International Journal of Engineering Research and General Science Volume 4, Issue 2, March- April, 2016
ISSN 2091-2730

ANALYSIS OF MALWARES FOR ANDROID APK

Mr. Akash J. Wadate, ME 2nd year, CSE department , G.H.Raisoni COE&M, Amravati, Maharashtra,India,

akashwadate007@gmail.com

Prof. N. R. Chopde,Assistant Professor,CSE Dept.,G.H .Raisoni COE&M,Amravati,Maharashtra,India

Nitin.chopde@raisoni.net

Prof.D.R.Datar, Assistant Professor,CSE Dept.,G.H. Raisoni COE&M,Amravati,Maharashtra,India

Dinesh.datar@raisoni.net

ABSTRACT: The aim of this work is to check whether an android application is having malware or not. Day by day malwares are spreading very fast. Malwares are dangerous to any system. By this work user will be able to verify the presence of malwares in apk file, based on that he can avoid installing it in his mobile and thus can be protected from malware attacks. Here cloud computing is used to provide portability and other features.

Keywords: malwares, apk , cloud computing, android application.

1. Introduction

The increasing ability of attacks to avoid traditional security systems and remain undetectable was a prediction we got right five years ago, but we have seen only the early stages of this phenomenon. Malware is still very popular and growing, but the past year has marked the beginnings of a significant shift toward new threats that are more difficult to detect, including fileless attacks, exploits of remote shell and remote control protocols, encrypted infiltrations, and credential theft[1].

As endpoint, perimeter, and gateway security systems got better at inspecting and convicting malicious executables, attackers moved to other file types. Now they are experimenting with infections that do not use a file. Leveraging vulnerabilities in BIOS, drivers, and other firmware, they are evading defenses by injecting commands straight into memory, or manipulating functions in memory to install an infection or exfiltrate data[1]. These attacks are not easy to execute and are not as interchangeable as some of the most popular malware, so the number of known attacks is currently quite small. However, like other techniques, they will get simpler and commoditized over time, broadening their accessibility and fueling their growth. The security industry is developing active memory protection and scanning technology that detects memory not linked to a specific file, but we expect to see an escalation in this type of attack until these defenses are commonly deployed[1].

The merging of cybercrime and APT has emboldened financially motivated criminals who have gracefully transitioned from attacking end users to going after the financial institutions themselves. The past year has seen plenty of examples of attacks on point-of-sale systems and

ATM s, not to mention the daring Carbanak heist that pilfered hundreds of millions of dollars[2]. In the same vein, we expect cybercriminals to set their sights on novelties like alternate payment systems (ApplePay and AndroidPay) whose increasing rate of adoption should offer a new means of immediate monetization. Another inevitable point of interest is stock exchanges, the true mother lode. While frontal attacks may yield quick payoffs, we mustn't overlook the possibility of more subtle means of interference, such as going after the black-box algorithms employed in high-frequency trading to ensure prolonged gains with a lower likelihood of getting caught[2].

2. Literature Survey

Android is covering most of market today. There are millions of clients are available for android today. That's why attackers are targeting towards android more than other platforms. Malwares are the biggest threat to mobile applications as they get directly spread from internet or application stores. There are various reasons behind spread of malwares through application store. Some includes that the one who develops any application is not properly authorized before that application is uploaded in app store. Another reason includes that there are no signature testing included before uploading any application over through application market. Simply any attacker can develop an application and can upload it, where because there is no proper security checks are performed ,it get directly deployed in market and any normal user only see the fake purpose of whose from end shows different thing and at the back side it does malicious work.

Jianlin Xu[3] has worked on this and in his work, based on home-brewed cloud computing platform and data mining, they proposed a methodology to evaluate mobile apps for improving current security status of mobile apps, MobSafe, a demo and prototype system, is also proposed to identify the mobile app's virulence or benignancy. MobSafe combines the dynamic and static analysis methods to comprehensively evaluate an Android app, and reduce the total analyse time to an acceptable level. In the implementation, they adopted the two representatives

dynamic and static analysis methods, i.e. Android Security Evaluation Framework (ASEF) and Static Android Analysis Framework (SAAF) to evaluate the Android apps and estimate the total time needed to evaluate all the apps stored in one mobile app market, which provide useful reference for a mobile app market owner to filter out the mobile malwares[3].

3. Proposed Work

In this work an analysis system is proposed which will scan th given apk first, give its analysis report and based on that user can install it or can avoid it.For analysis it uses two techniques SAAF and ASEF, which are two framework . After applying this over the application, the results are taken out in form of percentages. There are three categories of analysis malware , spyware ad safe. The user will be given output divided into three categories and highest percentage result will be considered as an output , based on what user can accept or reject that particular application. The analysis application is basically divided into two phases-training and testing.The flowchart of training phase is given as follows.

By this training part, the developer will upload an apk file to record the byte codes of that apk for doing further analysis. The procedure of selecting that that particular apk for reference is omitted here as that work include the process of installing the newly launched apk over various devices and doing observation over number of days.

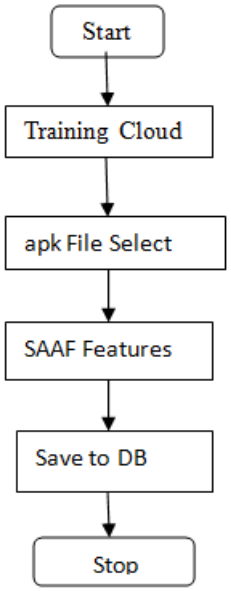

Fig. 1: Training phase

The testing phase of the system is shown in the fig.2, where apk file will be selected for analysis. From that apk again with the application of SAAF , features are extracted then are compared with the stored pattern of apk's , ASEF analysis is performed over that and the data mining technique Top-K rules are applied over that in order to get the best matching features. Finally the result is displayed. The time needed to do the analysis is also displayed in training and testing phase. This procedure is also depicted with the percentage analysis, the percentage of each category of type is displayed i.e. as in training there are following categories – malware, spyware , safe , so the percentage of these categories are displayed. Four kinds of output get displayed: safe or unsafe status, time analysis, path of that selected apk, percentage analysis.

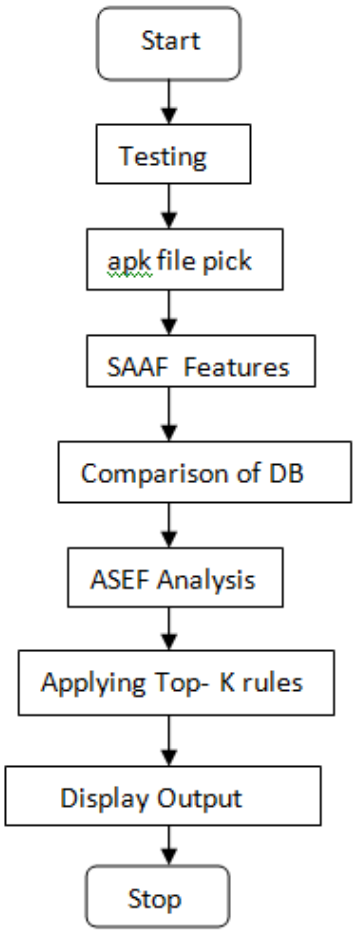

Fig. 2: Testing phase

4. Conclusion and Future Scope

By this work Security status of the application is analyzed in terms of malware, spyware and safe and the user will be notified and prevented from use of such applications having presence of malware and spyware. Here the total time needed to evaluate a mobile app is estimated. Here the apk is analyzed and percentage of features get displayed based on three types and based on that the one having highest percentage from three types will be declared as the result of analysis.

As the future perspective some web mining and more advanced data mining techniques will be implemented to get more optimized outputs. Machine learning is the issue which will be in the future work of this system. The system can be extended to include the facility of providing the user security by which the user will be notified about the safe and unsafe contacts numbers and generate the report of one user.

REFERENCES:

1. McAfee Labs Report 2016 Threats Predictions, 7 April 2016.
2. Kaspersky Security Bulletin 2015 2016 PREDICTIONS: IT'S THE END OF THE WORLD FOR APTs AS WE KNOW THEM, 10 April 2016.

3. MobSafe: Cloud Computing Based Forensic Analysis for Massive Mobile Applications Using Data Mining Jianlin Xu, Yifan Yu, Zhen Chen, Bin Cao, Wenyu Dong, Yu Guo, and Junwei Cao TSINGHUA SCIENCE AND TECHNOLOGY ISSNll1007-0214ll10/10llpp418-427 Volume 18, Number 4, August 2013