International Journal of Recent Development in Computer Technology & Software Applications
Vol. 1, Issue 1- 2017

REVIEW ON ANDROID SECURITY VULNERABILITIES

SATISH SMORE*, VAISHALI S SARDAR*, PRAMOD T TALOLE*

ABSTRACT

In our day to day life we are more careful about the security. As the day passes, We provide the different security technology to different applications. The security is playing the important role of our life, by using security we protect our data. We provide the best security to the money or important data. Starting from our handheld device to computers to smart appliances, our world is digitized. Thus a smart home would be the next step for a better future. With the popularity of Android smart phones everyone finds it convenient to make transactions through these smart phones. And the users of these smart phones, in most cases unaware of different types of threats. The purpose for this survey paper is to conduct a survey on users to get the information about the security vulnerabilities they are creating unknowingly, bringing forward some security frameworks for these threats & giving a basic knowledge to the new comer to the android about android OS architecture and the threats to this architecture. The security in android based device is based on the permission based security model. Android OS allows each developer to define their access to the resource and OS allows developer to do this. But this leads the vulnerability in OS functionality.

KEYWORDS*:* Android Security Framework, Security Vulnerabilities, Android App Permission, Malware.

INTRODUCTION

A smart phone is a complicated mix of a cell phone and a figuring stage with effective registering framework and fast availability. In the market of advanced mobile phones, android overwhelms the market with 78% offer. Advanced cells have turned out to be basic piece of our everyday lives as of late, since they are engaged with staying in contact with loved ones, working together, getting to the web and different exercises. Andy Rubin, Google's chief of portable stages, has remarked: "There ought to be nothing that clients can access on their work area that they can't be access on their mobile phone" [1]. We are keeping data which are private in nature inside our smart phone for easy access. Since users keep a huge amount of data in our smart phone, the hackers are targeting our smart phones more and more. In this paper a survey is done to see what vulnerabilities occur due to a user's unawareness. Some security frameworks are also discussed which will help to remove these vulnerabilities if these frameworks are adopted in an android phone.

*Department of Information Technology, Anuradha Engineering College, Chikhli.
Correspondence E-mail Id: editor@eurekajournals.com

ANDROID SECURITY ARCHITECTURE

- Android looks to be the most secure and usable working framework for mobiles by giving safety efforts to ensure client information, framework assets and it segregate applications. Google gives following security highlights to accomplish these goals
- Robust security at the working framework level through the Linux bit.
- The OS is sandboxed, keeping pernicious procedures from going between applications
- Secure mediates correspondence.
- User characterized authorizations.
- Application marking

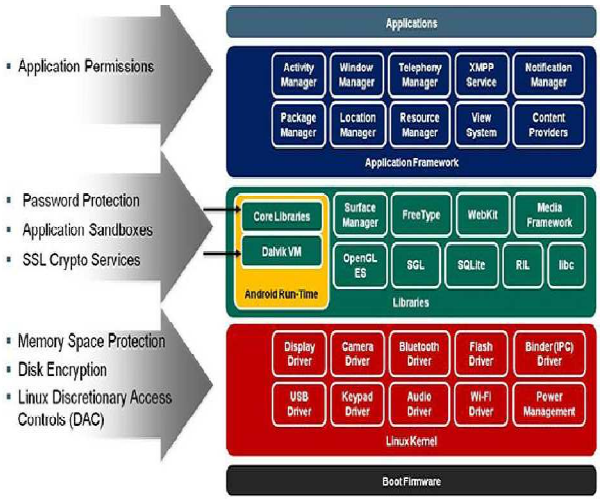

Figure 1.Summarizes security provided at various levels of Android. Every level assumes that level below is properly secured

In the Linux Kernel, the primary reason for memory space assurance is to keep an errand from getting to memory without legitimate access authorizations. Without memory insurance, memory fragment like code and information portion are powerless against memory related bugs and code infusion assaults. Plate encryption guarantees that documents are constantly put away on circle in a scrambled shape. Security Enhanced Linux (SELinux), an entrance control usage for the Linux part which was presented as of late has kept numerous vulnerabilities, and now it has been reinforced considerably more to address the issues of big business clients that have strict security necessities. All procedure keep running over the Linux bit is limited by the application sandbox.

In the Libraries, The Android stage exploits the Linux client based insurance as a methods for recognizing and separating application assets. This approach is not quite the same as other working frameworks (counting the customary Linux design), where various applications keep running with a similar client consents. This sandbox is unique than the sandbox found on the J2ME or Blackberry stages.

ANDROID THREATS

Security breach on this architecture may come in two routes from outside hostile exercises; assault because of client ignorance and assault because of framework surrenders. Most assaults misuse vulnerabilities of the PDA. The dangers we see showing up on portable are rootkits, Trojans, and even botnets. Since new malwares are showing up nearly consistently and it is difficult to supplant portrayed secured gadget with a most recent secured gadget in this pace by a client; client mindfulness can go in an incredible length to stop outside impedances to the present gadget.

SURVEY ON USER AWARENESS

- For research purpose we conducted a survey on a focus group of 20 people while they are using android device. We asked following questions:
- How often you install third party applications (third party means applications which are not from Google play store)?
- Have you rooted your android device at least once?
- How often you connect to an unsecured WI-FI network?
- Do you give your android device remote access to your PC?
- Do you maintain unsecured Bluetooth connection?
- When you separate an outside gadget from your android gadget how regularly you check for infection in the gadget after detachment?
- Do you tap on obscure messages and any spam substance in online networking?
- When you introduce an application do you check every one of the authorizations on the consent list?
- Do you know the danger of outsider destinations?

The synopsis aftereffect of our study on client mindfulness is appeared in the accompanying figures

VULNURBILITIES DUE TO USER UNAWARENESS

- After experiencing the review from center gathering; various vulnerabilities were recognized which are caused because of client ignorance. This vulnerabilities happen a when a client.
- Installs outsider applications.
- Roots a gadget
- Connects to an unsecured WI-FI network& keeping up unsecured Bluetooth association
- Gives remote access to PC
- Connects to SD card and outside gadget
- Clicks on spam messages/sms/mms
- Grants superfluous authorization to an application
- Studying these vulnerabilities, an android gadget is inclined to these following assaults because of client ignorance

SPAM

Spam is for the most part sent in SMS, MMS and email. VoIP and Instant Messaging (IM) have additionally turned out to be basic courses for spamming. It is seen from our study that 25% individuals tap on obscure connections or spam substance from their email or errand person.

MALWARE

Clients are not generally mindful of downloaded applications capacities. Regardless of whether applications have gained express client assent, clients might be unconscious that the applications are executing malevolent code[2].A contextual analysis demonstrates that, in AAMU in the wake of interfacing with the Wi-Fi utilizing Interceptor-NG, in the wake of running the output summon it demonstrated every one of the gadgets with IP delivers that are associated with AAMU WIFI. From that point onward, the application began to gather bundles that is sending and accepting through this WIFI. In most case, this application can gather the data like client name and secret key.

FRINGE INTERFACES ATTACKS

Advanced mobile phones as a rule have numerous fringe interfaces, for example, Wi-Fi, Bluetooth, USB, and so on. While fringe interfaces can expand PDAs correspondence abilities, shockingly, they likewise turn into a mainstream steppingstone for outside assaults. In our review it is seen that the vast majority of the general population don't associate with unsecured Bluetooth association. So it is

conceivable that clients are much mindful about this kind of security chance.

INFORMATION HIJACKING

Allowing superfluous authorization gives aggressors full control over the substance (e.g. photograph display) and data (e.g. area) of a telephone. Private and classified information can without much of a stretch be commandeered by an aggressor.

PERMISSION

Permissions are the rights that a specific application has that allow it to perform certain actions on a device. Examples of these actions include taking pictures, using the GPS, reading contacts, or making phone calls. All applications have their permissions available for users to check; many users do not check the permission properly and thus cyber criminals can exploit user information for their personal gain. In our survey, only 15% people said that they check all permission before installing an app.

THIRD PARTY INSTALLATION

One of the main advantages of android is that as it is an open source operating system, one can easily install applications which do not belong to the google play store from the internet. In 2012, researchers uncovered an increase in the number of malicious domain accounts related to Android apps. From approximately 3,000 domains in January 2012, the number jumped to almost 8,000 by the end of the year. These malicious domains host suspicious .APK files or files containing data needed in Android app installation. Just an example of these malicious apps is the recent fake versions of the popular Candy Crush app with features that can be abused by cyber criminals [3]. By using these features, they can get hold of your important data and aggressively push ads onto your device. In the survey, all the people in our survey keep personal data and private documents in the mobile. We have seen from our survey that most of the people often install app from third party websites.

ROOTING

Establishing an Android telephone basically intends to increase managerial benefits on the framework. For malware distributers, on the off chance that they gained the root power; they could plan malwares to get users private data and accreditation. [4] Researchers from University of California, Berkeley, have picked 6 most famous Android frameworks from 2010 to 2011 to tally the days that root misuses are uncovered, and demonstrated that the percent of time with known root abuse are high. The minimum one is 74%. It implies that the root misuse is uncovered just a single day for every five days. [5] It is seen from our overview that a large portion of the general population even don't have the foggiest idea about the correct importance of "Establishing."

UNSECURED CONNECTION:

We have effectively expressed that dominant part individuals don't keep up unsecured Bluetooth association. The vast majority of them know dangers of android security chiefly originate from the assaults amid information change. Noxious Applications additionally make unapproved activities when Android trading information through the innovation, for example, informing, remote system and near. Filed Communication. 45% People from our review advised that they never interface with unsecured Wi-Fi organize however 40% individuals told that they utilize unsecured system when they think that its important. 55% individuals said that they do give their android telephone remote access to their PC. 9 out of 20 individuals said they don't check for infection in their advanced mobile phones subsequent to disengaging from an outside gadget.

Getting to UNKNOWN CONTENTS:

"Stage dread" is the moniker given to a potential adventure that lives genuinely somewhere inside the Android working framework. Stage trepidation that is utilized to process, play and record sight and sound documents. A portion of the blemishes in android consider remote code execution and can be activated while getting a MMS message, downloading an exceptionally made video document through the program or opening a Web page with inserted sight and sound substance. On a discovering, analysts of FireEye [6], a security organization discovered a case of such endeavor. Clients simply need to tap on the included connection in the email and the noxious.apk (Android Package File) is downloaded. Analysts at Fire Eye experienced HTTP asks for and discovered almost two-dozen URLs serving up the .APK, some hidden as “LabelReader.APK”. As per them this malware isn't completely new. It’s surfaced before and is known for deluding clients into paying for cleanup of other non-existent diseases on their gadget. For whatever length of time that the client pays the expense, the telephone will purportedly stay uninfected with malware.

ADVANTAGES

- The biggest advantage of the software framework is that it reduces the time and energy in developing any software.
- Framework provides a standard working system through which you can develop the desired module of application or complete application instead of developing lower level details.
- Using frameworks, the developers can devote more time in developing the software requirement, not in preparing the environment and tools of application development.
- Framework follow design pattern, so when you see these frameworks you have to follow their coding conventions which makes your code clean and extensible for future purpose.
- Framework separates business logic from user interface making the code clean and extensible.
- Frameworks help you to develop the project rapidly, if you know one framework well then you will never worry about the project deadline.

DISADVANTAGES

- For the amateur client, it is harder to utilize the structure rapidly as it is enormous and complex conceptual and client needs to invest more energy in as sesing the idea, capacity and its uses in building up the program.
- Another disservice is that a bland "one-measure fits-all" does not work so viably for a particular programming. There is have to stretch out system with particular code to create particular programming.
- Frameworks support and improve the effectiveness and profitability of the application advancement however it faces a few issues in some particular area like.
- Artistic drawing, music structure and mechanical CAD Compilers for various programming dialect and target machine
- Financial displaying applications
- Earth framework displaying applications
- Decision supporting framework

CONCLUSION

This paper presented the existing research proposals for removing vulnerabilities caused due to user unawareness. It was found that the prime threat is install time granting access without reading the permission list. Fortunately from API level 23 Google introduced run time permission granting option. But runtime granting permission may be tedious to the user. So whether it would be fruitful is still uncertain. After studying the frameworks in this paper, there is a future scope

to build a new framework for tackling multiple threats to android phone.

REFERENCES

[1]. http://news.bbc.co.uk/2/hi/technology/7266201.stm.

[2]. Choosilp, Wichien, and Yujian Fu. "A Case STUDY OF MALWARE DETECTION AND REMOVAL IN ANDROID APPS."

[3]. http://blog.trendmicro.com/trendlabs-security-intelligence/the-hidden-dangers-in-third-party-app-sites.

[4]. Tse, Daniel, X. Liu, Christopher Nusaputra, B. Hu, Y. Wang, and M. W. Xing. "STRATEGIES IN IMPROVING ANDROID SECURITY." (2014).

[5]. www.acumin.co.uk.

[6]. http://www.konsultek.com/10/cyber-attacks-2/fireeye-discovers-emails-carrying-malware-in-android-devices.

[7]. Fuchs, Adam P., AvikChaudhuri, and Jeffrey S. Foster. "Scandroid: Automated security certification of android applications." *Manuscript, Univ. of Maryland, http://www.cs.umd.edu/avik/projects/scandroidascaa*2, no. 3 (2009).

[8]. Ongtang, Machigar, Stephen McLaughlin, William Enck, and Patrick McDaniel. "Semantically rich application-centric security in Android."*Security and Communication Networks* 5, no. 6 (2012): 658-673.

[9]. Barrera, David, H. GüneşKayacik, Paul C. van Oorschot, and Anil Somayaji. "A methodology for empirical analysis of permission-based security models and its application to android." In *Proceedings of the 17th ACM conference on Computer and communications security*, pp. 73-84. ACM, 2010.

[10]. Felt, Adrienne Porter, Erika Chin, Steve Hanna, Dawn Song, and David Wagner. "Android permissions demystified." In *Proceedings of the 18th ACM conference on Computer and communications security*, pp. 627-638. ACM, 2011.

[11]. Marforio, Claudio, and AurélienFrancillon. *Application collusion attack on the permission-based security model and its implications for modern smartphone systems*. Department of Computer Science, ETH Zurich, 2011.

[12]. Powar, Swapnil, and B. B. Meshram. "Survey on Android Security Framework." *International Journal of Engineering Research and Applications*3, no. 2 (2013): 907-911. p. 240-253. 2012.

