iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 **Issue:** XII **Month of publication:** December 2019

DOI: http://doi.org/10.22214/ijraset.2019.12024

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue XII, Dec 2019- Available at www.ijraset.com

Malware Detection & Prevention in Android Mobile by using Significant Permission Identification & Machine Learning

Ms. Kirti Reddy[1], Mr. Danesh Bastani[2], Ms. Charul Joshi[3], Prof. Rinku Badgujar[4], Mr. Abhijeet Singh[5]
[1, 2, 3, 5]*Computer Engineering, BSIOTR, Pune, India*
[4]*Assistant Professor, BSIOTR, Pune, India*

Abstract: ***Malware is today one of the biggest security threat to internet. People today use terms such as malware, spyware, or ransomware much more than the term "virus. "where Malware can steal your information, make your device send SMS messages to premium rate text services, or install adware that forces you to view web pages or Download apps.***
We need a robust malware detection solution to counter this serious malware project that can effectively and efficiently recognis e malware apps. We will present SIGPID in our proposed system to perform data extraction and as SIGPID design is effective in malware detection. Our method defines the substantial work permission needed by an application. And differentiate between essential and non-essential permissions and detect and remove the malware on such a basis.
Keywords: Machine learning, classification, malware, android, SIGPID, SVM

I. INTRODUCTION

The problem of malware infection was so severe that a recent report reveals that 97% of all Android devices target mobile malware. More than 3.25 million new malicious Android applications were discovered in 2016 alone. It translates roughly every 10 seconds into an introduction of a new malicious Android app. Such malicious apps are generated in the form of Trojans, worms, hacks, and viruses to perform various types of attacks. Many infamous malicious apps have over 50 versions, making finding them all extremely challenging. Researchers and researchers have used various approaches to build Android malware detection tools to tackle these growing security concerns.

RISKRANKER, for example, uses static analysis to identify malicious activities in Android apps. Fixed approaches to research, however, generally assume that more actions are probable than would actually be possible, which can lead to a large number of false positives. In order to improve the precision of the results, researchers have suggested different methods of dynamic analysis to capture the context of real-time execution. For example, TAINTDROID uses tainting analysis to simultaneously track several sensitive data sources. Nonetheless, approaches to dynamic analysis generally require adequate input suites to exercise execution paths adequately. Since we can use cloud computing to meet these requirements, the issues about privacy then become a potential problem. Petra et al. demonstrate that there is a wide range of anti-analysis techniques that advanced malware can use to successfully avoid malware detection based on dynamic analysis. More recent efforts have been made to examine the behavioral data of users in both the Android system and other data processing systems online. The required privileges from the applications were used to implement the least privilege, which to some degree shows the functionalities of the apps as well as the act ions of runtime.

As a result, researchers use techniques of machine learning and data mining to detect Android malware based on use of permission.

For example, DREBIN[6] uses multi-view technology to combine static analysis with supervised learning to detect malware accurately. SIGPID [7] strengthens DREBIN[6] through the use of many more learning and detection functions. Droidclassifier [8] uses information about traffic flow and unsupervised learning to detect malware and identify that malicious app's kin. We present SIGPID in this paper, an approach which extracts substantial permissions from apps and uses the extracted data to detect malware effectively using supervised learning algorithms. SIGPID's development aim is to efficiently and reliably detect malware. As stated earlier, there is an unprecedented increase in the number of newly introduced malware. It would encourage analysts to be more active in detecting and analyzing malware as such. Our method analyzes permissions and then defines only those that make a difference between malicious and benign applications. Specifically, they suggest a multi-level data pruning method involving negative rate permission ranking, association-based permission mining and support-based permission ranking to strategically extract

substantial permissions. Then, classification algorithms based on machine learning are used to identify various types of malware and benign applications.

Our empirical analysis results show that SIGPID can significantly reduce the number of permissions we need to evaluate to just 22 out of 135 (84 percent reduction), while retaining more than 90 percent accuracy in malware detection and Fmeasure when using Vector Machine Support (SVM) as a classifier. We also find that the number of important permissions listed by our approach is smaller than Google has identified the number of "dangerous" permissions. In contrast, only eight permissions appear on our list together with their list. This is because, as a data-driven method, SIGPID dynamically defines significant permissions based on the applications ' actual use rather than the static concept of dangerous permissions based on their intended services.

This fundamental difference helps us to find more malware than the method using the dangerous list alone. We also check SIGPID with 67 widely used supervised algorithms to show the generality of this method and consider that it retains very high precision with all these algorithms.

In addition, we contrast our approach's accuracy and runtime quality with two state-of-the-art methods, DREBIN, Permission-Induced Risk Malware Detection, and current virus scanners.

Also, with considerably less overhead, we find that our approach can detect more samples of malware than the other approaches.

In summary, our paper makes the following contributions:

1) We are creating SIGPID, an approach that recognizes an important subset of permissions (significant permissions) that can be used to detect Android malware effectively. By using our methodology, the number of permissions to be evaluated will be reduced when 84%

2) Using only a fifth of the total number of Android permissions, we determine the efficacy of our strategy. We consider that in precision, recall, reliability, and F-measure, SIGPID can achieve more than 90 percent. Those findings favorably contrast with those obtained through a strategy that uses all 135 permissions as well as the hazardous list of permissions. Compared to other state-of - the-art approaches to malware detection, we find that SIGPID is more effective by detecting 93.62 percent of malicious applications in the data set and 91.4 percent of unknown malware.

3) We use SIGPID with 67 widely used supervised learning algorithms and a much larger dataset (5,494 malicious and 310,926 benign apps) to demonstrate that the technique can generally work with a wide range of supervised learning algorithms. We consider that 55 out of 67 algorithms can reach F-measurement of at least 85%, while the average runtime can be reduced by 85.6% compared to the baseline method.

II. LITERATURE SURVEY

A. IDC, "Smartphone OS market share, 2017 q1." [Online].Available:https://www.idc.com/promo/smartphon market-share/OS:

Smart-phones which are used for operating small hand-held devices are increasing rapidly And became the functional part of the humans. Transparency of these current era has leads to domain market and application by enabling high affiliation with online assistance like digital banking. Competition under creators are in the terms of speed, processors, storage, features capacity, camera etc. but the main factor for the profitable market and popularity is smart-phone's operating system. This research meets with pros and cons of operating system and analysis the individual progression over other by proposing board enhancing focuses for making them acceptable as well as protected.

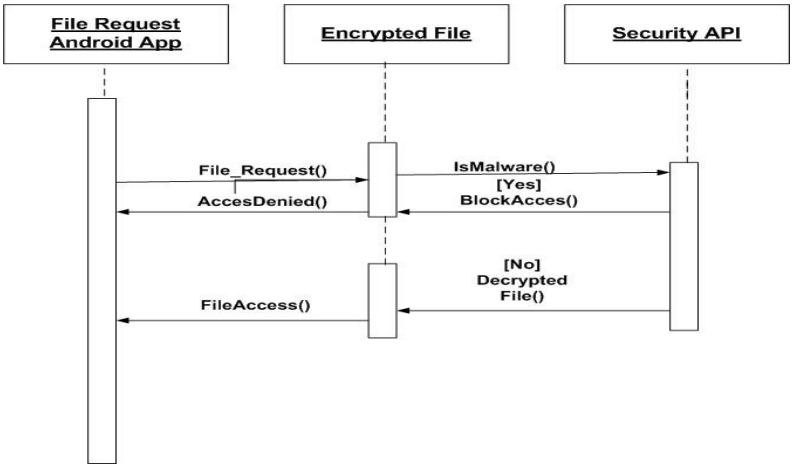


Fig2.1. Sequence Diagram of the System

B. M. Grace, Y. Zhou, Q. Zhang, S. Zou, and X. Jiang, "Riskranker: scalable and accurate zero-day android malware detection," in Proceedings of the 10th international conference on Mobile systems, applications and services. ACM, 2012, pp. 281–294.

Recent explosive growth has occurred in the sales of smart-phones. The success also promotes the deployment of malware authors to various mobile markets using malicious programs (or applications). Such malicious apps hide in a large number of other benign apps which are hard to detect. Existing mobile antivirus software is not adequate in its proactive nature to use known malware samples to collect signs. In this study, author had proposed a proactive scheme to spot zero-day malware for Android. Researchers are motivated to assess potential security threat posed by these unreliable programs, using malware samples and their signatures. The authors developed automatic system called Riskranker for scalable analysis of dangerous activities (e.g., root exploit release or the background sending of SMS). The output is then used to create a priority list of growing applications that need further analysis. Applied to 118, 318 apps obtained from different Android markets in September and October 2011, researcher system takes less than four days to process them all and Successfully reports 3, 281 risky apps. Among these reported apps, 718 malware samples (in 29 families) were successfully uncovered and 322 are zero-day (in 11 families). These findings show the effectiveness and scalability of Riskranker for Android policing in every way.

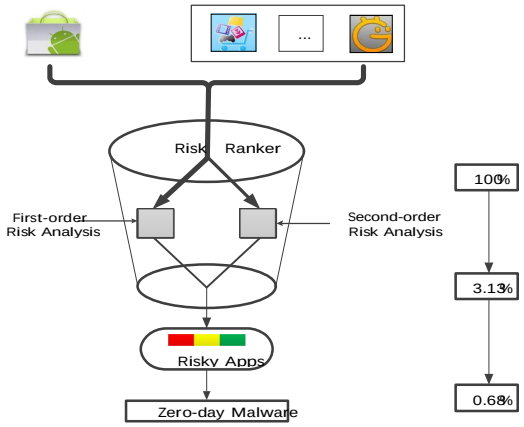


Fig2.2. The Riskranker architecture

C. S.Wang, Q.Yan, Z.Chen, B.Yang, C.Zhao, and M.Conti,"Textdroid: Semantics-based detection of mobile malware using network flows."

Android is the most common 80 % market share mobile operating system, But also the most malware-targeted platform, as a result. To counter the rising number of wild malicious Android apps, Malware researchers typically use analysis tools to automate the extraction of features of a device. While the research community has addressed the significance of such instruments, their analytical capacity and availability remain limited by the resulting prototypes. ANDRUBIS, a fully automated, open-to-public and detailed evaluation program for Android apps, was proposed by author in this study. At Dalvik VM and machine stage, ANDRUBIS incorporates static analysis and a dynamic analysis, as well as increase codes coverage via stimulation techniques. We received over 1,000,000 Android apps with ANDRUBIS including 40% malware platform. This dataset enables us to speak about malware patterns found in apps from 2010, and to gain insight into how ANDRUBIS has been used as a publicly available tool for the last 2 years.

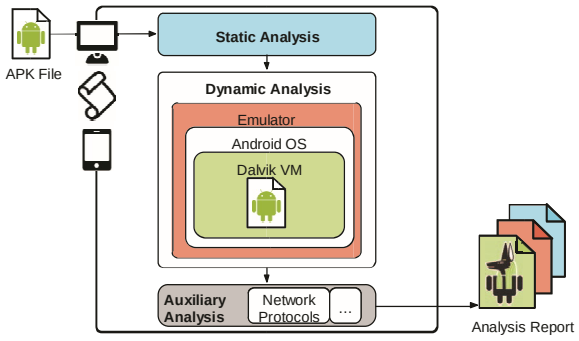


Fig2.3. Overview architecture of ANDRUBIS

D. J. Z. L. H. P. S. Y. Lichao Sun, Xiaokai Wei and W. Srisa-an, "Contaminant removal for android malware detection systems," in Proceedings of IEEE International Conference on Big Data, 2017

Threats are sensitive issue for the era of digitalization it seems to be more problematic when it could escape by copying the anti threat patterns of harmless applications after adopting same features like sending messages, and squashing the payload to minimize the risk of being caught by invoking payload during night. Their analyses of security sensitive resource which are being used like networks are on the marks. Indirect methods in viruses like these differentiate in sensitive and normal nature of applications by deep analysis, This proposal is based on targeting reasons which are used to violate security sensitive behavior of benign and malware applications. Basically App-context based security sensitive behavior are being extracted using static analyzing program for comparing benign and malware nature within apps. Implementation of a prototype is being proposed in this research by evaluating app-context on 202 malicious apps from malware dataset, and 633 existing benign apps on play store. With 95% of recall and 87.7% precision 192 malicious apps are being recognized via app-context. Research shows that intention of the behavior is closely related to maliciousness of security sensitive behavior(reflected via behavior) rather than type of he security sensitive sources that the behavior accesses.

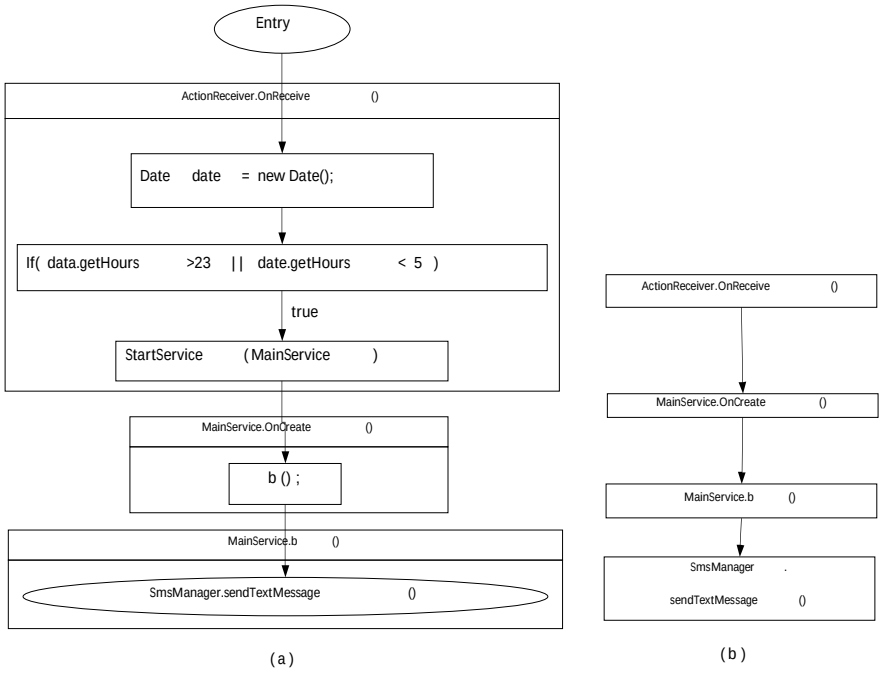


Fig 2.4(a). An RICFG (a) and its corresponding ECG subgraph

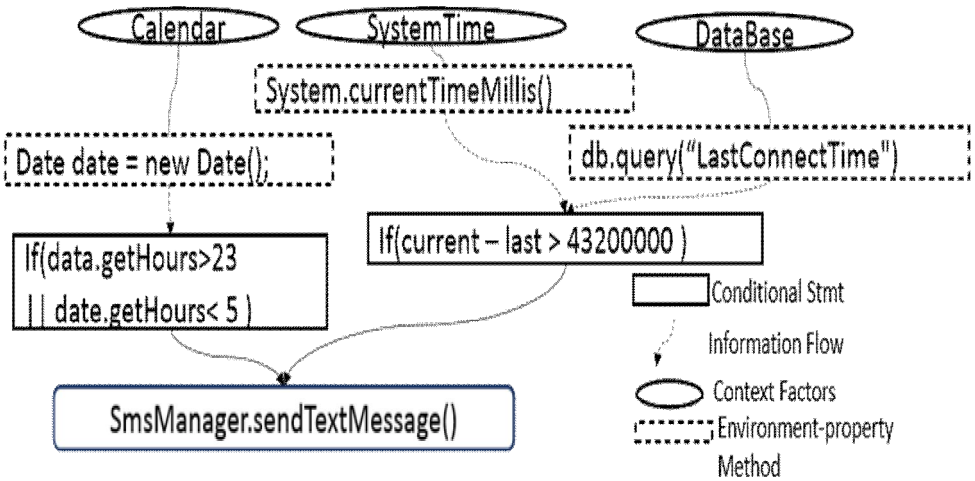


Fig 2.4(b). Context factors of Moon Sms

E. *X. Cao, L. Liu, W. Shen, A. Laha, J. Tang, and Y. Cheng, "Real-time misbehavior detection and mitigation in cyber-physical systems over wlans," IEEE Transactions on Industrial Informatics, vol. 13, no. 1, pp. 186–197, 2017*

Mass level threat could be observed increasing rapidly after the hike in demands of mobile networks currently. Networks interface are used as gateway to perform malicious actions, like stealing personal information of user, and accomplishing harmful activities on individual or organization. In this research they are proposing Text-droid, combining natural language processing and machine learning makes effective and automated threat analyses process. Charactering malware samples are done by extracting identifiable features(n-gram sequences) via Text-droid . Support vector machine classifier is been applied for developing malware detection model for detecting mobile threats. Advance SVM models represent powerful performance via two major data sets, Test set adopting threat detection rate occupying 96.36% and app set occupying 76.99% in the wild, in advance , flow header visualization process is developed for visualizing major text generated while applications network interactions, Applications complex actions are analyzed by security researchers for betterment.

F. *S. Wang, Q. Yan, Z. Chen, B. Yang, C. Zhao, and M. Conti, "Detecting android malware leveraging text semantics of network flows," IEEE Transactions on Information Forensics and Security, 2017.*

A recent report says that every 4 seconds a new malicious program is launched This quick malware delivery causes current malware detection systems to go far behind, enables malicious apps to escape testing efforts and even official app stores to distribute number of negative consequences occur when trusted platform circulate malware. First, the popularity of such malicious apps could allow devices to be infected rapidly and extensively. Furthermore, researchers and authorities relying on machine-based detection techniques can also install and accidentally mark such programs as harmless because they have not been exposed as malware. Such tests are then used to process training and testing as part of their benign dataset. Contaminants in benign data may affect their detection and identification techniques effectiveness and accuracy. In order to resolve this issue, author proposed PUDROID (Positive and Unlabeled Learning-Based Android Malware Detection) for auto and efficient reduction by machine-to-apprentice malware classifier and detectors. Allow machine learning to be more efficient and accurate in classifying and detects malware. The researcher applies a feature selection strategy to pick appropriate features from a variety of characteristics to further improve the performance of such detector systems. Researcher then measures detection rates and detection system accuracy using two datasets, with PUDROID for contaminant detection other without contaminants being removed. The results show that the Author can significantly improve malware detection levels as well as detection accuracy when contaminants are removed from the datasets.

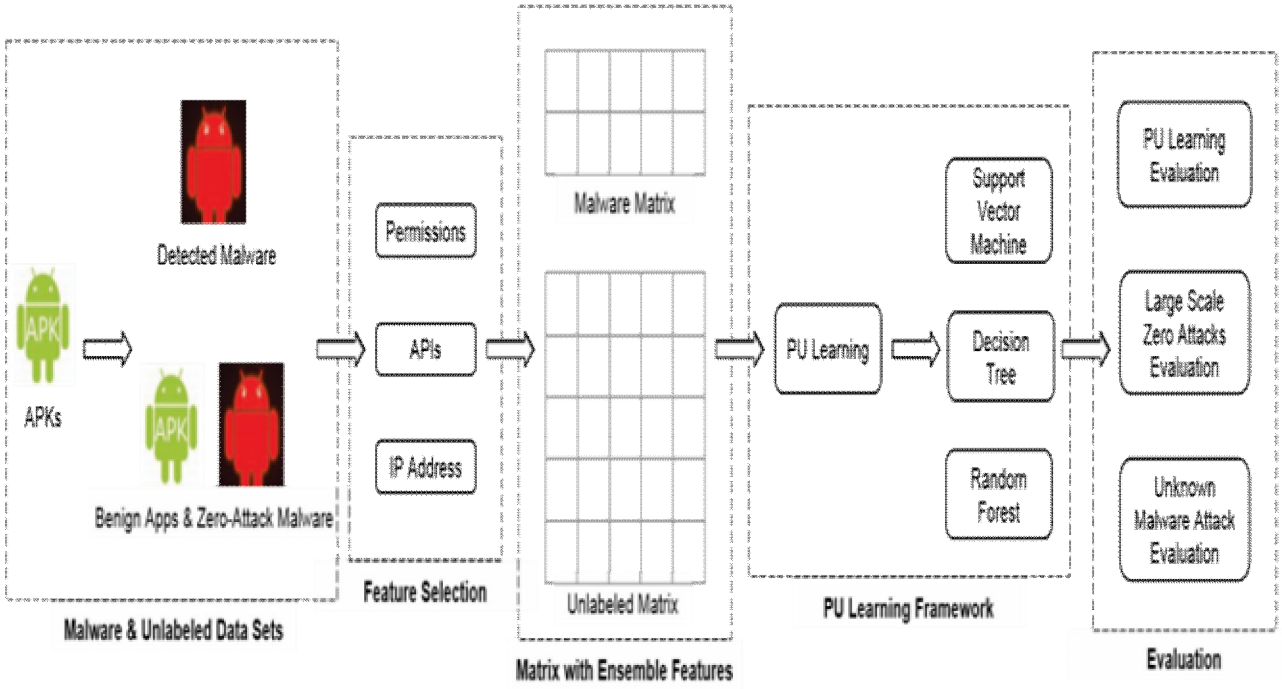


Fig 2.6. An Overview of PUDroid Approach

G. *T. Cruz, L. Rosa, J. Proenca, L. A. Maglaras, M. Aubigny, L.Lev,J. Jiang, and P. Simoes, "A cybersecurity detection framework for supervisory control and data acquisition on Industrial Informatics, vol. 12, no. 6, pp. 2236–2246, 2016.*

A deceptive node can gain significant advantage over other ordinary nodes in terms of resource sharing via the deliberate handling of its protocol parameters in cyber-physical systems (CPS) over IEEE 802.11e based local wireless zone networks (WLANs). Because of the random access to the protocol, it is difficult to reliably and in real time identify the misbehaving node. In the case of IEEE802.11e networks with heterogonal network configurations, several current misbehaviors, designed primarily for standard IEEE802.11 networks, are inapplicable. In this research, author proposed number of novel countermeasures for real-time and low-weight use, including a hybrid-share abuse detector and a packet-dropping system for CPS based on IEEE 802.11e. A develop mathematical models for the efficiency and mitigation mechanisms of the proposed detector. Extensive results of simulation show that the proposed mechanisms can achieve a high rate of detection and punish a malfunctioning node with a high rate of fall.

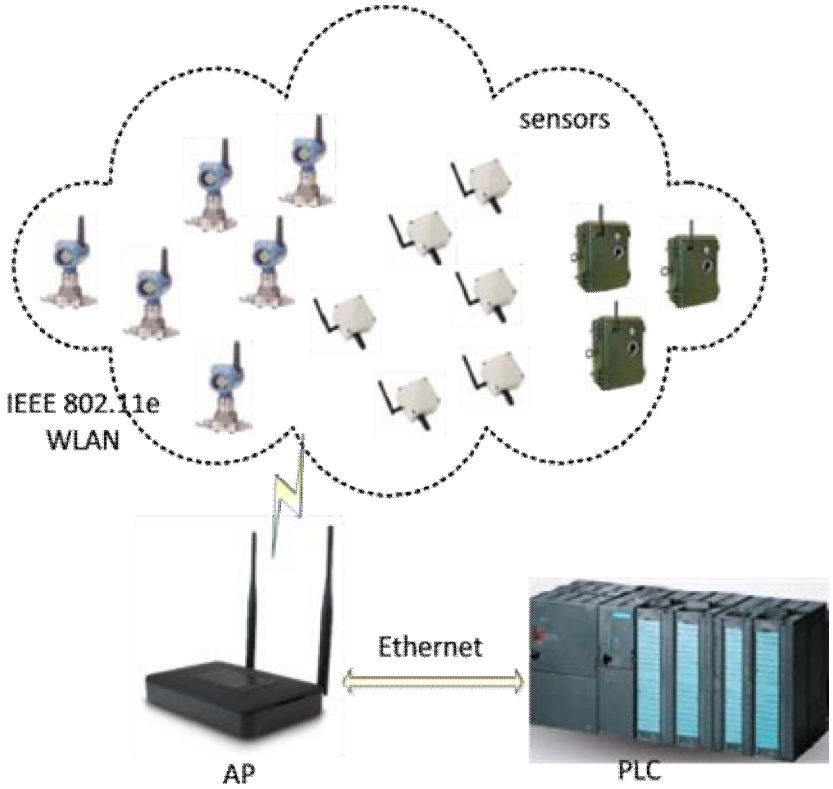


Fig 2.7. Illustration of the simulating CPS. All the sensors report data to the AP based on the IEEE 802.11e protocol.

III. CURRENT SYSTEM

Android is currently is the most used platform occupying 85% of the market share different android users install apps from various sites and the sites that provide malicious app. That consists of various malware present according to a survey on malware infection over 97% device were targeted. To address such problem some researchers has developed various approaches to encounter such problem.

1) *Riskranker:* This system uses static approach to discover malicious app
2) *Taintdroid:* This system uses dynamic approach to track sensitive data via tainting analysis

A. *Limitations Of Current System*

1) Static analysis used in risk ranker can mainly lead to false positive.
2) Dynamic analysis used in taint droid request adequate input to perform execution So privacy protection becomes a concern.
3) There is only detection and no removal of malware.
4) Easily prone to malware

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue XII, Dec 2019- Available at www.ijraset.com

IV. PROPOSED SYSTEM

In our proposed system we are going to present SIGPID to perform extraction of data and as design of SIGPID is effective in detection of malware. Our approach identifies the significant permission required by an application to work. And differentiate between essential and non-essential permissions and on such basis the malware is detected and removal is performed.
Some basic components of system are

A. MLDP (Multi-level data pruning)

The first component basically analyze the permission required by an app not every application require each and every permission and required permissions are listed in a package. package. And after performing various tests the app is analyzed and is any error id detected it is terminated.
Various tests are performed via multiple permission tests:

1) Permission for negative rate.
2) Permission for association rules.
3) Support based permission.

[2].MALWARE DETECTION USING SIGNIFICANT PERMISSION SVM (support vector machine) determines a hyper lane to separate classes with margin based on dataset that consist of benign and malicious software. We basically compare all result records to detect correctness via SVM

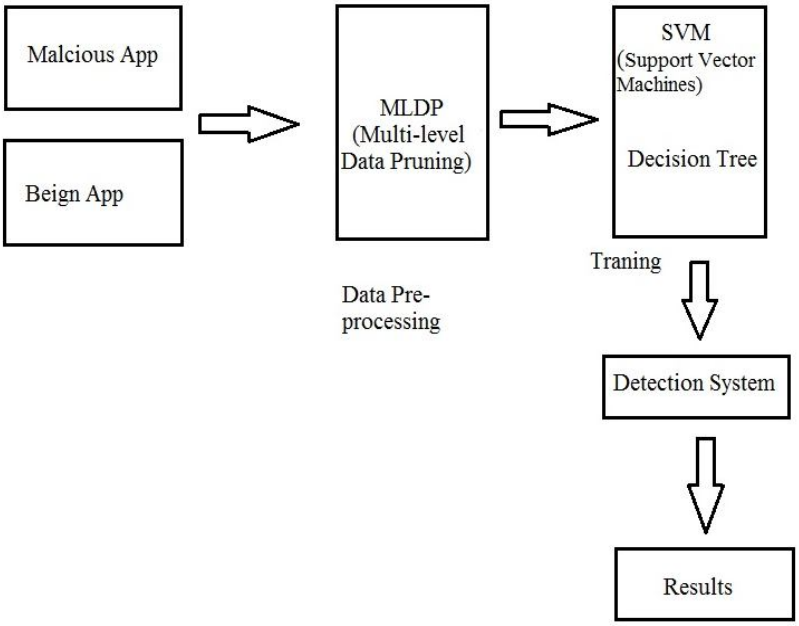


Fig. Proposed System Architecture

V. CONCLUSION

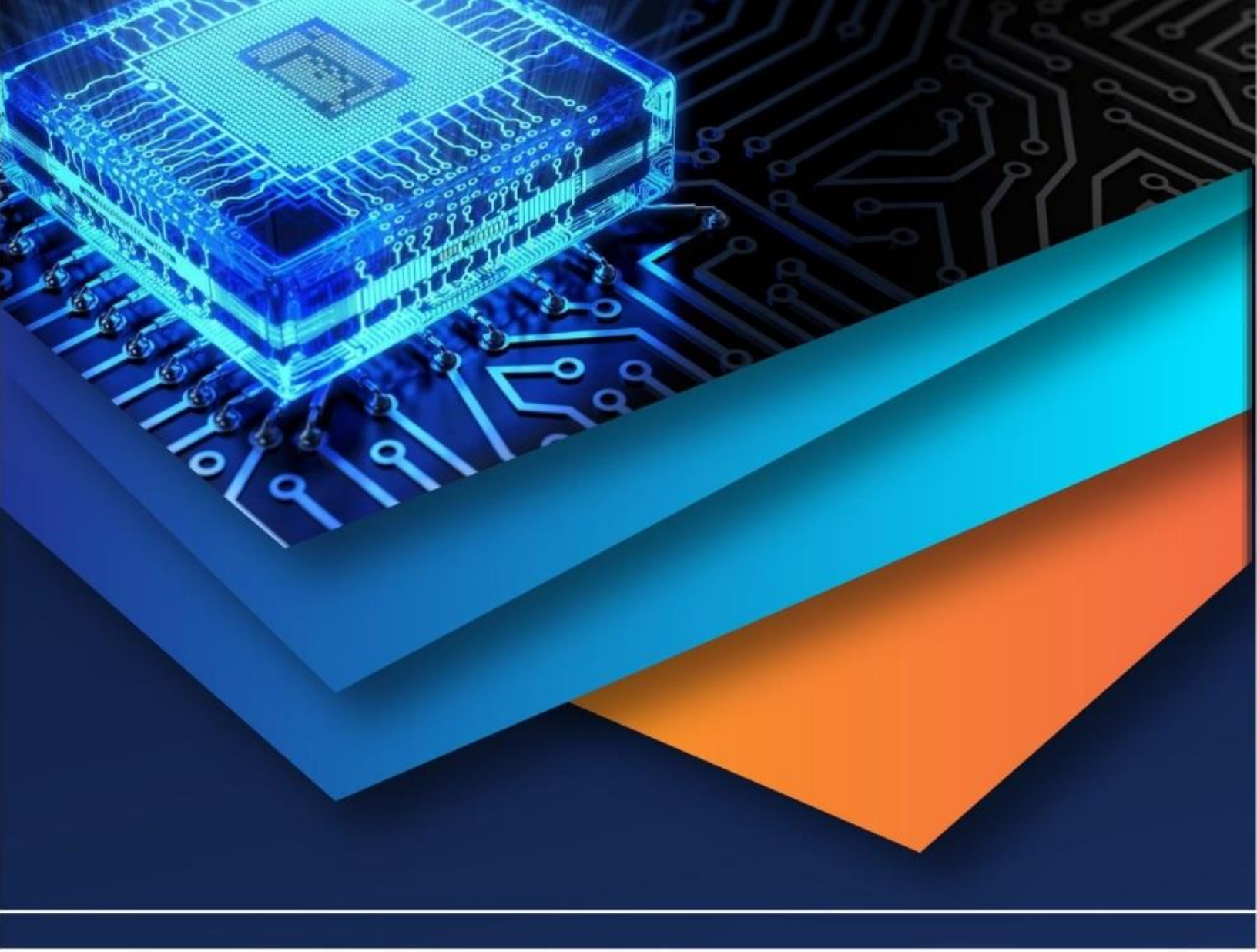
The existing systems has been examining the mechanism for classifying Android apps, whether they are malicious or ordinary.So Existing system's main focus is on detecting malicious apps, here we're preventing the malicious app by uninstalling thos e detected apps. Therefore, the malware detection system survey analyzes malware apps based on permission list using SIGPID and f ocuses on malware prevention.

VI. ACKNOWLEDGMENT

The All faith and honor to our HOD for his grace and inspiration. I would like to thank all the members of my Friends and Family for loving us. Thanks sincerely to our Department Head, Project Coordinator, our Project Guide and all other staff members to provide us with feedback for this journal.

REFERENCES

[1] IDC, “Smartphone os market share, 2017 q1.” [Online].Available:https://www.idc.com/promo/smartphone-market- Share/os .
[2] M. Grace, Y. Zhou, Q. Zhang, S. Zou, and X. Jiang, “Risk ranker: scalable and accurate zero-day android malware detection,” in Proceedings of the 10th international conference on Mobile systems, applications, and services. ACM, 2012, pp. 281–294.
[3] S. Wang, Q. Yan, Z. Chen, B. Yang, C. Zhao,and M. Conti, “Textdroid: Semantics-based detection of mobile malware using network flows,” in IEEE INFOCOM 2017.
[4] J. Z. L. H. P. S. Y. Lichao Sun, Xiaokai Wei and W. Srisa-an, “Contaminant removal for android malware Detection systems,” in Proceedings of IEEE International Conference on Big Data, 2017.
[5] X. Cao, L. Liu, W. Shen, A. Laha, J. Tang, and Y. Cheng, “Real-time misbehavior detection and mitigation in cyber-physical systems over wlans,” IEEE Transactions on industrial informatics,vol 13, no. 1,pp. 186-197, 2017.
[6] S. Wang, Q. Yan, Z. Chen, B. Yang, C. Zhao, and M. Conti, “Detecting android malware leveraging text semantics of network flows,” IEEE Transactions on Information Forensics and Security, 2017.
[7] T. Cruz, L. Rosa, J. Proenca, L. A. Maglaras, M. Aubigny, L.Lev,J. Jiang, and P. Simoes, “A cybersecurity detection framework for supervisory control and data acquisition on Industrial Informatics, vol. 12, no. 6, pp. 2236–2246, 2016.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY