IRA-International Journal of Technology & Engineering ISSN 2455-4480
Proceedings of the
International Conference on Science & Engineering for Sustainable Development (2017)
Pg. no.137-147
Published by: Institute of Research Advances
https://research-advances.org/index.php/IRAJTE

Light Weight Defense Mechanism Against Camera Based Attacks

Supriya Deshpande*[1], Shirish Pattalwar[2], Poonam Lohiya[3]

[1,2,3] Prof. Ram Meghe Institute of Technology & Research, Badnera, Amravati, India.
(Supriya.deshpande36@gmail.com) (shirish.pattalwar@rediff.com) (pblohiya@mitra.ac.in)

Type of Review: Originality Check & Peer Review under the responsibility of the Scientific Committee of the Conference and The Institution of Engineers (India).
DOI: http://dx.doi.org/10.21013/jte.ICSESD201714

How to cite this paper:
Deshpande, S., Pattalwar, S., Lohiya, P. (2017). Light Weight Defense Mechanism Against Camera Based Attacks. *Proceedings of the International Conference on Science & Engineering for Sustainable Development (2017)*, 137-147. doi: http://dx.doi.org/10.21013/jte.ICSESD201714

ABSTRACT

A number of android security and privacy vulnerabilities have been exposed in past several years. However, the mobile malware and privacy leakage remain a big threat to mobile phone security and privacy. It has been observed that phone camera would become a traitor in such attacks by capturing photos or videos secretly. Secret photography is not only immoral but also illegal due to invasion of privacy. Phone users themselves could also become victims and if the phone camera is exploited by malicious spy camera app, it may cause serious security and privacy problems. Most of the existing mobile antivirus apps are unable to monitor such camera based attacks. Hence, we demonstrate two camera based attacks through our proposal namely, Remote controlled real time monitoring Attack like Screen unlocking Attack, SMS Attack. We also propose a light weight defense scheme as a countermeasure for the camera attacks that can protect android phones against harmful spy Camera attacks.

Keywords: Android, Security, Privacy Leakage, Defense App, Spy Camera

Introduction

Smart phones are said to be a basic part in our everyday life, from individual connection stage to portable informal community hub. On one hand, carrying a 3G cell phone is verging on like having a modest PC in your pocket, as cell phones are frequently equipped with complete working frameworks (OSs) that give a standardized interface and stage for application engineers. On the other hand, a present pattern for Internet social networking Web destinations, like MySpace and Facebook are to turn portable. The small size of android devices, attached with people's careless usage, increases the chances of malicious software injection onto smartphones. They can be compromised in three respects: confidentiality, integrity, and availability. Smartphone security has not kept pace with traditional computer security. Technical security measures, such as firewalls, antivirus, and encryption, are uncommon on mobile phones, and mobile phone operating systems are not updated as frequently as those on personal computers. Recent advancements in mobile commerce have made users to conduct many transactions from their smartphone, such as purchasing goods over wireless networks, redeeming coupons and tickets, banking, and even paying at cash registers.

As the official application market, Google's Play store provides a platform of delivering apps for Android smartphones. There are many third-party app markets providing similar platforms. App developers publish their apps on the Google's play or on the third-party app markets, where end users download and install their interested apps on their Android smartphones. Obviously, how to detect and keep the large number of malware out of the application markets is an emerging, crucial, but challenging issue. Fig 1 shows various types of attacks that are possible on android devices. Attacks can be of many types amongst which our ideology focuses on Spywares available in the market for performing malicious activities on camera device of the android phone.

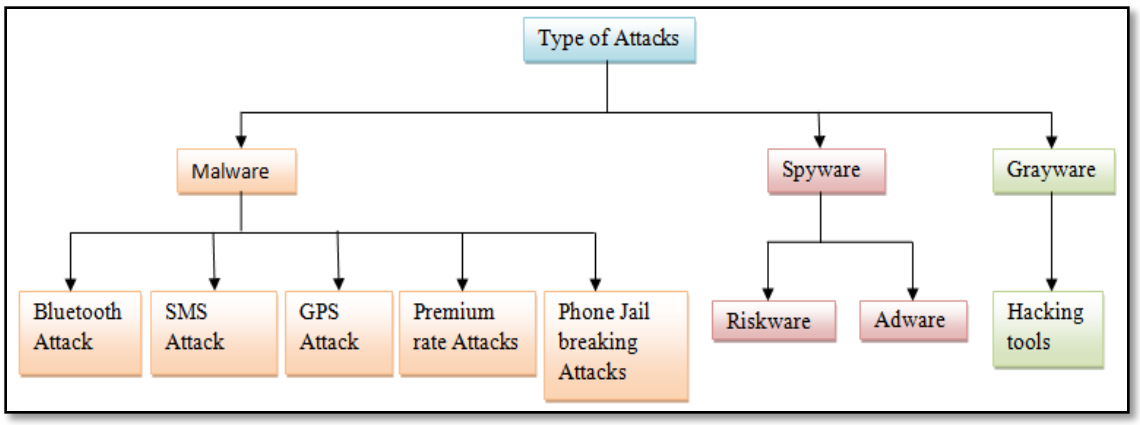

Fig. 1: An Example Review of Moto G2

Risk ware, a spyware can be defined as type of applications that can be modified for another purpose and used against the mobile or computer user or owner Its example can be VNC (Antivirus).

This paper can be summarized in following three points:

1. Remote controlled real time monitoring Attack for Android Camera on the triggering event of Screen Unlocking of the android device.
2. Real time attack on triggering event of SMS.
3. Countermeasure Mechanism based on machine learning to detect all types of maliciously behaving Applications based on permission analysis.

Permission Analysis is the study of android permissions used by various applications to detect whether the applications use permission for benign reasons or for some malicious activities. Detection of malicious applications is the integral part of the System without which the risk of using various applications cannot be reduced.

Literature Survey

A.Exploring Permission Induced Risk in Android Applications For Malicious Application Detection
Smartphones and mobile devices have become explosively popular for personal or business use in recent years. As reported by Digitimes research, global smartphone shipments are expected to reach around 1.24 billion in 2014. This number has increased 30% over the last year. Meantime, smartphone platforms have seen a massive surge in malwares. With Android accounting for 81 percent of all smartphone shipments globally in the third quarter of 2013, it has unsurprisingly become the major target for mobile malware. The volume of Android malware families and samples has been growing explosively. Here three feature ranking techniques to evaluate the risk of granting each permission are employed, based on which the permissions are ranked from most to least risky. Second, permission sets, instead of individual permission, are evaluated by feature subset selection methods for investigating the risk introduced by the collaboration of several permissions. Third, the detection of malapps based on risky permissions is formulated as a classification problem and executed by building classifiers [2].

B. Droid Chameleon: Evaluating Android Anti – Malware against Transformation Attacks
There may be Several kinds of transformations that may be applied to malware samples while preserving their malicious behavior. Each malware sample undergoes one or more 5 transformations and then passes through the anti-malware tools. The detection results are then collected and used to make deductions about the detection strengths of these anti-malware tools. The transformations are classified as trivial which do not require code level changes or changes to meta-data stored in AndroidManifest,those which

result in variants that can still be detected by advanced static analyses involving data-flow and control-flow analysis (DSA), and those which can render malware undetectable by static analysis (NSA) [3].

C. Mobile Attacks And Defence

Jail breaking disables code signing on iPhones to run apps not from the App Store. This breaks almost all the protections iOS offers. First, it disables code signing, which opens the platform up to malware. In addition, many of the added non signed applications run at the root level without a sandbox. The jail breaking patches also somewhat disable data execution prevention by allowing writable and executable memory, which isn't normally in iOS. So, the openness that jail breaking offers also introduces potential security problems. This will require attackers to have two exploits, such one to get code running and one to break out of the sandbox [4].

D. Rootkits on Smart Phones

Rootkits are malware that achieve their malicious goals by infecting the operating system. For example, rootkits may be used to hide malicious user space files and processes, install Trojan horses, and disable firewalls and virus scanners. Rootkits can achieve their malicious goals stealthily because they affect the operating system, which is typically considered the trusted computing base. The term "rootkit" originally referred to a toolkit of techniques developed by attackers to conceal the presence of malicious software on a compromised system. Although such rootkits only persist until the system is rebooted, they are effective on desktop computers, which are often not rebooted for several days or months at a time. Once infected, a rootkit can serve as the stepping stone for several future attacks. For example, rootkits are commonly used to conceal keyloggers, which steal sensitive user data, such as passwords and credit card numbers, by silently logging keystrokes [5].

E. Apps Playground: Automatic Security Analysis of Smartphone Applications

There is very less difference between PCs, Laptops, Note Pads & Smart phones as all these are connected technologies. Various services like Social networking & gaming provided by smart phones with the help of applications, these are exposed to gain confidentiality .Sidewinder Targeted Attacks use basic ad libraries to infect android phones. They use non android services to target a machine to infect it which is an android smartphone. Intensity of this attack is high because no user thinks that a non android service may infect an android phone. UI Redressing attacks mostly target the browsers in the desktop as well as mobiles [6]. Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. It focuses on the development of computer programs that can teach themselves to grow and change when exposed to new data. This paper uses the concept of machine learning in order to classify malicious applications and benign applications. Following is the Table I that Shows the comparison of various approaches studied in the literature survey.

Methodology

A. Proposed System

If any material that is included in a paper is under copyright, the authors of the paper are responsible for obtain1. Demonstration of Camera based Attack:

Camera Attacks like Remote controlled real time attack on triggering event of screen unlocking. Application oriented Attack through SMS are demonstrated in order to show the severity of such types of attacks.

2. Countermeasure of the Attack with Light Weight Defense Mechanism

Tofight against Camera based attacks a countermeasure is implemented based on permission analysis to detect such harmful attacks and the user has the freedom to freeze or block such type of attacks.

Table 1: Comparative Analysis

System	Features	Merits	Limitations
Exploring Permission -induced Risk in Android Applications [2].	It systematically ranks the permissions w.r.t. their risk to the Android system.	It identifies subsets of risk permissions.It employs several algorithms, namely, SVM, Decision Tree and Random Forest, to detect malapps based on the identified subsets of risky permissions.	There are some root-exploit type malapps do not need to request any permissions. In this case, relying on only permissions is not feasible for the detection of malapps.
Trivial Transformatio n attack[3].	Repacking Disass embling and Reassembling	Difficult to detect by antimalware software.	Transformation Attacks are Detectable by Static Analysis.
Mobile Attacks And Defence [4].	Jailbreaking Attack Performed on IOS	It Somewhat disables data execution prevention by allowing writable and executable memory. (Not present in IOS).	The openness that jail breaking offers also introduces potential security problems.
Rootkits on Smart Phones [5].	Spying on Conversations via GSM.	Rootkits modify system utilities and some kernel modules often leave a disk footprint, and can possibly be detected using user-space malware detection tools.	Rootkit detection mechanisms must reside outside the control of the operating systems that they monitor.
Automatic Security A nalysis of Smart Phone Apps[6].	Sidewinder Targeted Attacks.	They use basic ad libraries to infect android phones	They use non android services to target a machine Or smartphone.

B. Remote Control Real Time Monitoring Attack

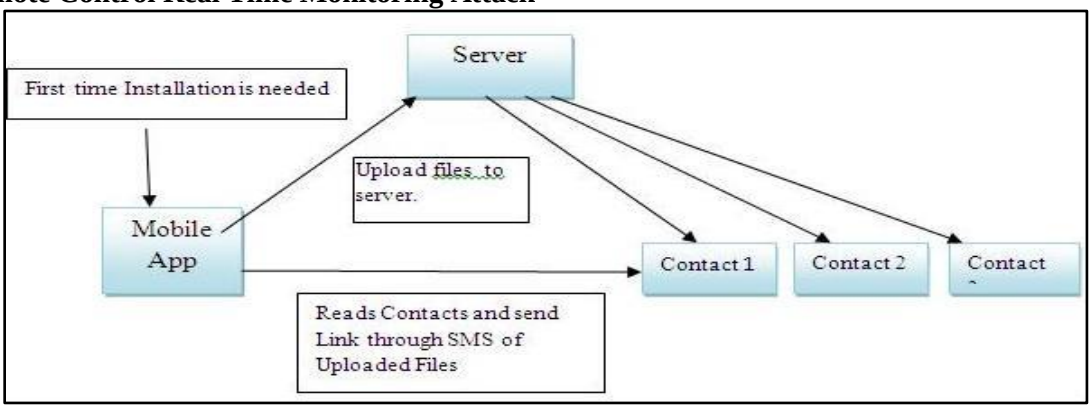

Fig. 2: Remote Control Real Time Monitoring Attack

Fig 2 shows the Attack performed on the event of Screen Unlocking. Once the application is installed on the mobile phone whenever the screen gets unlocked the camera starts automatically and captures images, It stores the images to a secret folder from where it gets uploaded to the server when the wifi connection or the internet is available. After the files are uploaded to server the contacts are read from the phone and a link of the files uploaded are sent to the contacts. Here the risk of leakage of information starts, as soon as the contacts open the link the files can be viewed by them. That is nothing remains private. Leaking out privacy is not only immoral but also a sort of cyber crime.

C. SMS Attack

In this Attack the Camera Starts automatically on the event of receiving of a particular type of secret message which the user may consider to be very genuine. Fig 3 shows the flow of the SMS Attack. This attack can also be called as application oriented Attack as it gets triggered by the help of an application. For demonstration purpose the images are stored in a secret file at some secret location. In actual spy cameras the users are completely unaware of the fact that these images get stored in devices' internal memory or the SD card if present.

D. Light Weight Defense Mechanism

This can be illustrated as a counter measure mechanism for detection of malicious Application. It is based completely on the information about the application permissions used by the developer in the formation of the application.

System is divided into two modules:

1. Operates on PC
2. Operates on Android Phone

Operations performed on PC are one time setups, while operation on android are executed when LWDM (Light Weight Defence Mechanism) application is opened. Weka tool is used compare different classifiers along with proposed Random Forest classifier. Input to this system is the matrix of permission while output its trained model is supplied as an input to classifier on the android device. Depending on prediction results applications are classified as benign or malicious. Only malicious applications are displayed as block list. User intervention is needed here to block the particular malicious applications.

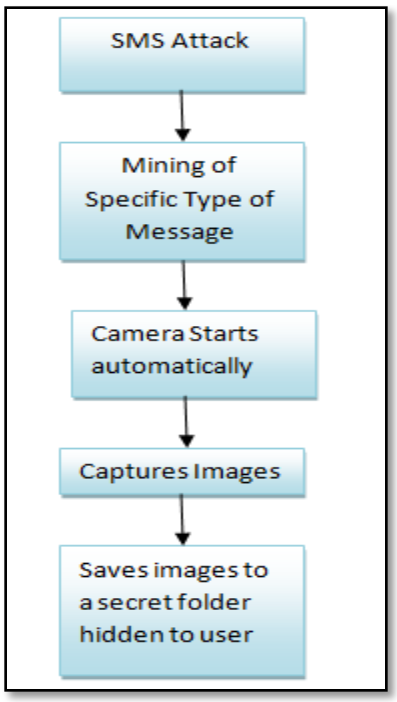

Fig. 3: SMS Attack

Fig. 4 shows a complete view of the architecture of the mechanism as follows

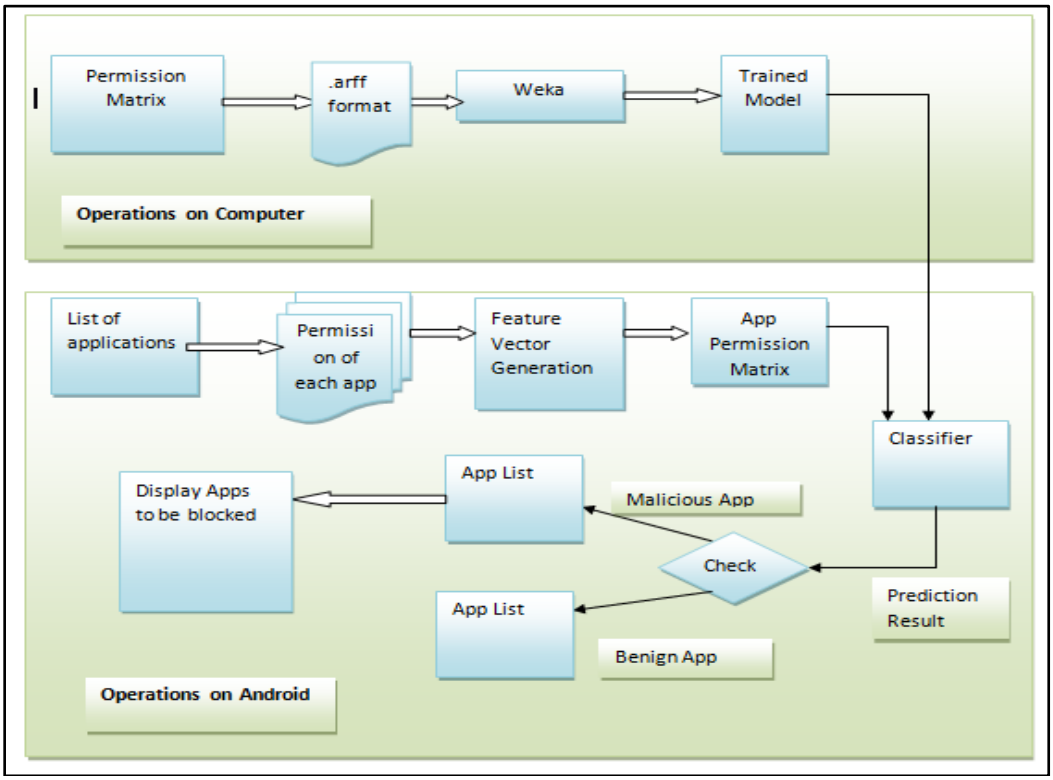

Fig. 4: Architecture of the Countermeasure Mechanism

Algorithm used to classify apps in system is Random Forest. It is a method in which an additional layer of randomness is added to bagging. In addition to constructing each tree using a different bootstrap sample

of the data, random forests change how the classification or regression trees are constructed. In a random forest, each node is split using the best among a subset of predictors randomly chosen at that node. This somewhat counterintuitive strategy turns out to perform very well compared to many other classifiers. These classifiers help to detect the malicious and benign applications.

Experimental Work

Various different types of experiments are performed to evaluate the performance of the proposed system for detection of attacks. For analysis and evaluation of algorithms there are some parameters on the basis of which performance of algorithms can be determined and compared. Some of the evaluation parameters are precision, recall and F-score.

Following is the dataset used for the experiments carried out:

A. Dataset

Dataset is important aspect when applying any classification algorithm. If dataset is not sufficiently large it has a drastic effect on the accuracy of classifier. In order to conduct extensive analysis on permission usage, large well-labeled app set needs to be collected. A total of 310,926 free apps from Google's play in June 2013 after removing malapps reported by an antivirus tool and by a free online service named VirusTotal. Two Malicious app sets (named Mal_Com1 and Mal_Com2) from two different antivirus companies were added. Total of 88 distinct permissions are used. Therefore, each app can be represented by a 88-dimensional Boolean vector, where 1 denotes that the app requests the permission and 0 otherwise. Each app contains some of the permissions from them.

Table 2: Statistics of Benign and Malicious apps permission matrix [2]

Sr. No	Permission_matrix Name	No. of Records
1	permmission_matrix_benign_google_apps	310926
2	permission_matrix_malicious_zhou	1026
3	permmission_matrix_malicious_com1	247
4	permmission_matrix_malicious_com2	154
5	permmission_matrix_malicious_VS	3207

B. Normalization of Dataset Using Permission

Total no. of apps considered are 10389 which includes malicious as well as benign apps. If the value of permission is one, it is assigned to that particular app where as if it is zero it is not assigned to that app. Value of permission can be either zero or one. The top risky permissions well discriminate malapps from benign ones by the frequency of their appearance.

C. Classification of Apps

Initially the permission matrix is converted into arff file format. An ARFF (Attribute-Relation File Format) file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files were developed by the Machine Learning Project at the Department of Computer Science of the University of Waikato for use with the Weka machine learning software.The ARFF file is feed to the Weka tool. The Weka tool outputs model of the Random Forest Classifier. This model is then feed to the Android version of weka i.e. weka-stripped.jar which synchronize the trained model and its instance can be used to classify instances of test dataset.

D. Generation of App List

After the results obtained from the SerializedClassifier instance of RandomForest model, Apps from App List is are categories as Malicious (0) or Benign (1). The Suspicious App List (predicted as malicious

app) is added to the ListView and user is provided with a block option to block it. Table 3 shows accuracy of different classifiers. Here, proposed algorithm Random Forest is compared with other three algorithms SVM, Naïve Bayes and J48.It proves that accuracy of Random Forest is better than other three algorithms. Proposed algorithm outperforms other three algorithms.

Table 3: Accuracy of Various Classifiers

Classifier	Accuracy (%)
SVM	95.12
Naïve Bayes	90.31
J48	96.31
Random Forest	97.20

Result and Discussions

This chapter discusses various results found after the experimentation done on various types of permissions taken by the white listed applications as well as the blacklisted applications. Camera attacks basically use the camera of the android device which reflects the battery usage of the phone. In the counter measure suggested above the application uses minimum battery that is shown in the following graph. The attacks are only possible when the battery of the device is above 20%. So the check on the battery usage can be maintained throughout the application. The results are evaluated and compared with the proposed Random Forest approach of App classification with existing machine learning algorithms. The graph of Precision and recall against different classification algorithms is shown in Figure no.5. It shows that the proposed method Random Forest for App classification obtained higher Precision and Recall.

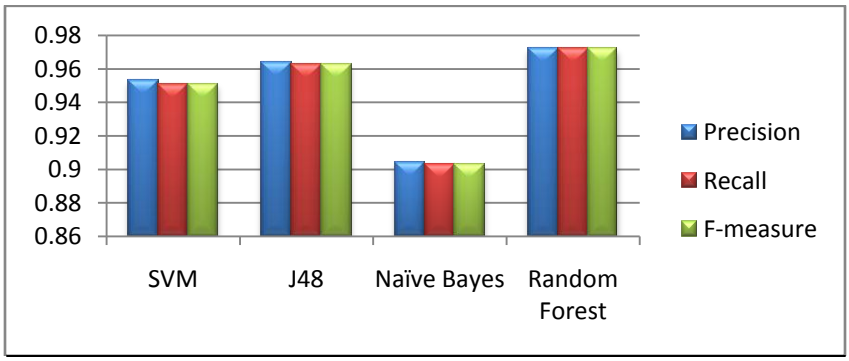

Fig. 5: Comparison of Parameters

ROC stands for Receiver Operating Characteristic. It is a technique for visualizing, organizing and selecting classifiers based on their performance. ROC graphs were used earlier for single detection theory to depict the tradeoff between hit rates and false alarm rate of classifiers. It has been extended for use in visualizing and analyzing the behavior of diagnostic system.

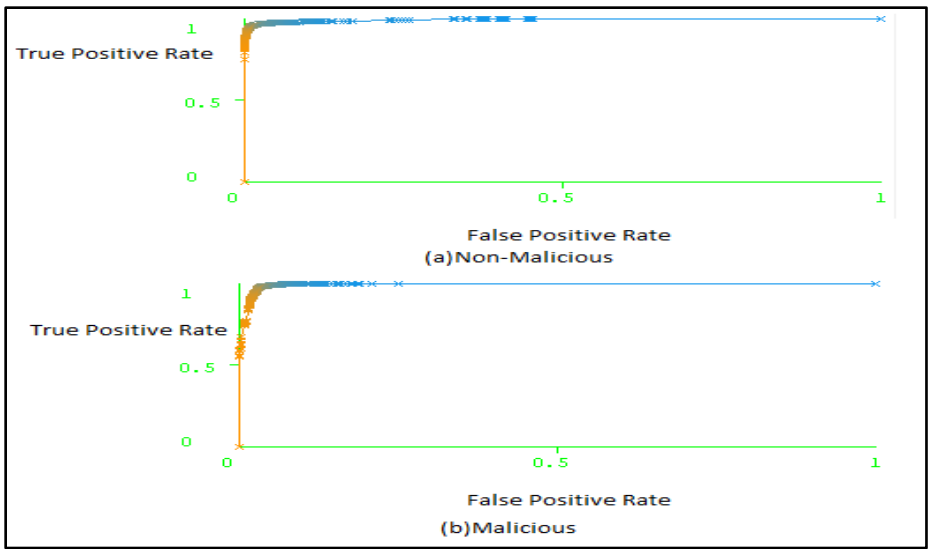

Fig. 6: ROC of Random Forest

The lower left point (0, 0) represents the strategy of never issuing a positive classification; such a classifier commits no false positive errors but also gains no true positives. The opposite strategy, of unconditionally issuing positive classifications, is represented by the upper right point (1, 1). The point (0, 1) represents perfect classification

Conclusion and Future Work

Spywares such as risk wares perform modifications in the normal functioning of the system or device and helps it to behave in an abnormal way. Phone users are mostly unaware of the fact that applications may be spying on the user. In this work, camera related attacks are demonstrated that may cause security threat to the user's privacy. The techniques used to perform camera attacks are based on the basic architecture of camera attack and a light weight defense mechanism is devised.

The countermeasure mechanism is dependent on the permission list available in the android. Another approach to do the same is by using the application set. The dataset used is the latest set of benign and malicious application available in bulk. Various types of classifiers are compared with each other in order to gain accuracy. Random forest strategy is implemented to counterattack camera vulnerabilities. This strategy proves to be more accurate when compared to others like naïve bayes, libSVM. User intervention is needed to block the applications which are predicted to be malicious. This proves to be effective as it alerts the user about the malicious applications present in the android device.

While the permission requests characterize the behaviors of apps to certain extent and the detection can be effective, only considering the permissions.The empirical results of the study indicates that risky permissions can be effective for the detection of blacklisted or malapps. In the future, it can investigate the feasibility of performing spy camera attacks on other mobile operating systems like Symbian OS and IOS. Also two types of techniques such as permission matrix and application list can be combined to obtain more accurate predictions.

References

[1]. Longfei Wu and Xiaojiang Du, Temple University Xinwen Fu, University of Massachusetts Lowell, "Security Threats to Mobile Multimedia Applications: CameraBased Attacks on Mobile Phones," IEEE Communications Magazine, March 2014

[2]. Wei Wang, Xing Wang, DaweiFeng, Jiqiang Liu, Zhen Han, and Xiangliang Zhang "Exploring Permission-induced Risk in Android Applications for Malicious Application Detection" IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY,Volume: 9, Issue: 11, Nov. 2014 , pp.1869 – 1882

[3]. VaibhavRastogi, Yan Chen, and Xuxian Jiang "DroidChameleon: Evaluating Android Anti-malware against Transformation Attacks", ACM, May2013.

[4]. Charlie Miller "Mobile Attacks and Defense" in IEEE computer and reliability societies,Volume 9, Issue: 4 July/August 2011,pp.68-70.

[5]. J. Bickford, R. O'Hare, A. Baliga, V. Ganapathy and L. Iftode, 'Rootkits on smart phones',Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications - HotMobile '10, 2010.

[6]. Bright Talk "Fire Eye", Online (Available) Internet: https://www2.fireeye.com/WBNR14Q4SidewinderAndroid.html (Accessed On: 15th May 2016).

[7]. D Stites and A Tadimla, A Survey of Mobile Device Security, Threats, Vulnerabilities and Defences,http//afewguyscoding.com/2011/12/survey-mobile-device-security-threats vulnerabilities-defences, 2011.

[8]. F Maggi, S Gasparini and G Borachhi, "A Fast Eavesdropping Attack against Touch screens", 7th International Conference on Information, Assurance and Security, 2011, pp. 320–325.

[9]. Trend Micro, Online (Available) Internet: http://www.trendmicro.com/us/home/index.html (Accessed On: 26th May 2016).

[10] Android manifest," http://developer.android.com/guide/topics/manifest/permission-element.html, (Accessed On: 12th June 2014).

[11] Andy Liaw and Matthew Wiener, "Classification and Regression by Random Forest", R News, Vol. 2, Issue No. 3, pp. 18-22, December 2002.