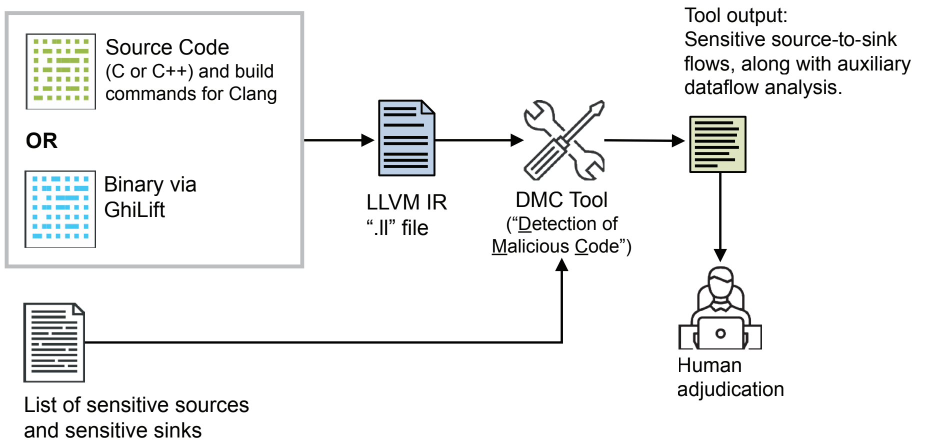
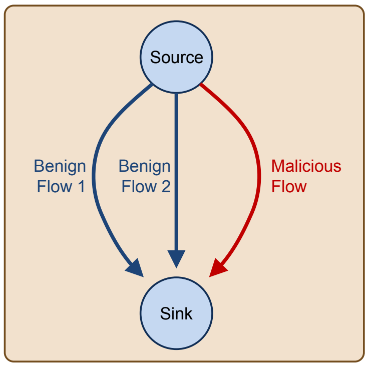
Carnegie Mellon University
Software Engineering Institute

Detection of Malicious Code: Taint Flow Analysis for Weapons Systems Software

for the 2024 DoD Weapon Systems Software Summit


Source Code
(C or C++) and build commands for Clang
OR
Binary via GhiLift
LLVM IR
".ll" file
DMC Tool
("Detection of Malicious Code")
Tool output:
Sensitive source-to-sink flows, along with auxiliary dataflow analysis.
Human adjudication
List of sensitive sources and sensitive sinks



Our tool detects two types of malicious code: (1) exfiltration of sensitive data and (2) timebombs, logic bombs, remote-access Trojans (RATs), and similar malicious code. We use only static analysis, not dynamic analysis, so we do not need to execute potentially malicious code. While our focus is on C/C++ codebases, we perform all analysis at the level of LLVM intermediate representation (IR), so we also have some support for binaries via lifting to LLVM IR. Our tool uses taint analysis to track the flow of sensitive and auxiliary information. A limitation of traditional taint analysis is that it conflates all flow paths from a given source to a given sink, so a malicious flow path can be "hidden" by a benign flow path. In contrast, in this project we use auxiliary information (e.g., whether a filename was specified by a local user or whether it came from network data) to help separate benign flows from malicious flows.

Our tool flags code as potentially malicious, but further human analysis is required. Whether behavior is malicious depends on the what the program is supposed to do. The goal for our tool is to produce output that concisely and precisely characterizes the potentially malicious behaviors of the codebase, so that a human analyst can quickly and accurately determine whether the behavior is benign or malicious.

Tool on GitHub

Project page:
https://github.com/cmu-sei/dmc

Contact us if you're interested in using our tool or working with us to extend it !

Will Klieber
weklieber@sei.cmu.edu
Software Security Engineer
Principal Investigator

Lori Flynn, PhD
lflynn@sei.cmu.edu
Senior Software
Security Researcher