ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 9(32), DOI: 10.17485/ijst/2016/v9i32/100721, August 2016* ISSN (Online) : 0974-5645

Granular Per(M)issions: Questions Unanswered

Akshay Bhardwaj and A. J. Singh

Department of Computer Science, Himachal Pradesh University, Himachal Pradesh, India;
akshay117@gmail.com, aj.hpucs@gmail.com

Abstract

Objectives: In android users do not understand the permissions structure and how apps utilize these permissions. **Methods/Statistical Analysis:** Study of App Ops system in android and finding out ways to check what permission android apps are using and also to check and derive various algorithms to see what levels of access can be granted to permissions. Also to see if an application can be developed so that the user could know what permissions app is using and whether they are required or not? **Findings**: After proposing four algorithms and trying to classify the level of protection access to permissions we have found that this classification yields desired results. The client can know what permissions are being requested by which apps and can therefore be enlightened to its hazards and security risks. Also although this approach is similar to what Google has done with the Marshmallow version howsoever it suffers from shortcomings and is not yet, a complete solution. Thus, our results will only substantiate and help in the creation of a more robust and secure android version. Whether this much power should be given to the naïve user, however is a question that needs some thought. **Application/Improvements**: Our algorithms can be applied to newer versions of android to make it more secure and ensure user privacy.

Keywords: Algorithms, Android, Granular, Permissions, Privacy, Security

1. Introduction

Cell phones and other cell phones have turned out to be dangerously well known for individual and business use as of late. Worldwide cell phone shipments are relied upon to stretch around 1.4 billion in[1], an expansion of 16% in the course of the most recent year. The business sector is right now overwhelmed by the Android working framework, representing 78% of all cell phone shipments all around in the principal quarter in[2]. Android underpins a blasting outsider application market. At the season of composing Google Play (the social Android application vault) included more than 1.5 million applications[3], which have been downloaded more than 150 billion times[4]. Sadly, the development in the Android stage has set off the enthusiasm of questionable application engineers. Android malware harvests information or sends premium SMS messages and other undesirable applications (gray ware) gather intemperate measures of individual data (e.g. for forceful promoting effort). Malware and gray ware have both been found in Google Play, and the measure of new malware is expanding after some time[5]. Google has attempted endeavours in the past to confine malware[6] - with restricted achievement[7] - however Google Play still contains immense measures of problematic code. Android likewise depends on the client to recognize protection or security-obtrusive applications by utilizing an authorization framework. At the point when a client starts the procedure of introducing an application, he is demonstrated the rundown of authorizations that the application demands. This rundown characters all of the telephone assets that the application will have entry to on the off chance that it is introduced. For instance, an application with the SEND_SMS consent can send instant messages, however an application without that authorization can't. In the event that the client is not happy with the application's consent demands, then he can wipe out the establishment. Security infringement can happen notwithstanding when a client stipends access to ensured information (e.g. contact rundown) to

* *Author for correspondence*

a benevolent application. Applications frequently come packaged with noxious publicizing libraries, and kind Android applications tend to demand a greater number of authorizations than required for their expected usefulness[8]. Clients are not demonstrated consents whenever other than establishment.

2. Problem Statement

The cell phone stages' security models delegate clients to settle on security choices while downloading applications from social application vaults. This designation ranges from essentially approving access to some ensured assets, to giving clients a chance to choose whether an application may debilitate his security and protection. The study consequences in[9] are not in accordance with the desire of cell phones' security models for a sensible security mindful client. Interestingly, the overview results propose that clients are not satisfactorily arranged to settle on suitable security choices. Further research by in[10] confirms that cell phone clients require mindfulness preparing particularly custom-made for cell phone security. In[11] examined the Android authorization framework specifically, and state that clients don't comprehend Android consents in their present structure. Their decision is that clients are as of now not all around arranged to settle on educated protection and security choices around introducing applications from Google Play. Concentrates on by in[12] show that Android authorizations neglect to illuminate the dominant part of clients. They found that clients are not ready to associate asset based notices to dangers, can't reason about the nonappearance of consents, and some of them experience cautioning exhaustion. In[13] affirm that as clients are unconscious of the ramifications of the asked for authorizations, consent demands seem, by all accounts, to be insufficient. Low rates of client consideration and perception recommend that critical work is expected to make the Android consent framework broadly usable.

3. Overview

Security is an expansive idea that catches different parts of normal life. Along these lines a few conceptualizations of protection exist. In[14] effectively made a traditional refinement in 1967, separating between different circles of security: isolation, closeness, store and namelessness. A later meaning of protection introduced by in[15] catches the most pertinent measurements that are generally found:

- the privilege to be not to mention (Warren and Brandeis' well known detailing for the privilege to security[16]),
- restricted access to the self (the capacity to shield oneself from undesirable access by others),
- mystery (the camouflage of certain matters from others),
- control over individual data (the capacity to practice control over data around oneself),
- personhood (the insurance of one's identity, distinction and poise), and
- closeness (control over, or restricted access to, one's personal connections or viewpoints of life).

3.1 Data Protection

In[17] take a gander at protection from a data and correspondence innovation point of view. The centre is then essentially on educational measurement of protection: the relationship amongst gathering and scattering of information, innovation, people in general desire of security, and the legitimate and political issues encompassing them. For instance, web clients might be worried to find that a large portion of the sites which they visit gather, store, and potentially share by and by identifiable data about them. In this setting protection can be characterized as the privilege to control the arrival of individual data around oneself, notwithstanding when that information is gathered and put away by an outsider. Understand that security is not the same as information security, despite the fact that the last can help in shielding individual data from unapproved get to and utilize. The privilege to security additionally expresses that individual information is just gathered when vital, that not any more individual information is gathered than required, and that this information ought to just be utilized for the reason for which it was initially gathered. Likewise, individuals have the privilege to view and upgrade individual data held by others about themselves.

3.2 Smartphone Privacy

Cell phones have components of both a cellular telephone and a PC, permitting individuals to talk, content, access individual and work email, skim the Internet, take pictures, make buys and oversee ledgers. They are

getting to be equipped for accomplishing increasingly consistently, with new functionalities, for example, unique finger impression distinguishing proof and contactless installments. Not at all like numerous PCs, cell phones are dependably with us and numerous individuals once in a while turn them off. Therefore clients should know about the sort of data that can be gathered by different elements from their cell phone. Versatile system administrators (MNO's, for example, T-Mobile and Vodafone gather information, for case for charging and execution estimation purposes. In any case, information accumulation by MNO's likewise happens in view of enactment (alluded to as information maintenance), with the goal that legislatures can get to the information for movement investigation and mass observation. Such enactment is right now on hold in Europe. The points of interest of what MNO's are gathering are generally not clear, but rather they may incorporate the accompanying:

_ Date and time for each correspondence:
{ Starting or getting telephone calls;
{ Sending or getting instant messages;
{ Internet access.
_ The initiator or beneficiary of the correspondence;
_ Location.

Be that as it may, information gathering by MNO's is not the essential focus in this proposal. Notwithstanding the information gathered by MNO's, clients ought to likewise know about the conceivable security issues encompassing the accumulation or exposure of:

_ Photos or video tackled their telephone;
_ Text messages and messages sent and got on the gadget;
_ Who called them, who they called, when it was put and to what extent it endured;
_ The contacts put away in their telephone;
_ Passwords;
_ Financial information;
_ What is put away in their telephone's schedule;
_ Their area, age, and sexual orientation.

Organizations, promoters, cybercriminals and even government offices would have an enthusiasm for the information put away in a cell phone. Applications can gather a wide range of information and transmit it to the application designer or offer it to outsider sponsors. Ads from promoting systems running on a few applications may change cell phone settings and take contact and other data without the client's express authorization. A few applications may track your area: area based administrations like Google Maps, Foursquare or Tinder require your area keeping in mind the end goal to work legitimately. Nonetheless, there are applications that needn't bother with area to work however may even now be following it. Applications may likewise be contaminated with malware. Numerous portable applications don't have security strategies, and when they do, they are regularly long and hard to get it.

3.3 Level of Control in Permission Access

The Android authorization framework offers clients a parallel decision: it is possible that they acknowledge the asked for consents and acquire the application, or the establishment is drop. Out of the container Android does not consider fine-grained consent control. The motivation behind why such control is required is on the grounds that numerous applications have an inadequately characterized set of authorizations. There are situations where an application demands a specific consent yet never really utilizes it (so the application is over privileged), which raises security issues. Be that as it may, the client does not know this, and there is no real way to tell if an application is really using every one of the consents it requested. Also, some of Android's consents are excessively coarse. For instance the READ_PHONE_STATE consent is expected to recognize when a call is being gotten, so that the application can quiet its own particular sound. Be that as it may, the same authorization additionally permits the application to gather the telephone's International Mobile Equipment Identity (IMEI). Thirdly, authorizations are likewise asked for on account of auxiliary usefulness that has nothing to do with the primary capacity of the application. Versatile publicizing stages depend on utilizing information like the client's area and the capacity to get to the Internet to serve promotions. In any case, publicists are generally not straightforward about what they gather and how they store and process that information. Versatile publicizing organizations frequently have a huge number of novel records about clients, which can track clients from their telephone to their desktop.

3.4 Application Ops

Application Ops is a system inside Android that empowers tweaking of individual application authorizations.

Application Ops was initially presented by Google in Android 4.3 as a concealed element. With the arrival of Android 4.4 Google made it harder to get to App Ops, yet kept on presenting new upgrades. At last in Android 4.4.2, Google expelled access to App Ops. In any case, there are still different approaches to introduce it, in spite of the fact that root access is required. Application Ops' code is open source, so engineers can coordinate it with their own Android work also. What App Ops does is permit a client to disavow singular authorizations on a for every application premise. At the point when such an application tries to make a framework call to get to something that the client has now illegal (e.g., the camera) then Android will give back a blunder and won't give access to that information or usefulness. In the most pessimistic scenario applications will crash, since they don't anticipate that authorizations will be denied.

4. Methodology

Clients won't not know about or might be confronting challenges in fathoming what the consents model would involve for them. In request to help them out we propose to have a few criteria but a graphical one which could bail them out of this problem, We name it the protection component or rating. The objective of the Privacy Impact is to help clients choose in the event that they need to utilize specific applications. It can serve as an extra property to consider while selecting applications, together with different properties, for example, measure of downloads audits and verbal. The best place to put the security effect is in an application store. Thusly the client can consider the security sway when hunting down applications. For instance, the client needs an electric lamp application, so he enters the pursuit question 'spotlight'. Electric lamp applications ought to oblige access to only the camera streak, so its protection effect ought to be zero. In any case, bunches of spotlight applications ask for extra consents keeping in mind the end goal to recover individual information, which is being sold to different gatherings to make income. In the event that the client could sort his indexed lists in view of the protection sway, it is anything but difficult to choose the one that has minimal effect on their security. Another proposal is to make a security board, as a major aspect of the Settings menu or as a different application, this would be our execution of App Ops, since it as of now contains a protection related element. In view of the protection affect the client can choose whether he needs to turn on specific consents for an application. Thusly the Privacy Impact would work as a trigger to conform application authorizations.

4.1 Outline

The Privacy Impact outline was part into two sections: the specialized configuration and the client interface plan.

4.2 Specialized Configuration

Android's bundle director permits you to recover the rundown of consents an application demands. To ascertain the security sway from these consents, a calculation is required. This calculation could run from an inconsequential estimation to a mind boggling calculation that likewise considers different variables. The accompanying calculations all figure a number somewhere around 0 and 100, however this can be scaled to whatever is required. Here I expect that the higher the number, the more effect the application has on protection. The variables utilized can be recovered from the working framework.

Calculation 1: all authorizations

This is only a minor check of consents. It expect that all authorizations are similarly hurtful to the client's protection, and that less consents utilized is better.

Privacy Impact = (amount Of Requested Permissions/ amount Of Device Permissions) * 100

The issue with this calculation is that by and by, not all consents are possibly hurtful: some permit access to individual information (e.g. READ_CONTACTS), while others don't (SET_WALLPAPER).

Calculation 2: unsafe consents

This calculation just takes a gander at authorizations that are known not hurtful to the client's security. It utilizes the main 40 most hazardous consents as found by in[18]. I expect that every one of those 40 consents are similarly unsafe, so they convey the same weight.

Privacy Impact = 0

For all requested Permissions

In the event that requested Permission is in harmful Permissions

Privacy Impact += 2.5

The issue with this calculation is that consents independent from anyone else are not that unsafe.

An application may have admittance to your contacts, however without a correspondence channel (like the Internet) it won't have the capacity to send them to a server. Typically it is blends of consents that prompt security encroachments.

Calculation 3: perilous blends

The third calculation takes a gander at mixes of authorizations known not unsafe. The accompanying tenets originate from work by in[18], who built up a five stage process for planning security rules.

An application must not have:

- the SET_DEBUG_APP consent mark.
- PHONE_STATE, RECORD_AUDIO, and INTERNET consent marks.
- PROCESS_OUTGOING_CALL, RECORD_AUDIO, and INTERNET consent marks.
- ACCESS_FINE_LOCATION, INTERNET, and RECEIVE_BOOT_COMPLETE consent names.
- ACCESS_COARSE_LOCATION, INTERNET, and RECEIVE_BOOT_COMPLETE consent marks.
- RECEIVE_SMS and WRITE_SMS consent marks.
- SEND_SMS and WRITE_SMS consent marks.
- INSTALL_SHORTCUT and UNINSTALL_SHORTCUT consent marks.
- the SET_PREFERRED_APPLICATION consent mark and get Intents

for the CALL activity string

Privacy Impact = 0

For all dangerous Permission Combinations

In the event that dangerous Permission Combination is in requested Permissions

Privacy Impact += 10

The computation is more refined now: rather than an insignificant consent check, I now take a gander at mixes of authorizations that could be conceivably unsafe.

Calculation 4: hazardous blends, protection

This calculation takes a gander at mixes of authorizations that identify with protection dangers. In[19] made a mapping of information resources, consents and dangers. They examined each conceivable consent blend from a major application set, bringing about sets of authorizations that ensure a benefit and a transmission channel. Their example contains 89 such matches, of which the twenty most continuous ones could be executed:

Channel Access to information

Web ACCESS_NETWORK_STATE

Web WRITE_EXTERNAL_STORAGE

Web READ_EXTERNAL_STORAGE

Web READ_PHONE_STATE

Web ACCESS_WIFI_STATE

Web ACCESS_COARSE_LOCATION

Web ACCESS_FINE_LOCATION

Web GET_ACCOUNTS

Web CAMERA

Web GET_TASKS

Web READ_CONTACTS

Web READ_HISTORY_BOOKMARKS

Web READ_CALL_LOG

Web RECORD_AUDIO

Web READ_LOGS

Web USE_CREDENTIALS

SEND_SMS READ_PHONE_STATE

Web RECEIVE_SMS

SEND_SMS ACCESS_NETWORK_STATE

SEND_SMS WRITE_EXTERNAL_STORAGE

Privacy Impact = 0

For all dangerous Permission Combinations

On the off chance that dangerous Permission Combination is in requested Permissions

Privacy Impact += 5

Again tests have demonstrated that applications like Facebook and Whatsapp score moderately low contrasted with basic or framework applications. The second part the client model is a genuinely clear question and stars or different images as of now been utilized as a part of play store, can be utilized.

4.3 Permission Usage

Consents asked for by applications are just shown at introduce time. After the client acknowledges them, the application has boundless access to those consents, while the client has no idea of how the application is really them. I acquaint a strategy with record authorization use by applications and present it to the client by method for an Android application. This gives clients understanding into how applications are really utilizing the consents they ask for, for instance when setting up a recently introduced application or when the telephone is on standby. Without

diving a lot into the sort of consents and the authorizations model which is not required right now the accompanying should be possible. The configuration can be part up into two sections: the logging of authorization use and displaying it to the client in an application. Specialized configuration the genuine giving of authorizations by the working framework happens inside the Android system: all the more particularly, inside App Ops Service. This administration contains two strategies that handle the allowing of authorizations: start Operation and note Operation. Start Operation plans for operations that will keep going for a more extended time, similar to GPS and vibrator. Finish Operation must be called when the operation is done. Note Operation is only for fleeting operations, such as sending or getting instant message. Both strategies inquire as to whether the asked for operation is permitted, provided that this is true (or not) a technique can be called to log the utilization of a consent. Android gives an inside SQLite database to applications that need to store information.

Such a database can be utilized to log the authorization use to. For this situation, the database can live either inside the Android structure or in the application that uses its information to present to the client. The database can be contained inside the application, to reduce coupling between the two. For presenting the database to the system a Content Provider can be utilized. The system checks for nearness of the Content Provider (in this way, if the application containing it is introduced on the gadget) and makes utilization of its capacities to keep in touch with the basic database. This can bring about a few issues however: authorizations are additionally utilized amid gadget boot up, yet then the Content Provider is not accessible yet. This causes the working framework to get stuck in a boot circle, and an appalling fix is expected to keep it from circling. A more strong arrangement would be to utilize Broadcast Intents for correspondence rather, in light of the fact that they couldn't care less if the recipient is available or not Client interface Other than the system an application that uses the database to exhibit the authorization use to the client can be produced. Upon startup the application demonstrates a rundown of introduced applications that the client can scan, barring framework applications that are not noticeable to or executable by the client. The client can look through this rundown and pick an application whose authorization use he needs to think about. At the point when the application really utilized consents a course of events is shown, containing a diagram of authorization utilization after some time.

5. Conclusion and Future Work

Over the span of this examination Google reported the following adaptation of Android, Android M. One of its new elements is an overhauled authorization framework. Applications will no more request authorizations at introduce time: rather, applications will request consents the first occasion when they require them. For instance, if an application needs access to the mouthpiece sooner or later, a discourse gives the client the choice to either permit or deny the authorization. On the off chance that clients alter their opinion later, they can flip switches for every authorization in an App Ops-like settings menu. Following the time when the shrouded combination of App Ops in Android 4.3, Android clients have been searching for granular application authorization controls. With the new authorization framework, Google needs to put protection and security once again under the control of clients.

Nonetheless, there are some remarkable contrasts between our proposed plan and the new authorization framework. Android engineers have been outlining their applications for the present consent model: applications get every one of the authorizations, or the client does not get the application. Designers could code applications to come up short smoothly when singular consent are denied, yet the App Ops highlight did not permit that. This would bring about entangled blunder messages or even application crashes, something the normal Android client is not set up for. Android Central found in their testing[19] that such cataclysmic disappointment in the updated authorization framework was entirely uncommon. Another distinction is that the modified authorization framework is less granular than App Ops.

The new framework packs existing authorizations into gatherings. The present rundown of gatherings incorporates Calendar, Camera, Contacts, Location, Microphone, Phone, Sensors and SMS. At the point when an application keeps running on Android M (and backings the new framework), it must issue a solicitation to the working framework for a particular consent, and soon thereafter clients are incited to permit access to whatever gathering that authorization has a place. Control is less

granular than with App Ops, where clients could, for instance, permit the perusing and composing of contacts independently. Android Police notes[20] that an essential consent bunch is by all accounts missing: the Internet authorization bunch. Things being what they are, clients will never be requested that permit access to the outside world, and it is unrealistic to deny it. The rationale for this choice depended on the off chance that the greater part of a client's information is secured as a matter of course, there is little motivation to be worried over applications speaking with the outside world. With respect to the particular INTERNET authorization, it is still required for applications that will get to the Internet. Distributed an application without characterizing the authorization in the Android show results in special cases being tossed the first run through an association endeavour is made, and the application could crash. The designer documentation for the Android M review[21] clarifies that consents with the `normal' insurance level are naturally conceded at introduce time and will never require approval from the client: Restricted Permissions Granted at Install Time: When the client introduces or upgrades the application, the framework allows the application all authorizations that the application asks for that fall under PROTECTION NORMAL. For instance, wake up timer and web authorizations fall under PROTECTION NORMAL, so they are naturally conceded at introduce time. This appears to be sensible, since the authorizations fit for interfering with a client's work process are set apart as `dangerous'. In any case, the INTERNET authorization has dependably been delegated perilous, in any event up to this point. The intelligent statement ought to be that the Internet authorization will be downsized for Android M. From an ease of use point of view it bodes well to bar a revocable Internet consent. Nowadays verging on each application appears to need Internet access for something. In the event that setting up a gadget would mean hitting `Allow' for each application, it would turn into a noteworthy irritation. Google has clarified that the new way to deal with consents is an exercise in careful control, where they need the discourse to be appeared to clients just as frequently as is important. A drawback to this methodology is that permitting access to a bit of data naturally implies that it can be conveyed over the Internet, notwithstanding when that is not required. Envision an application that deals with the location book: access to the Contacts authorization gathering is clearly important, however access to the Internet is not as a matter of course required. Clients would need to trust application engineers not to send that information to a server. In any case, the genuine purpose behind excluding a revocable Internet consent is publicizing. An expansive number of free applications on Google Play just depend on the Internet to download and show ads. In the event that each client could kill Internet access, advertisement upheld applications would lose their just technique for adaptation, which would genuinely demoralize engineers. During the time spent blocking notices, clients would likewise unconsciously incapacitate other helpful components like accident reporting and utilization measurements, that are imperative parts of enhancing applications. Android M's new consent framework is not culminate, but rather it offers new open doors for a more secure and more protection mindful environment. The framework is still a work in progress, so it will most likely experience changes before (and after) it hits the business sector. A strategy for disavowing the Internet authorization may emerge later on, or another way to deal with confine network may be taken. Generally custom ROMs and mods will venture up to assume that position, as we have seen some time recently. it is intriguing to examine the impact of enhanced computation calculations. Counting or expelling specific authorizations, allocating diverse weights to blends or utilizing an alternate scale could enhance exactness. Figuring out how to clarify to the client how the genuine estimation is done would likewise be intriguing, Another restriction of the proposed usage is the way that it is not ready to consider the goals of the engineer: are the asked for authorizations required for the usefulness of the application, or are they utilized for various purposes (e.g. following)? It is intriguing to consolidate diverse components in the calculation, other than simply the asked for consents. Thus it may be conceivable to distinguish malware or infections. From an ease of use point of view it would likewise be intriguing to gauge the impact of putting the Privacy Impact in an application store, to check whether clients consider it while selecting applications.

6. References

1. Digitimes Research. Global Smartphone shipments to reach 1.401 billion units. http://www:digitimes:com/news/a20150319PD213:html. Date Accessed: 19/03/2015.

2. Android and iOS Squeeze the Competition, Swelling to 96.3% of the Smartphone Operating System Market for Bot h 4Q14 and CY14, According to IDC. http://www.idc.com/getdoc.jsp?containerId=prUS25450615. Date accessed: 24/02/2015.
3. Statistics and facts about mobile app usage. http://www.statista.com/topics/1002/mobile-app-usage/. Date accessed: 22/01/2016.
4. Android Authority. Google Play Store vs the Apple App Store: by the numbers April 2015. http://www.androidauthority.com/google-play-store-vs-the-apple-app-store-601836/. Date accessed: 20/04/2015.
5. Yahoo Tech. Report: 1 in 5 Android Apps Is Malware.https://www.yahoo.com/tech/report-one-in-five-android-apps-is-malware-117202610899.html. Date accessed: 24/04/2015.
6. Inside Android 4.2's Powerful New Security System. http://www.computerworld.com/article/2473570/android/exclusive--inside-android-4-2-s-powerful-new-security-system.html. Date accessed: 1/11/2012.
7. An Evaluation of the Application ("App") Verification Service in Android 4.2. https://www.csc.ncsu.edu/faculty/jiang/appverify/. Date accessed: 10/12/2012.
8. Felt AP, Chin E, Hanna S, Song D, Wagner D. Android Permissions Demystified. In Proceedings of the 18th ACM conference on Computer and communications security, 2011, 627-638.
9. Mylonas A, Kastania A, Gritzalis D. Delegate the smartphone user? Security awareness in smartphone platforms. *Computers and Security*. 2013 May; 34:47-66.
10. Mylonas A, Gritzalis D, Tsoumas B, Apostolopoulos T. A Qualitative Metrics Vector for the Awareness of Smartphone Security Users. In: Trust, Privacy, and Security in Digital Business. 2013 Aug, 173-184.
11. Kelley PG, Consolvo S, Cranor LF, Jung J, Sadeh N, Wetherall D. A Conundrum of Permissions: Installing Applications on an Android Smartphone. *FC'12 Proceedings of the 16th international conference on Financial Cryptography and Data Security*. 2012 Mar, 68-79.
12. Felt AP, Ha E, Egelman S, Haney A, Chin E, Wagner D. Android Permissions: User Attention, Comprehension, and Behaviour. In: *Proceedings of the Eighth Symposium on Usable Privacy and Security, ACM*. 2012 Jul, 1-14.
13. Benton K, Camp LJ, Garg V. Studying the E_ effectiveness of Android Application Permissions Requests. In: *Pervasive Computing and Communications Workshops (PERCOM Workshops), IEEE International Conference on San Diego, CA, IEEE*. 2013 Mar; 291-296.
14. Android Central. Using App Permissions in Android M. http://www.androidcentral.com/using-app-permissions-android-m. Date Accessed: 14/06/2015.
15. Wang W, Wang X, Feng D, Liu J, Han Z, Zhang XL. Exploring Permission-induced Risk in Android Applications for Malicious Application Detection. *IEEE Transactions on Information Forensics and Security*. 2014 Nov; 9(11):1869-1882.
16. Solove DJ. Conceptualizing Privacy. *California Law Review*. 2002 Jul, 90(4):1-71.
17. Warren SD, Brandeis LD. The Right to Privacy. *Harvard Law Review*. 1890; 4(5):193-220.
18. Hoepman JH, Lieshout MV. Privacy ER. Leukfeldt, W Ph Stol (Eds). Cyber Safety: An Introduction, Eleven International Publishing, The Hague, 2012, 75-87.
19. Mylonas A, Theoharidou M, Gritzalis D. Assessing Privacy Risks in Android: A User-Centric Approach, In: *Risk Assessment and Risk-Driven Testing*. 2013 Nov, 21-37.
20. Android Police. Android M Will Never Ask Users For Permission to use the Internet, and that's Probably Okay. http://www.androidpolice.com/2015/06/06/android-m-will-never-ask-users-for-permission-to-use-the-internet-and-thats-probably-okay/.Date Accessed: 06/06/2015.
21. Android M Developer Preview - Permissions. http://android-developers.blogspot.in/2015/05/android-m-developer-preview-tools.html. Date Accessed: 28/05/2015.