****

**School of Computer Science and Engineering**

**J Component report**

**Programme          : B.Tech**

**Course Title          : Essentials of Data Analytics**

**Course Code         : CSE3506**

**Slot                        : G2**

**Title** **Malware Detection in Android Based Devices**

**Team Members:**

|  |  |
| --- | --- |
| Name | Reg. no. |
| B Sai Venkat Teja | 20BCExxxx |
| M V H Sai Sriraj | 20BAI1224 |
| Krotha Kayne Thambi | 20BAI1249 |
| Harikrishhan | 20BCExxxx |

**Faculty:  Dr. Sheik Abdullah A**

**Sign: Date:**

# Introduction

Malware, or malicious software, is a major threat to computer systems and networks worldwide. The proliferation of malware attacks has led to a growing need for effective malware detection and classification methods. One type of malware that poses a particular challenge for security analysts is obfuscated malware, which has been intentionally designed to evade detection by antivirus software and other security measures.

Android is the most widely used mobile operating system globally, with over 2.5 billion active devices. The Android platform's popularity has made it a prime target for malware attacks, which can lead to severe damage, such as data loss, financial loss, and identity theft. Therefore, detecting and classifying malware on Android devices is of utmost importance.

In recent years, machine learning and deep learning algorithms have shown promising results in malware detection and classification. Deep learning algorithms can learn and generalize from large amounts of data, enabling accurate and efficient detection and classification of malware. In this project, we aim to develop a system that can detect and classify malware on Android devices using deep learning algorithms.

The system will use a combination of static and dynamic analysis techniques to detect and classify malware on Android devices. Static analysis involves analyzing the code of the malware without executing it, while dynamic analysis involves executing the malware in a controlled environment and monitoring its behavior. The system will use a hierarchical classification approach to classify malware based on their behavior, features, and families.

The system will continuously monitor the Android device for any malware activity and alert the user if any malware is detected. The user will be able to see the status of their device's security on a user-friendly interface, which will display the types of malware detected and the actions taken to mitigate the malware. The system will use a privacy-preserving approach to protect the user's privacy and will be designed to be scalable and capable of handling a large number of Android devices.

The project's significance lies in its ability to prevent malware attacks on Android devices, ensuring the security of mobile devices. The system we develop will be an effective tool for detecting and classifying both known and unknown malware on Android devices, making it easier for users to protect their devices from malware attacks.

# Problem Statement

The CICInvesAndMal2019 dataset's problem statement requires the development and evaluation of efficient methods for Android malware detection and family categorization. To accurately discriminate between malicious and benign programmes, this includes combining static and dynamic characteristics, such as permissions, intents, API calls, log files, and battery conditions.

Adware, ransomware, scareware, and SMS malware were the four categories into which the dataset's researchers divided the malware samples, and they also discovered 42 distinct malware families. The objective is to create models that can accurately classify and categorise each sample of malware as well as find malware that hasn't been seen before.

This issue is significant since new malware types for Android are routinely created, posing a rising danger to mobile security. In order to safeguard users from potentially hazardous apps and to support malware investigation and threat intelligence, effective detection and classification approaches are required.

# About the Dataset

CIC InvesAndMal2019 Dataset: [Investigation on Android Malware 2019 | Datasets | Research | Canadian Institute for Cybersecurity | UNB](https://www.unb.ca/cic/datasets/invesandmal2019.html)

This is a second part of the CICAndMal2017 dataset publicly available namely CICInvesAndMal2019 which includes permissions and intents as static features and API calls and all generated Log files as dynamic features in three steps i.e during installation, before restarting and after restarting the phone. In this dataset, the authors combine the previous dynamic features (80 network flows using CICFlowMeter-V3) with 2-gram sequential relations of API calls to increase the malware category and family classification performance by about 30%. The authors also look at these properties in the two-layer malware analysis approach that is offered. In addition to these, the authors offer additional features that were collected, including battery statuses, log states, packages, process logs, etc.

Laya Taheri, Andi Fitriah Abdulkadir, Arash Habibi Lashkari; **Extensible Android Malware Detection and Family Classification Using Network-Flows and API-Calls,** The IEEE (53rd) International Carnahan Conference on Security Technology, India, 2019

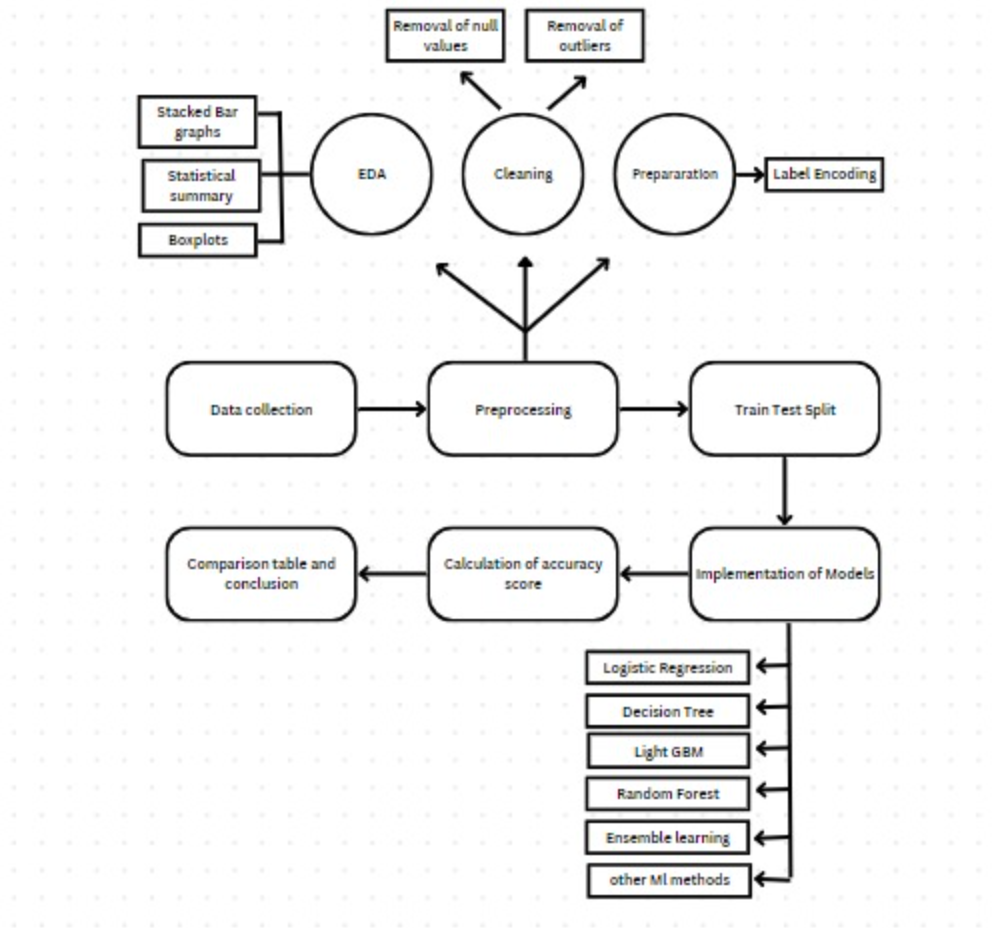
# Novelty

Android malware detection and classification is an important area of research due to the exponential growth of the Android platform and the increasing number of mobile devices. Malware attacks on mobile devices can lead to severe damage such as loss of data, financial loss, and identity theft. In this project, we will develop a system that can detect and classify malware on Android devices in real-time.

The novelty of our project lies in the following aspects:

1. **Real-time detection and classification**: Our system will detect and classify malware in real-time on Android devices. Real-time detection and classification are crucial for preventing malware attacks and ensuring the security of mobile devices. Our system will continuously monitor the Android device for any malware activity and alert the user if any malware is detected.
2. **Dynamic analysis of malware**: Our system will perform dynamic analysis of malware to detect any malicious activity on the Android device. Dynamic analysis involves executing the malware in a controlled environment and monitoring its behavior. This approach enables the detection of malware that may be difficult to detect using static analysis techniques.
3. **Multi-level classification:** Our system will classify malware at multiple levels to enable more accurate detection and classification. We will use a hierarchical classification approach to classify malware based on their behavior, features, and families. This approach will enable us to detect and classify both known and unknown malware.
4. **Scalable architecture:** Our system will be designed to be scalable and capable of handling a large number of Android devices. We will use cloud-based infrastructure to enable efficient and reliable detection and classification of malware on Android devices.

# Methodology Diagram



# Implementation

# Literature Survey

## Verma, Vinita, Sunil K. Muttoo, and V. B. Singh. "Multiclass malware classification via first-and second-order texture statistics." *Computers & Security* 97 (2020): 101895.

Verma et al. proposed a novel method for malware classification in their research, which involved converting the malware code into images. They then utilized first-order and second-order statistical equations to extract features by analyzing the distribution and correlation of various points on a one-channel image. To evaluate the effectiveness of their approach, they utilized the Malimg dataset along with a dataset that they gathered themselves. The results of their study demonstrated a precision rate of 98.04%, while recall and F1 measures were both 98.06% and 98.05%, respectively. However, the authors did not include any obfuscated malware in their analysis. One drawback of their approach was that the features were manually extracted, which resulted in the loss of a significant number of discriminative features. To address this limitation, the authors extended their work on malware detection to focus on reducing false-negative errors. They incorporated new statistical features to enhance the accuracy of their approach, testing it on a dataset of 10,000 samples that did not include any obfuscated data. Despite these improvements, the limitations of manual feature extraction remained, highlighting the need for further research in this area. Overall, Verma et al.'s approach provides a promising foundation for malware classification and detection, but there is still much to be done to fully exploit the potential of this method in the face of increasingly sophisticated malware.

## Marastoni, Niccolò, Roberto Giacobazzi, and Mila Dalla Preda. "Data augmentation and transfer learning to classify malware images in a deep learning context." *Journal of Computer Virology and Hacking Techniques* 17 (2021): 279-297.

The impact of obfuscation on the accuracy of machine learning methods for static and dynamic malware detection was investigated in a study are outlined. While a number of well-known obfuscation methods have been developed, the researchers found that obfuscation had a more significant effect on the accuracy of static methods as opposed to dynamic methods. In this study, Convolutional Neural Networks (CNNs) were used to classify malware codes that were converted into three-channel images. The researchers generated new images using various types of noise as augmentation and achieved 96% accuracy on a dataset obtained from GitHub. The malware was classified into different categories, along with a class of benign data. However, the method did not explicitly use obfuscated malware, which poses a more challenging problem for detection.

Another study of the same research paper addressed the problem of data deficiency by utilizing obfuscation. The authors sought to increase the amount of data by mapping a generated code to an image and applying augmentation to the resulting image. They then used transfer learning on a pre-trained network for malware classification. The researchers achieved 93.8% accuracy on the Microsoft2015 Dataset and 98.5% accuracy on the Malimg Dataset. The approach demonstrated the potential of using obfuscation and image-based classification to address data deficiency in malware detection. However, the effectiveness of the method in detecting obfuscated malware remains to be tested, and further research is needed to develop more robust solutions for detecting these types of malicious code. Overall, the use of obfuscation in conjunction with machine learning methods shows promise in enhancing the accuracy of malware detection and classification, but additional studies are required to fully explore its potential.

## Royal, Paul, Mitch Halpin, David Dagon, Robert Edmonds, and Wenke Lee. "Polyunpack: Automating the hidden-code extraction of unpack-executing malware." In *2006 22nd Annual Computer Security Applications Conference (ACSAC'06)*, pp. 289-300. IEEE, 2016.

The detection of obfuscated malware has been a topic of significant research interest in recent years, as malicious actors continue to develop new techniques to evade detection. One such study, outlined, focused specifically on Android applications and demonstrated that some malicious Android apps are re-released under new names and with obfuscated certificate information. To combat this, the researchers employed a novel approach that leveraged the correlation between package names and certificate owner names to classify applications as malicious or benign.

In their method, the researchers used stacking techniques to combine Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) to analyze information obtained from the package name and certificate owner. The resulting classification system was able to accurately detect obfuscated malware in Android apps with high precision and recall. However, it should be noted that the features used in this study were specific to the Android operating system and cannot be extended to other operating systems such as Linux or Windows.

While the study focused on the detection of obfuscated Android malware, the approach could be adapted to other types of malicious code and could potentially be extended to other operating systems with the development of new features. As the threat landscape continues to evolve, it is clear that the development of effective methods for detecting and classifying obfuscated malware will be crucial in ensuring the security of computer systems and networks.

## Roundy, Kevin A., and Barton P. Miller. "Hybrid analysis and control of malware." In Recent Advances in Intrusion Detection: 13th International Symposium, RAID 2010, Ottawa, Ontario, Canada, September 15-17, 2010. Proceedings 13, pp. 317-338. Springer Berlin Heidelberg, 2020.

As the threat of obfuscated malware continues to grow, researchers have explored various methods for detection. Miller et al .He developed a technique for detecting obfuscated malware on Android devices by identifying the presence of certain API calls, permissions, and opcode instructions. They used a Discriminative Adversarial Network (DAN) to train their system to classify obfuscated malware from benign applications. However, the main limitation of this approach is that it only trains the system to detect the specific types of obfuscation that were used in the training set, and cannot predict novel forms of obfuscation.

In recent years, obfuscated malware has become an increasingly prevalent type of malware, as attackers continue to find new ways to hide their malicious code. These malwares are often derived from other malwares, and can take many different forms. One of the most challenging aspects of detecting obfuscated malware is that it can change from one machine to another, making it difficult to identify with traditional signature-based methods.

 Generative models have emerged as a promising approach for obfuscated malware detection, by estimating the probability distribution of occurrence of any point using a trained network. However, due to the creative nature of obfuscation, generative networks may not be able to consider all possible forms of obfuscation, which limits their ability to fully achieve the goal of obfuscated malware detection.

## Yan, Ping, and Zheng Yan. "A survey on dynamic mobile malware detection." *Software Quality Journal* 26, no. 3 (2018): 891-919.

The susceptibility of deep neural networks to adversarial attacks is a major concern in many application domains, including malware detection. Yan et al**.  highlighted the fact that even minor modifications to input images can lead to a decrease in the accuracy of deep neural networks**. To address this issue, **generative networks have been proposed as a solution that can enhance the robustness of the models against such attacks**. However, the vulnerability of deep neural networks to adversarial changes has been highlighted, where it has been shown that malware can evade detection using this technique.

To improve the accuracy of malware detection, several methods based on machine learning techniques have been proposed in recent years. **MalNet**, introduced in, is an ensemble method that employs convolutional neural networks (**CNNs**) and long short-term memory (**LSTM**) networks to classify malware. **MalNet converts binary codes and opcode instructions to images** and then trains the models on the resulting data. The proposed method achieved high accuracy of **99.36% on the Microsoft2015 dataset**]. Similarly, a hybrid model composed of ResNet and GoogleNet has been proposed for malware detection on current, where the byte codes are directly converted to images for training. The model achieved an accuracy of 88%, demonstrating the effectiveness of the image-based approach for malware detection.

## Abusitta, A., Li, M. Q., & Fung, B. C. (2021). Malware classification and composition analysis: A survey of recent developments. *Journal of Information Security and Applications*, *59*, 102828.

Information security is seriously threatened by malware, a class of software that is more common and sophisticated that is intended to damage computer systems or networks. Information security now requires the detection and classification of malware. **Malware classification is the procedure of classifying malware into several categories according to its actions and traits**. Malware composition analysis, on the other hand, **examines the features and actions of a malware sample**. The purpose is to learn more about the motives of the attacker and the infection.

For malware categorization and composition analysis, many methods have been put out recently, ranging from signature-based approaches to methods based on machine learning. Although these methods have progressed, malware creators continue to create new evasion strategies to prevent detection. In order to increase malware detection and analysis, it is crucial to stay current with this field of study.

This study tries to categorise and contrast the key conclusions from malware composition and classification studies. Also, we go over malware evasion strategies and feature extraction approaches. Also, we describe each reviewed research based on the characteristics and algorithms utilised, highlighting their advantages and disadvantages. Lastly, we discuss problems, roadblocks, and potential future research directions in malware analysis.

Classification of Malware: Malware classification is the process of classifying malware samples according to their behaviour and traits into various kinds. The most popular techniques for classifying malware include signature-based detection, heuristics-based detection, and machine learning-based detection.

A malware sample is compared to a database of known malware signatures in signature-based detection. The sample is labelled as malware if it fits one of the signatures. Although quick and effective, this method has several drawbacks, such as the inability to identify brand-new virus.

Heuristics-based detection includes examining a program's behaviour to determine whether it is malicious. Compared to signature-based detection, this technique is more sophisticated, but it is also more complicated and prone to false positives.

With machine learning-based detection, massive volumes of data are analysed using algorithms to find patterns and categorise malware samples. This technique can identify new and unidentified viruses, but it needs a lot of data to adequately train the algorithm.

Malware Analysis by Composition: Analysis of the functionalities and behaviours of a malware sample is known as malware composition analysis. The purpose is to learn more about the motives of the attacker and the infection. The most popular techniques for analysing the composition of malware are dynamic analysis and static analysis.

Running a malware sample in a controlled environment and observing its activity is known as dynamic analysis. Although it might not cover all aspect of the malware's operation, this method can give thorough information about the functions and behaviours of the infection.

Static analysis involves reviewing a malware sample's source code without actually running it. Although it might not cover all aspect of the malware's behaviour, this method can give information about the features and actions of the malware.

Malware Avoidance Methods: Obfuscation, polymorphism, and metamorphism are only a few of the methods used by malware developers to avoid detection. Obfuscation includes burying a malware sample's source code to make it more difficult to find. Polymorphism is the process of altering a malware sample's code to produce new, more evasive forms. A malware sample can metamorphosize by altering its structure and behaviour to produce new, harder-to-detect variations.

Techniques for Feature Extraction: The technique of choosing and extracting pertinent features from a malware sample for use in composition and classification analysis is known as feature extraction. API calls, system calls, and file system activities are examples of frequently utilised functionalities. The approaches for feature selection and extraction include deep learning-based feature extraction, automatic feature selection, and manual feature selection.

Conclusion: Given the intricacy of malware creation and current advancements in communication and computer infrastructure, malware detection and categorization are becoming more difficult. The current malware classification methods allow reverse engineers to comprehend

## Awan, M. J., Masood, O. A., Mohammed, M. A., Yasin, A., Zain, A. M., Damaševičius, R., & Abdulkareem, K. H. (2021). Image-based malware classification using VGG19 network and spatial convolutional attention. Electronics, 10(19), 2444.

Malware proliferation has raised serious concerns in recent years, making it essential to find and eliminate these dangerous agents. **This study suggests using a deep learning-based SACNN (Spatial Attention and Convolutional Neural Network) to classify 25 well-known malware types based on images.** On the **Malimg** benchmark dataset, the proposed model was assessed using a number of evaluation criteria, including precision, recall, specificity, and F1 score. High scores on all of the evaluation indicators were found for the suggested model with class balancing, which demonstrated its excellent performance. It is also interesting that the suggested model can perform well even when the benign class is balanced.

The suggested model takes a novel method to classifying image-based malware by utilising spatial attention and convolutional neural network. The findings of this study show that even in the face of cutting-edge, automated malware production techniques, the suggested approach is highly effective at spotting and classifying malware images. The paper contends that the proposed model can be less complex than existing options while yet providing equivalent or superior performance.

Overall, the results of this study have significant implications for the cybersecurity sector and show how deep learning techniques might help with problems brought on by the growing threat of malware. The suggested model can be applied to other malware detection tasks in future research, and its effectiveness can be assessed using larger and more varied datasets.

## Agarap, A. F. (2017). Towards building an intelligent anti-malware system: a deep learning approach using support vector machine (SVM) for malware classification. arXiv preprint arXiv:1801.00318.

Malware mitigation has been a long-standing goal of the information security community. The creation of an anti-malware system that can combat unidentified malware is a common endeavour that could be advantageous to many industries.

The authors hope to develop an intelligent deep learning (DL)-based anti-malware system. By applying such models, it would be possible to mathematically generalise the detection of recently released malware. Identifying the connection between a specific malware x and the malware family that it belongs to, f:xy.

The Malimg dataset (Nataraj et al., 2011) of malware pictures extracted from malware binaries was employed by the authors to achieve this accomplishment. The authors then trained the **CNN-SVM (Tang, 2013), GRU-SVM (Agarap, 2017), and MLP-SVM DL models 1 to categorise each malware family. The GRU-SVM stands out among the DL models with a prediction accuracy of 84.92%, according to empirical data.**

This makes sense given that, of the models shown, the aforementioned model had the comparably most complex architecture design. The next step in creating an intelligent anti-malware system is to investigate an even more ideal DL-SVM model.

## Raff, E., & Nicholas, C. (2020). A survey of machine learning methods and challenges for windows malware classification. *arXiv preprint arXiv:2006.09271*.

**------**

Malware classification is a difficult problem, to which machine learning methods have been applied for decades. Yet progress has often been slow, in part due to a number of unique difficulties with the task that occur through all stages of the developing a machine learning system: data collection, labeling, feature creation and selection, model selection, and evaluation. In this survey we will review a number of the current methods and challenges related to malware classification, including data collection, feature extraction, and model construction, and evaluation. Our discussion will include thoughts on the constraints that must be considered for machine learning based solutions in this domain, and yet to be tackled problems for which machine learning could also provide a solution. This survey aims to be useful both to cybersecurity practitioners who wish to learn more about how machine learning can be applied to the malware problem, and to give data scientists the necessary background into the challenges in this uniquely complicated space.

## Explainable Malware Detection System Using Transformers-Based Transfer Learning and Multi-Model Visual Representation

[Sensors | Free Full-Text | Explainable Malware Detection System Using Transformers-Based Transfer Learning and Multi-Model Visual Representation (mdpi.com)](https://www.mdpi.com/1424-8220/22/18/6766)

Due to its accessibility and adaptability, Android has become the dominant mobile ecosystem due to its broad adoption. That has, however, also turned it become the main target of malicious programmes. Hence, the necessity for a reliable malware detection system is critical. This work suggests a transfer learning and malware visual features-based explainable malware detection system. The suggested method increases malware detection precision by making use of both textual and visual data.

A pre-trained Bidirectional Encoder Representations from Transformers (BERT) model is used to extract trained textual features. The malware-to-image conversion mechanism is additionally presented to render network byte streams in a visual manner. The FAST (Features from Accelerated Segment Test) extractor and BRIEF (Binary Robust Independent Elementary Features) descriptor are used to effectively extract and mark key features. The proposed malware detection system's effectiveness is ensured by the combination of these techniques.

The suggested approach is advantageous since it effectively detects malware using both textual and visual features. Additionally, the trained and texture characteristics are combined and balanced using the Synthetic Minority Over-Sampling (SMOTE) technique. The ensemble model is employed for effective malware classification and detection, while the CNN network is used to mine deep features. Two open datasets, CICMalDroid 2020 and CIC-InvesAndMal2019, are used to conduct a thorough analysis of the suggested technique. In order to clarify and validate the suggested methodology, an interpretable artificial intelligence (AI) experiment is also carried out.

Despite its benefits, the suggested strategy has several drawbacks. The malware detection system's overall performance may be impacted by the inaccuracy of the conversion of network byte streams into visual representations. Also, the performance of the suggested strategy can be impacted by the quantity and calibre of training data. The final drawback of the suggested method is that it can need a lot of processing power and time for training and detection.

# References