Dynamic Control System Based On Context for Mobile Devices

Kavana M.S
Department of Information Science and Engineering
GSSS Institute of Engineering and Technology for Women, Mysuru
kavana.8193@gmail.com

Meghashree P
Department of Information Science and Engineering
GSSS Institute of Engineering and Technology for Women, Mysuru
megha.51293@gmail.com

Keerthi M
Department of Information Science and Engineering
GSSS Institute of Engineering and Technology for Women, Mysuru
keerthimanjunath.20@gmail.com

Anand M
Assistant Professor
Department of Information Science and Engineering
GSSS Institute of Engineering and Technology for Women, Mysuru
anandm@gsss.edu.in

Abstract*:* "To render the accurate information, at correct place in real period with custom-made setup and locality sensitiveness" is the inspiration for every location based information scheme. Android applications in mobile devices may often have access to susceptible data and resources on user device. "Location Based Services" can only provide services that give a data and information to person, wherever he might be through various android applications. To avoid the data misuse by malicious applications, an application may get privilege on the specific user location and thus a Context Based Access Control Mechanism (CBACM) is needed so that privileges can be established and revoked vigorously. A very interesting application include shadowing where immediate information is required to choose if the people being monitored are valid intimidation or an flawed object. The execution of CBACM differentiates between the narrowly located sub-areas within the distinct area. Android operating system is modified such that context based access restriction can be precise and imposed.
Keywords*: Android, CBACM, Location based services, mobile applications.*

*****

I. INTRODUCTION

Smartphone's usage has become mandatory in the current trend and one cannot imagine a life without it. Nowadays Smartphone's communication and computational capabilities are essential and becoming more powerful, it has become an advantage for application developers to enhance or create new services to their applications. The tasks such as email, internet banking, ecommerce activities and delicate records such as photos, videos and communications logs are maintained through Smartphone's, security plays a crucial task. Hackers through malicious applications can use the sensitive data that may have exposed without the user knowledge for unlawful activities; this becomes a major threat for the user. The control of the applications should laze in the hands of the user to foil the threats that can occur. The privileges of the applications can be restricted according to the user sensitive context. Applications and services are designed in such a way that the usability problem of the device can be tackled by predefined policy configurations of the applications. The current policy system in the applications does not build a wall for protecting user data and device resources. Even though the existing security for mobile operating system restricts the applications by accessing the sensitive data and resources, but lacks in enforcing those restrictions. For a particular context the selected privileges can grant the permission to access the data for a picky application. The user should have power over the usage of data and resources to configure device policies based on particular context. The context which extends from high profile employees to regular smart phone users is the basic need for configurable device policies. For example, government employers, such as in national labs [5], restricts their workers to use camera-enabled device including Smartphone at their work place, so the user cannot have their devices all the time with them even though they need. Hence context based device policies, provides the control to the user so that the user can restrict the access of camera enabled applications.

Locations can be fetched through various positioning methods such as Global Position System (GPS), Cellular Triangulation (Cell ID), WIFI Positioning. Finally, existing location-based policy systems are not accurate enough to differentiate between nearby location without extra hardware or location devices [6, 8, 9]. Sometimes without location positioning methods context may be faked by assuming the context by itself, in order to overcome the problem the context based policy system is designed in such a way that it should rise above the following challenges:

1) The policy boundaries that have applied on the device for a particular context should not be avoided. Hence the application cannot fake the location or time of the device.
2) As predefined policy for a particular location is identified the restriction is applied automatically. So the location accuracy should be greater as the different policies are applied in different spots.

3) Delays in the system performance should not be incurred by the applied policies and the need to modify the source code should not be caused when the context-based policies are enforced.

The user can set the privileges over their applications based on the usage of system resources as well as services at different context using Context Based Access Control mechanism (CBAC). For example the student can restrict the privileges while being in college and can regain the original privileges when the device is at home. The action repeats every time when the user device matches predefined context of the user defined policies. For the locations that are not defined previously can also be assigned with the set of default policies. CBAC policies on android operating system are implemented along with the tool that enables the user to define substantial places such as college or home using location positioning methods. The user can be more specific by differentiating between subareas within the same location, such as library, internet lab and class. Hence privileges for the applications can be set for the fine grained context.

II. PROBLEM STATEMENT

The permissions of the applications in accessing the user resources and data does not lie in the hands of user, the request for access is of the type *all or nothing.* The user cannot opt for particular set of privileges given by an application. So the application may have access to large amount of data and resources which may lead for privacy breaches and sensitive data leakage. To overcome the security issues the user can set the customized privileges in which access privileges for an application can dynamically granted or revoked based on the location.

III. RELATED WORK

A location-based service (LBS) is a mobile application that is reliant on the location of a mobile device. The inspiration for the user to use location based system is "To aid with the particular data, at the exact place at real time with personalized setup and location context". To eradicate the problems of bulky desktops, in this era we are mainly dependent on the palmtops and iPhones for computational purposes. The large number of applications are provided and usage where a person travelling in unknown area needs to get relevant data and information. So the needs of the person can only be accommodated with the help of LBS. An upcoming application includes surveillance where current information is needed to decide if the people being monitored are any real threat or an invalid target. There are many applications which are developed to provide the user with information about the place to visit, but these applications are restricted to desktops only. So these applications need to be imported on mobile devices. All the information for the user must be available in the mobile device and it must be in user customized format. LBS services can be categorized into LBS services and user-request LBS service. In the triggered LBS service, the location of users mobile device is retrieved when a condition set in advance is fulfilled. In the user-requested LBS service, the user decides whether and when to retrieve the location of mobile device and use it in the service [2].

Role Based Access Control (RBAC) model is developed for spatially-aware extensions of combining the administrative and security advantage of RBAC with the dynamic nature of mobile and persistent computing systems. The implementation is based on an enhanced RBAC model which supports location based access control policies by incorporating spatial constraints. In order to enforce spatially-aware RBAC policies in a mobile device requires addressing many challenges. Firstly, one must ensure about the integrity of a user location during an access request, so a proximity-based solution using Near-Field Communication (NFC) technology is adapted. The next challenge is to verify whether the user position is continuously satisfying the location constraint. For the implementation of RBAC models many protocols are generated and the evaluation of the security measures is taken. The first challenge can be addressed by a novel proof-of-location protocol, based on the assumption that a number of location devices are predefined in known physical location positioning methods [8].

Smartphone's are vast becoming an integral part of life for many users and they mainly contain personal data like photos, videos and contacts. In the mobile devices configuration of access control policies are quite boring and unintuitive for users. So the software developers have attempted to address this problem by setting up default policy configurations, but such global defaults may not be sagacious for all users. So the problem can be overcome by using modern Smartphone which are capable of sensing a variety of information about the surrounding environment like Bluetooth devices, Wi-Fi access points, temperature, ambient light, sound and location co-ordinates. Protection mechanisms serve their purpose only when they are configured with sensible policies for accessing and sharing data. Application and the software designers attempt to tackle the usability problem by providing users with a default policy configuration. But a global default policy may not be suitable for the needs of every user. So context profiling framework is implemented and dynamically the decisions are taken based on the perceived safety of current context[1].

Android Operating System will serve as a flexible and effective environment to instantiate different security solution, so generic security architecture is generated to solve security issues. The security architecture, termed as Flask Droid provides mandatory access control concurrently on both Android's middleware and kernel layers. The assignment of policies on both these layers is non-trivial due to their completely different semantics. So an efficient policy language (encouraged by SELinux) customized to the goals of Android's middleware semantics. The flexibility of the architecture is shown by policy-driven instantiations of selected security models such as fine-grained per-app access control. The evaluation on the implementation of SE Android illustrates its efficiency and effectiveness.

The rising Grid infrastructure presents many challenges due to its inherent heterogeneity, multi-domain characteristic and highly dynamic nature. The critical challenges faced are providing authentication, authorization and access control guarantees. So the SESAME dynamic context-aware access control mechanism for pervasive Grid applications is generated. SESAME has a great advantage to complement current authorization mechanisms to dynamically grant and adapt permissions to users based on their current context to the

particular application. The emerging Dynamic role based access control (DRBAC) model enlarges the classic Role based access control (RBAC) [4].

IV. METHODS

The implementation of the Context Based Access Control Mechanism is done using context provider, access controller, policy manager and policy executor. Along with the important components such as the location of the user, access privileges the necessary requirements for implementing the CBAC mechanism is classified into three sets where one set contains the package name of all the installed application on the user devices and can be symbolized using any of the unique character. The second set contains the objects or the resources that are protected which may refer to the services, permissions or available functionality for that particular application. The actions or the permissions that can/are restrict through CBAC mechanism are grouped in the separate set. The simple architectural pattern of the CBAC mechanism is as shown.

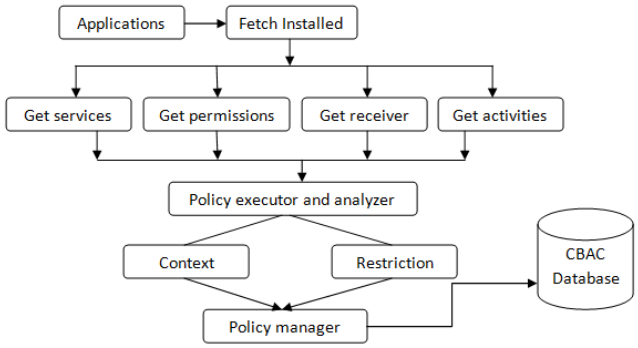

Fig1: Architecture of CBAC mechanism

The first and foremost thing to implement the CBAC mechanism needs the location of the user device which can be fetched through the context provider. Context provider collects the location of the user through various device sensors such as GPS, WI-FI and others, associates the fetched substantial location with that of the valid location defined by the user and updates automatically whenever the location of the devices changes.

The applications that make use of user's resource and data need to be authorized, so to check the authorization of applications Access controller is used. Access controller put a stop for unauthorized usage of the data and resources by the malicious applications. The security is very much enhanced by access controller which prevents the escape of the sensitive data and resources into the hands of unauthorized applications without the knowledge of the user.

The interface used by the user in order to set the restriction to an application via activities. An activity helps in defining physical location and setting restrictions to be applied on, for example the student can disable the application during 9AM to 4PM in college location which is pre-defined. Policy manager are checked with CBAC datasets for the user defined restrictions and policies are executed. Policy manager practices four main Android application components: Activities, Service, Context Provider and Broadcast Receivers. Application Events, Permission Access, Resource Access, System Preferences and Time Restriction are constituent of Activities. The Broadcast Receiver helps in fetching the device location information and updating the device location with intent. Service is used to know the up-to-date device location.

Policy executor imposes restrictions by comparing configured policies based on context; through policy executor the user can create CBAC policies. When an application is running the authorized access is checked and given the control to policy executor. Access controller verifies the user permissions, after creation of policies by the policy executor, a request for context from context provider at time of application request is sent. If the context matches the pre defined location then resultant policy is reported back to access controller so that restrictions are applied to applications.

V.EXPERIMENTAL RESULTS AND DISCUSSIONS

The approach requires users to configure their own set of policies; the difficulty of setting up these configurations requires the same expertise needed to inspect application permissions listed at installation time. However we plan to extend the approach to give user an option which will displayed the severity and security status for each permission for an application and if user wants to see which permission provide the high security threat to their device and data. So the reason the displaying of permission list in three filters, High security threat level (red icon), middle level (yellow icon) and Low security threat (green).

CONCLUSION

The enhanced edition of the Android operating system is utilized for supporting context-based access control policies. According to the context of user the policies restrict the applications from accessing specific data and/or resources. As pre-defined context matches with the current user location the specified restrictions are pertained. The security is enhanced as CBAC mechanism provides the custom privileges even for the undefined location. The complete restriction of the application to utilize services and permission. In future one can add utilization of each service by the individual application and then depends on the feature and time restricts the service utilization for fixed time in a day.

REFERENCES

[1] A.Gupta,M.Miettinen, N.Asokan and M. Nagy, "Intuitive security policy configuration in mobile devices using context profiling", *IEEE International Conference on Social Computing, ser. SOCIALCOM -PASSAT '12. Washington, DC, USA: IEEE Computer Society*, 2012, pp. 471–480.

[2] A. Kushwaha and V. Kushwaha, "Location based services using android mobile operating system", *International Journal of Advances Engineering and Technology*, vol. 1, no. 1, pp. 14–20, 2011.

[3] Bilal Shebaro, Oyindamola Oluwatimi, Elisa Bertino "Context Based Access Control Systems for Mobile Devices", *Computer Science, Cyber Center and*

CERIAS, Purdue University, West Lafayette, IN 47907, USA

[4] G. Zhang and M. Parashar, "Dynamic context-aware access control for grid applications," in Grid Computing, 2003. Proceedings. Fourth International Workshop on,2003, pp. 101–108.

[5] L.L.N. Laboratory, "Controlled items that are Prohibited on llnl proporty,"https://www.llnl.gov/about/controlleditems.html.

[6] M.Conti,V.T.N. Nguyen, and B. Crispo, "Crepe: context-related policy enforcement for android," *in Proceedings of the 13th international conference on Information security, ser*. ISC'10. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 331–345.

[7] M.Nauman.S. Khan, and X. Zhang, "Apex: extending android permission model and enforcement with user-defined runtime constraints," *in Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, ser*. ASIACCS '10. New York, NY, USA: ACM, 2010, pp. 328–332.

[8] M.S.Kirkpatrick and E. Bertino, "Enforcing spatial constraints for mobile rbac systems," *in Proceedings of the 15th ACM symposium on Access control models and technologies*, ser. SACMAT '10. New York, NY, USA: ACM, 2010, pp. 99–108.

[9] S. Kumar, M. A. Qadeer, and A. Gupta, "Location Based services using android," *in Proceedings of the 3rd IEEE international conference on Internet multimedia services architecture and applications , ser*. IMSAA'09, 2009, pp. 335–339.

[10] W.Enck, D.Octeau, P. McDaniel, and S. Chaudhuri, "A study of android application security*," in Proceedings of the 20th USENIX conference on Security, ser.*SEC'11. Berkeley, CA, USA: USENIX Association,2011,pp.21–21.