International Journal of Combined Research & Development (IJCRD)
eISSN:2321-225X;pISSN:2321-2241 Volume: 5; Issue: 5; May -2016

Implementing Context Based Access Control Systems for Mobile Devices

Dr. Sabera Begum[1], Rumana Parveen[2], Mehvish Fatima[3], S Mahaslaxmi[4]

Associate Professor, Dept. of CSE,KCT College of Engineering, Kalaburgi ,Karnataka, India[1].
UG Student, Dept. of CSE,KCT College of Engineering, Kalaburgi,Karnataka,India[234].

***Abstract*— Now-a-days, android applications have become a craze in people. Mobile applications are increasingly being deployed and used by enterprises, military and other secured industries. As mobile applications have so many advantages but every application comes with issue of data security. Mobile Android applications have access to sensitive data and resources on the user device. Misuse of this data by malicious applications may result in privacy breakage and sensitive data leakage. Protecting this sensitive data from leakage is a critical issue. In android mobile applications, the chances of such critical data leakage will be on higher side. For example, in many applications at the time of installation the application ask for system privileges. The fact is that Android users do not have control over the application capabilities once the applications have been granted the requested privileges upon installation. Mobile devices cannot be protected by physical security the same way as stationary systems can be protected. In this paper an access control mechanism is proposed which will avoid data leakages and misuse of user privileges.**

***Keywords*— Context-based access control, smartphone devices, security and privacy, policies, mobile applications**

I. INTRODUCTION

Android became very popular because of its various advantages and capacities. The first point is multitasking, meaning it can run many applications or services at the same time making the time factor feasible. Secondly, the process of notifying the user is made really easy because of high-end user interface. Third, there is easy access to millions and billions of applications in the Google Play Store and most of them are free [3]. This made a vast majority of the population to buy android based mobile phones. Though there are so many advantages, the problem of security is a crucial point to take note of. There are many ways to get information or data of the user from a service on their mobile phone. Most of these services can collect confidential data without the user's knowledge and can cause risks for the user. It is possible for an application to spy and release private data without the approval or even consent of the user.[4] Due to this reason, users carrying their devices in common places risk security problems by releasing personal information without their acceptance because they are unaware of such badware in their devices. The common solution to this is to not take the smartphone when going to certain confidential places, but this is easier said than done. In the case of certain government organizations, they restrict their employees from bringing any device having camera, video and recording facilities- which is most of the phones these days- even though their devices may contain private information which the user may be in need of. So the next step which can be possible is to have a good control over the capacities and capabilities of their devices. This can be done by reducing certain service privileges while being in private and confidential places based on context with more stress on location and time [2]. In the existing system, with context-based policies it can benefit most of the population by making certain applications disabled based on the location restrictions and enable it back when the user is out of such private locations. This is the case for government officials and law enforcement agents who are not supposed to bring the mobile devices during confidential meetings.[5]This requires the user to set their own policies to restrict applications based on the location. However, the difficulty of setting up these configurations requires the same knowledge needed to inspect service and resource permissions listed at the time of installation of the application [2]. In this paper, we give the network manager the role to block badware from using or even accessing the data that if exposed will affect the security of the network. This is important to achieve security in the network of corporate organizations and government bodies.

Users of Smart phones and other mobile devices are found everywhere. The organizations like IT industries, government sectors, military, aviation industry and e-commerce are using mobile applications to make work simpler. But at the same time the data which is transferred or shared during application installation has become a critical issue. This means sensitive data will invariably be placed on mobile devices (see Table 1). Security will be the limiting factor for deploying mobile devices in many scenarios where they would otherwise be extremely useful.

Table 1 Sensitive Data on Mobile devices on the basis of organization under which device is used.

Device Type	Sensitive data
IT industries	Company Bonds, Tender Data
Government	Tax Files, Police Records
Aviation industry	Plane Routes, Passengers information

As smart phones are becoming more powerful in terms of computational and communication capabilities, application developers are taking advantage of these capabilities in order to provide new or enhanced services to their applications. The danger arises when a device application acts intentionally harmful and uses device resources to keep a watch on user's activities or leak the user's personal data without the user's consent. Moreover, users carrying their Smartphone in public and private places may unknowingly expose their private information and can put their personal security in danger. The smart phone user is not aware of the existence of malicious activities getting performed on their devices. To prevent such harm and leakages, users must be able to have a better control over their device capabilities by reducing certain application privileges while being in public places e.g. At work place, shopping malls. To achieve such type of security mechanism, Smartphone systems must provide device owners with configurable policies that enable users to control their device usage of system resources and application privileges according to context, mainly location and time. Since such a feature is still missing in popular Smartphone systems, such as in Android systems, it is crucial to work on such systems and make them more secure and efficient.

II. Architectural Design

We studied the some architectures and one of them we are going to implement more efficiently. Studied framework consists of an access control mechanism that deals with access, collection, storage, processing, and usage of context information and device policies. To handle all the aforementioned functions, given framework design consists of four main components. The Context Provider (CP) collects the physical location parameters (GPS, Cell IDs, Wi-Fi parameters) through the device sensors and stores them in its own database, linking each physical location to a user-defined logical location. It also verifies and updates those parameters whenever the device is re-located. The Access
Controller (AC) controls the authorizations of applications and prevents unauthorized usage of device resources or services. Even though the Android OS has its own permission control system that checks if an application has privileges to request resources or services, the AC complements this system with more control methods. The AC enhances the security of the device system since the existing Android system has some permissions that, once granted to applications, may give applications more accessibility than they need, which malicious code can take advantage of.

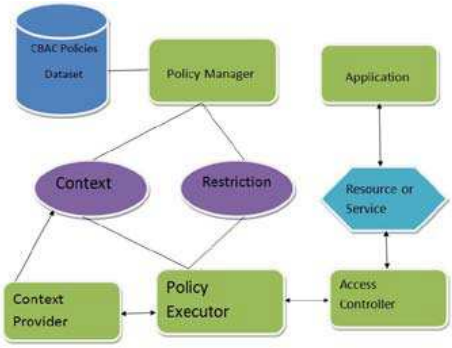

Fig. 1 Proposed Block diagram

The Policy Manager (PM) represents the interface used to create policies, mainly assigning application restrictions to contexts. It mainly gives control to the user to configure which resources and services are accessible by applications at the given context provided by the CP.

In this section we will introduce the architecture of the system capable of incorporation. Given below are the list of modules that are present along with the diagrammatic representation of the proposed system:

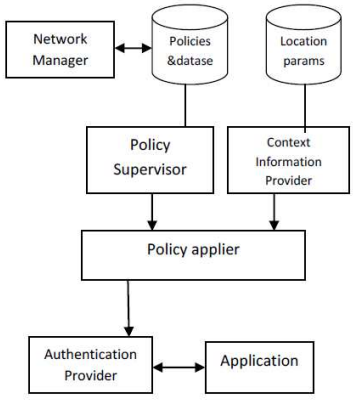

Fig. 2 The system architecture

A. Context Information Provider:
The first step is where the context information is discovered, the location parameters are discovered with the help of Global Positioning Satellite and Wireless Fidelity parameters. The second step involves the acquisition [8] where the collected information about physical parameters are stored in a database or repository. Linkage between physical and logical locations is done. If relocation occurs, updation is possible.

B. Authentication Provider:
This module performs authentication and authorization purposes so that there is no misuse or exploitation of the
services and data of the device. Android has a good checking mechanism for the grant or revoke signal but the authentication mechanism performs a second layer of security.

C. Policy Supervisor:
The creation of policies is done here i.e which restriction should be present for one specific location. For example, the College conference hall has a restriction of the camer a, so for this location, the resources to use the camera will be revoked permission.

D. Policy Applier:
The Policy Applier performs the process of comparison between the location and the restriction. When a service or resource is requested the policy applier checks for any restriction and based on the restriction will accept or deny the access. The result is sent to the authentication provider. The Policy Applier checks if there is a match between the corresponding location and restriction. The authentication provider then applies the restrictions, if there is no match it is considered as a new location and there will be default restrictions for the new location defined in Policy Supervisor.

E. Network Manager:
The registration of all the mobile numbers on a server is the main duty of a Network Manager. The policy setting mechanism is done by the manager for restricting the application on a mobile device when the user enters a sensitive area. As the service starts, the policy is set for the mobile and the control is passed to the 4 modules.

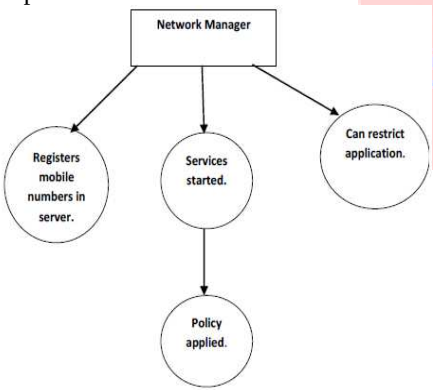

Fig.3 The network manager

II. CONCLUSIONS

In this paper, an android application has been implemented which support context based access control policies. This will help the application to restrict the malicious data and allow the system to access the specific data and/or resources based on user context. The proposed CBAC mechanism for android systems allows smartphone users to set configuration policies over their applications' usage of device resources and services at different contexts. For example, set of restricted privileges for device applications can be set when using the device at public place, and device applications may re-gain their original privileges when the device is used at private place.

REFERENCES

[1] A. Kushwaha and V. Kushwaha, "Location based services using android mobile operating system," Int. J. Adv. Eng. Technol., vol. 1, no. 1, pp. 14–20, 2011.
[2] D. Kulkarni, "Context-aware role-based access control in pervasive computing systems," in Proc. 13th ACM Symp. Access Control Models Technol., 2008, pp. 113–122.
[3] Wikipedia, (May 2013). Samsung galaxy s4 specifications. [Online]. Available: http://en.wikipedia.org/wiki/Samsung_Galaxy_S4
[4] W. Enck, P. Gilbert, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth, "Taintdroid: An information-flow tracking system for realtime privacy monitoring on smartphones," in Proc. 9th USENIX Conf. Oper. Syst. Des. Implementation, 2010, pp. 1–6.
[5] J. Leyden, (Apr. 2013). Your phone may not be spying on you now—but it soon will be. [Online].Available:http://www.theregister.co.uk/2013/04/24/kaspersky_mobile_malware_infose
[6] R. Templeman, Z. Rahman, D. J. Crandall, and A. Kapadia, "Placeraider: Virtual theft in physical spaces with smartphones," in Proc. 20th Annual Netw. Distrib. Syst. Security Symp. (NDSS), Feb. 2013.
[7] R. Schlegel, K. Zhang, X. Zhou, M. Intwala, A. Kapadia, and X. Wang, "Soundcomber: A stealthy and context-aware sound Trojan for smartphones," in Proc. 18th Annu. Netw. Distrib. Syst. Security Symp., Feb. 2011, pp. 17–33.
[8] Erlingsson, F. Schneider, "IRM enforcement of java stack inspection," in Proc. IEEE Symp. Secur. Privacy, 2000, pp. 246–255.
[9] D. Evans and A. Twyman, "Flexible policy-directed code safety," in Proc. IEEE Symp. Secur. Privacy, 1999, pp. 32–45.
[10] L. Bauer, J. Ligatti, and D. Walker, "Composing expressive runtime security policies," ACM Trans. Softw. Eng. Methodology, vol. 18, no. 3, pp. 9:1–9:43, Jun. 2009.
[11] G. Edjlali, A. Acharya, and V. Chaudhary, "History-based access control for mobile code," in Proc. 5th ACM Conf. Comput. commun. Security, 1998, pp. 38–48.
[12] C. A. Ardagna, M. Cremonini, E. Damiani, S.D.C.di Vimercati, and P. Samarati, "Supporting location-based conditions in access control policies," in Proc. ACM Symp. Inform., Comput. Commun. Security, 2006, pp. 212–222.
[13] E. Bertino, B. Catania, M. L. Damiani, and P. Perlasca, "Geo-RBAC: A spatially aware RBAC," in Proc. 10th ACM Symp. Access Control Models Technol., 2005, pp. 29–37.
[14] F. Roesner, T. Kohno, A. Moshchuk, B. Parno, H. J. Wang, and C. Cowan, "User-driven access control: Rethinking permission granting in modern operating systems," in Proc. IEEE Symp. Security Privacy, 2012, pp. 224–238.
[15] A. P. Felt, M. Finifter, E. Chin, S. Hanna, and D. Wagner, "A survey of mobile malware in the wild," in Proc. 1st ACM Workshop Security Privacy Smartphones Mobile Dev., 2011.
[16] W. Enck, "Defending users against smartphone Apps: Techniques and future directions," in Proc. 7th Int. Conf. Inf. Syst. Security, 2011, pp. 49–70.

[17] W. Enck, D. Octeau, P. McDaniel, and S. Chaudhuri, "A study of android application security," in Proc. 20th USENIX Conf. Security, 2011, pp. 21–21.
[18] J. Ligatti, B. Rickey, and N. Saigal, "Lopsil: A location-based policy- specification language," in Secur. Privacy Mobile Inform. Commun. Syst., Springer Berlin Heidelberg, vol. 17, 2009, pp. 265–277.
[19] M. Nauman, S. Khan, and X. Zhang, "Apex: Extending android permission model and enforcement with user-defined runtime constraints," in Proc. 5th ACM Symp. Inform., Comput. Commun. Security, 2010, pp. 328–332.
[20] Y. Zhou, X. Zhang, X. Jiang, and V. W. Freeh, "Taming information- stealing smartphone applications (on android)," in Proc. 4th Int. Conf. Trust Trustworthy Comput., 2011, pp. 93–107.