17 June 2019 Study · Privacy & Crypto

Topological Analysis of Program Code

The full text is currently only available in the Deutsch version. A translation will follow.

Archived. This content is no longer updated. It remains available for reference.

Version 2019-06-13
Source https://technology2.a-sit.at/en/publications/topological-analysis-of-program-code/

The findings from this project show an innovative approach to hierarchically prepare program code and to map this hierarchy „flat" again as a result. The concept pursued here can be used to simplify the analysis of large applications through abstraction and to find certain implementation patterns in programs.

In the security-oriented analysis of applications, it frequently happens that the properties of programs that are actually to be examined are becoming increasingly difficult to find and trace in the code. Increasing complexity and more extensive range of functions of programs contribute to the fact that manual checks of the implementation can only concentrate on certain aspects (e.g. code with cryptographic function routines), but other, potentially also security-relevant aspects are ignored or cannot be analyzed in all occurring usage scenarios.

In the course of this project, an approach was sought to prepare program code in such a way that a topological search is possible. In view of the given challenges, the graphical representation of code as AST (Abstract Syntax Tree) for the search of related code fragments was obvious. The aim was to simplify the analysis of large applications and to efficiently determine where safety-critical program parts are located.