IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 5, May 2023

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.423

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2023 ||

| DOI:10.15680/IJIRSET.2023.1205238 |

To Study Different Algorithms of Machine Learning to Detect Mobile Botnets

Yash Birdavade, RamTodkar, AashishWalke, Sambhaji Shinde Dr.Neelamkumar

Department of Computer Engineering, Shree Ramchandra College of Engineering, Wagholi, Pune, India

ABSTRACT - Botnets pose a significant threat to Internet security, and their constant sophistication and resilience have led to a new trend of shifting from desktop to mobile environments. Detecting mobile botnets is crucial to mitigate their impact. Among various strategies, identifying patterns in their anomalous behavior has shown promising and generalized results. In this research, we provide a host-based and anomaly-based method for mobile botnet detection. Our suggested approach makes use of machine learning methods to spot unusual patterns in statistical characteristics taken from system calls. We evaluated the performance of our high accuracy The widespread adoption of Android as the most prevalent mobile operating system has made it a prime target for malware authors. The open-source nature of Android, coupled with its unrestricted app installation capabilities, has contributed to the growth of malware specifically designed to infect Android devices and transform them into malicious bots. These Android bots can be enlisted in larger botnets, enabling various types of attacks such as DDoS attacks, spam generation, phishing, click fraud, and data theftThis research provides a deep learning method that uses Support Vector Machine (SVM) for Android botnet detection to counteract this expanding danger. Our approach utilizes 342 static features extracted through automated reverse engineering of apps to classify new or previously unseen apps as either 'botnet' or 'normal'. These features are directly fed into the SVM model without requiring additional pre-processing or featureselection.

KEYWORDS: Mobile Botnet, Support Vector Machine, Machine learning, neural networks, Image processing.

I. INTRODUCTION

Theprevalenceof Androidasthemostwidespreadmobileoperatingsystemhasledtoanincreaseinmalware targetingthis platform.TheopennatureofAndroid anditsability to installappsfromvarioussourceshavemadeitanattractivetargetfor malware authors. Malicious apps capable of turning Android devices into bots have been discovered in the wild. These Android bots can be organized into botnets that facilitate a range of malicious activities. The term 'mobile botnet' refers to compromised smartphones and other mobile devices remotely controlled by malicious actors through Command and Control (CC) channels. The sophistication and evasive techniques employed by malicious botnet apps necessitate more effective detection approaches. In this paper, we propose a deep learning approach utilizing Support Vector Machine (SVM) for Android botnet detection. By automating reverse engineering, we extract 342 static features from apps to create feature vectors that directly feed into the SVM model, eliminating the need for additional pre-processing or feature selection steps The Android operating system's wide adoption has made it an attractive target for malware authors, resulting in a significant increase in Android malware. The openness of the Android platform and the freedom to install apps from any source contribute to its vulnerability. Malware families capable of turning Android devices into malicious bots have emerged, posing serious threats. These mobile botnets can be utilized for various malicious activities, highlighting the need for effective detection approaches. In this paper, we present a deep learning approach that utilizes Support Vector Machine (SVM) for Android botnet detection. Our approach leverages automated reverse engineering to extract 342 static features fromappsandfeedsthemdirectlyintotheSVMmodelwithoutadditionalpreprocessingorfeatureselection.[1]

II. DOMAIN

Code that can be used to deliver messages, attack other networks, or launch DDoS attacks is frequently found in mobile botnets. Researchers have also found that specific versions of the Android operating system are required for mobile botnets to infect a smartphone.

A subfield of artificial intelligence (AI) and computer science called machine learning focuses on using data and algorithms to simulate how humans learn, gradually increasing the accuracy of the system.With regards to artificial intelligence, IBM has a lengthy history. By studying the game of checkers, one of its own, Arthur Samuel, is credited with coining the term "machine learning" (PDF, 481 KB) (link outside of IBM). On an IBM 7094 computer, Robert Nealey, a self-described checkers master, played the game in

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2023 ||

| DOI:10.15680/IJIRSET.2023.1205238 |

1. Supervised Machine Learning

Supervised machine learning is based on supervision, as its name suggests. In the supervised learning technique, this means that we train the machines using the "labelled" dataset, and then the machine predicts the output based on the training. Which inputs have already been mapped to which outputs are shown here by the data that has been annotated. To put it more precisely, we may say that we ask the machine to anticipate the results using test datasets after training it with input and related output.

2. Unsupervised Machine Learning

Because it does not require supervision, unsupervised learning differs from the supervised learning approach. This is the definition of unsupervised machine learning, when the system makes output predictions without any human oversight after being trained on an unlabeled dataset.

3. Semi-Supervised Learning

Semi-supervised learning is a type of machine learning technique that sits in between supervised and unsupervised learning. It uses a combination of labelled and unlabeled datasets during the training phase and stands in the middle of supervised learning (with labelled training data) and unsupervised learning (without labelled training data) techniques.

4. Reinforcement Learning

By striking and trailing, acting, learning from experiences, and improving performance, an AI agent (a software component) automatically explores its surroundings through reinforcement learning. Reinforcement learning operates on a feedback-based method. The objective of a reinforcement learning agent is to maximize the rewards since the agent is rewarded for each good activity and penalized for each negative one.

III.TYPES OF ALGORITHMS FOR DETECTION OFMOBILE BOTNETS

Naive Bayes

Naïve Bayes classification is indeed a straightforward and powerful algorithm commonly used for classification tasks. It is based on applying Bayes' theorem under the premise that the features are highly independent of one another.This assumption implies that the presence or absence of a particular feature is independent of the presence or absence of other features, given the class variable. Naïve Bayes classification is particularly effective in analyzing textual data, making it popular in Natural Language Processing (NLP) applications. It works well for tasks such as sentiment analysis, spam filtering, document classification, and text categorization. The algorithm is often referred to as Naïve Bayes, simple Bayes, or independent Bayes because of its reliance on Bayes' theorem and the assumption of feature independence. All these terms describe the same concept of applying Bayes' theorem in the decision rule of the classifier. The Naïve Bayes classifier utilizes Bayes' theorem to calculate the probability of a certain class given a set of features. It calculates the conditional probabilities of the features given each class and combines them with the prior probabilities of the classes to make predictions. By comparing the probabilities for different classes, the classifier assigns the most likely class to an input sample Naïve Bayes models are computationally efficient, require less training data compared to other algorithms, and can handle high-dimensional feature spaces effectively. However, their strong independence assumption can limit their performance in situations where feature dependencies significantly impact the classification outcome. Overall, Naïve Bayes classification is a valuable tool in machine learning, particularly for text analysis, due to its simplicity, efficiency, and applicability to a wide range of NLP tasks

Advantages of Naive Bayes

This algorithm is efficient and can greatly reduce processing time.

Naive Bayes works well for multi-class prediction issues.

It can outperform other models and needs a lot less training data if its assumption about the independence of characteristics is correct.

For categorical input variables as opposed to numerical variables, Naive Bayes is more appropriate.

Disadvantages of Naïve Bayes

Naive Bayes makes the uncommon but unfounded assumption that all predictors (or features) are independent. This restricts the algorithm's usability in practical usage cases.

This approach encounters the "zero-frequency problem," where it gives a categorical variable with zero probability if its category was not present in the training dataset but was present in the test data set. To solve this problem, it would be

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2023 ||

| DOI:10.15680/IJIRSET.2023.1205238 |

better if you employed a smoothing method.

You shouldn't take its probability outputs too seriously because its estimations may occasionally be incorrect.

Logistic Model Trees

Both for the prediction of nominal classes and numerical values, tree induction methods and linear models are prominent approaches for supervised learning tasks. The two approaches have been combined to create "model trees," or trees with linear regression functions at their leaves, which are useful for forecasting numerical numbers. In this study, we describe a logistic regression-based approach that extends this concept to classification issues as opposed to linear regression. We demonstrate how this method can be used to build the logistic regression models at the leaves by incrementally refining those constructed at higher levels in the tree. We use a stagewise fitting process to build the logistic regression models that can select relevant attributes in the data in a natural way. By contrasting theUtilizing 36 benchmark UCI datasets, we compare the performance of our method to a number of different cutting-edge learning techniques and demonstrate that it generates accurate and small classifiers. The algorithm that combines the concept of model trees with logistic regression for classification problems. Model trees are decision trees where each leaf node contains a regression model, and in this case, logistic regression models are used. The algorithm follows a stagewise fitting process, which means that it iteratively refines the logistic regression models at different levels of the tree. The process involves selecting relevant attributes in the data to build the logistic regression models in a natural way.The performance of the proposed algorithm is compared to several other state-of-the-art learning schemes on 36 benchmark datasets from the UCI (University of California, Irvine) repository. The evaluation demonstrates that the algorithm produces accurate and compact classifiers.In summary, the paper presents an adaptation of model trees for classification problems by using logistic regression models at the leaves. The algorithm utilizes a stagewise fitting process to construct these models, and experimental results indicate that it achieves accurate and compact classifiers compared to other learning schemes on benchmark datasets.[3]

Advantages of Logistic Model Tree

The training of logistic regression is very effective and easier to implement and analyse.

It does not make any assumptions about how classes are distributed in feature space.

Multinomial regression and a natural probabilistic perspective of class predictions can be simply added to it.

It only works for forecasting discrete functions. As a result, the discrete number set is confined to the dependent variable of the logistic regression.

Disadvantages of Logistic Model Tree

Logistic regression should not be employed if there are less data than features because this could result in overfitting.

It constructs linear boundaries.

The major limitation of Logistic Regression is the assumption of linearity between the dependent variable and the independent variables.

Only discrete functions can be predicted with it. The discrete number set is thereby confined to the dependent variable in a logistic regression.

Support Vector Machine (SVM)

Classification and regression issues are resolved using Support Vector Machine, or SVM, one of the most used supervised learning techniquesHowever, it is primarily utilized in Machine Learning Classification issues.. Both classification and regression are performed using supervised machine learning techniques known as Support Vector Machines (SVM). The most relevant phrase is categorization, even though we often talk about regression problems. The SVM method aims to find a hyperplane in an N-dimensional space that clearly classifies the data points. The size of the hyperplane depends on the quantity of features. When there are only two input features, the hyperplane effectively looks like a line. The hyperplane turns into a 2-D plane if there are three input features.A hyperplane is the name given to this optimal decision boundary. [4]

Advantages of SVM

Effective in high-dimensional cases.

Due to the decision function's use of support vectors, a subset of training points, its memory usage is effective.

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2023 ||

| DOI:10.15680/IJIRSET.2023.1205238 |

Different kernel functions as well as custom kernels can be provided for the decision functions.

Disadvantages of SVM

Unsuitable to Large Datasets.

Large training time.

More features, more complexities.

Bad performance on high noise.

Does not determine Local optima.

VI. CONCLUSION

In this paper we study different types of machine learning algorithms for detection of Mobile Botnets. It is concluded that naive Bayes algorithm works quickly and can save a lot of time but all features of this algorithm are independent which rarely happening in real life. Due to this applicability of this algorithm is limited. Further we study logistic tree model algorithm which is easier to implement and very efficient to training purpose. but it has used to predict discrete functions only in the last we study support vector machine algorithm which is effective in high dimensional cases and it is very efficient as compare to other algorithms used for Mobile Botnet Detection.

REFERENCES

1. CinziaBernardeschi,Exploiting Model Checking for Mobile Botnet Detection,23rd International Conference on Knowledge-Based and Intelligent Information & Engineering
2. Hong Chen,"Improved naive Bayes classification algorithm for traffic risk management",EURASIP Journal on Advances in Signal Processing volume 2021, Article number: 30 (2021),
3. Published: 22 June 2021
4. Darren Dancey,Logistic Model Tree Extraction From Artificial Neural Networks,IEEE Transactions on Systems, Man, and Cybernetics, Part B:2012.
5. Falguni Ghatkar, Sakshi kharche, Priyanka Doifode, Jagruti Khairnar, Prof. Neelam Kumar," To Study Different Types of Supervised Learning Algorithm" May 2023, International Journal of Advanced Research in Science, Communication and Technology (IJARSCT), Volume 3, Issue 8, May 2023,PP-25-32
6. Systems,Procedia Computer Science 159 (2019) 963–972,10.1016/j.procs.2019.09.263
7. S. Y. Yerima and S. Khan "Longitudinal PerfomanceAnlaysis of Machine Learning based AndroidMalware Detectors" 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), IEEE.
8. Kadir, A.F.A., Stakhanova, N., Ghorbani, A.A., 2015. Android botnets: What urls are telling us, in: International Conference on Network and System Secu- rity, Springer. pp.78–91.
9. ISCX Android botnet dataset. Available from https://www.unb.ca/cic/ datasets/android- botnet.html. [Accessed 03/03/2020]
10. M. Eslahi, R. Salleh, and N. B. Anuar, "Bots and botnets: An overview of characteristics, detection and challenges," in Proceedings of the IEEE Interna- tional Conference on Control System, Computing and Engineering (ICCSCE), 2012, pp.349-354.
11. J. Dae-il, C. Kang-yu, K. Minsoo, J. Hyun-chul, and N. Bong-Nam, "Evasion technique and detection of malicious botnet," in Proceedings of the Internet Technology and Secured Transactions (ICITST), 2010 International Confer- ence for, 2010, pp.1-5.
12. Z. Yajin and J. Xuxian, "Dissecting Android Malware: Characterization and Evolution," in Proceedings of the Symposium on Security and Privacy (SP), 2012, pp.95-109.
13. M. Eslahi, M. Rohmad, H. Nilsaz, M. Var Naseri, N. Tahir, and H.Hashim, "Periodicity classification of HTTP traffic to detect HTTP Botnets," in Pro- ceedings of the Computer Applications Industrial Electronics (ISCAIE),2015

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor: 8.423

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details