Ensemble Learning for Effective Run-Time Hardware-Based Malware Detection: A Comprehensive Analysis and Classification

Hossein Sayadi, Nisarg Patel, Sai Manoj P D, Avesta Sasan, Setareh Rafatirad, Houman Homayoun
George Mason University, Fairfax, VA, USA 22030
{hsayadi, npatel33, spudukot, asasan, srafatir, hhomayou}@gmu.edu

ABSTRACT

Malware detection at the hardware level has emerged recently as a promising solution to improve the security of computing systems. Hardware-based malware detectors take advantage of Machine Learning (ML) classifiers to detect pattern of malicious applications at run-time. These ML classifiers are trained using low-level features such as processor Hardware Performance Counters (HPCs) data which are captured at run-time to appropriately represent the application behaviour. Recent studies show the potential of standard ML-based classifiers for detecting malware using analysis of large number of microarchitectural events, more than the very limited number of HPC registers available in today's microprocessors which varies from 2 to 8. This results in executing the application more than once to collect the required data, which in turn makes the solution less practical for effective run-time malware detection. Our results show a clear trade-off between the performance of standard ML classifiers and the number and diversity of HPCs available in modern microprocessors. This paper proposes a machine learning-based solution to break this trade-off to realize effective run-time detection of malware. We propose ensemble learning techniques to improve the performance of the hardware-based malware detectors despite using a very small number of microarchitectural events that are captured at run-time by existing HPCs, eliminating the need to run an application several times. For this purpose, eight robust machine learning models and two well-known ensemble learning classifiers applied on all studied ML models (sixteen in total) are implemented for malware detection and precisely compared and characterized in terms of detection accuracy, robustness, performance (accuracy×robustness), and hardware overheads. The experimental results show that the proposed ensemble learning-based malware detection with just 2 HPCs using ensemble technique outperforms standard classifiers with 8 HPCs by up to 17%. In addition, it can match the robustness and performance of standard ML-based detectors with 16 HPCs while using only 4 HPCs allowing effective run-time detection of malware.

KEYWORDS

Malware Detection, Hardware Performance Counters, Ensemble Learning

1. INTRODUCTION

Malware is a piece of code designed to perform various malicious activities, such as destroying the data, stealing information, running destructive or intrusive programs on devices to perform Denial-of-Service (DoS) attack, and gaining root access without the consent of user. According to a 2017 McAfee threats report [12], 57.6 million new malware samples have been recorded in the third quarter of 2017, an all-time highest number with an increase of 10% from the second quarter. Furthermore, the overall counts of new malware samples grew by 27% in 2017 to 781 million samples. The recent proliferation of computing devices in mobile and Internet-of-Things domains further exacerbates the malware attacks and calls for effective malware detection techniques.

Malware detection can be simplified as a binary classification problem regardless of what detection method is being used. It is basically envisioned as distinguishing whether the running application has malicious intent or not. Traditional malware detection approaches such as signature-based detection and semantics-based anomaly detections are considered as software-based solutions and incur significant computational overheads [10]. Recent studies have demonstrated that malware behavior can be differentiated from benign applications by classifying anomalies in the low-level feature spaces such as microarchitectural events collected by Hardware Performance Counter (HPC) registers [3,4,5,11,13,15,16,24]. HPCs are CPU hardware registers that count hardware events such as instructions executed, cache-misses suffered, or branches mispredicted. Performance counters data have been extensively used to predict the power, performance, and energy efficiency of computing systems [14,20,22], and recently drew attentions to be used for detecting the malicious pattern of running applications to improve the security of systems. Thus, malware detection using HPCs microarchitectural events has emerged as a promising alternative to traditional malware detection methods [3,4,11,13,24]. As learning the underlying patterns of these microarchitectural events can aid in detecting malware, machine learning (ML) techniques are widely deployed for malware detection. The HPC microarchitectural features are used to train ML-based classifiers. In addition, such ML-based malware detection methods can be implemented in microprocessor hardware with significantly low overhead as compared to the software-based methods, as detection inside the hardware is very fast (few clock cycles) [4].

Recently, there has been a number of work on hardware-based malware detection using HPCs information [3,4,11,13,14,24]. However, these works performed a limited study on malware classification accounting for the availability of a large number (e.g. 16 or 32) and diverse type of HPCs. While, modern processors in the high-performance domain have a small number of HPCs (2 to 8), due to several reasons including the design complexity and cost of concurrent monitoring of microarchitectural events [17,21,23]. Due to deep pipelines, complex prefetchers, branch predictors, modern cache design etc., HPCs implementation becomes a great challenge in terms of counting multiple events and maintaining counter accuracy at the same time under speculative execution [17]. Better accuracy requires better and more complex hardware design hence increasing the number of counters with limited accuracy doesn't appear to be a good trade-off. Even modern Intel Xeon architectures houses only 4-6 performance counters, compare to 2 in Pentium 4 and server class Intel

DAC '18, June 24–29, 2018, San Francisco, CA, USA

ACM ISBN 978-1-4503-5700-5/18/06…$15.00