A Survey on Various Malware Detection Techniques on Mobile Platform

Aashima Malhotra
Dept. of Computer Science and engineering
Chitkara University
Himachal Pradesh, India

Karan Bajaj
Dept. of Computer Science and engineering
Chitkara University
Himachal Pradesh, India

ABSTRACT
With the rapid arrival of mobile platforms on the market, android Platform has become a market leader in 2015 Q2, according to IDC. As Android has ruling most of the market, the problem of malware threats and security is also increasing. In this review paper, a fastidious study of the terms related to mobile malware and the techniques used for the detection of malware is done. Some proposed methods and type of approaches used in those methods are also summarized.

General Terms
Pattern Recognition, Permission based detection Technique.

Keywords
Malware, Types of malware, Detection techniques, Permissions.

1. INTRODUCTION
Now-days smart-phones are becoming very famous all around the world. As the study says, among all platforms, Android is the widely used platform. With the rising pervasiveness of using these mobile platforms in delicate applications, there is a problems associated with malware targeted at mobile devices. A malware could be any code which is added, changed or removed from an application in order to purposely cause harm the intended function of the system.

A malware has threatened mobile devices for many years and android is gaining popularity with time. Seeing this most of the discovered malwares are aiming at android platform. The main purpose of intruder is to steal data, personal information, gaining access to user's accounts and establishing control channels. The performance of a device also depends upon the type of malware. There are lots of different kinds of malware such as Ransom-ware, Spyware, Worms, Trojan-horses etc.

Ransom-ware*:* Ransom-ware hits android in 2014. This is a type of malware that holds a device to ransom, by clasping it down so that it can't be used until the owner of the device pay the hostage-takers.

Spyware: It usually enters the device when free or trial software is downloaded and installed in any device. It is

installed without any user's consent. It poses a threat to device by using and spreading sensitive information of user.

Worms: Worm is a program whose objective is to perceptually reproduce it-self and spread from one device to another by transmitting its own copy via network without any interaction or authorization of user.

Trojan-Horses: Trojans always requires the interaction of user. Trojans are usually inserted into apparently attractive and non-malicious executable files or applications that are downloaded and executed by the user. It oftenly destroy data or extract private information. Once activated, it causes a serious damage by deactivating applications or the phone it-self, rendering it crippled after sometime.

Adware: Its purpose is to just advertising the products or websites that are annoying but doesn't cause any harm. Android dowgin is a adware that install itself on an android device as a bundle with the other applications. After that it displays ads in the notification bar and cannot be removed easily. It is estimated that between 10000-50000 users are infected with this adware.

In table I, we show the list of some malware and their behaviour. The intruders sometimes have financial pepping up. During installation of applications on mobile devices, some applications send SMS's without user's knowledge that revert itself in user's bills. Such applications have been piling up for years. Some attackers earn money via such malware.

Moua-bad is a malware that have gone further by making phone calls secretly in such a way that it waits until a while after the devices screen goes off and screen becomes locked. And then it start calling premium numbers, as soon as the user interact with the device, the malware disconnects the call.

2. MOBILE MALWARE DETECTION METHODS
Malware detection Techniques are widely divided in two categories: Anomaly Based Detection and Signature Based Detection (Misuse-based). Any malware detection system can use one or combination of these techniques for detecting malware.

Table I. Mobile Malware and their behaviour

Malware	Operating System	Behaviour
Walkingwat	Android	Its purpose is to just generate the purposeless destructions to the user. These malwares are mostly developed for fun.
Ikee	iOS	
NMPlugin	Symbian	
DroidLight	Android	This class secretly gather user's personal details and information and selling these details to marketers.
Privacy-A	iOS	
SPIsSaga	Symbian	
Geinimi	Android	Sends Span messages to mobile phone that generally contains phishing links and ads.
Shurufa	Symbian	
FakePlayer	Android	Generates premium rate calls and

Floker	Symbian	SMS.
Ikee.B	iOS	Steal user's credentials such as account details by secretly listening to text messages, capturing key logging etc.
InSpirit	Symbian	

Figure 1. shows different approaches which comes under these techniques. A specific analysis or approach of both the techniques is determined by how a particular technique gather information to detect malware.

An anomaly based detection apply its knowledge on program under inspection to decide its maliciousness. Knowledge consist a set of valid or normal behaviour. A specification based technique is a special type of anomaly based. This technique uses some specification or rule set of what is a valid behaviour in order to find behaviour of program under inspection. Programs violating specification are considered malicious. Anomaly based systems assume that all anomalous activity as malicious. Hence basing on the normal activities, system creates a normality model which then enables it to indicate normal activity [1].

Anomaly based technique generally works in two phases: training or learning phase and detection or monitoring phase. In training phase, the detector aim to learn the normal behaviour. The behaviour of host or program under inspection could be learnt under this phase. The detection of zero day attack is the prime advantage of anomaly based detection. The attacks which are not known to detector previously are known as zero day attack. High false alarm rate and difficulty in determining of what features should be learned causes problem in this technique.

A signature or misuse based detection technique uses its characterization of predefined patterns or signatures that can be matched with the data under inspection. A signature can be strings execution stack or binary information. Signatures require a repository, like any large quantity data requires some storage. All the knowledge, signature based technique has is represented by the repository. Repository is searched when this method is applied on program under inspection, to check whether the PUI contains a known signature. One of the limitations of signature based is that it cannot detect zero attacks.

Static, dynamic and hybrid approaches are used by intrusion detection system. Static approach detects malware before the execution of program under inspection whereas dynamic approach detects malware after or during the execution of the program under inspection. Hybrid approach is the combination of static and dynamic approach.

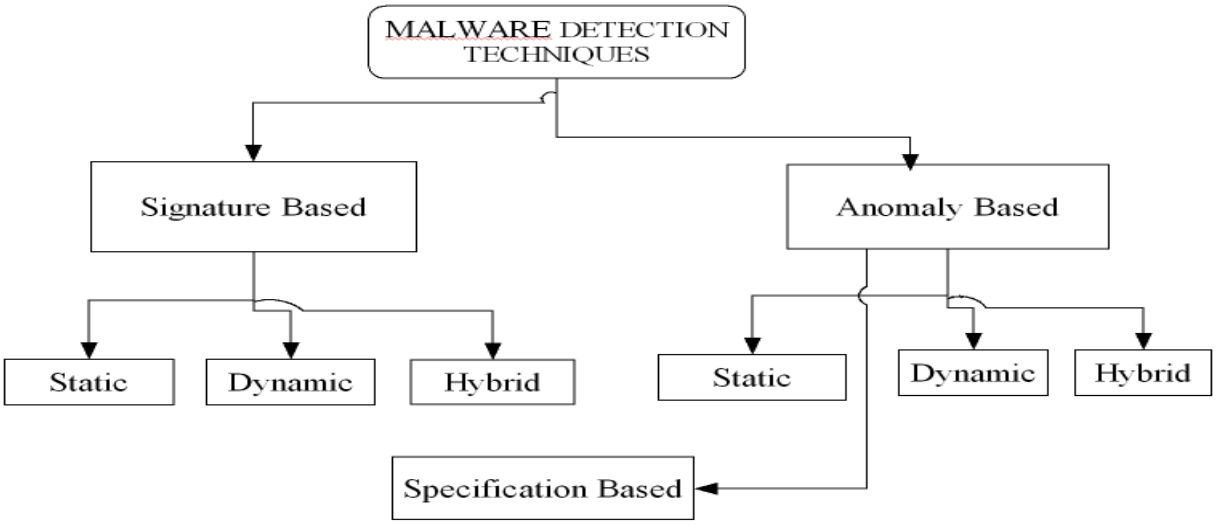

Fig 1. Malware Detection Technique

2.1 Static Malware Detection Technique

Static analysis approach is a fast and inexpensive approach of detecting maliciousness or bad code segments in a program without executing them. It can be applied on many representations of a program. Static analysis tools can applied on source code to detect memory corruption faults. It can also be used on the binary representation of a program. Some information gets lost while compiling the code into binary executables. It further complicates the job of analyzing code.

The techniques defined in Figure 2 are used in primary analysis, when programs are first checked out to find any threat. Figure 2(a) shows static malware detection technique for symbian OS. This technique uses IDA Pro to disassemble mobile application then extract system calls or features. Then it uses clustering mechanism to analyze application as malicious or non malicious. Figure 2(b) shows static technique used for performing static taint analysis on iOS. And Figure 2(c) shows static technique proposed for Android [3].

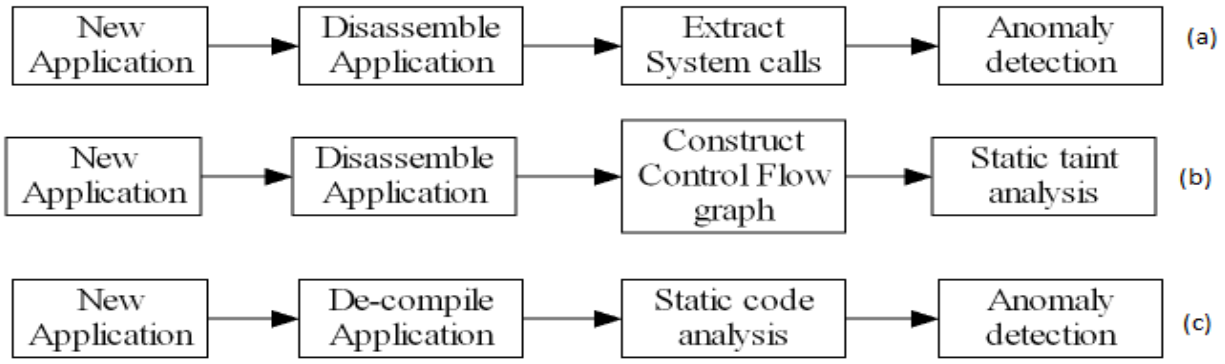

Fig 2. Static analysis based on system calls, taint analysis and source code

2.2 Dynamic Malware Detection Technique

To monitor dynamic behaviour of an application, this technique requires isolated environment for executing mobile applications, such as virtual machine or emulator. As dynamic analysis is done in runtime environment, it cop out the limitations of static analysis which are unpacking and obfuscation. Large scale analysis is another advantage of dynamic malware analysis.

But dynamic analysis suffers from partial code coverage as it usually monitors only single execution path. This drawback is known as dormant code. Also if the environment is not correctly isolated there is the risk of harming nearby systems. Primary use of dynamic analysis is in taint tracking and system call tracing. Enck [15] states a misuse based detection system which provide system-wide dynamic taint tracking for android. Figure 3 shows the system wide dynamic analysis in which mobile application passes to the dalvik machine to perform some granularities of taint propagation. Then dynamic analysis screens impacted data for any data loss before it leaves the system.

2.3 Permission Based Analysis

Several permissions are required by an android application to work and to install any application in mobile user has to allow all permissions requested by the application [5]. Permissions play a very important role in mobile application. It tells about the application's intention and back end activities to the user. Permissions are clearly defined in smart phones, so that application creator must get hands on appropriate permissions. In spite of that, some creator purposely hide the permissions the use in the application,

leading to application vulnerability [3]. Permissions include requested and required permissions. Android permissions are categorized into 4 types: normal, dangerous, signature and signatureOrSystem. Therefore, idea to determine a harmful

application is to check whether an application requires a dangerous permission.

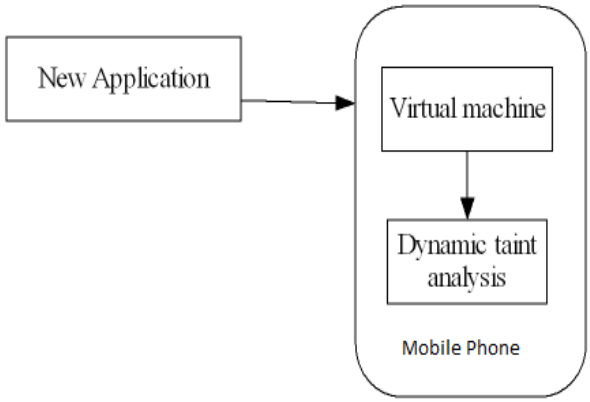

Fig 3. System wide Dynamic analysis

Access permissions are categorized as dangerous. Some dangerous permissions are: ACCESS COARSE LOACTION and ACCESS FINE reads the location of user, BLUTOOTH access Bluetooth devices, Access INTERNET etc. an application having more than one dangerous permission doesn't means that it's a baleful or unfriendly application. A location based application would require some dangerous permissions like ACCESS INTERNET or ACCESS COURSE LOCATION. Developer declares permissions manually and all declared permissions are not actually required by the application. This will increase difficulty in finding malicious application based on the permissions.

3. LITERATURE SURVEY

So far study of different malware detection methods and approaches are done. Different malware detection systems are proposed earlier. Table II shows an academic research on different malware detection systems. A solution based on monitoring events occurring on Linux-kernel level is proposed by Schmidt et al. [9]. They analyze Linux based tools for improving security and extracting features from the Linux kernel. After that these features were used to create a model for the smart-phone behaviour. But they were not able to test their system because on that time there were no real Android devices available. Same authors in [11] proposed an Android application sandbox. They aim to perform static and dynamic approaches on android applications. In this an android application sandbox is proposed to detect mistrustful application.

Table II. Academic Research On Malware Detection System

Author	Approach	Detection Method	Platform	Description
Schmidt et al. (2009) [10]	HIDS	Signature based Detection	Android	Host based static detection approach, which analyze executables to extract function calls. Classification is done by comparing function calls with executables.
Blasing et al. (2010)[11]	HIDS	Signature based Detection	Android	Static and dynamic analysis is performed on android application. Static analysis is performed to detect patterns by scanning source code. Dynamic analysis is done in isolated environment on application.
Portolakidis et al. (2010)[12]	HIDS,NIDS	Anomaly based Detection	Android	Paranoid Android, is a system where complete malware analysis is done in the cloud using mobile phone replicas. It requires a secure virtual environment.
Shabtai et al. (2010)[13]	HIDS	Anomaly based Detection	Android	Uses KBTA methodology to detect suspicious temporal patterns and issues a alert if an intrusion is found.
Shabtai et al. (2011)[14]	HIDS	Anomaly based Detection	Android	Andromaly: It is a host based intrusion detection system, which monitors features of mobile phone and apply machine learning algorithms to classify malicious and non-malicious data.
Enck et al. (2010)[15]	HIDS,NIDS	Anomaly based Detection	Android	Taint-Droid: It is a system which monitors android application and alert user when sensitive data is found. To monitor sensitive data information it uses taint tracking.
Grace et al. (2012)[16]	HIDS	Signature based Detection	Android	RiskRanker tool is a based on signature based detection of known exploits. Static analysis is done on symbolic execution.
Egele et al. (2011)[17]	NIDS	Signature based Detection	iOS	PiOS: Detects leaked sensitive phone-related information. Firstly, it decrypts Objective-C binary and generate control flow graph. Presence of leaks - paths arising from functions obtaining sensitive resources are checked in the graph.
Burguera et al. (2011)[18]	HIDS	Behaviour based Detection	Android	Crow-droid: A framework is proposed which analyze mobile phone application. It checks the anomalous behaviour of known application and in collaboration with android user community, it will be able to distinguishing between malicious and non malicious application.

Schmidt et al. [10] proposed a host based malware detection system for android platform. It is a signature detection method which applies static approach on executables Portolakidis [12] proposed another system where security checks are applied on remote security servers. At the same time they implement security model prototype for android phones. Shabtai et al.[13] proposed a host-based intrusion detection system in which time-stamped security data is repeatedly monitored and then the (KBTA) knowledge based temporal abstraction methodology is proposed. Central management capabilities for android mobile developed to evaluate KBTAmethod and combine with light weight Intrusion detection system (IDS).

Andromaly, a framework to detect malware on android devices is developed by Shabtai[14]. It monitors different events and features obtained from mobile devices. then it applies machine learning to determine collected data as malicious and non malicious. Dynamic analysis method uses application emulation or execution. Enck et al. in 2010 provides a dynamic taint tracking for Android. It integrates variable, method, message, file-level. It tracks multiple sources of sensitive data. Virtual execution environment is used for real time analysis. TaintDroid acquires 32% performance overhead and enforces minor overhead on related third party application.[15] Tracking the flow of acute data with TaintDroid generates beneficial input for android users android users and security service associations chasing to

find applications not working properly. Risk-Ranker [16] tools is implemented by authors in which static analysis is done on detect paths of user unexpected actions. It also uses signature based analysis to find known exploits. Applications which are using encryption and decryption methods examined high risk.

Egele et al. in 2011 provides a network intrusion detection system PiOS. It uses signature based detection method. It was developed to detect tenable leaks of sensitive phone-related data and information on smartphone platform. [17]. Burguera et al. in [18] provides a behaviour based system crowdroid. It consists of a client application which is installed on central server and android devices. Client application transfer system calls to the server executed by the observed application. For every application behaviour datasets are created by server. After collecting sufficient data partial clustering algorithm is used to cluster each dataset. It produces two sets: benign programs behaviour and Trojan-like behaviour.

Yujie Fan et al. in 2016 propose a detection framework based on data mining called Malicious Sequential Pattern based Malware Detection (MSPMD) sequence mining algorithm. This framework is the combination of sequential pattern matching algorithm and All-Nearest-Neighbor (ANN) classifier. This framework gave favorable experimental results on real data collection. MSPMD attain better outcomes in Detection Rate(96.17%), False Positive Rate(6.13%) and ACCURACY(95.25%) as compare to other Malware detection framework [19].

4. CONCLUSION

Various literatures related to mobile malware detection has been thoroughly studied and analyzed in this paper. The various pros and cons of the different techniques have been discussed and listed. Two major techniques, namely, anomaly based and signature based techniques, are usually taken up by researchers. In the signature based techniques, the pattern of instruction sets is studied and analyzed while in the anomaly based techniques, the unusual activities are detected. The review gives an idea of research gaps available in the field. In future signature based techniques can be enhanced using DNA matching techniques applied in other domains. Also permission based strategy can be utilized and a hybrid technique can be developed for improved performance in terms of accuracy, precision, recall, etc.

5. ACKNOWLEDGMENT

I would like to express my gratitude to Mr**.** Karan Bajaj for his support, help and guidance during my research work. He has been a constant guiding force and source of illumination. Finally thank the almighty god with whose grace I am always motivated and deeply engrossed with my thesis work during the entire duration from its conception to success.

6. REFERENCES

[1] Schmidt A.D. 2011. Detection of Smartphone Malware, doctoral diss., Berlin Institute of Technology, Berlin, Germany.

[2] Landage, J. and Wankhade, M. P. 2013. Malware and Malware Detection Techniques: A Survey, International Journal of Engineering Research and Technology, Vol. 2, no. 12, 2013.

[3] Felt, Porter, A., Finifter, M., Chin, E., Hanna, S. and Wagner, D. 2011. A survey of mobile malware in the wild, Proc. 1st ACM workshop on Security and privacy in smartphones and mobile devices, ACM, New York, 2011, pp. 3-14.

[4] Dini, Gianluca, Martinelli, F. and Sgandurra, D.2013. MADAM: A Multi-level Anomaly Detector for Android Malware, Journal of MMM-ACNS, vol. 12, 2013, pp. 240-253.

[5] Huang, C.Y. , Tsai, Y.T. and Hsu, C.H. 2013. Performance evaluation on permission-based detection for android malware, Advances in Intelligent Systems and Applications, Springer, Berlin Heidelberg, vol. 2, 2013, pp. 111-120.

[6] Shabtai, A., Moskovitch, R., Elovici, Y. and Glezer, C. 2009. Detection of malicious code by applying machine learning classifiers on static features: A state-of-art survey, Journal of Information Security Technical Report, vol. 14, 2009, pp. 16-29.

[7] Christodorescui, S. and Jha, S. 2006. Static analysis of executables to detect malicious patterns, Wisconsin Univ-Madison Dept Of Computer Sciences, 2006.

[8] Schmidt, A.D. 2009. Static analysis of executables for collaborative malware detection on android, In Communications IEEE International Conference, Dresden, Germany, 2009, pp. 1-5.

[9] Schmidt, A.D. , Schmidt, H.G., Clausen, J., Yuksel, K.A., Kiraz, O., Camtepe, A. and Albayrak, S. 2008. Enhancing security of linux-based android devices, Proc. of 15th International Linux Kongress, Lehmann, Hamberg, Germany, 2008.

[10] Schmidt, A.D., Camtepe, A. and Albayrak, S. 2010. Static smartphone malware detection, Proc. of the 5th Security Research Conference (Future Security), 2010, pp. 146.

[11] Blasing, T., Schmidt, A.D., Batyuk, L., Camtepe, S. A. and Albayrak, S. 2010. An android application sandbox system for suspicious software detection, 5th International Conference on Malicious and Unwanted Software (Malware), Nancy, France, 2010, pp. 55-62.

[12] Portokalidis, G., Homburg, P., Anagnostakis, K. and Bos, H. 2010. Paranoid android: versatile protection for smartphones, Proc. of the 26th Annual Computer Security Applications Conference, *ACSAC, ACM*, New York, 2010, pp. 347-356.

[13] Shabtai, A. , Kanonov, U. and Elovici, Y. 2010. Intrusion detection for mobile devices using the knowledge-based, temporal abstraction method, Journal of Systems and Software, vol. 83, 2010, pp. 1524-1537.

[14] Shabtai, A., Kanonov, U., Elovici, Y., Glezer, C. and Weiss, Y. 2012. Andromaly: a behavioural malware detection framework for android devices, Journal of Intelligent Information Systems, vol. 38, 2012, pp. 161-190.

[15] Enck, William, Gilbert, P., Han, S., Tendulkar, V., Chun, B.G., Cox, L. P., Jung, J., McDaniel, P. and Sheth, A.N. 2014 TaintDroid: An information-flow tracking system for realtime privacy monitoring on smartphones, Journal of ACM Transactions on Computer Systems (TOCS), vol. 32, 2014, article 5.

[16] Grace, M., Zhou, Y., Zhang, Q., Zou, S. and Jiang, X. 2012 Risk-Ranker: scalable and accurate zero-day

Android malware detection, Proc. of MobiSys, New York, NY, USA, 2012, pp. 281-294.

[17] Egele, M., Kruegel, C., Kirda, E., and Vigna, G. 2011. PiOS: detecting privacy leaks in iOS applications, In NDSS, 18th Annual Network and Distributed System Security Symposium, San Diego, CA, USA, 2011.

[18] Burguera, I., Zurutuza, U. and Tehrani, S.N. 2011. Crowdroid: behaviour-based malware detection system for android, Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices, *ACM*, Chicago, Illinois, USA, 2011, pp. 15-26.

[19] Fan, y., ye, y. and Chen, L. 2016. Malicious sequential pattern mining for automatic malware detection, Expert systems with application (Jan. 2016).