Call graph based Secure Malware Detection using Machine Learning



The objective of this project is to develop a robust and scalable solution for malware detection, a critical area of research within the cybersecurity domain. This project initially aims to develop a tool designed to statically extract call graphs[1] from Executable (PE[2] - Windows or ELF[3] - Linux) files without requiring execution. Subsequently, The extracted call graphs will be leveraged to improve malware detection capabilities.

Project Objectives

- Develop a Python-based tool to parse PE/ELF files and extract call graph information statically.
- Generate call graph representations in desired formats (CSV, JSON, Graphviz DOT)
- Prepare dataset from the extracted features to facilitate machine learning applications and train the model using Machine Learning
- Investigate the achievement of zero false negatives (detect all malware) and the minimization of false positives (false malware alarm) through the use call graph features.

Profile/Required Skills

- Be interested or motivated in reverse engineering tools/framework, e.g., LIEF, Capstone, r2.
- Be familiar with Python, Scikit-Learn, Auto-Sklearn

Practicalities

Promotor:	Bart Preneel		
Daily supervisor:	Enrique Argones Rúa	Enrique.ArgonesRua@esat.kuleuven.be	office 0x.xx
	Aysajan Abidin	Aysajan.Abidin@esat.kuleuven.be	office 0x.xx
	Irfan Bulut	Irfan.Bulut@esat.kuleuven.be	office 0x.xx
Nature of the work:	30% literature, 20% theoretical work, 50% software		
Number of students:	1 or 2		

[1] `https://en.wikipedia.org/wiki/Call_graph`
[2] `https://en.wikipedia.org/wiki/Portable_Executable`
[3] `https://en.wikipedia.org/wiki/Executable_and_Linkable_Format`

 © 2026 – 2027 COSIC