Identifying ATT&CK Tactics in Android Malware Control Flow Graph Through Graph Representation Learning and Interpretability

Cameron Noakes

February 2022

1 Sentence Overview

This paper is the first in providing an automation methodology to locate TTP from the ATT&CK framework matrix in a sub-part of the control flow graph (CFG) that describes the execution flow of a malware executable (exe). This methodology merges graph representation learning and tools for machine learning explanation.

The proposed solution presented by the researchers also associates the TTP with a sub graph of a CFG. They use the Graph Neural Network (GNN) and SIR-GN node representation learning approach to process the CFG.

The researchers collected 3250 malware apks, providing 3250 graphs with an average of 5775 nodes and 12581 edges per graph.

2 Introduction

Summary: Malware applications create significant monetary damages and represent a menace for people and businesses.

Problem: The challenging task of mitigating the effects of a malware application requires deep understanding of what the application does.

Connecting malware actions with the specific macro tactics, techniques, and procedures (TTP) enumerated in the ATT&CK framework ontology helps with understanding the malware actions which is what the researchers express interest in solving.

The related work for this specific topic was said to only focus on the detection of malware or on the classification of the malware family. ATT&CK framework matrix TTPs are applied to associate a malware to a specific family of malware, this association is often done by a human interaction.

The aim of the proposed paper was to use the control flow graph to identify which subset of the actions in the graph has a high likelihood of being responsible for each specific TTP.

The proposed solution was a novel approach to identify ATT&CK framework TTPs in a control flow graph (CFG) by applying Graph Machine Learning techniques on Android Malware samples. The proposed solution presented by the researchers also associates the TTP with a sub graph of a CFG. They use the Graph Neural Network (GNN) and SIR-GN node representation learning approach to process the CFG.

3 Methodology

3 sections specified:

- (1) data collection and processing for training a graph TTP classifier.

- (2) SIR-GN graph representation learning procedure integrated into the graph TTP classification.

- (3) an attribution that explains the TTP classification, results through the graph representation learning procedure to identify a CFG sub graph.

The sampled malware that was used in the form of creating and developing the research and proposed solution was a sub sample of the Android malware (apk) provided by Virus Total and use the Virus Total API to collect the human-curated list of TTP from the ATT&CK framework and then convert each android malware into its corresponding control flow graph (CFG).

The apk android malware are passed into a sandbox in order to collect the ATT&CK TTPs. The sandbox used in this process was Hybrid-Analysis Sandbox as it allows the raw Android .apk executable samples to be passed in. The API allows for 100 uploads to the sandbox every 24 hours.

4 Experiments

The expressed results in this literature section describe the data collection statistics, the TTP classification results, and qualitative evaluation of the sub graph identification in the control flow graphs.

The researchers collected 3250 malware apks, providing 3250 graphs with an average of 5775 nodes and 12581 edges per graph. The dataset has 136993 nodes and 333854 edges.

the distributions of the nodes and edges in all the control flow graphs extracted are reported through figure 5 shown in the literature.

The provided classification results are compared combining SIR-GN and standard classification models, with Graph Attention Network (GAN). The comparisons are made in terms of F1-score and accuracy in classifying the TTP.

F1-Scores are commonly used within research using ATT&CK and is a measurement evaluating classification performances that is robust to unbalanced data. All F1-Scores and accuracy's are given in tables.

5 Conclusion

The researchers provide an automated procedure based on graph representation learning and interpretation, able to well classify the different TTP of malware and detect their related sub graphs in the CFG where it also associates the TTP with a sub graph of a CFG and use the Graph Neural Network (GNN) and SIR-GN node representation learning approach to process the CFG.

6 What problem does this solve?

Problem: The challenging task of mitigating the effects of a malware application requires deep understanding of what the application does.

7 How do they solve it?

Solving the outlined problem at hand within the literature was done by a proposed solution presented by the researchers associating the TTP with a sub graph of a CFG and using the Graph Neural Network (GNN) and SIR-GN node representation learning approach to process the CFG.

Where the aim was to use the control flow graph to identify which subset of the actions in the graph has a high likelihood of being responsible for each specific TTP from the ATT&CK framework.

8 How do they assess it?

A qualitative evaluation of the sub graph identification in the control flow graphs was given based on the data.