Jordan University of Science and Technology

Towards the Memory Forensics of OOP Execution Behavior

Authors: Ziad A. Al-Sharif, Mohammad I. Al-Saleh and Luay Alawneh

Abstract: Perpetrators might employ computer programs (software) to perform their offenses or to cover their wrongdoings. Program's source code, in particular variables and their values diverge in their scope and duration in RAM. Object oriented languages encapsulate variables and operations in classes and allow for objects to be instantiated, which simplify software design but add to the complexity of its execution behavior and its data and control flow. This paper explores execution behaviors and information left by program execution in support for legal actions against perpetrators in the court of law. Our investigation model assumes no information is provided by the operating system; only raw RAM dumps. Our methodology employs information from the presumed program source code and its object oriented design. It explores various execution states and scenarios to uncover the evidence of potential software usage. These scenarios are designed to show that scope, access modifiers, and storage information of various source code variables can be used to identify program's activities. Results show that investigators have good chances locating various variables' values that are uniquely corresponded to the presumed software and its execution states. In some cases, values are successfully identified in memory dumps even after the process is stopped.