



**Security and Trust  
in  
Mobile Applications**

**October 2008**

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>3</b>
<b>2</b>	<b>Secure Mobile Applications.....</b>	<b>4</b>
	2.1 Financial Applications .....	4
	2.1.1 Near Field Communications (NFC).....	4
	2.1.2 Mobile Financial Use Cases .....	6
	2.2 Access Applications.....	8
	2.2.1 Secure Pass.....	8
	2.2.2 Protected SMS.....	9
	2.2.3 One Time Password (OTP) Applets.....	9
	2.2.4 VPN and Secure Storage of Data.....	10
	2.3 Protection of Downloaded and Broadcasted Content.....	11
	2.3.1 Mobile DRM Use Cases.....	11
	2.3.2 Service and Content Protection for Mobile Broadcast Services .....	12
	2.3.3 Security Requirements.....	14
	2.4 Authentication Applications.....	15
	2.4.1 Extensible Authentication Protocol (EAP).....	15
	2.4.2 Generic Bootstrapping Architecture (GBA).....	18
	2.4.3 Public Key Infrastructure (PKI) and Certificate-based Authentication.....	20
	2.5 Identity Selection Applications.....	23
	2.5.1 Introduction .....	23
	2.5.2 Security and Trust Model of Identity Selector .....	24
	2.5.3 Identity Selector Applet Details .....	24
<b>3</b>	<b>Security Feature Requirement Matrix.....</b>	<b>26</b>
<b>4</b>	<b>Key Differentiators .....</b>	<b>27</b>
<b>5</b>	<b>Conclusion.....</b>	<b>28</b>
<b>6</b>	<b>Acknowledgements.....</b>	<b>29</b>
<b>7</b>	<b>References .....</b>	<b>29</b>

# 1 Introduction

Traditionally, the mobile application environment has been relatively easy to manage. Operators preloaded devices with some applications, and other applications were made available only via purchase directly from the service provider. In this paradigm, the applet space could be easily controlled by validating each applet on new devices before launch, thus ensuring the user a consistent and trusted applet experience, resulting in few legitimate security threats.

As the mobile applet space evolves, so does the complexity of managing the available end user applications. What was once a “Walled Garden” which the operators could easily control has now become a data connection rich space that offers the end user infinite application choices. Today, it is virtually impossible to be absolutely certain that every application run on a mobile device comes from a reliable source, and the introduction of open access networks magnifies the problem.

As a result, subscribers face threats such as identity theft, phishing, pharming and other attacks designed to compromise secure data, pirate content or even for example, take money directly from bank accounts. If the users cannot trust that their applications are secure and come from a trusted source, they will lose confidence not only in the applications, but also in the operator providing the service.

Ultimately, this can have a detrimental impact on the service provider’s business model. If subscribers do not see the applications as secure and trusted, interest in the application offering will be minimal or nonexistent. Consequently, the value of a content offering is directly proportional to the application’s security and the trust placed in the application providers. By extension, if security and trust in one network access technology is seen as far superior to others, it will have a competitive advantage in the mobile application marketplace.

A great deal of work has been done recently to address security and trust concerns in the mobile world. Most notably, the Open Mobile Terminal Platform published a paper entitled, “OMTP – Advanced Trusted Environment (TR1)” [1]. This paper discusses a comprehensive security environment for the mobile industry that addresses threats and business opportunities, highlighting currently available technology and guidance for terminal requirements so that operators and service providers can offer secure and trusted mobile applications.

One of the OMTP’s core security elements is the UICC, identified as the “...primary operator Asset in the UE...provid[ing] a trusted execution environment...and secure data storage facilities” [1]. The UICC ensures that sensitive data and applications are properly stored and processed. Furthermore, the UICC provides a portable and low cost trusted security token that is used ubiquitously across GSM evolutionary access technologies. Cryptography is at the core of the technology, and the security mechanisms are constantly improved to withstand the most strenuous attacks as verified by independent certification processes [2].

This paper illustrates the state of mobile applications in today’s market. The following sections highlight the key requirements for successful secure and trusted mobile applications, give specific examples of secure applications that are available today and

in the near future and identify the key differentiators for the GSM family of access technologies as compared to other networks.

## **2 Secure Mobile Applications**

### **2.1 Financial Applications**

The 3GSM infrastructure is well-suited to mobile financial applications, due in part to the robust user authentication mechanisms used to access the network, and in part to the UICC's highly secure storage for sensitive keys and security credentials.

Mobile financial applications take several forms, such as: banking applications, money transfers, balance checking, payment applications via either credit, debit, or prepaid methods; loyalty applications, including e-coupons and vouchers; or various types of ticketing applications. Transactions can occur remotely, as with web-based purchases, or at the point of sale using near-field communications (NFC) technology.

Mobile NFC is discussed below with emphasis on the UICC's role and relevant use cases that describe mobile payment, mobile banking and mobile ticketing applications.

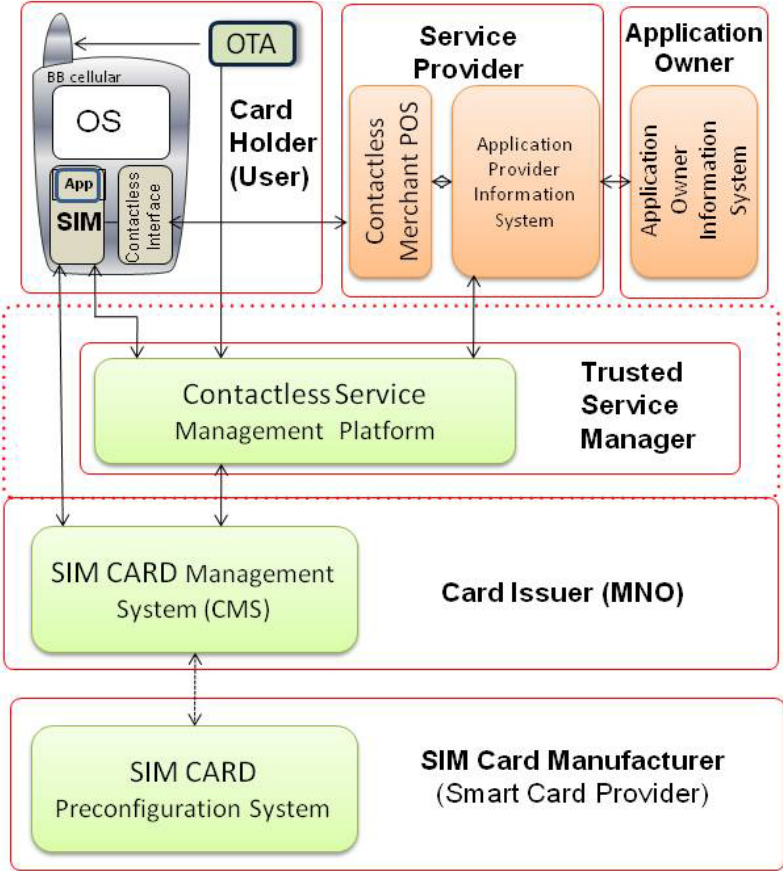
#### **2.1.1 Near Field Communications (NFC)**

NFC technology is evolving as a key enabler for mobile payment services. An outgrowth of the contactless card industry, mobile NFC allows the handset to communicate with close-proximity "card readers" allowing point-of-sale payments and/or authorizing the device owner to acquire services, such as access to public transportation. Mobile NFC business models are still being worked out among the stakeholders involved.

The mobile NFC ecosystem consists of the following elements:

- Mobile Phone with NFC Chipset and Secure Element
  - NFC Chipset contains the logic and interfaces to communicate with external card readers, NFC 'tags', or other NFC devices
  - Secure Element contains NFC 'card applications' from service providers and storage for content
- Mobile Network Operator (MNO)
- Service Providers
- Trusted Service Manager – single point of contact between service providers and MNOs. May perform life-cycle management of NFC applications on the phone

The diagram below shows the elements of the mobile NFC ecosystem (from the GSMA Mobile NFC reference architecture):



**Figure 1 – Mobile NFC Ecosystem**

The Secure Element (SE) is contained in embedded memory on the device, on a removable memory card, or on the Universal Integrated Circuit Card (UICC). The GSM Association has issued a position paper stating that the collective GSM community believes the UICC is the preferred option for the Secure Element [3][4]. Part of the reasoning for this recommendation is the well established security of the UICC, its portability between devices, and the ability of the mobile network operator (MNO) to access the SE in the event a device is lost or stolen. If the Secure Element is embedded in the UICC, the MNO can access and disable the SE through existing Over-the-Air (OTA) mechanisms, even if the stolen or lost UICC is used in any device. Such access to the SE is not generally possible if the SE is embedded in the phone or contained on a removable memory card.

Multiple applications from different Service Providers may be hosted within separate domains in the Secure Element. The device contains a User Interface to allow selection of a particular NFC application, or in some cases the appropriate application will be

automatically selected via the NFC protocol. The User Interface may be implemented via a Java applet, Subscriber Identity Module (SIM) Toolkit application, or the Smart Card Web Server.

Incorporating NFC into the GSM world offers advantages over conventional contactless cards. If the Secure Element is contained within the UICC, the contactless application can be portable from device to device and the content, certificates, applications, and keys within can be made secure from physical or electrical compromise. The mobile NFC application can be integrated with the mobile data network as an additional security measure. This means that applications can be updated and maintained over the air, and more importantly, disabled in the event a device is lost or stolen. Static card readers are often not connected to any network (e.g., mass transit turnstiles) and cannot authenticate a static card holder. A mobile NFC device, however, can provide user authentication.

There is some discussion within the industry regarding the certification of the security mechanisms employed by the systems that will be used to remotely manage the payment applications and the tokens stored in the Secure Element. As of the writing of this paper, there is no published specification for the certification of the remote management of these applications. It is currently being developed and should be available in the near future.

### **2.1.2 Mobile Financial Use Cases**

The existing GSM infrastructure enables a variety of mobile financial applications. A selected few are described below. These applications either use the UMTS/HSPA (Universal Mobile Telecommunications System/High Speed Data Access) infrastructure or NFC or a combination of the two.

#### **2.1.2.1 Mobile Payment**

Sometimes referred to as mPayment or mCommerce, mobile payment includes many types of payment transactions. There are well-established examples, like digital services and MNO-related content purchases, such as ringtone purchases, handset wallpaper, or downloading and purchasing music, video, or games. Information services are another example, wherein a purchaser subscribes to a news, weather or traffic service. These latter services may be offered in conjunction with location-based service (LBS). A good example is traffic information where an accurate location is required, however for weather information, the user's location could be a rough estimate.

Typically the purchases are aggregated to the mobile subscriber's monthly bill (Direct Operator Billing), but in other cases payment is made during the transaction. Examples of this are internet-based purchases using WAP (Mobile Web Payments), or purchases through online auctions or brokerage services. These types of payments can be made via a credit or debit "card" which are actual credentials stored on the user's UICC, or via debiting a pre-paid account (possibly stored on the UICC).

There are Point of Sale purchases made using NFC and credentials that are stored in the Secure Element. An interesting paradigm that is emerging with regard to point of sale purchases is Just in Time shopping. This is where a subscriber uses their mobile

device to “book” an item to be picked up later. Payment can be made at the time of booking or at pickup.

Mobile payment models can be categorized into two broad areas: Micropayment and Macropayment. Conceptually, it is difficult to assign a monetary threshold to differentiate micro and macropayments, so typically the distinction between the two is that micropayments are of an amount small enough that user authentication is not required to complete the transaction. Micropayments combined with Direct Operator Billing provide a potentially high-value revenue source for MNOs and service providers.

#### 2.1.2.2 Mobile Banking

Mobile Banking gives users access to their bank accounts from their mobile devices. They may check balances, transfer funds electronically, top-up prepaid tokens and possibly even obtain cash from ATMs. Mobile Banking is still in the early stages of being implemented in Europe and North America.

However, there are several Mobile Banking applications deployed, primarily using either J2ME applets on the mobile device or using a SIM Toolkit Application resident on the UICC.

One such example from a Mobile Banking solution currently in service in Latin America allows subscribers to manage their personal account information via a mobile application. The application allows the user to transfer funds between accounts, retrieve balance and transaction information, make credit card payments, pay bills and recharge their mobile prepaid account. All of these services are managed via a Card Application Toolkit that is PIN protected to ensure that only the subscriber can access the application functions and the bank account information.

#### 2.1.2.3 eTicketing

eTicketing is a rapidly growing area for mobile financial applications, as a well-established precedent exists with widely deployed NFC mass transit cards. Extending this to the mobile domain is obvious; the device is used in NFC mode to pay for access to mass transit and ground transport systems. Typically the device will contain either a prepaid ‘eWallet’ which is debited on each use, or the user pays for a monthly subscription and the NFC transaction would verify the validity of the stored access credentials.

The mobile device is used to receive and render eTickets for airlines, sporting events, parking, etc. Ticket purchase is made either using the mobile device or ‘offline’ via some other means and the ticket voucher is sent to the device for display or transfer to the admitting agent. One method for transfer and display is bCODE, which uses SMS and the device screen. The voucher is stored in the UICC, where it is transferred from device to device and stored securely.

This same technique is used to send and store coupons, vouchers and loyalty tokens. Both businesses and consumers benefit from eliminating bulky paper coupons or plastic cards and are more inclined to patronize the participating businesses. Loyalty tokens work like a coffee shop punch card or airline frequent flyer miles; where the patron is

rewarded with freebies, after multiple purchases. Loyalty tokens have tangible value and are securely stored on the UICC enabling portability from device to device.

## **2.2 Access Applications**

Access Applications are any applications that access service via password, login or other user credential before service is granted.

Included are secure pass applications (storage of a subscribers passwords, login credentials, etc.), protected SMS applications (secure messaging), One Time Password (OTP) applications (single secure identity that opens access to a multiple services/applications), and Virtual Private Network (VPN) applications (secure access to a corporate intranet via the public domain).

The following sections briefly describe these access applications, highlight their key requirements and explore the value proposition from the service provider and end-user's perspective.

### **2.2.1 Secure Pass**

Secure Pass surely and simply stores all employees' passwords, IDs, and personal information in one single folder, thereby preventing security-focused enterprises from having to use multiple and complex levels of access and authorization codes.

This application allows employees to store confidential information in a special folder that is secured through a master password, which can then be added, modified, or deleted at any given time. It also stores information such as bank accounts, internet banking passwords, account numbers, credit card data, PIN information, email account information or any other protected user data.

#### **2.2.1.1 Key Features**

These applications manage a user's identities and passwords thus simplifying access to protected services. The subscriber's confidential information is stored in a special folder secured through a master password. The subscriber's information can be added, modified or deleted at anytime once the special folder is accessed using the master password.

#### **2.2.1.2 Operator Benefits**

The operator benefits associated with Secure Pass applications include increased service usage by simplifying access to the service and building user loyalty, since the storage of sensitive data augments the mobile phone's value. Finally, these applications are easy to implement with minimal operator investment required.

#### **2.2.1.3 End User Value**

First and foremost, Secure Pass applications are easy to use and make accessing secure data effortless for the user. Additionally, they minimize the number of access codes, passwords and locations where secure data is stored on the device. The application can be configured so that with one PIN or password, diverse confidential data is stored in one secure folder.

## 2.2.2 Protected SMS

Protected SMS or Secure Chat application provides the end users with secure “one-to-one” communication via binary SMS messages. These messages are protected using different encryption methods (e.g. 3DES) and stored in a specific inbox with limited access through an application PIN or password.

### 2.2.2.1 Key Features

The Protected SMS messages use data encryption to secure the message from spying or snooping. Access to the secure message is protected via a specific PIN code. The generation of the encryption key used for transport is based on user input, guaranteeing a unique key for each user’s service.

### 2.2.2.2 Operator Benefits

From an operator’s perspective, Protected SMS applications increase services usage by allowing subscribers to use SMS in circumstances where message security is a concern. Furthermore, these applications build subscriber loyalty, since the storage of sensitive data adds more value to the mobile phone. Since these applications rely on existing technology, they are easy to implement with limited operator investment required.

### 2.2.2.3 End User Value

The essential value of Protected SMS applications for the end user is end-to-end SMS protection for the text or data that is exchanged during the session. An added benefit for the subscriber is the secure storage of these messages which can be PIN protected for added local access security.

## 2.2.3 One Time Password (OTP) Applets

OTP solutions are scalable network identity platforms that deliver the strong authentication that consumers require to use sensitive on-line services confidently. It uses a PIN and the handset SIM to authenticate the user and generate a one time password for an individual transaction. When the user enters the password, the validation server also generates a password for comparison, if it matches, access is granted.

There are three basic interfaces for OTP applications:

- a) **Voice Channel:** As a call is placed the destination number is routed to the application for the addition of an OTP. As the call is routed, the session password is validated by the network or the called party before the call context is established.
- b) **SMS Channel:** This is the standard channel for the exchange of encrypted information.
- c) **MMI Channel:** An OTP is displayed on the mobile screen, and later copied into certain web-forms (depending on the application size, this feature might be later enhanced with the possibility of sending an OTP via IrDA or Bluetooth).

Additional interfaces are available using SCWS technology, e.g. bar code.

#### 2.2.3.1 Key Features

OTP applications transform the handset into a secure authentication token. It uses OATH OTP applet and secret UICC stored keys to generate a one-time password with one keystroke. The UICCs secure file system and crypto-coprocessor provide strong security (HMAC-SHA-A cryptographic standards, Strong Authentication Agent Software, etc) that are attack resistant.

#### 2.2.3.2 Operator Benefits

OTP solutions position the operator at the leading edge of on-line security and pave the way for new partnerships with banks and other institutions. Furthermore, the applications generate new potential revenue opportunities in the corporate sector by securing access to sensitive information.

#### 2.2.3.3 End User Value

OTP applications inspire user confidence by protecting application access with a PIN code and by virtue of the single session password. Additionally, OTP solutions make life simple since there is no need to remember special procedures, carry tokens or memorize complicated passwords. Finally, these applications open the online world, allowing users to make the most of internet and mobile services.

### **2.2.4 VPN and Secure Storage of Data**

When an organization has to connect networks that contain sensitive and proprietary data to the Internet for remote access, the increased connectivity exposes a significant security risk. In the potentially hostile Internet environment, the VPN solution becomes critical because in addition to potential operational savings it maintains the security associated with a private network infrastructure. A VPN solution provides security because it uses a secure tunneled connection, encrypting data and allowing only authenticated users to access the corporate network.

VPNs support a wide range of authentication methods, tunneling protocols, and encryption technologies to maintain business data security.

A VPN application package is comprised of handset software and a smart card combining UICC and enterprise security functions (in a PC environment, a smart card reader will be also required in addition to the software).

#### 2.2.4.1 Key Features

VPN applications store a user's public and private keys and the associated public key certificate, retrieve the public key certificate and perform private key operations on behalf of the user. These keys and certificates are managed via a standalone PKCS#11 application typically running on the UICC.

#### 2.2.4.2 Operator Benefits

VPN applications position the operator at the leading edge of on-line security and open new revenue streams in the corporate sector that were unavailable to the operator previously.

#### 2.2.4.3 End User Value

For the end-user, VPN applications allow for direct secure access to the corporate data from the handset or from the PC without keeping track of any additional token as is the case with most solutions available today.

### **2.3 Protection of Downloaded and Broadcasted Content**

Today, multimedia mobile phones, carrying audio and video players, gaming and mobile TV platforms enable consumers to download protected content, such as ring tones, logos, games, music, video, and/or receive broadcasted content, such as TV or movies, at will. In the absence of proper “Digital Rights Management” (DRM) systems providing controlled consumption of digital content, authors and producers risk losing important revenues if their intellectual property rights are not sufficiently protected.

To control the distribution and consumption of downloaded digital media objects, the Open Mobile Alliance (OMA) has provided appropriate “Digital Rights Management” (DRM) specifications to protect authors and content providers’ intellectual property rights from unauthorized attempts to copy and reuse it by the customer or any third party.

The specification Mobile Broadcast Services Enabler Suite of OMA (BCAST) provides the requirements for protection of broadcasted content and associated services.

Those specifications define the fundamental building blocks and complete security infrastructure necessary for robust end-to-end DRM and broadcast content systems, designed to support various business and usage models.

#### **2.3.1 Mobile DRM Use Cases**

OMA DRM version 2 [5] enables several use cases covered by the specification. This paragraph outlines few of them.

##### 2.3.1.1 Basic Download Use Case

A DRM system’s fundamental use case is content downloading. The end user navigates to a content provider portal and then initiates the download and purchase. Afterwards, the DRM Agent located on the mobile device contacts the corresponding Rights Issuer (RI) to get authorization to download the Rights Object (RO). By the introduction of a DRM Agent, the OMA DRM specifications enforce permissions and control access to DRM Content at the point of consumption i.e. the Mobile Equipment. Ultimately, the DRM Agent extracts the Content Encryption Key (CEK) from the RO, decrypts the DRM Content and sends it to the customer’s multimedia player for rendering, applying the permissions specified in the RO. Once the end user has downloaded the content, he may wish to install it for automated use (e.g. ring or alarm tone).

##### 2.3.1.2 Streaming of DRM-protected Content

The previous download use case assumes that content is packaged and delivered in its entirety. Alternatively, content may be packaged and delivered as a stream to see, for example, a concert or TV show. In such a case, the stream itself is encrypted. It can be controlled through the same procedure as described previously. A RO is generated, the

encryption key to access the encrypted stream is put in the RO, and the RO is then bound to a DRM Agent.

### 2.3.1.3 Sharing Content use case (Super Distribution and Domains)

An end user may forward protected content to a friend via Bluetooth®, IrDA®, messaging or other means, if associated permissions allow forwarding it. The friend receives the protected content but not the rights to render it. A browser session is opened when the friend attempts to unlock the content and he will be directed to the Rights Issuer Portal from which rights can be obtained. The RI controls whether release of a new RO to the new DRM Agent is necessary.

The basic model of OMA DRM binds the RO and CEK to a specific DRM Agent. The domain model based service offered by the Service Provider extends this notion, allowing a RI to bind RO and CEK to a group of DRM Agents. An end user may then share a specific DRM content off-line with other mobile equipment, either owned by him or belonging to friends.

## 2.3.2 Service and Content Protection for Mobile Broadcast Services

To control distribution and consumption of broadcasted content, the Digital Video Broadcasting (DVB) organization has specified several standards defining a common interface (DVD-CI) at both the transmission site and the receiver, scrambling algorithms (DVB-CSA), and communication protocols.

OMA BCAST has specified the Service Protection and Content Protection systems enabling broadcast services delivery [6]. Content Protection ensures the protection of the content (files or stream) during the complete life time of the content, i.e. at the time of delivery, playing and storage. Meanwhile, Service Protection ensures the content access control protection at delivery.

In traditional pay-TV, the concept of conditional access (CA) is used for content protection. The CA approach gives the content owners assurance that their material is not illegally accessed. It doesn't, however, control the reuse of received and possibly stored content. While this may be adequate in fixed location TV viewing, for mobile TV a more sophisticated approach might be required.

Two different security mechanisms are used to provide service and content protection: the DRM Profile and the Smartcard Profile. Both protection mechanisms use a 4-layer model using different key management systems. In order to ensure maximum interoperability, OMA BCST defines a common layer for traffic encryption (Layer 4) and implements the other key management layers using either the DRM Profile or the Smartcard Profile.

### 2.3.2.1 DRM Profile

OMA BCAST DRM Profile relies on OMA DRM v2.0 for the key material exchange based on Public Key Infrastructure (PKI) and RO management.

Layer 1 is for registration and uses the private/public key pair stored in the BCAST terminal. The public key secures the delivery of the Rights Encryption Key (REK), and with the corresponding private key processes the Generalized Rights Objects (GROs).

The registration is done over the interactive channel using the OMA DRM v2 mechanisms or using a specific protocol set for Broadcast and out-of-band channels. This profile supports terminals without an additional interactive channel.

The Long Term Key Messages (LTKM) are delivered on Layer 2 over the broadcast or interactive channel. The LTKM transports the Service or Program Encryption Key (SEK/PEK), as well as permissions and attributes. SEK/PEK is encrypted using keys delivered or broadcasted during the Layer 1 registration procedure.

Layer 3 transports short term keys, i.e. the Traffic Encryption Keys (TEK), in the Short Term Key Message (STKM). STKMs are distributed over the same channels used by the corresponding content. The TEKs are encrypted by a SEK or PEK.

The actual content is encrypted on Layer 4 using the TEKs and using different mechanisms depending on the actual encryption method used. For file download delivered over the broadcast channel, Content Protection must follow the OMA DRM v2.0 specification. Service protection of download data uses DCF (DRM Content Format) and encryption by TEK or IPsec (Internet Protocol Security standard). For real time broadcast streaming using RTP (Real-time Transfer Protocol), protection is based on IPsec, SRTP or ISMACryp.

#### 2.3.2.2 Smartcard Profile

OMA BCAST Smartcard Profile is a smartcard based solution relying upon 3GPP Multimedia Broadcast Multicast Service (MBMS) [7] for the keys management system based on the symmetric key model. It requires an interactive channel at all times to manage registration and key material.

Two variants of the Smartcard Profile are specified: the (U)SIM Smartcard Profile and the (R-)UIM/CSIM Smartcard Profile. They differ in Layer 1 where Subscriber Key is established. Layers 2, 3 and 4 are the same.

Layer 1 establishes the keys securing communication between the BCAST Subscription Management (BSM) and the Smartcard. The layer 1 key, i.e. the Subscriber Management Key (SMK) is generated using the Generic Bootstrapping Architecture (GBA) for the (U)SIM Smartcard Profile or is derived from the "SmartCard Key" (SCK) for the (R-)UIM/CSIM Smartcard Profile. The SCK stored on a Smartcard based identity module, is a pre-provisioned secret key that is shared between the Smartcard and the Smartcard issuer. The SMK is stored on the Smartcard or the terminal depending on the Smartcard Profile key management implemented.

Depending on the service configuration, within the LTKM a Program Encryption Key (PEK) or a Service Encryption Key (SEK), used respectively for pay per view or subscription customers, is delivered protected by SMK in Layer 2.

Layer 3 delivers the Short Term Key Message (STKM) within Traffic Encryption Keys (TEKs) which are protected using SEK or PEK, and optionally by a Terminal Binding Key (TBK).

Layer 4 encrypts traffic using the TEK for stream or file delivery respectively for both service and content protection. For file downloads over the broadcast channel, the protection is based either on DCF or on IPsec as in the OMA BCAST DRM Profile. For

real time broadcast streaming using RTP, protection is based on either SRTP or ISMACrypt encryption.

BCAST DRM and Smartcard Profiles, when relying upon either transport layer security with SRTP, or IP layer security with IPsec, allow content format flexibility since it is protected at a lower OSI stack layer.

BCAST DRM and Smartcard Profile, when relying upon ISMACrypt content layer security, allow the solution to be completely independent of the intrinsic transport or IP layers security.

### **2.3.3 Security Requirements**

Even if a solution is compliant with DRM standards specifications deployed in the mobile world, its implementation is still sensitive and vulnerable to security threats. To facilitate consistency in DRM implementations, Open Mobile Terminal Platform (OMTP) – the mobile operator led forum – provided a set of specific terminal requirements to facilitate improved security consistency for service deployment of based on the OMA DRM v2 specification [8].

In order to achieve the full DRM commercial potential, while not impacting the end user experience, it is important to provision the mobile equipment with keys and certificates, to maintain the security of Rights verification (authenticity, integrity), content decryption and rendering (confidentiality), and to sustain the multimedia performances compatible with the most advanced formats (e.g. HD).

The RO is protected using a Rights Encryption Key (REK) for the sensitive parts such as CEK. The RO is digitally signed by the RI. During delivery, the REK is cryptographically bound to the target DRM Agent. Only in this manner does the authorized target DRM Agent access the RO and thus the CEK.

In addition, the DRM Agent requires updating of state permissions, i.e. remaining plays authorizations, rights expiration date or play time constraints. Thus, a RO containing such permissions and content may be stored in a mobile device's secure non-volatile memory or in the (U)SIM Card. It is assumed that the mobile device is provided with efficient and secure cryptographic mechanisms, some monotonic counters and trusted date and time sources.

In May 2008, the OMTP published in the document: "Advanced Trusted Environment (TR1)" [1], a set of advanced security terminal requirements. The combined expertise of operators, manufacturers and chipset vendors, software and hardware suppliers provided a secure foundation on which to build services that can run with confidence.

OMTP TR1 extends the work of the "Basic Trusted Environment (TR0)" published in 2006 [9], by outlining the requirements for new security sensitive customer deployment applications, such as mobile broadcast of premium TV and film content.

OMTP TR0 provides the basic security requirements for debugging, the mobile device ID (e.g. IMEI) protection, subsidy lock protection, secure booting and secure Flash updating and the basic DRM security requirements. OMTP TR1 defines an enhanced level of security addressing the pertinent threats posed by embedded hackers.

OMTP TR1 defines an enhanced set of security requirements addressing the pertinent threats applicable to sensitive assets of the platform, including technological enablers, such as Trusted Execution Environments, Secure Storage facility or Secure Link between the Mobile Equipment and the UICC, and use cases, such as Broadcast Service Protection.

## **2.4 Authentication Applications**

### **2.4.1 Extensible Authentication Protocol (EAP)**

The EAP [10] is a generic protocol, which supports multiple authentication methods. In other words, with EAP the user and network can authenticate each other using any authentication mechanism that supports mutual authentication. With that the specific mechanisms are so effective that introducing a new mechanism requires no protocol change. The authentication methods specify the informational elements relevant to specific mechanisms to be carried in the EAP message. This makes the EAP protocol extremely flexible in that once it is implemented and tested the software needs no changes. Thus, EAP is a true plug-and-play protocol.

Historically Internet authentication occurs in the link layer (such as Point-to-Point Protocol [PPP]) between the Internet and Network Access Server (NAS). For this reason EAP still typically runs directly over data link layer protocols (e.g., PPP, Ethernet). The IETF has standardized the EAP protocol. The EAP methods are being standardized in the IETF EAP Method Update (EMU) working group. The group has standardized only the EAP-TLS method so far, but the industry has developed several authentication methods.

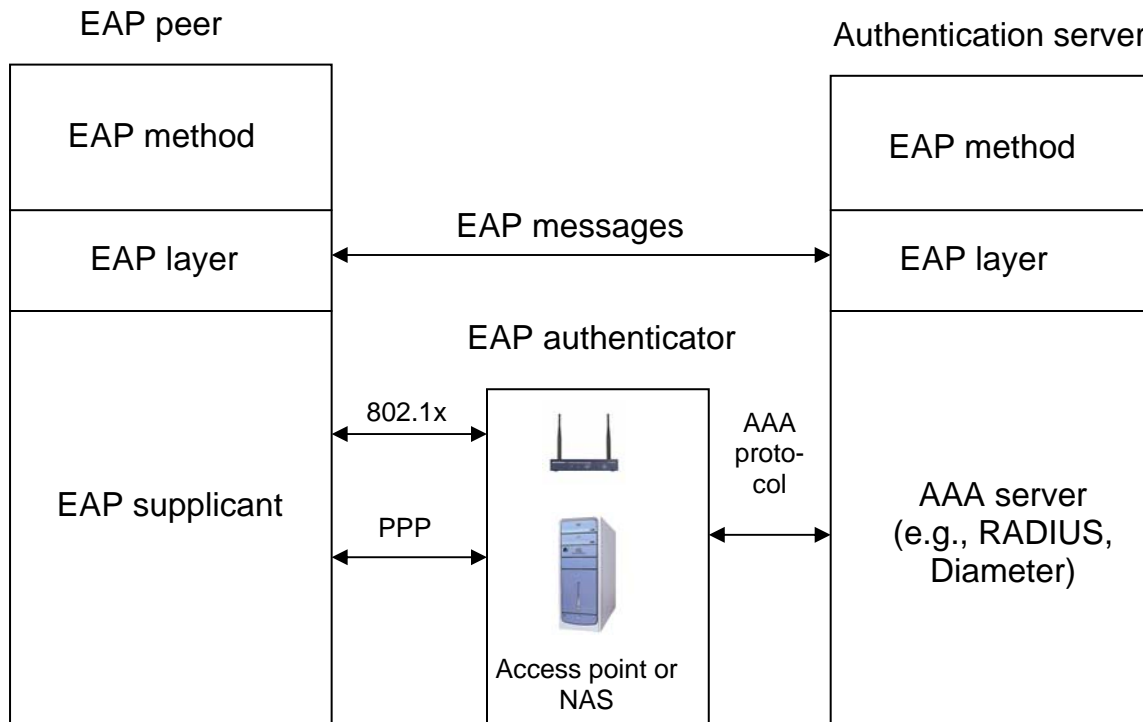
The sections following describe EAP architecture and EAP methods.

#### **2.4.1.1 EAP Architecture**

The peer and authenticator entities start the EAP exchange. The latter is the link end point (e.g., access point or NAS) that initiates EAP authentication, the former is the link end point that responds to the authenticator using a supplicant, a software component, which enables an EAP peer to communicate the EAP packets over a link layer protocol (e.g., PPP or 802.1x).

The actual authentication is done by the EAP server, the entity that implements an authentication method and terminates the authentication exchange with the peer. The EAP server can be a part of an authenticator or a part of a back end authentication server, an entity that runs an Authentication, Authorization and Accounting (AAA) protocol. Typically the AAA server is a Remote Authentication Dial-In User Service (RADIUS) or a Diameter server.

Figure 2 below depicts a typical EAP architecture where the EAP server is a part of the authentication server.



**Figure 2 – EAP architecture**

#### 2.4.1.2 EAP Methods

Multiple EAP methods have been developed by the industry. This paper describes the EAP-SIM, EAP-AKA, EAP-TLS, and EAP-TTLS authentication methods.

##### 2.4.1.2.1 EAP-SIM Authentication Method

EAP-SIM was developed by the IETF in support of 3GPP, and is described in the RFC 4186. The document specifies the use of a SIM for mutual authentication and session key distribution in GSM. The EAP-SIM mechanism specifies enhancements to GSM authentication and key agreement. It also includes network authentication, user anonymity support, result indications, and a fast re-authentication procedure. The use of the EAP-SIM method is described in 3GPP Technical Specification TS 33.234, *Wireless Local Area Network (WLAN) Interworking Security*.

##### 2.4.1.2.2 EAP-AKA Authentication Method

EAP-AKA specifies an EAP method that is based on the Authentication and Key Agreement (AKA) mechanism used in third generation mobile networks. The method has also been developed by the IETF in support of 3GPP, and is described in the RFC 4187. The document specifies an EAP method for mutual authentication and session key distribution that uses the third generation Authentication and Key Agreement

mechanism. The use of the EAP-AKA method is described in 3GPP Technical Specification TS 33.234, *Wireless Local Area Network (WLAN) Interworking Security*.

The AKA is specified for Universal Mobile Telecommunications System (UMTS) in 3GPP Technical Specification TS33.102 and for CDMA2000 in 3GPP2 standard S.S0055-A. The UMTS and CDMA2000 are global third generation mobile network standards that use the same AKA mechanism.

#### 2.4.1.2.3 *EAP-TLS Authentication Method*

The EAP-TLS method has been developed by the IETF, and is specified in the RFC 5216.

The EAP-conversation typically begins with the authenticator and the peer negotiating the EAP as described in the EAP specification. After that the authenticator sends an EAP-Request/Identity packet to the peer and the peer will respond with an EAP-Response/Identity packet to the authenticator containing the peer's user identifier. From this step forward, the EAP converses between the peer and the EAP server, which may be located at the back end authentication server or be a part of the authenticator. When the EAP server is located at the back end server, the authenticator encapsulates the EAP packets in the protocol packets that run between the authenticator and the back end authentication server (e.g., RADIUS, Diameter).

The EAP-TLS method leverages the handshake protocol defined in the Transport Layer Security (TLS) protocol. The protocol enables the peer and the server to authenticate each other using digital certificates. Providing support for digital certificates, in addition to mandating support for Certificate Revocation Lists (CRLs), EAP-TLS also strongly recommends support for the Online Certificate Status Protocol (OCSP), which defines a mechanism that outperforms traditional CRLs interrogation.

After a successful authentication, the peer generates a pre-master secret key by encrypting a random number with the server's public key and sends it to the server. Both peer and server use this key to generate the shared keys providing confidentiality and integrity.

#### 2.4.1.2.4 *EAP-TTLS Authentication Method*

Similarly to EAP-TLS, the EAP-TTLS method employs the TLS handshake protocol for authenticating an authentication server to a peer, but the method does not require peer authentication to the server using the digital certificates. Such authentication is done after a secure connection (TLS tunnel) is established between the peer and the server as a result of the handshake. This secure connection enables the server to authenticate the peer using other authentication mechanisms (e.g., legacy password-based mechanism). Thus, EAP-TTLS allows the use of the widely-deployed legacy mechanisms by providing additional protection (i.e., against the eavesdropping and man-in-the-middle attacks).

The EAP-TTLS method also establishes the keying material that secures the data connection between the peer and the access point or (Network Access Server) NAS.

## 2.4.2 Generic Bootstrapping Architecture (GBA)

The Generic Bootstrapping Architecture (GBA) specifies a framework for bootstrapping authentication and establishing key agreement. The GBA, leveraging the 3GPP Authentication and Key Agreement (AKA) mechanism, enables authenticated User Equipment (UE) access to the Network Application Function (NAF) services.

The GBA authentication procedure results in a secret key being shared by the UE and NAF. This shared key can be used for mutual authentication of the UE and NAF, and protected data exchange between these entities. In addition, the GBA protects user's privacy by supporting authenticated access to the NAF services without revealing the user's identity to the NAF.

The GBA also supports Single Sign On (SSO). The SSO is accomplished by repeating the GBA authentication procedure for authenticating the UE to multiple NAF's without requesting a user's login credentials.

The following section provides a high-level description of the GBA.

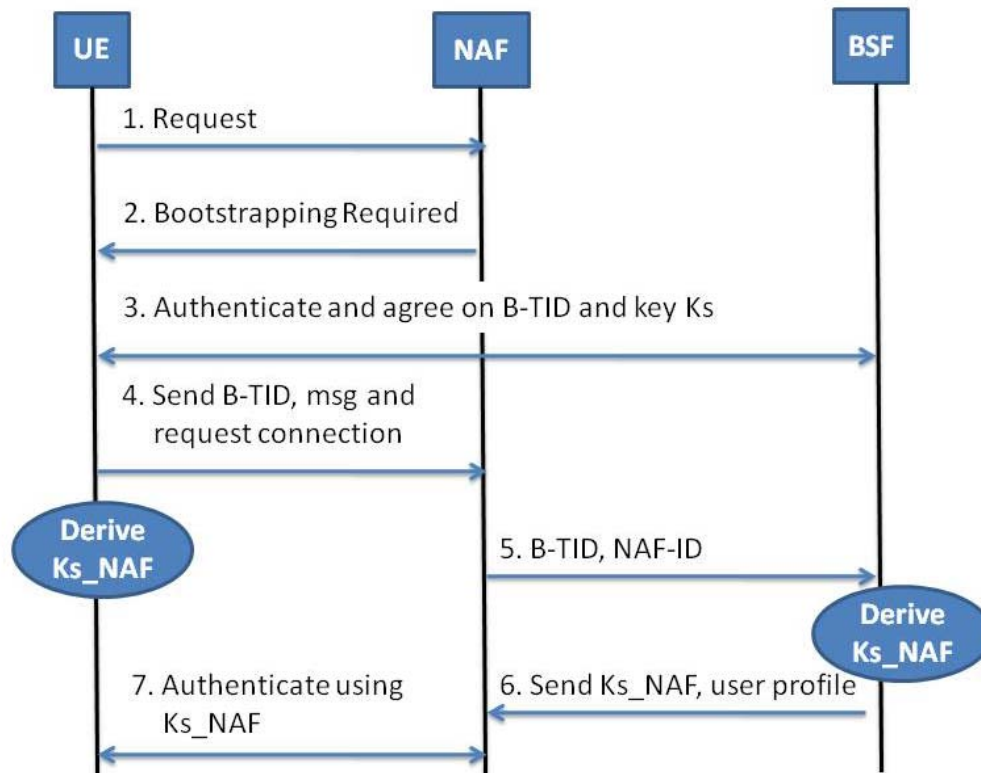
### 2.4.2.1 The Basics of GBA Authentication

The GBA includes three major entities:

- An end-user who is trying to obtain network services using User Equipment (UE)
- An application server (called Network Application Function or NAF)
- A trusted entity (called Bootstrapping Server Function or BSF), which authenticates and shares keys between two other entities.

The GBA authenticates a user, who is using UE, to an application server (NAF) without revealing the user's long-term credentials and secrets to the NAF by using a trusted entity BSF.

The Technical Specification ETSI TS 133 220, *Generic bootstrapping architecture* specifies the GBA. The GBA authentication and key agreement process basics are illustrated in Figure 3 and described below.



**Figure 3 – The GBA authentication process**

1. User initiates communication with NAF by sending request from UE without any GBA-related parameters.
2. The NAF replies with a bootstrapping initiation message.
3. The UE contacts the Bootstrapping Server Function (BSF) and both parties proceed with authenticating each other using long-term credentials and the end-user's profile. The authentication process involves communication between the BSF and the back-end authentication databases, Home Subscriber System (HSS). To locate the HSS that holds the user's information, the BSF may need to query the Subscriber Location Function (SLF), and this step is not shown. The HSS provides the user's authentication information to the BSF. As a result of a successful authentication procedure, which is based on the 3GPP AKA protocol, UE and BSF share a secret key (Ks) and a Bootstrapping Transaction Identifier (B-TID).
4. The UE sends an application request to NAF, which includes B-TID and msg (msg denotes here the application-specific data). It also derives from the key Ks a key Ks\_NAF, which it uses in communications with the NAF.
5. The NAF sends B-TID to the BSF along with its own ID (NAF-ID).

6. In the authentication answer, BSF sends to NAF a key  $Ks\_NAF$ , which it had derived from  $Ks$ , and the application-specific user security settings.
7. Finally, UE and NAF can authenticate each other using a shared key  $Ks\_NAF$ . The exact authentication procedure depends on the protocol between the UE and NAF. For instance, the GBA specifies that HTTP-based applications can use either HTTP Digest authentication (RFC 2617) or TLS pre-shared key ciphersuites (RFC 4279).

### **2.4.3 Public Key Infrastructure (PKI) and Certificate-based Authentication**

The certificates are commonly used for authentication of the network entities.

This section describes the fundamentals of authentication based on the certificates defined in ITU-T Recommendation X.509.

#### **2.4.3.1 Description of Authentication Certificate**

A certificate is a digital document that includes an entity's identifier, its attributes, an entity public key, and other authentication information (i.e. information on the certificate issuer, Certificate Revocation List [CRL], starting and ending dates and times of the certificate's validity, etc.).

The description of the fields of X.509 certificate is provided in Table 1. A certificate is digitally signed by a trusted third party, which is called the Certification Authority (CA). The CA computes a hash (i.e. SHA-1) of all the fields except the field Signature Value, signs it with its own private key, and then adds the signature to the certificate in the Signature Value field.

Name of a field	Description
Subject	Identifier of the certificate holder
Serial Number	A unique identifier of the certificate
Issuer	The name of the party issued the certificate (the name of the CA)
Valid From	Starting date and time of the certificate validity
Valid To	Ending date and time of the certificate validity
Public Key	The public key of the certificate holder
Version	Version of the X.509 on which the certificate is based
Subject Alternative Name	Another certificate holder identifier
CRL Distribution Points	URL of the Certification Authority's CRL
Authority Information Access	URL to CA information
Enhanced Usage Key	Description of the certificate uses (list of the ISO-defined object identifiers [OIDs])
Application Policies	The applications and services that can use the certificate (specified by the OIDs)
Certificate Policies	CA policies and mechanisms used to receive a request for, handling, authorizing, issuing, and managing the certificates
Signature Algorithm	Algorithm used by the CA for computing the certificate's signature (e.g., RSA, DSA)
Signature Value	The actual certificate signature

**Table 1 – Fields of an X.509 certificate**

A recipient can validate the certificate signature using the certificate specified algorithm and the CA's public key, and then compare the result to a computed digest of the certificate. If they match, the recipient can trust the certificate's holder information as long as it trusts the CA that has issued the certificate.

### 2.4.3.2 Certificate-based Authentication

The use of authentication certificates is demonstrated in an example where user A wishes to send an authenticated message to user B.

This example relies on the assumption that user B has obtained a certificate from a trusted CA, which identifies user A as the certificate's holder and that B has verified the CA's signature, checked the validity dates, and the CRL. The paragraph below describes how user B can verify that the message is from user A.

For B to authenticate the message, user A encrypts a digest  $MD(P)$  of his plain text message  $P$ , with his private key  $D_A$ .

Then user A sends the result of the encryption,  $D_A(MD(P))$  to user B along with the plain text  $P$  itself. That is, user A sends the message  $P, D_A(MD(P))$  to B.

Upon receiving this message B decrypts the encrypted part using A's public key  $E_A$  to obtain a value  $MD'(P) = E_A(D_A(MD(P)))$ .

After that B computes the digest  $MD(P)$  and compares it to the value  $MD'(P)$ . If they match, B is assured that the digest had been signed with user A's private key, because there is only one private key  $D_A$  that corresponds to a public key  $E_A$ , and B knows that  $E_A$  belongs to A. In addition B has verified that the text  $P$  has not been altered.

### 2.4.3.3 Hierarchy of the Certification Authorities

The ITU-T Recommendation X.509 defines the Public Key Infrastructure (PKI) model, which specifies the CAs hierarchical relationship. The top-level CAs (known as root CAs) do not have any CAs above them and sign their own certificates. These CAs are well-known and trusted by the users.

The CAs below the top-level CAs have their certificates signed by the root CAs. The users that trust a root CA should trust one of its subordinate CAs at a level  $n$  if they can verify all certificates in the chain above the level  $n$ , which ends with the root CA.

The concept of the hierarchical relationship of the CAs is illustrated by Figure 4, where the red arrows indicate a chain of trust. The figure also illustrates the fact that there are many Root CAs. Each Root CA may have its own tree of the subordinate CAs, but only one bunch is depicted by the figure.

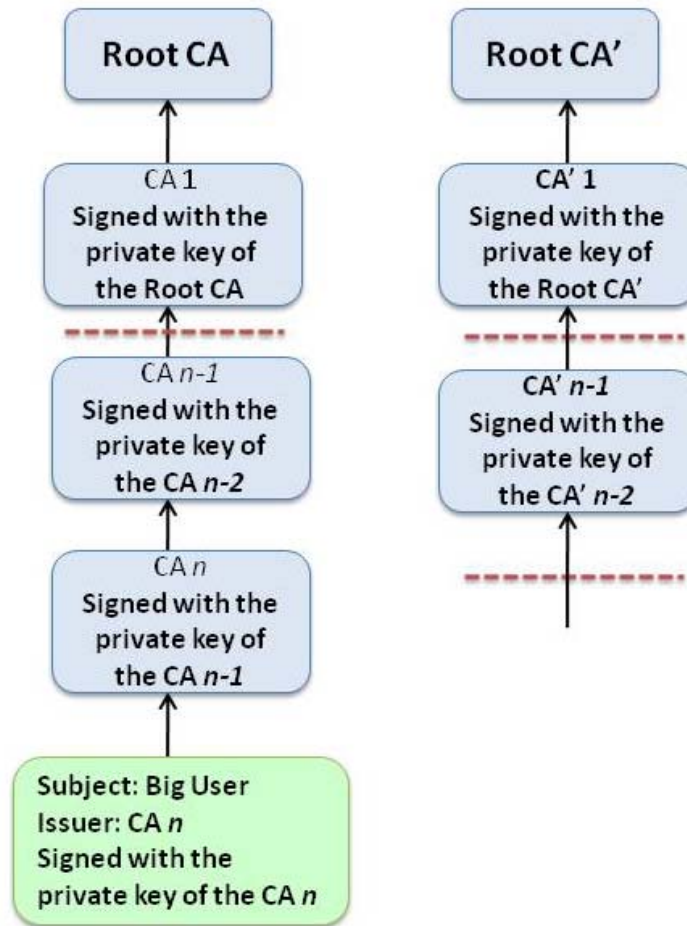


Figure 4 – Hierarchical relationship of the certification authorities

## 2.5 Identity Selection Applications

### 2.5.1 Introduction

An identity selector is a user-centric identity management application that provides a consistent user experience for authentication with a service provider or Relying Party (RP). The identity selector design concept was originated from the Windows CardSpace, which is part of Microsoft's new identity metasystem architecture. A few open source options, such as Project Higgins and Bandit DigitalMe, also provide vendor neutral identity selector development options. It is user-centric as users can manage their identities the same way they manage credit cards in their wallet. MNO can deploy various security tokens with different assurance levels in the identity selector based on different security needs of mobile applications. The users don't need to remember different passwords once the identities have been created. The identity selector design can also help solve Internet phishing problems which are described below.

Identity selector is a consistent way to work with multiple digital identities using Information Cards (InfoCards). Each InfoCard is an XML document containing identity metadata that processes authentication with an Identity Providers (IP). The RP provides a policy on the credentials required to access its services. The OASIS WS-Trust standard is the de-facto identity selector standard used to manage the security and trust, as well as achieving interoperability among different identity selector platforms.

### **2.5.2 Security and Trust Model of Identity Selector**

Since InfoCard only contains identity metadata, there is little security risk if the InfoCards are stolen. The identity selector architecture also provides a solution to the phishing problem as the user is not directly conducting the authentication process with the RP. The authentication process takes place between RP and IP where the identity selector is a trust broker for the authentication process using security tokens. The identity selector also requires mutual authentication between PR and IP to enforce the trust.

The identity selector is complementary to both SAML-based and OpenID-based identity federation models.

The user-centric based OpenID has gained significant momentum since its inception in 2005 because its simplicity of design, initially designed to support the social networking and blogging communities. Since the OpenID was not originally designed with a security and trust model in mind, there are security concerns. OpenID's phishing issue is the highest concern. The combination of the identity selector and OpenID provide the most promising solution for fixing the phishing problem, because the identity selector can mitigate the middle-man security attacks by using the OpenID as the authentication process isn't between the user and RP.

Although SAML 2.0 has been the de-facto identity federation standard for a few years, mass market adoption is very limited, the major reason being the complex layers of the SAML standard specifications. Intricate pre-existing business agreements between IdP (aka IP) and SP (aka RP) require supporting the SAML based identity federation, which doesn't encourage SAML adoption in the consumer space. Also no standard SAML mechanism supports IdP discovery, which is required in the inter-federation or dynamic federation scenarios. On the contrary, the OpenID built into an IP discovery model doesn't require any pre-existing business agreement, however trusting an IP is another issue. The integration of SAML and the identity selector provides an IdP discovery function, since the InfoCard metadata includes IP end point location information. MNOs may support both SAML and OpenID in the identity selector applet to address different mobile application requirements and customer needs.

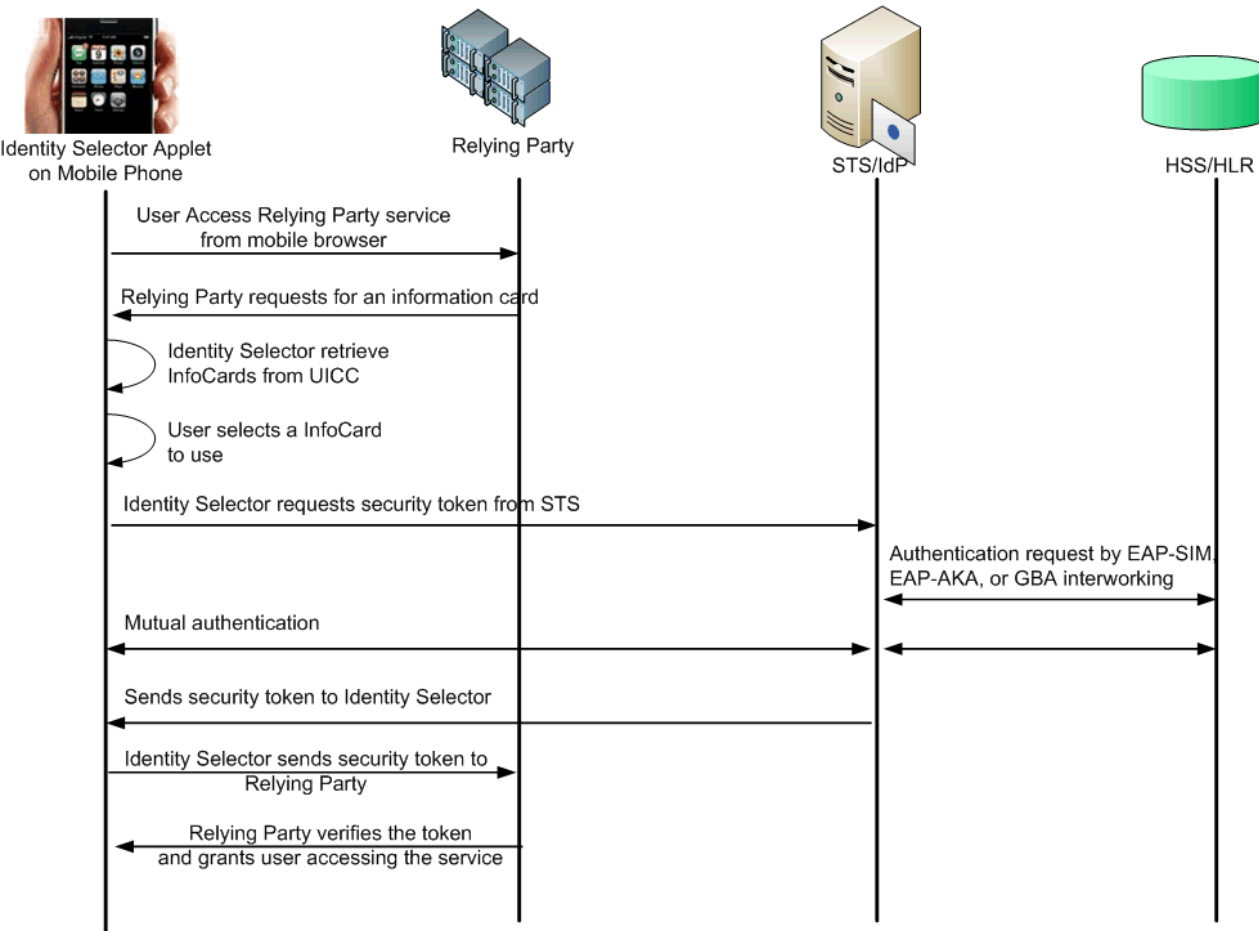
### **2.5.3 Identity Selector Applet Details**

Identity selector supports two types of InfoCards: managed cards, which are issued by the same or different Identity Providers, and self-issued cards, which are created by the users and asserted by a Personal IP (PIP). A MNO can issue different InfoCard assurance levels in the user's identity selector while the users can create their own self-issued cards with lower assurance levels.

The main components of the Identity Selector Applet include an Identity Selector UI client or the InfoCard manager, InfoCards and Security Token Service (STS).

In a common scenario, the user clicks on an identity selector icon in the mobile phone screen to bring up the identity selector UI. The InfoCards are stored in the UICC or a trusted module in the mobile equipment. The STS is the main component of WS-Trust to issue, validate and exchange the security tokens. Figure 5 shows the identity selector authentication flow where the MNO stores the InfoCards in the UICC, and supports a remote STS in its identity provider infrastructure.

Another interesting deployment option that MNO may consider is to provide a local or delegate IdP in the mobile equipment or UICC. NTT has recently demonstrated a concept trial implementing an IdP in the mobile equipment, so the users can use mobile phone as an IdP. Applying such a concept to the identity selector applet, by deploying a local STS in the mobile equipment or UICC provides powerful identity service capabilities for the mobile applications. The Liberty Alliance Advanced Client specification, which is part of the ID-WSF open standard, provides another option as a way to enable the IdP and the identity selector using open standards.



**Figure 5 – Identity selector authentication flow**

### 3 Security Feature Requirement Matrix

It is evident that the application spaces addressed above are very divergent. Consequently, correlating the security requirements across this space is a challenging endeavor.

The table below helps the reader concisely visualize the key security requirements associated with the mobile application space:

	Financial	Access	Content Protection	Authentication	Identity Selection
Certificate Storage/Management	NFC application credentials	Access credentials	DRM Rights Objects	Authentication credentials	Identity credentials
Authentication	User auth to financial institutions	Authentication of user to access service	Authentication of content access rights	User Authentication	Authentication of user to access service
Digital Signature			Generation of digital signature for validation		
PIN/password protection	User access to payment apps	Automatic PIN/password validation	User access to content		Enables User to select identity
Remote applet management	Life cycle mgmt of payment apps by TSM		Content protection algorithms management		
Content storage/encryption			Secure Storage of content		
Identity management	User identification for secure transactions	User identification for service access			Identity life cycle management
Secure data exchange	Exchange of financial transaction records	AKA	AKA, Content Decryption	AKA	Exchange of identity credentials

**Table 2 – Key security requirements**

## 4 Key Differentiators

Mobile technologies must address the key requirements presented in the previous section in order to provide secure applications for subscribers. The GSM evolutionary family of mobile access technologies has a clear advantage over other access technologies as far as security features are concerned.

At the heart of most of these advantages is the UICC. As previously mentioned the UICC is a secure portable token that is recognized in both the mobile and banking industries as the most secure, tamper resistant device for the storage and use of credentials and secret data.

The Milenage algorithm commonly used in UMTS network access authentication is also viewed as superior to the authentication mechanisms used in other access technologies. Most other access technologies perform the authentication function in the user equipment, making the algorithms and credentials vulnerable to attack via the various interfaces in the device itself. The UICC is engineered to include a number of physical and logical countermeasures that make compromising its secrets virtually impossible [2].

The algorithms used to generate digital signatures and encrypt data are stored in the same secure, tamper resistant device as the access credentials; consequently they are protected by the same security features and preventative countermeasures.

To ensure that only the correct user can access the secrets stored in the UICC, the information is protected using a PIN or password, which enables two-factor authentication.

Another key advantage enjoyed by GSM evolutionary access technologies is the mechanisms used to manage the sensitive information stored within the UICC. When the cards are initially produced, the data personalized onto the UICC is loaded in a secure facility that is approved by stringent certification boards appointed by the mobile and banking industries.

Once in the field, secure data and applications can be remotely managed via the Global Platform [11]. Global Platform allows operators to manage sensitive data in the field using the same methods as the secure personalization facility. The distinct advantage enjoyed by the GSM family of operators is that the network platform used to remotely access secure data are likely already installed in their network. Most OTA platforms currently deployed utilize Global Platform for remote management of data and applications on the UICC. Furthermore, the security features in place in Global Platform already adhere to the strict standards imposed by the banking industry. In order for other access technologies to be compliant, they will have to integrate a new remote management platform into their framework, which will take time and come at considerable cost to the operator community.

A collateral advantage of strongly tying security credentials to the wireless account is the power that this gives the network operator to manage the features and credentials used by a subscriber. In cases of loss or fraud, the credentials can be very easily terminated or modified before they can be used without the subscriber's knowledge. By tokenizing all security features on the UICC, the subscriber enjoys the ability to seamlessly move

their credentials from one device to another, and the operator and service providers enjoy the benefits of the user's ability to easily access all services from any device that they choose.

The differentiating factors associated with application security and trust in the GSM evolutionary family make these access technologies a clear leader in the mobile industry. The devices and network infrastructure implemented in today's networks give them a considerable advantage over competing technologies, and place these operators at the leading edge of trusted mobile application offers.

## **5 Conclusion**

Security and trust are real concerns for consumers and application providers alike. Attempts to compromise services and applications such as identity theft, phishing and pharming threaten to limit the types of applications that are provided. Consequently, there is a segment of society that does not trust transactions solely machine based, particularly where mobile devices are concerned.

This paper discusses key applications and services that are on the leading edge of those offered today and in the near future. Not surprisingly, the applications that have the greatest revenue potential are also those that are obvious targets for hackers to steal information and use it maliciously. The resulting goal is simple; the system that provides the greatest security and trust will be seen as superior to other implementations.

Fortunately, GSM evolutionary access technologies have key security features in place today that give them a decided advantage over those implemented in other access technologies. The authentication mechanisms employed in the UICC are seen as industry proven, and the network platforms used to remotely manage these secrets are already compliant with the most stringent security standards in the world today. Furthermore, most GSM, UMTS and HSPA operators have these platforms in place today.

When the Secure Element is collocated with the user's mobile network access identity on the same physical device, it ensures that the Mobile Operator is a part of the identity value chain and validates that the subscriber using the application is also the owner of it, regardless of the device being used to access the network or the service itself.

These advantages place the GSM evolutionary access technologies at the forefront of the mobile application space and give them a decided advantage over the competition as these applications and services become a reality in the near future.

## 6 Acknowledgements

The mission of 3G Americas is to promote and facilitate the seamless deployment throughout the Americas of the GSM family of technologies including LTE. 3G Americas' Board of Governors members include Alcatel-Lucent, AT&T, Cable & Wireless, Ericsson, Gemalto, HP, Huawei, Motorola, Nortel Networks, Nokia, Openwave, Research in Motion (RIM), Rogers (Canada), T-Mobile USA, Telcel (Mexico), Telefónica and Texas Instruments.

We would like to recognize the significant project leadership and important contributions of Yale Vinson of Gemalto as well as the other member companies from 3G Americas' Board of Governors who participated in the development of this white paper.

## 7 References

- [1] *"Advanced Trusted Environment, OMTP TR1"*, version 1.0, 22nd May 2008
- [2] *"Smart Card Security – Why is a Smart Card Secure?"* Gemalto, 7 April 2008.
- [3] *"Mobile NFC Technical Guidelines, Version 2.0"*, GSM Association White Paper, November 2007.
- [4] *"Pay-Buy-Mobile Business Opportunity Analysis, Version 1.0"*, GSM Association White Paper, November 2007.
- [5] *"DRM Architecture"*, OMA, version 2.0.1, 26 Feb 2008, OMA-AD-DRM-V2\_0\_1-20080226-A
- [6] *"Service and Content Protection for Mobile Broadcast Services"*, OMA, version 1.0, 26 Feb 2008, OMA-TS-BCAST-SvcCntProtection-V1\_0-20080226-C
- [7] *"Security of Multimedia Broadcast/Multicast Service (MBMS), release 7"*, 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; 3GPP TS 33.246 v7.5.0, 2007-09
- [8] *"Mobile Content Security, Requirements for OMA DRM V2 Enabled Terminals"*, version 1.3, 16<sup>th</sup> May 2007
- [9] *"Trusted Environment, OMTP TR0"*, OMTP Hardware Requirements and Defragmentation, version 1.1, 27 March 2006
- [10] *"RFC 3748, Extensible Authentication Protocol (EAP)"*. B. Aboba, L. Blunk, J.Vollbrecht, J. Carlson, H. Levkowitz, Ed., June 2004
- [11] *"Global Platform Card Specification v2.2"*, March 2006