International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-1, November 2019

Anomaly Detection on Android System

D. Usha, V. R. Niveditha, K. Sharadha Priya

Abstract: ***Due to the proliferation of mobile phones in recent years there are many advantages and some disadvantages in smart-phones usage. Many applications like E-commerce, Social media and games get access of user's permission to use some privileges. Normally while installing any application the users grant permission to the particular application to access all the privileges. While installing the app in smart- phones, it has tie-up with some advertisement company. So that to benefit the app developer these ads will run in the app background. This paper describe the advertisements can behave maliciously some times. This malicious behavior extracts personal information like user's contact and files by using the permission access policy of particular app. This data theft may happen without the knowledge of user. The malicious ad will upload the collected information to the desired server. In this paper proposed third party server is used to validate the advertisement before the ad gets loaded into the app.***

Keywords: Adcapsule, app, Adagency, App user, Android

I. INTRODUCTION

In the modern era mobile phones plays a major role in our life. The main reason is that the providers are offering various applications to the users. For example, Google Play Store is offering 2,600,000 applications. As many as 90 percent of these devices are available. Users are free to use these features. App designers have opportunities to incorporate one or many ad libraries into their software and earn payments to make revenue. To make ease, ad service suppliers provide SDKs or libraries to app developers. Such libraries were implemented with little effort by the App developers and also by adding some lines of code.

The ad libraries then interact with the ad network and collect ad content. At the time of this process, the user have the tendency to propose Ad Capsule, a user level resolution to limit ads in golem applications. The resolution of the user does not have to amend the golem framework, and it does not need the foundation privilege and hence it may be promptly deployed. It uses two sandboxes, the sandbox authorization, and the sandbox folder. The sandbox access isolates permissions from those used by the host device used by ad libraries. In this case, ad libraries are operating with their own permissions and contexts within an independent sandbox, so it cannot misuse the host app's permissions to carry out disruptive operations. The file sandbox distinguishes ad library and ad content related directory operations. In a sandbox, the whole file read and write processes are limited. Outside this sandbox users can't bit any folder. Thereby, the system make sure that advertisement libraries and advertisement material cannot extract any non-public information directly or indirectly in accessing host device files.

Experts have therefore suggested multiple ways to address libraries security and privacy issues. There are two representatives of AdDroid and AdSplit. AdDroid is a privilege-specific Android platform ad framework. It introduced new ad library APIs and corresponding Android application permissions. In doing so, ad libraries ' privileged operations are performed in a device process and are completely separated from the host program. In order to follow the method, it is necessary to change the ad libraries software with ad libraries and the basic Android application. Adsplit into another framework with different user authentication on the other segregates advertisement libraries. Advertisement libraries and host attach are two separate applications, making it possible to use the feature of the Android UID framework to distinguish them from one another. One drawback, however, is that the Android code adjustment impedes the realistic implementation of these devices. In order to adopt these systems, users have to flash the custom ROMS that is impossible for normal users. Even if mobile manufacturers choose to incorporate these technologies, the well-known Android fragmentation issue makes integrating different mobile variants and Software updates a time-consuming operation. PEDAL leverages byte code redrafting method in the agreement to recreate ad databases and add security measures to the security-related APIs. There is no need to adjust the advertisement library or the basic Android structure, so it is not hard to deploy. However, because of its nature, this method is not a perfect solution and can be ignored. Because of its design, the initial static code rewrite can be avoided. First, rewriting static code could be through the use of adaptive code execution technique in ad libraries, or through the JavaScript interface exported. Existing ad libraries have shown this. Next, malicious ads could infer sensitive information by reading the host app's private files or publicly accessible SD card files, without appealing any privacy-related API. Therefore, they need a realistic and comprehensive solution to restrict ads, including media libraries and marketing material.

In this research, a user-level approach for confining advertisements throughout Android apps is introduced for AdCapsule.

Revised Manuscript Received on November 30, 2019.
***** Correspondence Author
D. Usha*, Department of Computer Science and Engineering, Dr. M.G.R. Educational and Research Institute, Chennai, India. Email: usha.cse@drmgrdu.ac.in
V. R. Niveditha, Department of Computer Science and Engineering, Dr. M.G.R. Educational and Research Institute, Chennai, India. Email: vrniveditha@gmail.com
K. Sharadha Priya, Department of Computer Science and Engineering, Dr. M.G.R. Educational and Research Institute, Chennai, India. Email: ksharadhapriya@gmail.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

The workaround doesn't have to modify the Android code and it doesn't need the root rights and can therefore be implemented quickly. In addition, it leverages for this function two sandboxes, the authorization sandbox and the folder sandbox. The access sandbox isolates privileges from those used by the host device used by ad libraries. In this case, advertisement libraries operate within an independent sandbox with their own authorization and environments, so the host app's privileges for disruptive operations cannot be manipulated. The file sandbox divides data library and ad content based directory operations. All operations of file reading and writing are confined within a sandbox. Outside this sandbox they can't touch any file. The proposed system guarantees that ad libraries and ad material cannot extract any private information directly or indirectly when accessing host device archives.

The model of AdCapsule has been incorporated. Execution is focused on three main strategies to efficiently control authorization-related and file-related activities which involve GOT hooking, binder hooking, and in-VM API hooking. In general, AdCapsule suggests a technique called binder hooking to intercept the privacy-related APIs that influences Java dynamic proxy to efficiently hook Android system APIs and enforce security measures. Compared with frameworks that use byte code rewrite to interrupt application APIs, AdCapsule offers a reliable and efficient approach to target such APIs, even when code is obfuscated, Java reflection, or complex software execution. AdCapsule then influences in-VM API hooking to pass matching APIs to our own application in order to normalize file-related activities. Furthermore, it works in ad libraries for the APIs, but it cannot work for ad content that accesses local files via WebView module and JavaScript GUI. AdCapsule also hooks the WebView component's GOT table to normalize the access of rich media ads to files. Following the interception of privacy-related and file-activities APIs, AdCapsuletries to enforce safety procedures to authorize or reject such entry, or in certain cases have false values.

II. RELATED WORKS

A. Privacy Concerns of Mobile Advertisements

In recent years, mobile advertising has raised concerns about security and privacy. AdRisk is the first program in the in-app ad libraries to reveal possible safety and confidentiality problems. The researchers reviewed in March-May 2011 100,000 applications from the official Android Market. We found 100 characteristic in-app ad libraries from these implementations and thereby built a system called AdRisk to recognize possible threats consistently. The results presented that furthermost of the current advertisement libraries actively gathered personal information, including location, phone numbers, call logs, etc. A list of 114,000 applications were obtained by another research and then the embedded ad libraries were retrieved and graded. It is observed that the use of advertisement library privileges has grown, and additional libraries use credentials that pose specific threats to the confidentiality of users. Recently, the arc of digital ads has been extensively analyzed by scientists using machine learning and data mining techniques. Researchers find that ad networks, such as preferences, ages, medical conditions, etc., could access users ' private information through app data collection.

B. Solutions to Mitigate Risks of Mobile Advertisement

Scientists have suggested many ways to reduce the threats posed by ad libraries. One of them is AdDroid. This added new ad-related APIs in the Android system that are supported by the ad service, meaning ad content can be viewed without requiring specific permissions for privacy. AdSplit modifies the Android system by running it in different processes to distinguish the host device and ad libraries. Zhang et. al. introduced a frame that provides authorization, view and output separation to segregate ad libraries from the host application. All these frameworks, however, demand that the Android architecture be updated, which hinders functional implementation. For each Android app, Data-Sluice is a system for managing incoming and outgoing data, including ad libraries and ad information. It can delete unnecessary ads for advertisements or irritating popups, but this can reduce the income for app developers. By detecting attacks via JavaScript, ADSandbox chooses whether a site is malicious or not. Then AdSentry uses a JavaScript shadow engine for advertisements that are not supported by the sandbox. There are also works focused on source that contain ad content and the JavaScript interfaces available. All of them can reduce the threat from ad content, but they left the threat from advertisement libraries.

C. Smartphone Apps Security and Privacy

Apps also induce serious confidentiality and safety concerns in addition to ad libraries. Scientists have suggested different techniques to help Orasses understand the risks involved with smartphone apps. Acquired and described Android malware by the Android Malware Genome Project. It gathered more than 1,200 samples of malware that targeted and described the bulk of known Android malware families at the time from different features. TaintDroid uses the methodology of dynamic information flow to monitor the data flow within the device. This program identified many instances where private information was leaked via smartphone apps. On the other hand, PiOS uses the methodology of static analysis to investigate the applications and identify the possible routes between the software for extracting confidential data to those sent out data.

III. ANALYSIS AND DESIGN

A. System Architecture Overview

The purpose of this project is to limit ads, containing in-app advertisement libraries and marketing content. To this end, recommend to sandbox advertisements a user-level fix that does not need to modify the Android code underlying it. Our system contains explicitly of two separate sandboxes. The initial one is the sandbox authorization, which controls access rights that ad libraries may abuse. This sandbox is meant to control ads' privacy-related operations. Another one is the sandbox file, which offers advertisers with a different directory area. Advertisements are unable to read or write files beyond the sandbox. This sandbox forbids advertisers from accessing the host app's private files explicitly, or from gathering information from user based on assessing the presence of such files. The system architecture of AD capsule are illustrated given below in figure 1.

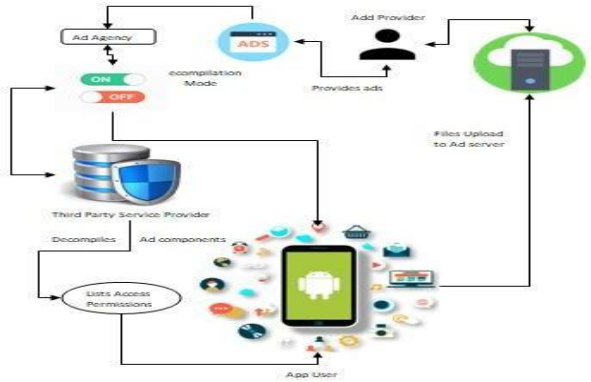

Fig. 1. Ad capsule System Architecture

IV. IMPLEMENTATION AND RESULTS

A. Adding Ad Libraries

The Ad Capsule contains many Ad libraries where the Ad Broker system has to register with the Ad Library. The broker has to get the authorized access to the Ad Libraries by doing required payment procedure. The payment procedure has to be carried on here. After the payment got completed the Ad Broker will have the authorized access to some requested ads. App registration are illustrated in figure 2.

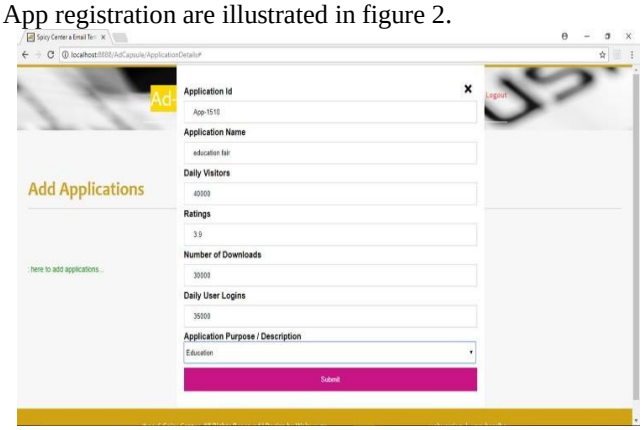

Fig. 2. App registration

B. Third Party Service Provider

The Third Party Service Provider gets the ad privileges from Ad Broker based on the trend are shown infigure.3. So the app developer can earn through the hosted advertisements. The app developer has to segregate some predefined space to allocate the advertisements. The advertisement will broadcast into the specific area where the point of location is specified.

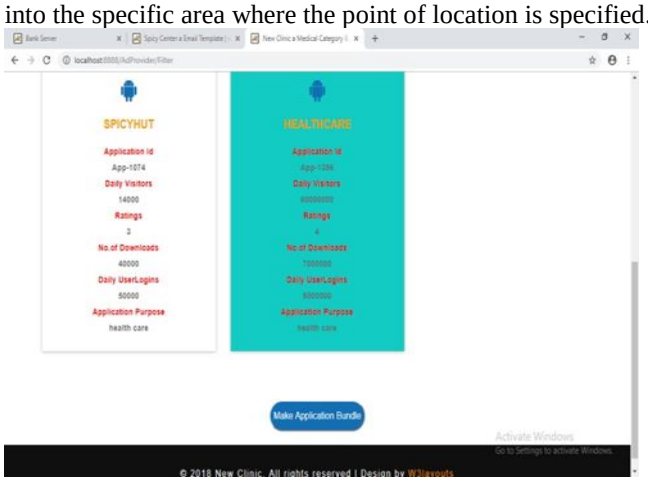

Fig. 3. Bundle Formation

C. Malicious Ad Component:

The Ads in the ad library has to be loaded with running components. This running component will be designed with some functionality that can access the user's internal storage, contacts and some additional privileges. This ad component will be accessing the permissions of the hosted app. Then access the data and periodically uploads the user's sensitive contents to its AdServer are shown in figure 4.

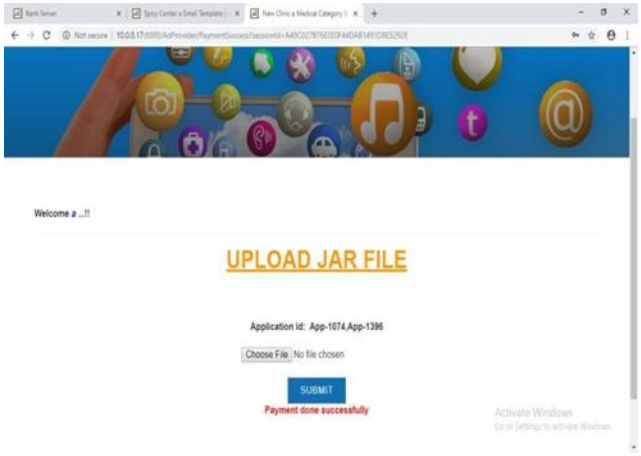

Fig. 4. Uploading malicious component

E. Deciding Access Permissions:

So, whenever a new application is loaded into the empty space allocated for specific ads, the ads get loaded into certain locations inside the host apps. We need to do a preprocessing technique in our proposed Third party Service Provider. So when an advertisement is loaded first into the application the Service provider will check or process all the running components and find out the possible permission access that the ads going to get from the hosted application are illustrated in figure 5. Then the Third party application will send a request to the application user in order to grant permission for authorized access. A popup will be shown to the app user and get the authorized access. Hereby we can restrict the ad components from getting access to user's sensitive data. Thus we can handle the ads from misbehaviors. This is known as confinement of advertisements.

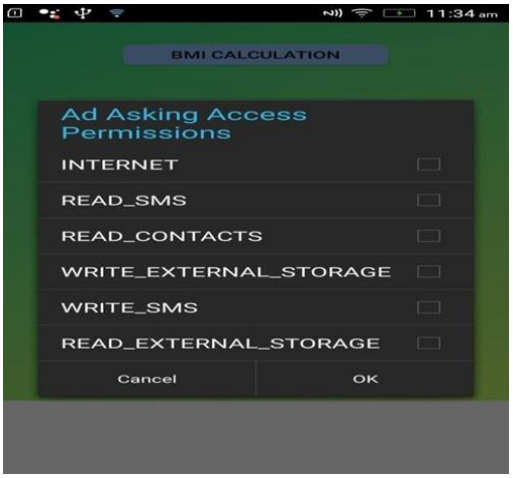

Fig. 5. Getting Access Permission

V. CONCLUSION

The framework suggested a user-level result in this research to restrict advertisements, including advertisement libraries and advertisement content. The program can be implemented easily because it does not need to modify the Android code, nor does it need the root privilege. This uses two sandboxes to restrict security-related activities and commercial folder activities. A prototype has been implemented in the system and the results of the test show the system's reliability and functionality, as well as low overhead performance.

REFERENCES

1. S. Poeplau, Y. Fratantonio, A. Bianchi, C. Kruegel, and G. Vigna, "Execute this! analyzing unsafe and malicious dynamic code loading in android applications.," in NDSS, vol. 14, pp. 23–26,2014
2. M. Grace, W. Zhou, X. Jiang, and A.-R. Sadeghi, "Unsafe Exposure Analysis of Mobile In-App Advertisements," in Proceedings of the 5thACM Conference on Security and Privacy in Wireless and Mobile Networks, ACMWiSec, 2012.
3. Sundaram, P. Sakthi Shunmuga, N. Hari Basker, and L. Natrayan. "Smart Clothes with Bio-Sensors for ECG Monitoring." International Journal of Innovative Technology and Exploring Engineering 8.4 (2019): 298-301.
4. Kumar, M. Senthil, et al. "Processing and characterization of AA2024/Al_2O_3/SiC reinforces hybrid composites using squeeze casting technique." Iran. J. Mater. Sci. Eng 16.2 (2019): 55-67.
5. N. Reddy, J. Jeon, J. Vaughan, T. Millstein, and J. Foster, "Application-centric Security Policies on Unmodified Android," UCLA Computer Science Department, Tech.Rep, 2011.
6. S. Demetriou, W. Merrill, W. Yang, A. Zhang, and C. A. Gunter, "Free for All! Assessing User Data Exposure to Advertising Libraries on Android," in Proceedings of the 23rd Annual Symposium on Network and Distributed System Security, NDSS, 2016.
7. X. Dong, M. Tran, Z. Liang, and X. Jiang, "Adsentry: comprehensive and flexible confinement of javascript based advertisements," in Proceedings of the 27th Annual Computer Security Applications Conference, pp. 297–306, ACM, 2011.
8. Natrayan, L., M. Senthil Kumar, and Mukesh Chaudhari. "Optimization of Squeeze Casting Process Parameters to Investigate the Mechanical Properties of AA6061/Al_2O_3/SiC Hybrid Metal Matrix Composites by Taguchi and Anova Approach." Advanced Engineering Optimization Through Intelligent Techniques. Springer, Singapore, 2020. 393-406
9. S. Son, D. Kim, and V. Shmatikov, "What Mobile Ads Know about Mobile Users," in Proceedings of the 23rd Annual Symposium on Network and Distributed System Security, NDSS, 2016.
10. L. Natrayan and M. Senthil Kumar. Study on Squeeze Casting of Aluminum Matrix Composites-A Review. Advanced Manufacturing and Materials Science. Springer, Cham, 2018. 75-83. (https://doi.org/10.1007/978-3-319-76276-0_8.)
11. A. Zarras, A. Kapravelos, G. Stringhini, T. Holz, C. Kruegel, and G. Vigna, "The dark alleys of Madison avenue: Understanding malicious advertisements," in Proceedings of the 2014 Conference on Internet Measurement Conference, pp. 373–380, ACM, 2014.
12. L.Natrayan et al. Optimization of squeeze cast process parameters on mechanical properties of Al_2O_3/SiC reinforced hybrid metal matrix composites using taguchi technique. Mater. Res. Express; 5: 066516. (DOI: 10.1088/2053-1591/aac873, 2018).
13. T. Book and D. S. Wallach, "An empirical study of mobile ad targeting," CoRR, vol. abs/1502.06577, 2015.
14. L Natrayan, V Sivaprakash, MS Santhosh, Mechanical, Microstructure and wear behavior of the material AA6061 reinforced SiC with different leaf ashes using advanced stir casting method, International Journal of Engineering and Advanced Technology, 2018,8(2S);366-371.
15. L.Natrayan , P. Sakthi shunmuga sundaram, J. Elumalai. Analyzing the Uterine physiological With MMG Signals using SVM, International journal of Pharmaceutical research, 2019, 11(2); 165-170.
16. M. Senthil Kumar et. al, Experimental investigations on mechanical and microstructural properties of Al_2O_3/SiC reinforced hybrid metal matrix composite, IOP Conference Series: Materials Science and Engineering, Volume 402, Number 1, PP 012123.
17. J. Seo, D. Kim, D. Cho, T. Kimy, and I. Shinz, "Flexdroid: Enforcing in-app privilege separation in android," in Proceedings of the 23rd Annual Symposium on Network and Distributed System Security, 2016.
18. S.Yogeshwaran, R.Prabhu, Natrayan.L, Mechanical Properties of Leaf Ashes Reinforced Aluminum Alloy Metal Matrix Composites, International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 13, 2015. .
19. S. Shekhar, M. Dietz, and D. S.Wallach, "AdSplit: Separating Smartphone Advertising from Applications," in Presented as part of the 21st USENIX Security Symposium, USENIX Security, 2012.
20. X. Zhang, A. Ahlawat, and W. Du, "A Frame: Isolating Advertisements from Mobile Applications in Android," in Proceedings of the 29th Annual Computer Security Applications Conference, ACM ACSAC, 2013.

AUTHORS PROFILE

Dr. D. Usha is currently working as Associate Professor in Department of Computer Science and Engineering, Dr. M.G.R. Educational and Research Institute, Chennai. She has 12.4 years of teaching experience. She completed her Doctorate of Philosophy in the year 2017 from Hindustan University. She has published more than 20 papers in various Conferences and Journals. Her area of research is Data Analytics and Data Mining. She is a member in IEEE and CSI. She is also playing the role of reviewer in International Journals.

Ms. V. R. Niveditha has completed B.E in Comp. Sci. Engineering in PB college of Engineering and M.Tech Information Security and Cyber forensics in Dr. M.G.R Educational and Research Institute. She has also published two papers in International journals and six papers in National conferences.

Ms. K. Sharadha Priya completed her B.Tech CSE from Meenakshi College of Engineering and M.Tech ISCF from Dr. M.G.R. Educational and Research Institute in the year 2019. She has also published paper in the National Conference on Innovative Computing Techniques 2019.