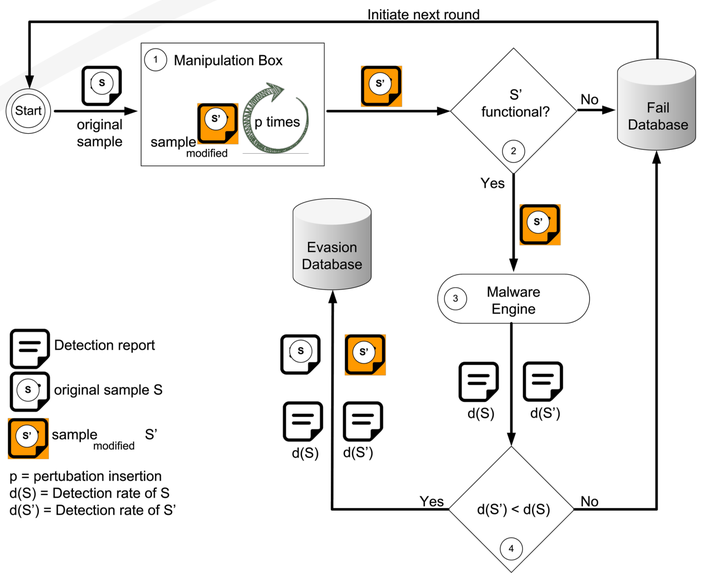
AIMED: Genetic Programming to Evade Malware Detection

Raphael Labaca Castro, Corinna Schmitt, and Gabi Dreo Rodosek
Research Institute CODE
Bundeswehr University Munich, Germany

Introduction

Genetic Programming (GP) has previously proved to achieve valuable results on the fields of image processing and arcade learning. Similarly, it can be used as an adversarial learning approach to evolve malware samples until static learning classifiers are no longer able to detect it.



Objectives

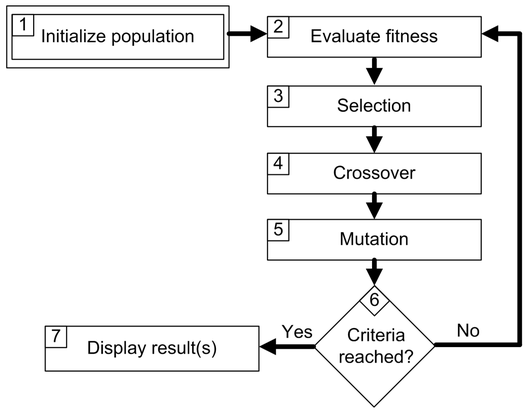
AIMED – **A**utomatic **I**ntelligent Malware **M**odifications to **E**vade **D**etection – was designed and implemented using genetic algorithms to evade static malware classifiers.

Automatically achieving evasion on static malware scanners is interesting to understand how byte-level modifications impact malicious samples without rendering them corrupt.

Methods

AIMED consists of three main components:

1) The manipulation box where perturbations are injected to the malware sample.
2) A sandbox where new malware mutations are tested to make sure they are valid.
3) A malware scanner that provides the detection result for every new malware mutation.



Step 1: It starts with a number of random malware mutations that is called *population*.
Step 2: Each member of the population will be evaluated in terms of fitness and receive a score. Hence, the new mutation runs through the sandbox to check whether the file is not corrupt and then through the detection stage.
Step 3: The two fittest members will be selected to breed the next generation.
Step 4: The selected members mate with each other to generate offspring, which are more prone to result in evasions.
Step 5: All malicious files receive random genetic mutations in unexpected ways similar to evolution in nature.
Step 6: The whole process is repeated over many generations until a number of evasive samples is achieved or a threshold of generations is surpassed.

Results

- Required time to generate evasive mutations can be reduced up to 50% compared to random approaches.
- The number of corrupt mutations generated shows to decrease leading to more functional mutations.
- In case of small sample size the random approaches scale better than GP-based approaches, due to the higher probability of finding an evasion in a shorter time.

Conclusions

- AIMED generates large number of functional adversarial examples in less time than random approaches.
- GP is a relatively simple yet powerful option to create adversarial examples of malware mutations.

Next steps

- Adjusting mutation rates during step 5 can help identify optimal conditions to achieve faster evasions.
- Extending investigations with different malware types and finding correlations between structure and evasions.
- Searching for an optimal population number, because small populations are faster in generation 1 but provide only a limited number of genes.

Acknowledgements

The authors would like to thank the Chair for Communication Systems and Network Security, headed by Prof. Gabi Dreo Rodosek, and the research institute CODE for their comments.

der Bundeswehr Universität München
Werner-Heisenberg-Weg 39 . 85577 München–Neubiberg . Telefon: +49 89 6004 2244 . E-Mail: code@unibw.de . **www.unibw.de/code**