CAGDEEP: Mobile Malware Analysis Using Force Atlas 2 with Strong Gravity Call Graph And Deep Learning

Nur Khairani Kamarudin
Faculty of Computing
Universiti Malaysia Pahang Al-Sultan Abdullah
26600 Pekan, Pahang, Malaysia
nurkhairani@uitm.edu.my

Ahmad Firdaus*
Faculty of Computing
Universiti Malaysia Pahang Al-Sultan Abdullah
26600 Pekan, Pahang, Malaysia
firdausza@ump.edu.my

Azlee Zabidi
Faculty of Computing
Universiti Malaysia Pahang Al-Sultan Abdullah
26600 Pekan, Pahang, Malaysia
azlee@ump.edu.my

Mohd Faizal Ab Razak
Faculty of Computing
Universiti Malaysia Pahang Al-Sultan Abdullah
26600 Pekan, Pahang, Malaysia
faizalrazak@ump.edu.my

2023 IEEE 8th International Conference On Software Engineering and Computer Systems (ICSECS) | 979-8-3503-1093-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICSECS58457.2023.10256350

***Abstract*—Today many smart devices are running on Android systems. With the increasing popularity of Android, mobile malware continuously evolves as well, and further attacks Android operating systems. To address these shortcoming issues many security experts use different approaches to detect malware based on various static features. However, by considering only the statistical features, the potential semantic information such as the behavioral feature of the code is overlooked. To leverage the existing static analysis techniques, this study proposes CAGDeep, to reflect deep semantic information of malware samples. The novelty of our study lies in the Force Atlas 2 call graph development to capture malware behavior patterns. Afterwards, this study adopts Convolutional Neural Network (CNN) for malware detection and classification algorithm. We compare CAGDeep with a state-of-the-art Android malware detection approach. Our evaluation results demonstrate that CAGDeep can achieve 80% accuracy for detecting malware.**

Keywords—feature selection, machine learning, mobile malware, call graph

I. INTRODUCTION

Malware refers to any program or code that is intentionally designed to harm computer systems, networks, or devices. It is one of the many threats that are posed to Internet users. Applications in the Android market conceal a large amount of malware, posing a serious threat to Internet security. 560,000 new malwares are discovered daily and that there are currently over 1 billion malware attacks in circulation [1]. Also, McAfee 2023 Mobile threat reported, most common tricks attempted by malicious apps are fraudulent advertisements, obtaining user credentials, and skimming personal information, much of which occurs without the users' knowledge [2].

Malware development has prompted many security practitioners to conduct malware analysis, which aims to examine malware characteristics and behavior. Despite the development of numerous malware analysis approaches, the threat of malware attacks continues to rise [3]. Here, the researchers studied at Android malware in which new features were regularly added [4]. Many kinds of malware used similar features to each other, whereas some other malware used more recent and enhanced features. Consequently, to fight against the rapid growth of Android malware, it is important for the researchers to keep their focus on detecting this malware.

As Android malware continues to grow and evolve, machine learning and deep learning has been proposed as effective malware detection tools at scale, classifying a given application as benign or malicious according to various potential malicious features. Among these, deep learning (DL) has demonstrated its effectiveness in performing large-scale feature learning compared to classical machine learning methods that have limitations in the number of features they can handle. Examples of deep learning approaches include Convolutional Neural Network (CNN) [5]–[7], Recurrent Neural network (RNN) [8], [9], Auto-encoder (AEC) [10], [11], Long Short-Term Memory (LSTM) [12], [13], and Multi-Layer Perceptron (MLP) [14]. In each approach, the methods for extracting features are different from one another. Examples of features considered in these approaches include user permissions, API calls, system call sequence and opcode sequence. Although various features and features selection approaches have been proposed, recent complex malicious behaviors require more complicated features and learning models.

Many malware detection systems are proposed to detect malware based on variety of static features by taking different static features (permission, opcode, API). The extracted static features then will act as input in machine learning algorithms to detect and classify malware. The rise of interest in static malware analysis provides deeper understanding about malicious patterns and promotes efficiency in detecting malicious software. Despite the fact that their promising results with high accuracy have been reported, by focusing on the statistical features of the code, the potential semantic information, such as malicious behavioral features, is overlooked.

In this study, we present a critical call graph development to present a deeper presentation of static features to reflect malware behaviors, by investigating data flow of the program. As part of our approach, we incorporated the Convolutional Neural Network (CNN) as the model for training. To leverage the full potential of CNN in image classification tasks, we transformed each Android application source code into an RGB graph image with node attributes. Then, the neural