

Introduction to protocol discovery mechanism for secure elements applications

Sarabjeet Singh

Research and Development, Syscom Corporation Limited

Abstract- Communication between Secure Elements and the Service providers have always been a matter of concern in terms of security for exchanging sensitive information in secure manner. Most common requirement for this information exchange is the protocols that are supported by different applications on secure elements or applications that interact with the secure elements. This paper describes how protocol discovery mechanism proposed by global platform works with SIM based applications (for example NFC application interacting with SIM and other applications) or application interacting with the cards to exchange information about supported protocols in the applications and provide/deny access to services based upon discovery of protocols

Index Terms- SIM, Protocol, REST, OTA, Secure Element

I. INTRODUCTION

With evolution in GSM technology, use of secure elements have evolved from just storing information of user to performing interactions with outer world applications and exchanging sensitive data. Example of such interactions is NFC applications that interact with both secure elements via NFC controller installed on it and NFC terminals. For example, banking and payment applications are using SIM Cards to store information like passwords, card information, account information because SIM cards are considered as secure elements.

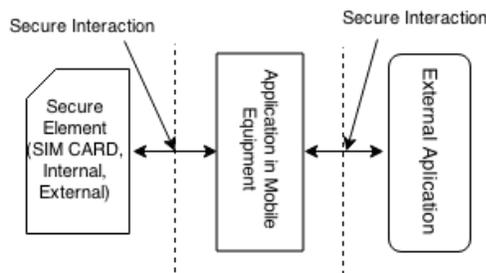


Figure 1: Simplified View of Interaction of application with secure element

A 'Secure Element' is tamper-proof entity that can be used to host applications securely. These elements can be of three types: embedded secure elements, universal integrated circuit cards a.k.a UICC or micro SD cards. Out of these three, micro SD and UICC are removable. These are not necessary to be a SIM card. A micro SD card or a security chip can also be used as a secure element. Which type of secure elements is to be used is dependent upon business needs because each element satisfies a

different market need. For example, for NFC applications on mobile equipments, UICC or SIM cards are used as secure elements.

Global Platform has defined specifications that allow different service providers to store different types of services on secure elements. A service can be defined as a composition of a set of applications to be deployed on the end user secure elements. This composition can optionally contain a User Interface application to be deployed in the device like Mobile equipment. This deployment is managed by using Over the Air Technology. This technology allows service providers to deploy, upgrade or remove applications and services on the secure elements. According to their requirements, these applications or services require communication with each other and exchange messages in a reliable and interoperable way. For example, for a bank, it has to request the deployment of a payment application in to an SE which has Over-The-Air (OTA) capability; for a Mobile Network Operator it may be required to notify the various life cycle events of the end-user mobile environment (device lost, etc.) to the service providers. Now, how these interactions are implemented are purely up to service providers implementations. But these interactions should be completely in compliance with Global Platform system messaging core specification. Since there are different protocols available in the market that can be implemented in the applications interacting with each other.

Research Elaboration – In order to complete a task, these applications need to communicate with each other and exchange messages in a reliable and interoperable way. There are cases when the interacting applications are supporting different protocols of communications or have different versions of same protocol implemented and applications needs to know about capabilities of each other to check which service should be loaded onto other. For such scenarios, these applications requires a method to exchange information of their capabilities and protocols to each other, this may be required only during first interaction. GP has introduced a new specification of protocol discovery mechanism between two interacting application based upon RESTful services consumptions (with exchange rules and format of protocol exchange). Currently a draft is released for public review.

With implementation of protocol discovery mechanism between applications following can be achieved:

- 1) Identification of protocol in client/server applications (client application must also support protocol discovery)
- 2) With identification, it can be found which protocols are supported by client/server application.
- 3) If client application does not support or implement required protocols for data exchange from/to the application

interacting with SE, access can be denied thus decreasing chances of unauthorized access to data on SE,

- 4) With protocol identification, the requesting application can know the protocols that it is not supporting (Client application may have updated protocol version that our application is not supporting), thus it can request the client application access based upon protocol that our application supports.

Use case examples:

First Case:

- 1) User mobile contains a web based application that interacts with SIM card present on mobile. It requires interaction with another application (say a website) for certain activities.
- 2) Application A (WEB application) required both X and Y to be checked for granting access.
- 3) User application discovers that application A requires X+Y protocol but it has only support for Y (The protocol information in extracted from secure element).
- 4) It can request Application A to grant access to services that requires only protocol Y (if Application A allows to do so)
- 5) In this way, user application will be able to use services of application A (not all but some, instead of denial of service usage)

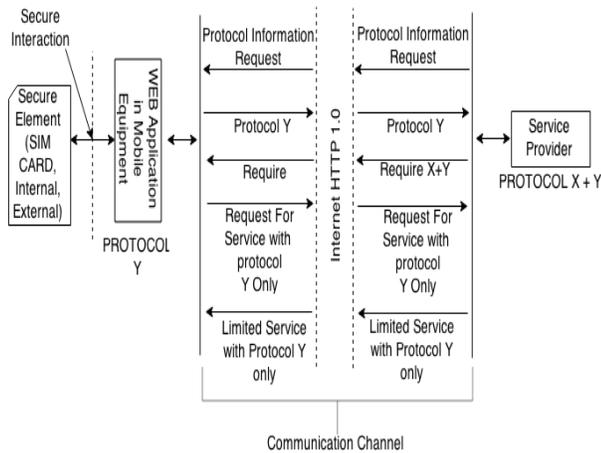


Figure 2: Protocol Discovery and Service request

One point to note here is that the request for service with only protocol Y is made by the Web application present on the Mobile equipment, not by the secure element. The web application when receives service, it is the responsibility of that application to load and manage the service on the secure element.

Second Case:

- 1) User has banking application that implements protocol X with version 1.1. It is interacting with application A that supports also supports protocol X. But the supported protocol version of communication in case of application A is 1.0
- 2) Application A requests banking application for its capabilities. Banking application publishes its capabilities to application A
- 3) Application A discovers that banking application have updated version of protocol X. It then undergoes upgrade of protocol X in itself.

- 4) In this scenario, Application A undergoes upgrade of its capabilities based upon capabilities of user application.

II. PROTOCOL EXCHANGE MECHANISM

The exchange of protocol information will happen with the help of HTTP REST Requests. GP System protocol discovery mechanism provides complete format and requirements for the request process, required REST URI formation, attributes and response exchange mechanism. An example of HTTP REST request for protocol information request is provided below:

**spdm/protocols?version=<version>&requesterId=<Discovering actor Id>
&secureComponentType={SE|TEE}&secureComponentId=<SC Id>**

Parameter information in the request URI:

Table 1: Parameter in HTTP REST request

Parameter	Information	Mandatory/Optional
Version	Version of the System Protocol Discovery Mechanism used by discovering application	Mandatory
requesterId	It is the identity of the Requester	Mandatory
secureComponentType	Type of secure component	Mandatory
secureComponentId	It is the identity of the secure component	Mandatory

Here secure components can be either secure element (SE) or Trusted Execution Environment (TEE).

Response Information: When a request for protocol/capability information will be received, the standard response of HTTP 200 OK code will be sent by the receiving entity with a response of required information. This response will be in standard JSON format that can be parsed by the requester application. The structure of the response will be pre-determined in accordance with the Protocol Discovery mechanism specification. Example of a standard JSON response for the request is provided below:

```
{
  "version": "<version>",
  "supportedVersions": ["<version #1>", "<version #2>"],
  "providerId": "<SPDM Provider Id>",
  "supportedProtocols": [
    {
      "protocolId": "<Protocol #1 Id>",
      "protocolVersion": "<Protocol #1 Version>",
      "protocolInfo": {
        "<Protocol #1 Info>"
      }
    }
  ],
  "referAlso": [
    {
      "additionalProviderURL": ""
    }
  ]
}
```

Response attributes are described in table 2 below:

Table 2: Response Parameter to be sent to requester application

Field	Information Provided by Field	Mandatory /Optional
version	Version of System Protocol Discovery Mechanism	Mandatory
supportedVersions	list of versions of System Protocol Discovery Mechanism protocol	Conditional
providerId	ID of Application providing security protocol discovery mechanism	Mandatory
supportedProtocols	List of supported protocols	Optional
protocolId	Protocol ID that might be used by requester for dialog exchange	Mandatory
protocolVersion	Version of identified protocol for dialog exchange	Mandatory
protocolInfo	Information specific to discovered protocol	Optional
referAlso	List of other applications present on secure element that can be contacted with the discovered protocol	Optional
additionalProviderURL	URL for additional application on which protocol discovery mechanism can be initiated by requester.	Mandatory

The identified protocols will be sent in format of MAJOR.MINOR.MAINTENANCE format.

For example, if a protocol has major version 1 minor and maintenance version as 0 then information in JSON field will be sent as: "1.0.0". For the URL to be formed, first requirement for the application is to get a base URL from the secure protocol

discovery mechanism. GP specification for protocol discovery mechanism has detailed information on how to get the base URL and form the request URLs for protocol discovery mechanism initiation.

III. CONCLUSION

With the emerging new applications in the field of secure elements, there soon will be requirement of a proper and common mechanism of information exchange and capabilities discovery between applications on secure application, application interacting with secure elements and applications that interact with applications that interact with secure elements. The discovery mechanism introduced and explained in this paper will allow the future applications an opportunity to work in collaboration with each other to provide versatile applications to users and increasing the use of secure elements for secure transactions and data storage

REFERENCES

- [1] GlobalPlatform System System Protocol Discovery Mechanism Specification(draft version)
- [2] GlobalPlatform Card Specification v2.2.1.
- [3] GlobalPlatform System – Messaging Specification for Management of Mobile-NFC Services v1.1.2.
- [4] 3GPP "Over the Air Technology" S3-030534
- [5] GSM 03.48: "Security mechanism for the SIM application toolkit.

AUTHORS

First Author – Sarabjeet Singh, Masters of Computer Applications, sarabjeet.singh2610@gmail.com.