Mobile Application Profiling using Secure Element

Harminder Singh

Research and Development, Syscom Corporation Limited

Abstract- This paper proposes the use of GSMA defined approach of profile switching for UICC to store and switch between multiple profiles on mobile applications. This paragraph is enough for abstract

Index Terms- Access Rule Applet, UICC, Java card applet, Profile Management, Secure element

I. Introduction

Today, Device support multiple Application operating simultaneously and each application supports multiple user profiles. The profile may contain user data, user information (secure, unsecure both) or configuration settings for application itself.

Multiple users on a single mobile application can be managed through the use of profiling where users switch to their desired profiles during the use of application. Mobile application maintains a different type of profile data w.r.t. a user on device storage. Therefore, this can lead to loss of data in case of corrupted storage apart from possible risk of leakage of personal information.

Some the examples of application profiles on device are:

For example, we will have the following two scenarios of profiling in applications. Check paragraph spacing rule from template

1) Parental Control: Parents want their kids to view only a specific type of information during the use of an application; so a specific profile for kids could be created and another profile could be created for general users. Kids and general user can switch to their respective profiles before using the application.

2) Payment Applications: Sensitive user information is stored in form of different profiles in the device. The profile may contain information of credit cards, credentials, balance information to be synched with the server. For this purpose the concept of "Trusted Environment", is created on the mobile devices.

In these examples, whenever multiple users are operating on the applications, the application will be using the profiles according to user and their saved information in the profile.

II. Framework of Profiles on UICC [2]

In UICC (or eUICC), multiple profiles can exist on a single platform. Every profile with its individual ISD-P contains specific information that is required to be stored. Each profile is controlled and created under ISD-R. There can be as many as profiles present on UICC depending upon the EEPROM storage available.

Condition imposed by the framework: There should be at least one default t profile which should always be present on the storage.

Every profile management request is sent to ISD-R by any external entity. Here external entity can be a device, remote server etc. ISD-R will create a new profile according to the requirement and availability of storage on UICC.

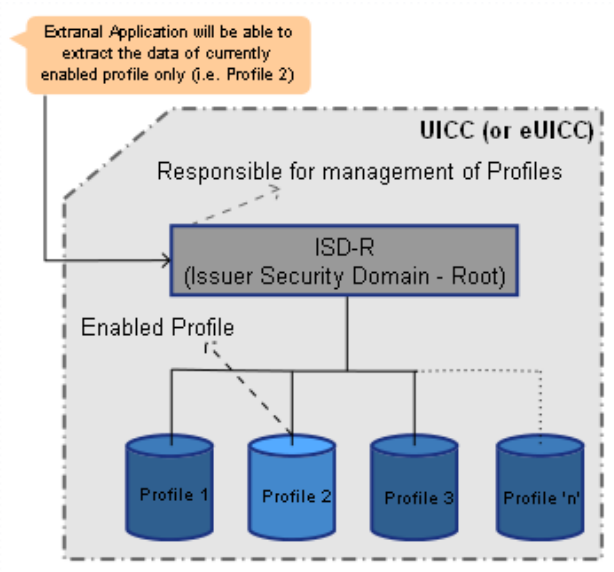

Figure 1: Framework of Profiles on UICC

Profile Management by the Framework: Each profile space is identified through its AID Profile space is used to store profile data. Each profile and its data are completely independent and isolated from any other profiles present on the storage. This framework enhances the security among different profile data.

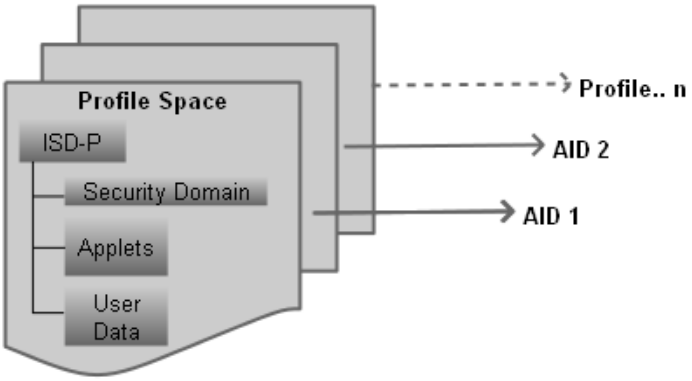

Figure 2: Isolation of Profiles on UICC

Request for enabling the profile is sent via ISD-R. ISD-R will check if the requested profile exist on the storage and if it exist, then the requested profile is enabled. Now every file data on UICC corresponds to the data from that particular enabled profile. On receiving the request for disabling the profile, UICC switches to a default profile known as fall-back profile. Therefore at least default data is always available for external entity to fetch.

III. Structure of Profile

A Profile is further divided in components and controlled by its ISD-P [2]. These components can be any of the following:

1. File System
2. Security Domains
3. Generic Applications
4. Policy Rules

File system in the form of elementary files [1] can be used to store data, which can be of different type and these files managed according to user preferences. Security domains are use to store user specific keys, which can use to encrypt user sensitive data.

IV. Management of Profile on the Devices

During profile creation, application creates a user space on mobile storage through its underlying framework. This profile (or user space) will hold user data or related configurations, this data ranges from user names, personal information's or even banking cards information etc.

In figure 3:

1. An application user has created two different profiles on mobile application. All the data of user is stored in user space (user data 1, user data 2...).
2. Applications have framework of profiling which manages the creation and enabling/disabling of profiles.
3. This profile management framework may be present inside an application or provided as an abstract layer from underlying native operating system.

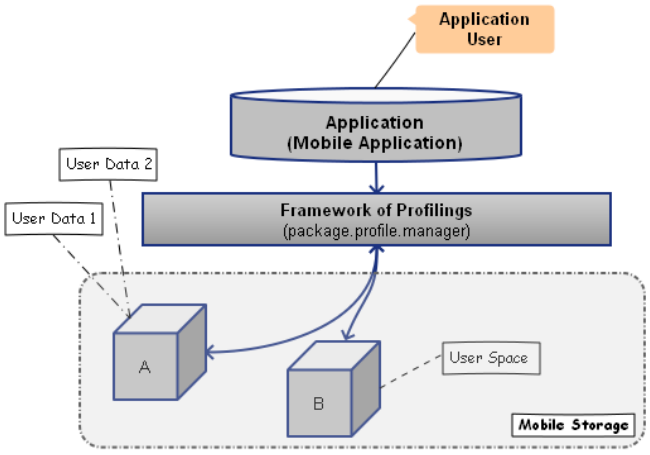

Figure 3: Management of Profiles in Mobile Application

The storage of profile related data on mobile phone and their switching holds the following pitfalls.

Security of Profile Data: Data stored on mobile devices is vulnerable to theft or leakage. After all mobile devices are attractive targets for stealer always. Therefore some sensitive information may be leaked along with profile data.

Does not adhere to Portability without Internet: Mobile application does offer the use of same profiles on a different mobile phone, but only if phone is connected to internet. In this way user could not use the same profile on different platforms such as mobiles or tablets.

Data loss due to Mobile Storage Degradation: User profile data will be loss if mobile storage is degraded or corrupted.

Framework for Profile Switching: Developer needs to implement profile switching framework in mobile application from start. This framework could leak profile data of user due to some programming bug.

V. Interface with Application Data

Interface of mobile/device application with profile data or user data can be explained with the help of figure 4.

A. Interface with UICC

Mobile application will deploy a client applet known as Java card applet on UICC. This client applet handles all the commands received from mobile application for further processing and will be in the form of secure element (SE) application. A mobile application on android or windows platform can access secure element by the help of smart card access API [3] [4]. Framework of profiling present inside the application could be used as an abstraction layer on transition from mobile based storage to UICC based storage. Framework of profiling will be used to send/received commands from UICC.

B. Access to application on UICC [5]

Smart card access API is designed to provide access on an applet on secure element by keeping restriction and also provide security to the application on secure element (UICC) with the use of access control enforcer (ACE).

Any external device application will have to gain access of SE application before start communication with it. This access is provided on the basis of access rules stored in ARA (Access Rule Applet). These rules are stored on the basis of hash of certificate of mobile application known as device identifier. In this solution, rule containing "full access" to mobile device for client application will be stored in ARA. Therefore mobile device will be able to send APDU's [6] to its client application on SE element stored in ARA (Access Rule Applet).

These rules are stored on the basis of hash of certificate of mobile application known as device identifier. In this solution, rule containing "full access" to mobile device for client application will be stored in ARA. Therefore mobile device will be able to send APDU's [6] to its client application on SE element.

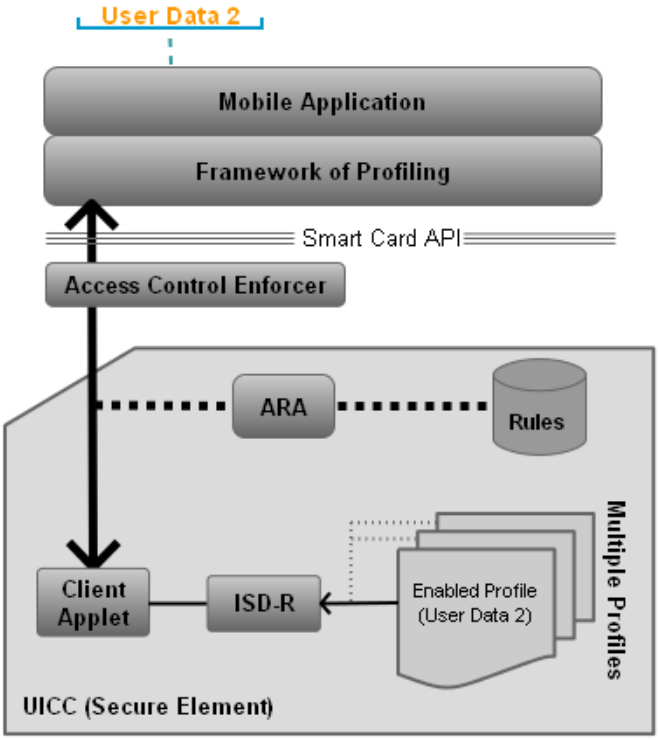

Figure 4: Interface of Device Application with Profile

C. Access to application on UICC [5]

Any new profile on UICC storage can be created with the use of "Profile Download and Installation procedure"[2]. In this command includes file structure that will contain user data and keys of security domain that could be used for encryption/decryption of data for additional security of user data. The constituent commands in this will be sent to client application on UICC and they will be routed to ISD-R for creation of new profile space and consequently a new ISD-P for the profile space. An identifier of profile known as application identifier of ISD-P will be returned to device application for further management of profile space.

D. Profile Management Commands

All the profile management commands will be sent to client application on SE element and it will route the commands to ISD-R for further processing. There could be multiple profile spaces under every ISD-R on UICC. But external mobile application will be able to retrieve the data of only currently enabled profile. Therefore enable profile command [2] is used before getting user data of from the profile space. A default profile space is also available (by default) and will be used in an initial installation of mobile application. This default profile can also be set using disable profile command [2]. On disabling a profile, its data will stay as it is on UICC storage for further use when the profile will be re-enabled. There may be a use-case when a profile space is now longer required by parent mobile application, then it can be deleted with the use of delete profile command [2]. All the user data will be deleted after execution of this command and will not be available for future user.

E. User Data Read/Update

User data modifications and reading can be done with the use of personalization commands [7] and will be sent to client applet; it will parse the commands from device application and provide the required output as a response to the received command.

VI. Conclusion

The usage of UICC for profile data storage and switching will eliminate the risk of leakage in personal information by any other application on mobile phone; enhance the concept of portability and android application can utilize the available framework of profile management and switching provided by GSMA. Reformat the line – Don't write the content only in one single line

References

[1] GSM 11.11 Version 5.3.0, July 1996, ch. 6.
[2] GSMA - Remote Provisioning Architecture for Embedded UICC Technical Specification, version 2.0.
[3] http://www.microsoft.com/en-us/download/details.aspx?id=43681
[4] https://code.google.com/p/seek-for-android/wiki/AccessControlIntroduction
[5] Global Platform, Secure Element Access Control – Public Release v1.0, Pg 12-15.
[6] ISO/IEC 7816-3, Cards with contacts -- Electrical interface and transmission protocols.
[7] Global Platform, Card Specification – Public Review v2.2.1.10.

Authors

First Author – Harminder Singh, Bachelor of Technology, harminder_jassal@yahoo.co.in