Preventing Malware Attacks on Android Mobile Platform

[1]Prof. Chetan J. Shelke
[1]*Head, Dept of IT*

***Abstract*—**As in recent era of internetworking, Smartphone's show business role in human day to day life. . These Smartphone's have a variety of apps and features but these new budding feature's of these devices give prospect to new malwares & threats. Android is new in use system since its make very stiff to detect and prevent these viruses and malware's attacks by means of some old fashioned mechanisms.
So protection of these Smartphone's be at the present appropriate an issue of researchers. The deficiency of standard security mechanism in Android applications is very reward to hackers. So to overcome these various pitfalls we proposed an smart malicious app detector as a security concern. It uses user feedback report and makes the app malicious free.

Keywords*—Android OS;Smartphone's; Malwares; User feedback; Applications Security.*

I. INTRODUCTION

Mobile malware spreads quicker on Android than on iOS devices for the reason that the people know how to install applications from the App Store only[1].a place illicit by Apple whereas many can install applications from a USB device as well as from the Android Market. Root exploits[1], called as root an Android device or jail breaking an iOS device, are another medium that mobile malware uses to contaminate devices. Such exploits give a user or application super user privilege: the hacker can install all sorts of applications, many of which are either malware as well as have been infected by malware, giving the attacker full freedom to the device remotely without user detection.

II. LITERATURE REVIEW

G A Jacoby implements Battery-based intrusion detection technique be based on inspection and monitor mobiles power utilization and compare them with the normal power utilization pattern to detect anomalies. Schmidt et al take up static analysis on executables to remove their function calls using the read elf command. lasing uses both static & dynamic analyses on Android applications to repeatedly detect mistrustful applications. cloud-based intrusion detection and response architecture Houmansadr The architecture emulate a smartphone in the cloud and uses a substitute to copy all passage between the smartphone and the Internet.
Permission Watcher: create User Awareness of Application Permissions in Mobile Systems. Struse developed an Android application which provide users with awareness information about other applications and allow to check on the permission set established to individual applications.

III. PROPOSED METHODOLOGY

According to the limitations and shortcoming stated in the literature survey it has been found that the in most of the work the security mechanism deploy on smartphone itself rather on mobile network.

The following methods would be used in our project for development of the security provider application on Android.

User Feedback Methods: Android leverages a vast amount of users which are actively using applications and facing issues with these applications. We would develop a feedback model where the users would be able to report malicious applications on mobile network servers and this would define the score of these applications. If the number of reports for a given application are above a certain level, then we would mark the application as a malware application. In future if any other user tries to download and install the same application then security provider would actively scan the application and recommend to the user that this application has a potential security threat.

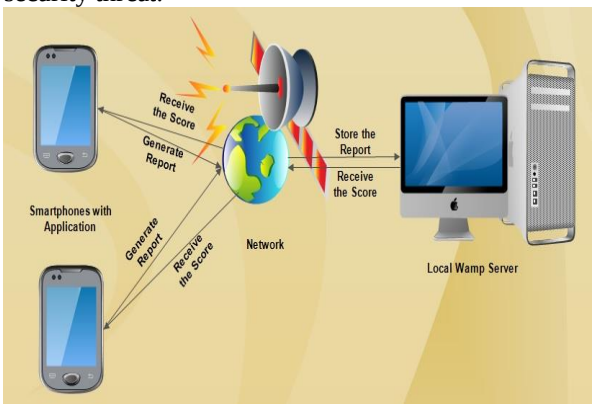

Fig.1:Storing the user feedback on mobile network

Permission Based detection: In Permission based detection permission are extracted from android manifest xml database is created which contain permission required for malicious app.system extract the permission and then matched with permission database

Application offline scanning: Each android application is made up of the following components,

A. Activities: The number of screens the application has
B. Services: The number of background processes by these applications
C. Broadcast receivers: The number of event receivers for this application
D. Permissions: The number of components this application has been granted access The user is shown the

permissions before the application is installed, and if the user feels that the application has an unwanted behaviour, the application installation can be cancelled by the user. Most malware affected applications take advantage of the user's negligence and ask for permissions which are not even needed by these applications. Example, a game might ask for permissions to access the messages, call logs and the internet, even though it's normal functionality does not depend upon these parameters. Thus, the user might install the application and it might send all the device's messages and call logs to an unknown server online, which is like spying on the user To avoid it, this work develop an offline application scanner program which would scan the application signatures and show the level of maliciousness for the given application, there by the user decides either to keep the application or to remove it

Online application signature check with assistive user feedback: In this method, online signature database will be developed, and updating it on the user's phone as soon as new entries are added to the database.

Fig.2: Proposed Methodology

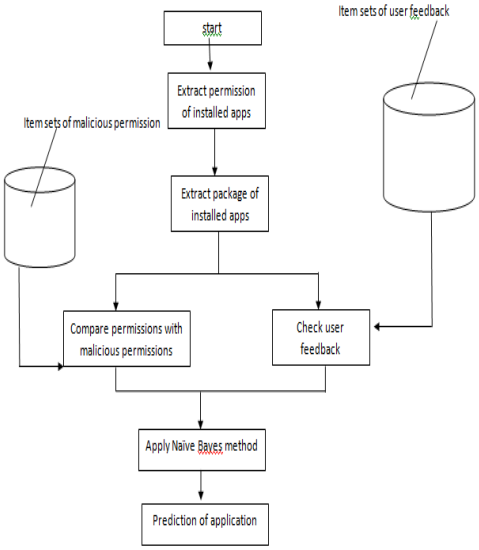

Fig .2: shows the integrated approach of userfeedback and permission based detection

In above figure for getting better result system integreting two approach that is user feedback method and permission based detection first of all system chech for the permission based detection in permission based detection we create two dataset one is malacious dataset and other is non malacious permission dataset if the extracted permission .

If(fmm > fgm)

Where fgm = Permission match factor with genuine matching, fmm = Permission match factor with malicious matching ,

If the above condition is matched then the application is considered to be malicious and it will further check for user feedback method for better accuracy.If the above condition is false then the application is considered to be genuine application.

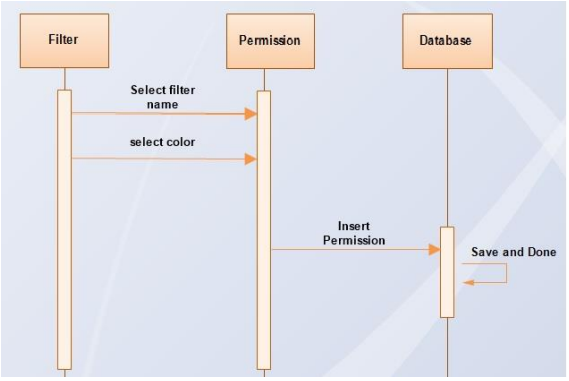

Fig.3: shows Inserting malacious permission on mobile network

If a downloaded application is detected as malacious by both permission and user feedback method then it is categorized as Malicious. If a downloaded application is detected as non malacios by permission-based detection and is detected as malacious by user feedback method or vice-versa then it is suspicious application. If a downloaded application is detected as non malicious by both permission-based and user feedback then it is a benign application.

IV. PERFORMANCE EVALUATION

Performance evaluation of the proposed approach is done based on classification context scenario. Precision, Recall, Accuracy and F-measure plays a major role in classification based performance.

Precision: It is the ratio between the number of relevant apps returned originally and the total number of retrieved apps returned after eliminating irrelevant apps. Here the relevant apps indicate the required documents which satisfy the user needs.

Precision = {Relevant Apps} ∩ {Retrieved Apps} / {Retrieved Apps}

Recall: It is the ratio between the number of relevant Apps returned originally and the total number of relevant Apps returned after eliminating irrelevant Apps.

Recall = {Relevant Apps} ∩ {Retrieved Apps} / {Relevant Apps}

F-measure: It is a measure of a test's accuracy and is the harmonic mean of precision and recall. It reaches its best value at 1 and worst score at 0.

F-measure = 2*Precision *Recall / Precision + Recall

Accuracy: Accuracy is the measure which matches the actual value of the quantity being measured. Accuracy = Relevant Apps / Total Apps

Install App for User 1

Extraction of package name and signature

Package Name	Signature Found
Com.sec.android.app.factorykeystring	58616037
Com.sec.android.app.samsungapps	115777466
Com.sec.android.app.emergencymode.service	146333291
Com.sec.android.configupdater	205356488
Com.sec.android.app.wlantest	53587553
Com.microsoft.office.excel	256834950

Com.sec.android.app.billing	25491015
Com.sec.android.app.minimode.res	177254004
Com.sec.android.app.daemonapp	260926482
com.sec.ims	235974627
Com.sec.enterprise.knox.attestation	256130528
Com.android.vending	103336089
Com.android.pacprocessor	198326110
Com.dsi.ant.service.socket	102000447
Com.sec.android.app.popupuireceiver	132077580
Com.sec.android.autoPreconfig	204756309
Com.sec.android.app.voicenote	55028804
Com.sec.android.app.easylauncher	211370285
Com.samsung.knox.rcp.components	82486882
Com.monotype.android.font.foundation	140960755
Com.sec.android.widegetapp.easymodecontactsswidget	238373040
Com.samsung.android.email.provider	16106281
Com.samsung.android.intelligenceservice2	80791214
Com.samsung.android.commucationservice	127928841
Com.samsung.smt	237581326
android	104570415
Com.android.conctacts	149871676
Com.samsung.hs20provider	120995269
Com.sec.android.autobackup	3343570
Com.android.location.fused	228407715

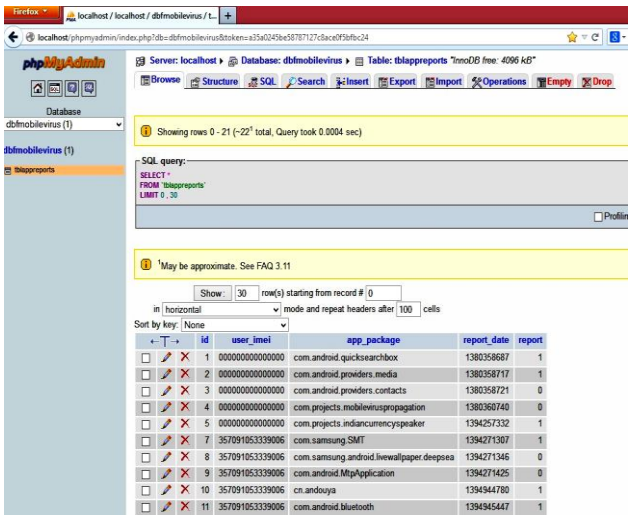

Fig.4: shows the snap of server displaying the content of table.

The sql query is executed for browsing the content of table This table display's the report id , imei number from which the report has received, also the date and time of report in time stamped format and at last the report type

App name	Positive Feedback ration	Negative Feedback Ration	Is it Spy app or malicious
WhatsAp	80%	20%	No
Gmail	85%	15%	No
Xion	40%	60%	Yes
Face book	78%	22%	No
Ruing	45%	55%	Yes

Fig. 3: Result Analysis as per user feedback method

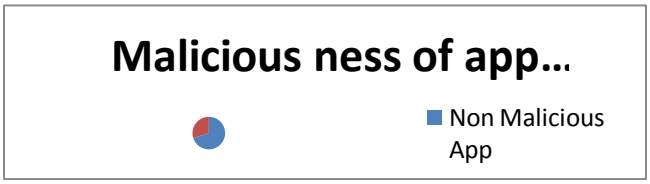

Fig .4: Result analysis as per permission based detection

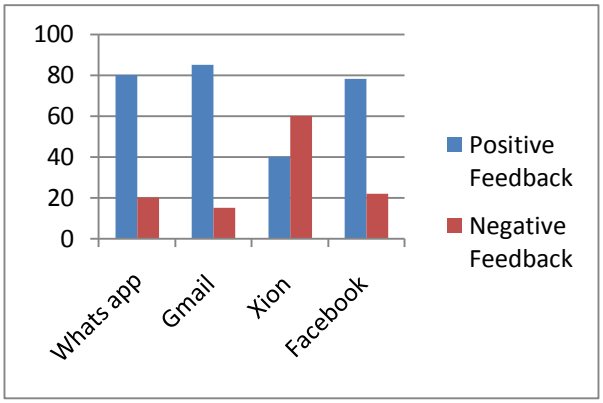

Fig .5: shows the feedback given by different user+

V. CONCLUSION

Proposed an android application framework for analysing permissions of android applications. It uses user feedback report generated by user, to check whether the application is malicious or not .proposed framework is an integrated approach of permission based detection and user feedback method for enhancing the result also to minimize the false positive result. A prototype, implementing the architecture on both the smartphone as well as inside the mobile network, is used to show the feasibility of the proposed architecture, and to demonstrate shortcoming of the approach. Hence the research work will emphasis security mechanism on to mobile network as well as on smartphone to prevent and detect attacks on smartphone .

This work introduces an architecture for finding malicious software on smartphones. The architecture utilizes on the advantages of the mobile network and offers smartphone user's the possibility to check their smartphone without doing changes to the physical device. Main design goals for the architecture were the possibility for easy exchange and expandability of the detection concept and the centralized appliance of the security scanning. as well as on smartphone to prevent and detect attacks on smartphone .

REFERENCES

[1] Qing Li, Greg Clark, "Mobile Security: A Look Ahead", *IEEE Security & Privacy*, vol.11, no. 1, pp. 78-81, Jan.-Feb. 2013, doi:10.1109/MSP.2013.15.

[2] G. Lawton. "Is it finally time to worry about mobile malware?", *Computer*, 41(5):12–14, 2008.

[3] Google Mobile Blog, "An Update on Android Market Security", March 2011. [Online].Available:http://googlemobile.blogspot.com/2011 /03/update-on-android-market-security.html

[4] D. K. Goldhammer, D. A. Wiegand, D. Becker, and M. Schmid. Goldmedia mobile life report 2012, "mobile life in the 21st century, status quo outlook",http://www.b\itkom.org/60376.aspx?url=081009_bitkom_goldmedia_mobile_life_2012(1).pdf.[Online; accessed 01-May-2010].

[5] M. Becher, F. Freiling, and B. Leider. "On the effort to create smartphone worms in windows mobile", *Information Assurance and Security Workshop, 2007. IAW '07. IEEESMC*, pages 199–206, 20-22 June 2007.

[6] C. Mulliner. Advanced attacks against pocketpc phones. 2006. [Online; accessed 04-Sep-2013].

[7] C. Mulliner. Exploiting symbian: "Symbian exploitation and shellcode development", http://mulliner.org/symbian/feed/CollinMulliner_Exploiting_ Symbian_BlackHat_ Japan_2008.pdf, 2008. Talk on BlackHat Japan 2008, visited 15.6.2009..

[8] M. Becher, F. C. Freiling, J. Hoffmann, T. Holz, S. Uellenbeck, and C.Wolf, "Mobile Security Catching Up? Revealing the Nuts and Bolts of the Security of Mobile Devices" ,in *Proceedings IEEE Security and Privacy*, May 2011. [Online; accessed 04-Sep-2013].

[9] J. Bergeron, M. Debbabi, J. Desharnais, M. M. Erhioui, Y. Lavoie, and N. Tawbi. "Static detection of malicious code in executable programs", In *Proceedings of the Symposium on Requirements Engineering for Information Security (SREIS'01)*, 2001.

[10] M. A. Bishop. "The Art and Science of Computer Security", *Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA*, 2002.

[11] Jerry Cheng, Starsky H.Y. Wong, Hao Yang, and Songwu Lu. Smartsiren: "malware detection and alert for smartphones", In *Proceedings of the 5th international conference on Mobile systems, applications and services, MobiSys '07, pages 258–271, New York, NY, USA, 2007. ACM*

[12] C. Mulliner and G. Vigna. "Vulnerability analysis of mms user agents",In *ACSAC '06: Proceedings of the 22nd Annual Computer Security Applications Conference on Annual Computer Security Applications Conference*, pages 77–88, Washington, DC, USA, 2006. IEEE Computer Society.

[13] D. Mutz, W. K. Robertson, G. Vigna, and R. A. Kemmerer. "Exploiting execution context for the detection of anomalous system calls", In *RAID*, pages 1–20, 2007.

[14] J. Oberheide, E. Cooke, and F. Jahanian. Cloudav: "Nversion antimalware in the network mobile network", In *Proceedings of the 17th USENIX Security Symposium (Security'08)*, San Jose, CA, July 2008.

[15] Amir Houmansadr, Saman A. Zonouz, and Robin Berthier. "A cloud-based intrusion detection and response system for mobile phones", *In Proceedings of the 2011 IEEE/IFIP 41st International Conference on Dependable Systems and NetworksWorkshops, DSNW '11, pages 31–32,Washington, DC, USA, 2011*. IEEE Computer Society.

[16] Iker Burguera, Urko Zurutuza, and Simin N. Tehrani." Crowdroid: behavior-based malware detection system for Android", *In Proceedings of the 1st ACM workshop on Security and privacy in smartphones mobile devices, SPSM '11, pages 15–26, New York, NY, USA*, October 2011. ACM.

[17] Adam P. Fuchs, Avik Chaudhuri, and Jeffrey S. Foster. Scandroid: "Automated security certification of android applications*," 2009.*

[18] Sven Bugiel, Lucas Davi, Alexandra Dmitrienko, Thomas Fischer, and Ahmad-Reza Sadeghi. Xmandroid: "A new android evolution to mitigate privilege escalation attacks." *Technical report, Technische Universit¨at Darmstadt*, 2011.

[19] Thomas Bl¨asing, Aubrey-Derrick Schmidt, Leonid Batyuk, Seyit A. Camtepe, and Sahin Albayrak. "An android application sandbox system for suspicious software detection", in *5th International Conference on Malicious and Unwanted Software* (Malware 2010) (MALWARE'2010), Nancy, France, France.