DOI: 10.21917/ijsc.2016.0180

CONTEXT BASED ANDROID APPLICATION ADMINISTRATIVE ACCESS CONTROL (CBAA–AAC) FOR SMART PHONES

S. Sharavanan[1] and R.M. Balajee[2]
[1,2]*Department of Computer Science and Engineering, Annapoorana Engineering College, India*
E-mail: [1]sharavanan33@gmail.com, [2]balajee.rm@gmail.com

Abstract

Android applications in smart phones are generally towards provide greater flexibility and convince for users. Considering the fact that the Android applications are having privilege to access data and resources in mobile after it gets installed (one time permission provided by end user on the time installation), these application may also lead to issues in security for the user data as well as issues relate smart phone with peripheral environment. A practical example for an issue which relates smart phone with peripheral environment can be even an Android smart phone application of a college student use camera resource to capture photos of R&D cell and transfer without user or organization permission. The security of the organization and user should be prevented by providing an adoptable solution. The proposed concept of CBAA-AAC (Context Based Android Application Administrative Access Control) is used to control the privileges of any Android application over a corresponding longitude and latitude by the organization administrator. In this way, administrator is able to block malicious application of every individual smart phone which can have activity towards utilizing services and resources that may affect the security of the organization, such an move is must for assuring security of any organization and educational institutions while they allow users to "bring their own smart phones/mobile devices" into the campus.

Keywords:

Android Application, Administrative Access Control, Smart Phones and Organization Administrator

1. INTRODUCTION

The smart phones using Android OS are getting smarter in day to day life. These smart phones are capable of providing lot of resources like Wi-Fi connectivity, 3G & 4G supports, advanced camera, cloud storage, bunch of sensors, synchronization support to external devices and environment, etc. These features are getting updates on every new release. This provides greater convenience to end users on their frequent mobility. Android application applications are also capable of detecting current user location, transferring any kind of data anywhere, utilizing any resources of smart phone when required after one time user permission. The above mentioned capabilities of Android application can also cause security issues. Few security issues are addressed on this paper are given below.

i) Android application may take secure data in the mobile phone and transfer it to hacker.

ii) Android application may utilize the resources in mobile phone without user knowledge to accomplish its own task.

iii) Users may also involve in suspected activity with the help of their own individual smart phone within the campus of any organization.

The organization can take control over the user's smart phone when organization is superior to user on a particular location. It is also made sure user should hold on control over Android application based on location to access any data and available resources in a smart phone.

In this paper, section 2 concentrated with the analysis of existing work and mechanism. The proposed work is carried out under section 3 and section 4 is followed with results and graphs of the proposed work. The paper concluded on section 5.

2. LITERATURE SURVEY

Many smart phones are inward to the market with enriched features according to their standard and price tag [1]. These resources are also utilizing and generate sensitive data; it exposed the security issues when any Android application misleads it [2]. Few Android applications are malicious [3] [4] [5], which can use the resources of individual smart phone to stolen resources without user knowledge. The organizations may also face the issue of preventing their secured data which can be transferred out even from their employees [6]. Few organizations /institutions are not allowing their employees/students to bring in their smart phones to make their resources and data secure. Most of the solution blocks the specified resources completely to a particular application; this may affect the application performance with nearly half the requirements are utilizable [7]. The restrictions are not made context based is a major consideration. Most of the research focus was given to developing policy system to restrict resources [8]. Even though focus shifted to context based policy restrictions, it is not accurate enough to differentiate nearby places for its job [7] [9] [10]. When system cannot able to identify the current location, then system sets with default location which is already loaded on it [7] [11]. The kernel of Linux will always provide better security than other kernels and Linux kernel is taken as a base to develop Android OS [12]. This extraction of Linux based kernel to mobile Android OS will result in better security provided by the operating system. The context based restriction is based on identifying the exact location of the device. The accuracy measurement on predicting the location of the device / smart phones is depending on the number of towers it gets surrounded with [13]. When GPS system is not available or signal strength is weaker to detect the exact location then Wi-Fi based location prediction can be done with the help of setting up few access points for every 200m [14] [15]. Few study referred also to usability techniques [16] [17] [18] [19] that can be efficiently used to offer end users with preset and adaptable policy configurations.

In smart phone operating system, especially android allow applications to share user ID, when those applications are belong to same developer. When using same used ID, the applications can access same resources with same privilege [20]. The user control access control gadget based on user intent is developed and there user can restrict resources of the device on the time of the usage [21]. In study [22] [23] [24] [25] they focused about

protecting application and users data plotted at the middleware and kernel layers of android operating system.

The study above is stated clear that any of the surveyed paper is not up to the mark to locate the location accurately and they don't allow the administrator to set the policy over the user mobile to restrict the application access mobile phone resources over a particular location where the organization is superior to the user. The above study has not focused on security of organization with group of users.

In this paper we focused on the security of organization against its group of users by allowing the administrator to set the restriction policies on the user's smart phone.

3. PROPOSED SCHEME (CBAA-AAC)

Context Based Android Application Administrative Access Control (CBAA-AAC) is a mechanism for Android system that allows smart phones users to setup policies which are configurable over their applications usage of device resources and services at different contexts. Through by using the CBAA-AAC mechanism admin can restrict privileges for application when using the device at organization (restricted location) and the application on device may re-gain their original status when it is away from restricted location. The user can also specify default set of policies which can be applied when the user is away from organization which is superior to him.

Such policies define which services are offered by the device to a particular application and limit the Android application from user information accessibility. Policy restrictions are linked with context. The context is defined in two parameters which are time and location. Location is figured out basically through visible Wi-Fi fixed access points and their corresponding signal strength values that allow us to differentiate between nearby sub-areas within the name work space, in addition to GPS (Global Positioning System) and co-ordinates of cellular triangulation whenever available.

The CBAA-AAC mechanism is implemented on the Android operating system and includes a tool that captured Wi-Fi parameters. Once the policies are configured on the device the admin can restrict the application privileges according to the context, the policy will be automatically applied whenever the user is within a predefined physical location and time interval.

3.1 LOCATION DETECTION TECHNIQUE

Three sources are used for location detection GPS, Cell Towers and Wi-Fi. When GPS and Cell Towers are not available in indoor environment then Wi-Fi method gets enabled to determine the device location. The captured location is compared with saved data to check whether the captured location is within the specified location.

The policy restrictions of detected locations are applied when it satisfies Eq.(1)

$$\text{Captured location} = \text{Specified Location} \quad (1)$$

The unregistered location based policy restrictions are applied when it satisfies Eq.(2)

$$\text{Captured location} \neq \text{Specified Location} \quad (2)$$

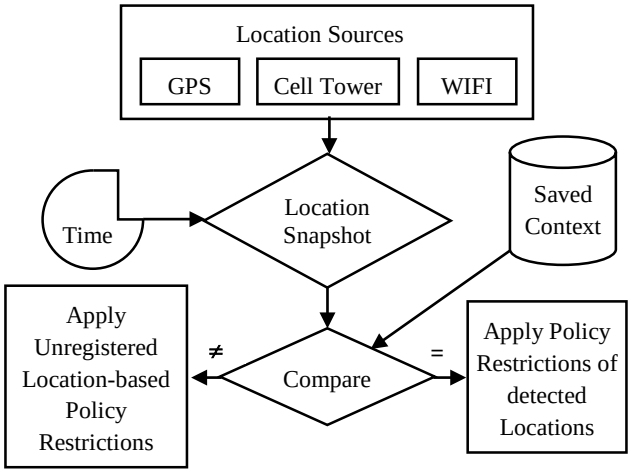

Fig.1. Location Detection Technique

3.2 POSITION CALCULATION USING HYBRID POSITIONING TECHNIQUE

The optimum estimation of mobile user location holds grip on the location-related information offered which is extracted from the following sources

- Measurement of received signal characteristics.
- Collateral information that indicate the relative probability of MS position.

Using relation of Byes probability, the occurrence of relative probability on measurements under condition of a priori state is depicted as,

$$P(X/Z) = [P(Z/X) \times P(X)]/P(Z) \quad (3)$$

X is the state vector of location parameters is a vector set of position measurements and $P(X/Z)$ indicates the probability of the state vector components are evaluated for x under condition that the interpretation have the values of measurement values Z.

Whereas $P(Z/X)$ represents the probability that the values of vector Z would be pragmatic under circumstance that the state variables are of the values in X.

$P(X)$ is the marginal probability on that the state values of X resulted. Whereas, is the entire probability of occurrence of measured parameter values for the observation vector Z.

3.3 BLOCK DIAGRAM

In this System the Network-Admin set policies which are necessary to the institution or organization to restrict student or employee smart phone accessibility. The CBAA-AAC mechanism consists of Network-Admin, set policy, policy manager, policy executor, context provider & access controller. The Policy Executor checks the policies through Policy Manager, when an application request for any resources. The Policies are extracted by Policy Manager from the CBAA-AAC Polices dataset. Once the privileges are generated its gives control to the Access Controller for further action.

3.4 MODULES OF THE PROPOSED SYSTEM

3.4.1 Context Provider:

The Context Provider (CP) gathers the arguments of physical area/location (GPS, Cell IDs, Wi-Fi parameters) through the device sensors and immediately stores them in its private database, relating each physical longitude and latitude to a user-defined logical longitude and latitude. It also clearly verifies and updates those arguments whenever the device changed its location.

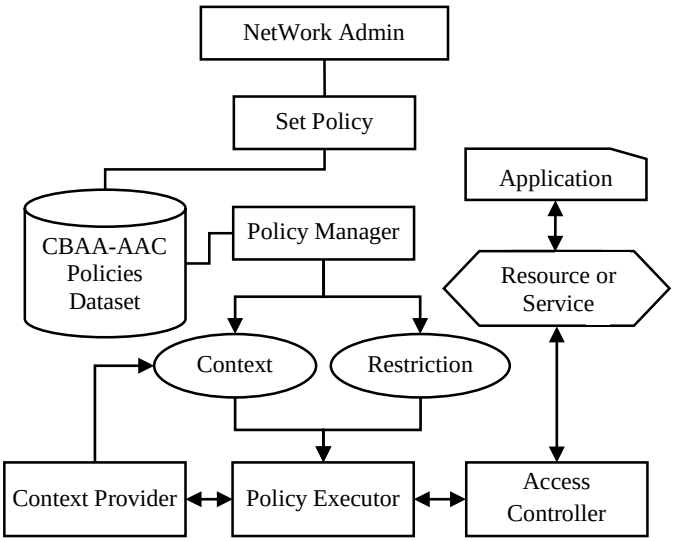

Fig.2. Block Diagram of CBAA-AAC Mechanism

3.4.2 Access Controller:

The Access Controller (AC) controls the privileges of applications and prevents security over unauthorized usage of device resources or services. Even though the Android OS has its unique permission/privilege control system that checks if an application has privileges to request resources or services, the AC complements this system with much more control methods and specific fine-grained control permissions that better replicate the Android application capabilities and reduces down its accessibility to resources. The AC enhances the security of the device (smart phone) system while the existing Android system has some permission that, once granted to applications, may give applications better accessibility than they required, which malicious code can take advantage of. For example, the permission READ PHONE STATE provides authorized Android applications a set of information such as the phone number, IMEI/MEID identity parameter, subscriber identification based on network, phone state (busy/available), SIM serial number, etc.

3.4.3 Policy Manager:

The Policy Manager (PM) represents the edge used to create policies, mainly conveying application limitations to contexts. It mainly provides control to the end user to configure which are all the resources and services accessible by applications at the given context provided by the Context Provider (Example: the user through the Policy Manager can form a policy to enable location services only when the user is at specific work during weekends between 9 am and 1 pm).

3.4.4 Policy Executor:

The Policy Executor (PE) enforces device/smart phone restrictions by comparing the device's context with the policies already configured. Once an application requests access to a particular resource or specific service, the PE checks the user-configured restrictions set at the PM to either grant or block, providing access to the smart phone application demand. The PE acts as policy enforcement by sending the authorization information to the AC to handle application specific requests, and is also had responsibility to again solve policy conflicts and apply the strictest restrictions. Through the PM, users can create CBAA-AAC policies through configuring application restrictions and linking them to the available contexts. When an application requests a particular resource or specific service, the AC verifies at run-time about the status, whether the application request is authorized and further forwards the request to the PE. If the request is clearly authorized, the PE then checks if there is any policy that corresponds to the application request. If such a policy is on a role, the PE requests from the CP to retrieve the context at the time of the application request. The PE then immediately compares the retrieved context against the context defined in the policy. In case of a similarity exists, the PE enforces the corresponding policy restrictions by reporting back to the AC to apply those restrictions on the application request. Special care had taken on the process of designing the access control framework so that the user-configured policies are securely implemented with lesser processing steps and even with better execution time to keep away from any significant delays in responding back to the requesting application. As our design shows efficiency on the way it securely handle policy execution, also maintain the context data provided by the CP to make sure it is much more accurate, specific and most often up-to-date.

3.4.5 Network Administrator:

The network administrator set policy to any user devices who is working under the control of any co-operative networks. Our approach to give network administrators of organizations the capabilities of providing access or denying access once a mobile device connects to their network. In this way, network administrators are able to block all malicious applications from accessing the resources and services that may affect the security of their network. The network-admin are also able to deny access to any application and control user in accessing them during work hours, such an approach is must for assuring good security of organizations when they allow users to "bring their own smart phones/mobile devices" into the campus.

The network administrator can able to register, login, add policy, view policy, delete policy and change password.

3.4.6 Set Policy:

Set Policy is the method of setting network-admin defined policy. In which network administrator are able to set policy and block malicious application accesses to particular resources and specific services that may affect the security of their organization. The network-admin is also able to deny access to any application and control user in accessing them during work hours.

4. EXPERIMENTAL RESULTS

The proposed approaches Context Based Android Application Administrative Access Control (CBAA-AAC) mechanism for Android systems and it provides mechanism for network administrator of organization, the capabilities of

providing access or denying access once a mobile device connects to their network.

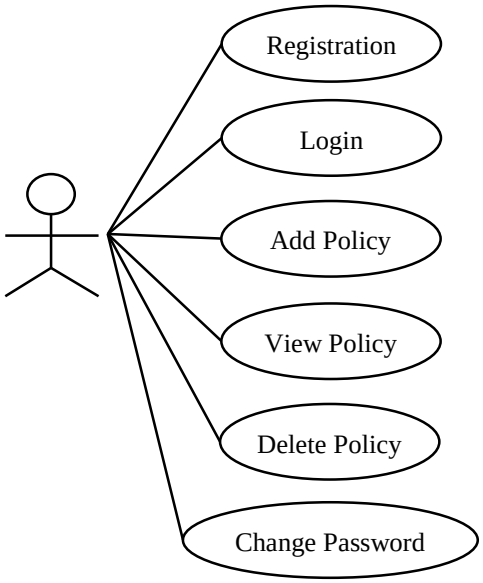

Fig.3. Network Administrator Use Case Diagram

4.1 LOGIN AND REGISTRATION

The Network-Admin Make a Registration and Log into the Page. The Network-Admin can able to set the user name and password for user's Android smart phones. Example: College administrator set the user name and password for smart phones which are hold by students.

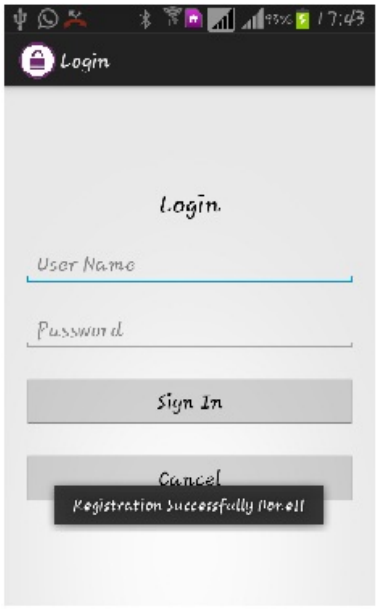

Fig.4. Admin Log-In Page

4.2 SET APPLICATION PRIVILEGES WITH IN CAMPUS

The Network-Admin set privileges with Policy Name, Restriction and Location. Example: The Policy Name can be set to anything by administrator for his/her identity, such as "Disable camera". Three restrictions that can be selected are: Resource Restriction Policies, Data Access Policies and Inter communication and Multitasking policies. Example: Here administrator can set the restriction as "Resource Restriction Policies".

In finishing step on setting of application privilege, administrator can set the location by Latitude and Longitude for Particular Location. Example: Administrator set location as Latitude: 13.094536 and Longitude: 80.205141.

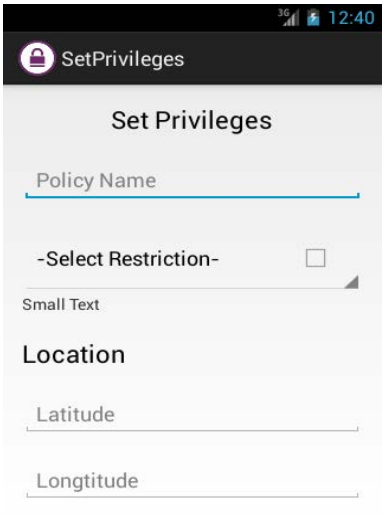

Fig.5. Set Privilege Page

4.3 VIEW POLICY / PRIVILEGE

The Assigned policies are viewed through the View policy. Once Network-Admin set the policies then those policies can be viewed.

Fig.6. View Policy Page

4.4 ACTIVATE POLICY

After viewing the policy, click on it to activate the policy. Once it gets activated, the camera cannot be used by any application on the Android mobile phone with in the specified location in terms of latitude and longitude.

Fig.7. Activate Policy Page

4.5 BLOCKING ACCESS TO APPLICATION

After activating the policy to block camera with in a specific longitude and latitude location, one of the smart phone Android application tried to open camera. During this process, as a result CBAA-AAC mechanism blocks the camera access with in specified location.

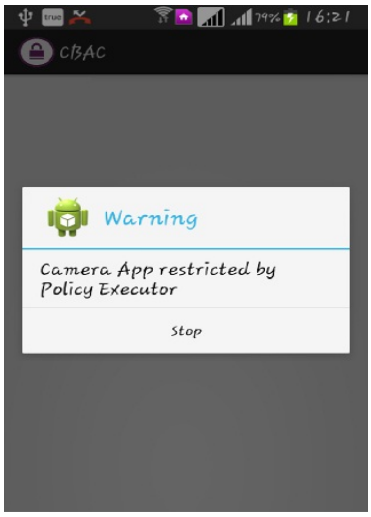

Fig.8. Resource Restricted Page

4.6 MEMORY OVERHEAD WITH CBAA-AAC MECHANISM

The memory overhead is taken into account after adding CBAA-AAC mechanism to android applications. The measured memory overhead is compared with the actual application memory overhead without CBAA-AAC mechanism and it is clear that only lesser deviation is present.

4.7 BATTERY POWER CONSUMPTION WITH CBAA-AAC MECHANISM

The consumption of battery power is absorbed to be lesser for policy enforced application comparatively to normal base application.

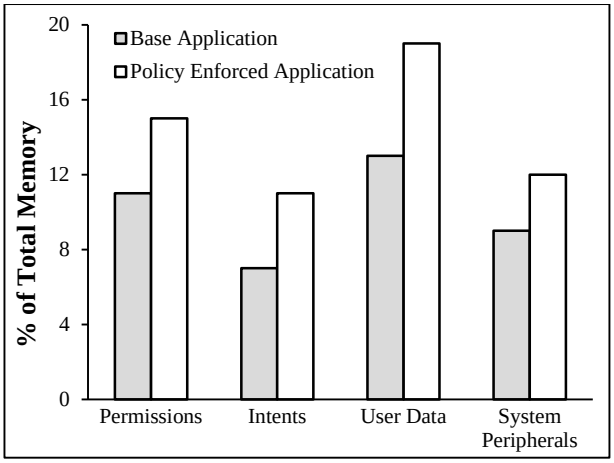

Fig.9. Comparison of memory overhead

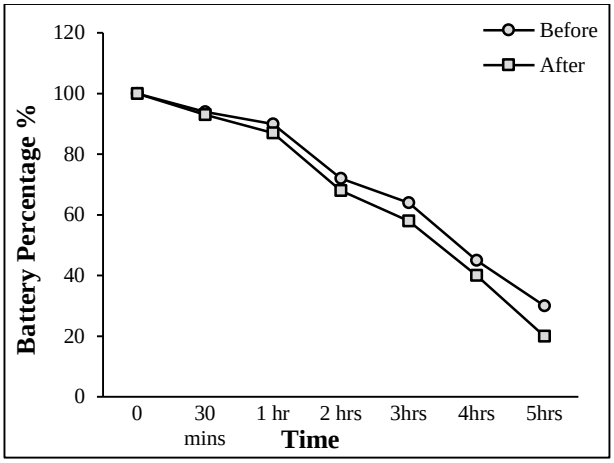

Fig.10. Battery Consumption before and after CBAA-AAC mechanism applied

5. CONCLUSIONAND FUTURE WORK

The proposed Context based Android Application Administrative Access Control (CBAA-AAC) mechanism is used to specify the restriction policies based on location. These policies restrict applications from accessing specific data and/or resources which are depending on the user context. The restrictions specified in a policy are automatically applied as soon as the user smart phone gets similarity with the pre-defined context associated with the policy. Our experimental results clearly show the effectiveness of these policies on the Android system/smart phone applications and the accurateness on locating the device within a user-defined context.

It is also be clearly shown that the consumption of battery power is also lesser for CBAA-AAC mechanism applied applications, when compared with normal base applications.

The restrictions are set by administrator where the administrator is superior to user and also in other aspect; the user set the restrictions to block resources and data for an application with respect to location.

In this way, network administrator blocked the Android application from accessing the resources and services that may affect the security of their organization, such an approach is must

for assuring security of institutions/organizations when they allow users to "bring their own smart phones/mobile devices" into the campus.

In future, the application can be synchronized with multiple organizations with their respective longitude and latitude. The memory overhead will remain almost same even though it gets synchronized with multiple organizations. Due to addition of larger and more restricted locations, the battery consumption of the smart phone gets reduced.

REFERENCES

[1] Samsung Galaxy S4 Specifications, Available at: http://en.wikipedia.org/wiki/Samsung Galaxy_S4, Accessed on May 2013.

[2] William Enck, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel and Anmol N. Sheth, "Taintdroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones", *Proceedings of the 9th USENIX conference on Operating systems design and implementation*, pp. 393-407, 2010.

[3] John Leyden, "Your Phone May Not Be Spying on You Now-But It Soon Will Be", Available at: http://www.theregister.co.uk/2013/04/24/kaspersky_mobile_malware_infosec/, accessed on April 2013.

[4] Robert Templeman, Zahid Rahman, David Crandall and Apu Kapadia, "Placeraider: Virtual Theft in Physical Spaces with Smartphones", *Proceedings of 20th Annual Network & Distributed System Security Symposium*, 2013.

[5] Roman Schlegel, Kehuan Zhang, Xiaoyong Zhou, Mehool Intwala, Apu Kapadia and Xiaofeng Wang, "Soundcomber: A Stealthy and Context-Aware Sound Trojan for Smartphones", *Proceedings of. 18th Annual Network & Distributed System Security Symposium*, pp. 17-33, 2011.

[6] L.L.N. Laboratory, "Controlled Items that are Prohibited on Llnl Property", Available at: https://www.llnl.gov/about/controlleditems.html, Accessed on 2013.

[7] Mauro Conti, Vu Thien Nga Nguyen and Bruno Crispo, "Crepe: Context-Related Policy Enforcement for Android," *Proceedings of 13th International Conference on Information Security*, pp. 331-345, 2011.

[8] Amit Kushwaha and Vineet Kushwaha, "Location Based Services using Android Mobile Operating System", *International Journal of Advances in Engineering and Technology*, Vol. 1, No. 1, pp. 14-20, 2011.

[9] Sandeep Kumar, Mohammed Abdul Qadeer and Archana Gupta, "Location Based Services using Android", *Proceedings of the 3rd IEEE International Conference on Internet Multimedia Services architecture and Applications*, pp. 335-339, 2009.

[10] Michael S. Kirkpatrick and Elisa Bertino, "Enforcing Spatial Constraints for Mobile RBAC Systems", *Proceedings 15th ACM Symposium on Access Control Models and Technologies*, pp. 99-108, 2010.

[11] A. Gupta, M. Miettinen, N. Asokan and M. Nagy, "Intuitive Security Policy Configuration in Mobile Devices using Context Profiling", *Proceedings International Conference Social Computing*, pp. 471-480, 2012.

[12] William Enck,Machigar Ongtang and Patrick Mcdaniel, "Understanding Android security," *IEEE Security Privacy*, Vol. 7, No. 1, pp. 50-57, 2009.

[13] E. Trevisani and A. Vitaletti, "Cell-Id Location Technique, Limits and Benefits: An Experimental Study", *Proceedings of 6th IEEE Workshop Mobile Computing Systems and Applications*, pp. 51-60, 2004.

[14] Jimmy LaMance, Javier DeSalas, and Jani Jarvinen, "Innovation: Assisted GPS: A Low-Infrastructure Approach", Available at: http://www.gpsworld.com/innovation-assisted-gps-a-low-infrastructure-approach/, Accessed on 2002.

[15] Skyhook, Available at: http://www.skyhookwireless.com/, Accessed on 2003.

[16] Mohamed Shehab, Gorrell Cheek, Hakim Touati, Anna C. Squicciarini and Pau-Chen Cheng, "User Centric Policy Management in Online Social Networks", *Proceedings of IEEE International Symposium on Policies for Distributed Systems and Networks*, pp. 9-13, 2010.

[17] Robert W. Reeder, Lujo Bauer, Lorrie Faith Cranor, Michael K. Reiter and Kami Vaniea, "More Than Skin Deep: Measuring Effects of the Underlying Model on Access-Control System Usability", *Proceedings of SIGCHI Conference on Human Factors in Computing Systems*, pp. 2065-2074. 2011.

[18] Lorrie Faith Cranor and Simson Garfinkel, "*Security and Usability*", OReilly Media, 2005.

[19] Kathi Fisler and Shriram Krishnamurthi, "A Model of Triangulating Environments for Policy Authoring", *Proceedings of the International Symposium on Access Control Methodologies and Tools*, pp. 3-12, 2010.

[20] Sven Bugiel, Lucas Davi, Alexandra Dmitrienko,Thomas Fischer, Ahmad-Reza Sadeghi and Bhargava Shastry, "Towards Taming Privilege-Escalation Attacks on Android", *Proceedings of 19th Annual Network and Distributed System Security Symposium*, pp. 1-18, 2012.

[21] Franziska Roesner, Tadayoshi Kohno, Alex Moshchuk, Bryan Parno and Helen Wang, "User-Driven Access Control: Rethinking Permission Granting in Modern Operating Systems," *Proceedings of 33rd IEEE Symposium on Security and Privacy*, pp. 224-238, 2012.

[22] Sven Bugiel, Stephan Heuser and Ahmad-Reza Sadeghi, "Flexible and Fine-Grained Mandatory Access control on Android for diverse Security and Privacy Policies", *Proceedings of 22nd USENIX Security Symposium*, pp. 131-146, 2013.

[23] Giovanni Russello, Mauro Conti, Bruno Crispo and Earlence Fernandes, "Moses: Supporting Operation Modes on Smartphones", *Proceedings of the 17th ACM symposium on Access Control Models and Technologies*, pp. 3-12, 2012.

[24] Machigar Ongtang, Stephen McLaughlin, William Enck and Patrick McDaniel, "Semantically Rich Application-Centric Security in Android", *Proceedings of Annual Computer Security Applications Conference*, pp. 340-349, 2009.

[25] Sven Bugiel, Lucas Davi, Alexandra Dmitrienko, Stephan Heuser, Ahmad-Reza Sadeghi and Bhargava Shastry, "Practical and Lightweight Domain Isolation on Android," *Proceedings of the 1st ACM Workshop on Security and Privacy in Smartphones and Mobile Devices*, pp. 51-62, 2011.