Data mining the memory access stream to detect anomalous application behavior

Francis B. Moreira
Informatics Institute UFRGS
fbmoreira@inf.ufrgs.br

Matthias Diener
Informatics Institute UFRGS
mdiener@inf.ufrgs.br

Philippe O. A. Navaux
Informatics Institute UFRGS
navaux@inf.ufrgs.br

Israel Koren Department of Electrical
Computer
Engineering
University of Massachusetts
koren@ece.umass.edu

Abstract

Detecting anomalous application executions is a challenging problem, due to the diversity of anomalies that can occur, such as programming bugs, silent data corruption, or even malicious code corruption. Moreover, the similarity to a regular execution that can occur in these cases, especially in silent data corruption, makes distinction from normal executions difficult. In this paper, we develop a mechanism that can detect such anomalous executions based on changes in the memory access pattern of an application. We analyze memory patterns using a two-level machine learning approach. First, we classify the behavior of different memory access periods within applications using Gaussian mixtures. Then, based on these classifications, we construct matrix representations of Markov chains to obtain information regarding the temporal behavior of these memory accesses. Based on metrics of matrix similarity, we can classify whether the application behaves as expected or anomalously. Using gradient boosting on the metrics of matrix similarity, our technique correctly classifies more than 85of all executions, identifying instances of the same application and different applications. We can also detect a range of faulty executions caused by benign or malicious permanent bit flips in the code section.