Authentication and Authorization in Mobile Environment

Teppo Halonen
Helsinki University of Technology
Department of Computer Science
Teppo.Halonen@hut.fi

Abstract

In services offered over information networks, like electronic banking, validating the identity of a user and the authorities he has is a fundamental issue.

There are many ways to perform this operation, some of which provide a higher degree of certainty and are easier to use than others. Public Key Infrastructure (PKI) based solutions are generally considered to be the most secure and reliable.

In a mobile environment, where the same services can be used through different channels, like the web and the WAP, the issue of authentication and authorization is often more complex. For instance in a PKI managing the private key in such a manner that it can be used, without being compromised, on different device platforms is a challenge.

The standardization of technologies to be used to solve the problems is advancing at a fast and steady pace. The actual implementations are lagging a step behind, especially when it comes to developing overall solutions for authentication and authorization.

1 Introduction

1.1 Background

Authenticating the identity and authorization of users with a high degree of certainty in open and unreliable environments has been one of the most significant functional problems that have been struggled with, when developing services accessible over the Internet. Many different schemes for accomplishing the desired result have been devised. Some of which have been better than others in terms of user friendliness and the level of certainty and security.

Now that the era of mobile Internet is dawning there is a lot of rethinking of procedures that are used for authentication and authorization going on. The dilemma that is being faced is that there should exist a mechanism that is as convenient as possible for the users and that would give a perfect certainty of the result of the authentication.

1.2 Scope

This paper discusses the current methods for authenticating and authorizing users in mobile environment. This paper doesn't concentrate on only the wireless applications, but a broader definition (presented in section 3.3) of mobile environment is used. The discussion is focused on PKI based solutions, although also other means of authentication are presented. The actual need for authentication and authorization is also discussed briefly, as the matter doesn't always seem to be completely clear. The discussion of many issues, like for example public-key cryptography is done on a very superficial level.

On some areas it is not sensible to stick to describing what is currently possible, as technology is evolving very fast. This paper discusses in part technology that is actually not yet widely available commercially. The Wireless Identity Module (WIM) is an example of such a piece of equipment.

1.3 Goals

The goal of this paper is to clarify the concepts of authentication and authorization and to discuss how the concepts and procedures related to them can be implemented or carried out in mobile environment. The paper will also make a proposal for a new, universally applicable procedure of authentication that relies on the technologies discussed in the paper.

2 Abbreviations

ACL	Access Control List
CA	Certificate Authority
eCommerce	Electronic Commerce - Commerce transactions carried out in information networks
eStore	Electronic Store - A website where one can perform eCommerce transactions
RFC	Request For Comments
ME	Mobile Entity, Mobile Equipment
MeT	Mobile Electronic Transactions Initiative
MSISDN number	Mobile Subscriber Integrated Services Digital Network Number
PIN	Personal Identification Number
PKI	Public Key Infrastructure
SIM	Subscriber Identity Module
SPKI	Simple Public Key Infrastructure
SSL	Secure Sockets Layer
WAP	Wireless Application Protocol
WIM	Wireless Identity Module

3 Definitions

3.1 Authentication

"Identity authentication is the process whereby some chosen attribute of a real-world entity ('the distinguishing character or personality of an individual') is demonstrated to belong to that entity." [5] In other words, authentication is the process of determining whether someone or something is, in fact, who or what it claims or is declared to be.

There are five methods of authenticating an identity principal [5]:

1. Something the claimant knows
2. Something the claimant owns
3. Something the claimant is
4. Claimant is at a particular place (at a particular time)
5. Authentication is established by a trusted third party.

3.2 Authorization

Authorization is closely related to authentication. Authorization is in fact the process of giving someone permission to do or to have something. This implies that in order to be able to give someone access to something this someone needs first to be authenticated. In other words authorization is logically preceded by authentication. Actual authentication of the identity of an entity or individual is, however, not always required, in order to do the authorization, if the authorization can be validated in some other means (see e.g. 3.6).

Authentication by itself has little use - it is just a means of getting the information needed for authorization.

3.3 Mobile Environment

In the context of this paper the concept 'mobile environment' is used to describe *a setting where a user of systems accessible through the Internet or other open networks, is not bound to a certain place, device or access channel in order to use the services provided by the systems*. An example of a setting like this is a banking system that is accessible through the Internet and WAP access channels.

3.4 Public Key Infrastructure and Public-key Cryptography

Simply put, Public Key Infrastructure (PKI) is "a system for publishing the public-key values used in public-key cryptography" [2].

Public-key cryptography is a technique initially introduced by Diffie and Hellman in 1976 [3]. Public-key cryptography relies on mathematically complex problems dealing with prime number theory, that are impossible to solve in a reasonable time without the correct input - the key. Public-key cryptography is used in essence, like all cryptography used, only for encrypting and decrypting sequences of data. In the public-key cryptography there are fundamentally two keys (a key pair) involved: a key that is kept secret by its owner (the private key) and a key that is made public to everyone (the public key) between which there is no connection that can be discovered. Data that is encrypted using the public key can only be decrypted by using the private key and vice versa. This paper will not discuss the specifics of public-key cryptography and the algorithms related to it in higher level of detail than this. The assumption is made that the cryptographic algorithms that are being used are strong and the keys that are in use are long enough for the process to be reliable.

Even though, the PKI is not a completely new thing (see [3] from 1976) and it is widely in use almost all over the world, there are some, almost philosophical, debates still going on around it. Generally the fact that the government authorities especially in the United States want to have access to the private keys of all individuals for 'purposes of national security' is being widely debated [1]. Further discussion of these issues is outside of the scope of this paper.

There are actually a number of slightly different approaches taken to implement PKIs but there are at least two things that are common to all of them:

- **Certification**, which is the process of binding a public-key value to an individual or attribute
- **Validation**, which is the process of verifying the validity of the prior

In his paper "A Survey of Public Key Infrastructures " [2], Branchaud defines certification to be the "means by which public-key values, and information pertaining to those values, are published" and certificates to be "the form in which a PKI communicates public key values or information about public keys, or both".

A serious issue to be solved in a PKI and pubic-key cryptography system implementation is the management of the private keys of the individuals. The keys should be kept secret, but the owner of the key should always have access to it.

3.5 Certificate Authorities and Digital Certificates

Certificate Authority (CA) is an entity that issues and verifies digital certificates. Digital certificate is a piece of data, containing a public key and its attributes, as the information about its owner and the signature of a CA. The fact that the certificate is signed by a CA, makes it possible for a user of the certificate, e.g. a person who wants to sign something with the public key of the presumed owner of the certificate (i.e. the subscriber of the certificate), to easily gain certainty that the information contained in the certificate is valid. This is possible since the CAs are trusted entities and the signature of the CA makes also the certificate trusted.

In a PKI there normally exists also a network or hierarchy of CAs [2, 6]. This makes it possible, that as long as a CA is trusted by some CA that is trusted by the root CA (that is implicitly trusted) or a CA on the 'trust path' to the root, that CA is also trusted.

It should be noted that when a CA issues a certificate by digitally signing it, it has to be sure that the public key and the information that is being attached to it in the certificate, i.e. the information about the subscriber of the certificate are correct and coherent. The CA normally relies on a register authority (RA) to do so.

3.6 SPKI - Simple Public Key Infrastructure

The SPKI has been defined in IETF RFC 2693 [4]. Idea of SPKI certificates is to link an authority to a key. This differs from the traditional PKI thinking, where identification information is linked to a key in a certificate. In other words the SPKI certificates are more focused on binding rights and authorities to keys than the individuals' identification information to keys, which is the case in with conventional PKI-implementations as the X.509 [4, 9].

For the purposes of open environments this is convenient, since in this setting a degree of anonymity is provided - a user can present a certificate that issues him authority to access some services or resources, without necessarily revealing his identity. The trust is based on a trust network and delegated authorization.

Even though it is an ideal method for providing authentication and authorization mechanisms in open environments, there aren't actually any implementations or products that would support it on the market currently [9].

3.7 X.509 PKI

X.509 is the most widely used PKI standard. It defines all the aspects of the infrastructure, like the structures of certificates and the hierarchical relationships between the CAs.

The X.509 certificates have the following content [9]:

Version that describes the version of the encoded certificate
Serial number that is a unique identification number of the
CA's signature algorithm identifier
Issuer name that identifies the CA who signed and issued the certificate
Validity that indicates period of validity of the certificate
Subject identifying the holder of the private key
Subject public key information
Issuer unique identifier
Subject unique identifier
Extensions field(s)

It is useful to understand the structure of the X.509 certificates, as they are used also in the WTLS that is discussed later on in this paper.

There are several versions of the X.509 standard. The latest version is the X.509v3.

4 Need for authentication and authorization

Authentication of the identity of a person or an entity has to be made, whenever it is required to know explicitly who the person or entity actually is. Authorization has to be done, always when a decision has to be made, whether or not someone is allowed to do or have something.

For example, when a person visits a bank, to withdraw money from his account the clerk has to first authenticate the person, by e.g. his identity card. Then he has to check the account balance of the person in order to know how much money he is authorized to withdraw. In this example both authentication and authorization are required. The bank notes can be used as tokens of payments - they authorize the payment transactions. When paying with the bank notes the identity of the person using them has no longer to be authenticated. Rather the bank notes are signed by the government to certify their authority (i.e. the denomination). This is a convenient arrangement for all sides; 1) the consumer has a right to privacy - nobody will necessarily know where he spends his money, 2) the identity of a consumer doesn't have to be authenticated each time he pays for something, which makes doing transactions faster and easier and 3) the authorities, i.e. the government that issues the bank notes doesn't have to deal with the consumers directly.

In the information networks the same principals apply. The authentication of a user is not needed if only the authorization can be validated in some other way. Even authorization is not needed for services or resources that are accessible for everyone. In theory, if there is at least one person or entity that shouldn't be authorized to access a certain service or resource, everyone's authorization should be validated, before allowing access to the service or resource.

Authentication of the identity of a person isn't necessary nearly as often as it is used, e.g. in eCommerce, nowadays. Most of the eStores on the Internet, require you to authenticate yourself when making a purchase, e.g. for the hidden agenda of collecting 'vital' marketing information. How many ordinary stores require you to present your id when you check out your purchases of books or groceries - not many! Actually they normally only do it if they have to have some assurance of your credit. This same practice should also apply to the e-world - everybody should be entitled to privacy, authentication should only be required when there is no other way to get assurance of creditability. There are many initiatives for providing secure e-Money that can be used on the Internet (both wired and mobile) - they will be the tools to allow the authorization of payments without compromising privacy [8, 9].

5 Means of authentication and authorization

In this section the actual methods or processes that are used to authenticate the identities of users are discussed. The methods all make use of one or more of the five characteristics described in 3.1.

The authorization of the user to gain access to services or resources can be carried out in a system after the user has been authenticated and his identity is resolved through the use of access control lists (ACL), to determine what the particular user is authorized to do.

Authorization is thus at maximum as accurate and correct as the process of authentication. A mechanism like the SPKI could be used, to avoid authentication of the user, but to still provide a reliable authorization.

5.1 Passwords

Passwords associated to user names (something that the person knows) are a simple way of authentication. They can have a one-to-one or a on-to-many relationship to actual persons. I.e. every user of a system can have a different password, or a group can have the same password. Implementing a system where passwords are used to authenticate the users is quite straightforward - there just has to be a repository, which should be secure, where the passwords are stored.

The disadvantage of using passwords is that they are notoriously vulnerable to attacks. At least the following types of attacks are possible: external disclosure, guessing, communications eavesdropping, replay attacks and host compromise [5].

There are several authentication schemes that make use of passwords in combination with some other factor. A simple extension of passwords is one-time passwords. A user is provided with a list of passwords through a secure channel - e.g. registered post. He only uses each individual password one time to authenticate himself with the system. The use of this scheme already eliminates the possibility of communications eavesdropping and replay attacks.

5.2 Password with a token

Passwords can be used in combination with some physical object (something that the person owns). E.g. an ATM card with a PIN code is an example of this - the card itself is useless without the PIN, as is the PIN without the card.

This concept has been extended with the use of integrated circuit cards (ICC) or smart card. They are tamper proof devices that may interact with a device directly, as is the case with e.g. the SIM-card in the GSM-phones. The SIM cannot be used without the knowledge of the PIN-code, which the SIM-card verifies itself. The PIN related to the SIM is never revealed from the SIM. A challenge and response method is used in authenticating the user.

'Synchronous one-time passwords' [5] is another similar technique. The user has a tamper proof device that produces codes that can be used as passwords, often in combination with a PIN and a user name to produce an authentication triplet. The code is produced in the device by an algorithm known also by the authenticator, that thus knows which code has been produced this time and can carry out the authentication. This is a more elaborate scheme than just using lists of one-time passwords, since the use of the device eliminates the need to distribute the lists.

5.3 Biometrics

Biometric authentication techniques include fingerprint recognition, retinal scanning, hand-geometry scanning and handwriting and voice recognition [5]. These techniques are all based on the physical properties of a person (something he owns / is).

As the physical properties of different individuals differ sometimes only slightly from each other, but can differ from time to time on a single individual, it is sometimes impossible to get a high enough level of certainty in the authentication. Furthermore the precision of technologies available is limited and some of the technologies are very costly. In some applications these techniques are perfectly suitable, but in open environments they are still often too difficult to implement into practice.

5.4 Digital Signatures

When a PKI is put in place, digital signatures can be used to authenticate users. The following sequence of actions has to be carried out in order to authenticate a user by his digital signature:

1. The user requests access to the service or system
2. The system generates some data for the user to encrypt using his private key. Then the data is sent over to the user.
3. The user concatenates the data received from the system and a time stamp and encrypts the whole sequence. (*N.B. It is a good practice that e.g. a time stamp is concatenated to the data, so that the data to be encrypted cannot completely be decided by some untrusted party. This is to avoid the possibility of a 'Chosen plain-text attack' as described in [12].*) Then the encrypted data (the cipher text) is sent back to the system. Along with the encrypted data a link to the certificate (or the certificate itself) of the user is sent.
4. The system decrypts the received information with the public key of the user, found in the certificate.
5. The system verifies that the decrypted information is composed of the originally generated data and the a valid timestamp. If this seems to be OK, the system has successfully authenticated the user.

5.5 Properties of a good authentication and authorization mechanism

This section lists the properties that a good identity authentication and authorization mechanism should possess. Some of the features listed, are in contradiction of each other, but mostly it should be possible to reach an acceptable level of compliance with each of the criteria.

- **Correctness -** The results of each individual instance of authentication or authorization carried out should be correct. If it is possible to authenticate the user, the result should always be that either it is found, that the user is who he claims or he is found to be a fraud. Based on this perfectly correct authentication it is further possible to authorize the user to access those services and resources that defined to be accessible for him or to the group or groups he belongs to. In practice it is impossible to get an absolute certainty in authentication. Only a reasonable level of certainty can be gained.

- **Possibility to anonymity and privacy -** Identity authentication should only be done when absolutely necessary. Whenever authorization is possible without the users identity being revealed, it should be done that way.
- **Speed -** The process of authentication should be fast. The user shouldn't have to wait for the result for more than a second or two.
- **Attack resistance -** The perfect mechanism of authentication should be resistant against any known or unknown types of attacks.
- **Inexpensiveness -** The mechanism shouldn't require extensive investments from either the users or the authenticators.
- **User friendliness -** The mechanism should produce as little overhead to the user as possible. It should also be as easy to use and understand as possible. In the optimum situation the user doesn't have to perform any actions in order to become authenticated. The user shouldn't be forced to carry around any extra equipment, magnetic or smart cards, lists of passwords or other physical objects in order to use the system.
- **Universality -** It should be possible for the user to use the same means or method of authentication in all services and everywhere.

6 Technical possibilities offered by current mobile devices

There are already quite a few Internet enabled mobile devices GSM handsets available on the market today. The term 'Internet enabled' today means that the handset has an embedded WAP. The first such device that became commercially available was the Nokia 7110. After this Ericsson introduced the R320 and many others followed. In a few of the current models and versions the WAP browsers implement the WTLS (Wireless Transport Layer Security).

6.1 Wireless Transport Layer Security

The WTLS is the security layer protocol for the WAP. It has been specified by the WAP Forum initiative. The purpose of the WTLS is defined by the WAP Forum as follows: "The primary goal of the WTLS layer is to provide privacy, data integrity and authentication between two communicating applications." [14].

The WTLS handshake protocol establishes a secure connection between the client and the server. It also gives the possibility for the server to authenticate itself to the client, by sending over its certificate. The same applies for the client. It can authenticate itself by sending its certificate or a link to it, if the client authentication is requested by the server [14].

The table 6.1 presents the full WTLS handshake protocol. The parts marked with asterix (*) signs are optional.

Client		Server
ClientHello	--->	
		ServerHello
		Certificate *
		ServerKeyExchange *
		CertificateRequest *
	<---	ServerHelloDone
Certificate *		
ClientKeyExchange *		
CertificateVerify *		
[ChangeCipherSpec]		
Finished	--->	
		[ChangeCipherSpec]
	<---	Finished
Application Data	<--->	Application Data

Table 1: The Full WTLS HandShake [14]

The WTLS specification doesn't actually in itself set any requirements for the authentication, i.e. whether or not the client should be authenticated by its certificate. In fact the current mobile phones, their WAP browsers and the SIM cards that are delivered by standard don't even have proper certificates. In order to standardize the storage of the certificates and private keys in the WAP terminals, the WAP Forum has issued another specification, the "Wireless Identity Module Specification" [13].

The actual level of security provided by the WTLS is often being debated. The encryption and authentication of the users is only between the WAP gateway and the mobile equipment - not all the way to the service, i.e. the content server. This means that there is a security gap inside the WAP gateway - the data exists in clear text and its integrity could possible be thus compromised. The authentication and handshake isn't either by standard done from end to end.

6.2 Wireless Identity Module

Wireless Identity Module is defined by the WAP Forum to be a tamper resistant device that is used for "performing WTLS and application level security functions, and especially, to store and process information needed for user identification and authentication" [13]. The WIM is envisaged to be implemented on a smartcard - either on the SIM or a separate card.

The WIM is used mainly for the following purposes [13]:

- To store the permanent certified private key (the private key never leaves the WIM)
- To store the client certificate or a link to it
- To perform cryptographic operations during the WTLS handshake, especially those needed for client authentication
- To sign data with the private key for application purposes
- To 'unwrap' e.g. keys signed with the public key related to the private key in the WIM
- To store CA certificates, that can be changed
- To generate random numbers for purposes of cryptography

As it can be seen, WIM plays actually a crucial role, in the WTLS - without WIM, a proprietary mechanism for handling the keys, certificates and the signing operations has to be implemented. As the WIM is designed as a tamper proof device and provides all the functionality needed to make use of the PKI, there is shouldn't actually be any point in going about the implementation in a different way.

Currently there aren't any commercially available mobile phones that have a WIM capability. At least Ericsson has, however announced the release of such a model by end of 2000 or beginning of 2001.

7 Practices and standards in mobile environment

Use of Internet and services accessible through it, by devices such as mobile phones has been reality only for a few years now. The authentication models that are being used have largely been inherited from the wired counterparts. There are, anyhow, some methods of doing authentication in the mobile Internet that are distinctive to it and often not even possible elsewhere.

This section discusses the status quo as well as some of the most significant standards and developments that are becoming available in the near future.

7.1 Simple methods

The same means of authentication that have been used in the past are also being made use in the mobile systems. In WAP based services the most commonly used way to authenticate oneself is to key in the username and password. In some services authentication may also be based on the GSM-phone number, i.e. the MSISDN number, of the user accessing the WAP service. This method of authentication is comparable to using the IP-address of a machine attached to the Internet as the grounds of authentication - the same deficiencies apply to both cases.

7.2 Products on the market

There are several vendors of products that provide facilities for authentication and authorization in mobile environment. Many of them are providing proprietary solutions. This is partly due to the fact that many of the standards that there are, are still very young and the industry has difficulty in keeping up with the pace.

In the near future the Mobile Electronic Transactions Initiative, formed by the mobile equipment manufacturers, Nokia, Ericsson and Motorola, will probably be a significant standardizing body in this sector. It hasn't, however, yet revealed its complete agenda, and results of its work are not yet available. The MeT has anyhow declared that it will work together with the exiting standardization and specification initiatives, "where appropriate", and most probably it will try to keep the same direction in the work that has already been taken in the WAP Forum, with the WTLS and WIM specifications [8].

7.2.1 HST

HST is a PKI in Finland. The idea is that everybody is issued a smart card that contains a certificate and the associated private key. The card is intended to be used to authenticate the citizens when they deal with government authorities. A requirement for using the card, is that there has to be a smart card reader and a piece of software that knows how to use the services provided by the card. Using this card e.g. in a mobile phone is not currently possible. There is also a key escrow [1] mechanism put in place in the PKI, which makes the security of the system questionable.

7.2.2 RSA Security - SecurID

The RSA Security SecurID (formerly a product of Security Dynamics), is a system for authenticating users. It is based on the use of a PIN code (something you know) and a device, authenticator that produces on regular intervals a new token code (something the user knows) that the user can use as the passcode. On the service end, there is of course a server, that can validate the passcodes and PINs received from the users [10].

This system provides a quite high level of security, since it is based on two factors of authentication, the PIN and the authenticator tokens. The disadvantages of it are that it uses a rather closed technology - the authenticators have to be purchased from RSA Security, as well as the server software. The user also has to carry with him some extra piece of equipment, the authenticator, which serves no purpose other than producing the passcode values. The technology is actually only intended for use inside an enterprise as the authentication solution.

7.2.3 Sonera SmartTrust - SIM Security Client

Sonera SmartTrust is offering a security solution for the WAP. Their SIM Security Client product is a SIM Toolkit based product, that basically implements the functionality, that e.g. the 6.2 is specified to provide, but in a proprietary manner. Part of the product suite is server software, called "SmartTrust Security Server" that implements the PKI facilities that service providers need [11, 9].

The idea behind the SmartTrust offering is good. It is ideal for closed, intranet type of WAP usage environments, but is not suited often for wider operator environments, since a prerequisite for using it is the change of the GSM SIM card, as the product partly resides on it. For corporations that are willing to go through this trouble, the solution should work nicely.

The bad thing about the product, is that is proprietary. The WIM is basically the standard way of implementing the same functionality and will no doubt be a dominant solution in future.

7.2.4 Case: Merita Bank - Solo

Solo is the Internet banking and investment service offered by Merita Bank Ltd. There are Internet, Voice and WAP interfaces to the services. In the service users are authenticated by their user name, that is randomly assigned to each user and a four digit one-time password. The list of one-time passwords is delivered to the users via post in a sealed envelope. On the same sheet with the one-time passwords also a group of authentication codes. The user is requested to enter a certain one of them whenever he requests a payment, investment or other money transaction to be performed in the service.

The same mode of authentication is used for all of the access channels of the Solo: Internet, voice and WAP.

The communication in the Internet version is protected by 40 bit SSL. The WAP access mode is not secured by WTLS - the WAP users use Merita Bank's own dial-in modem pool to gain access to the service, so the connection is assumed secure.

Merita Bank acts also as a trusted party, in certifying eCommerce payment transactions to e-Stores on the Internet. It provides a facility to pay for purchases in the Solo, after which the Solo notifies the e-Store of the delivery of the payment. The information about the reception of the payment is signed by Merita Bank, which makes it possible for the e-Store to trust that the money has been transferred to its account.

8 A Universal Authentication and Authorization Mechanism

This section describes a possible authentication and authorization mechanism that can be used in combination with all possible access channels. It is actually not proven to work, as it is purely the result of the imagination of the author, but it should be possible to implement with mobile devices that have the capabilities specified in [14, 13]. The mechanism is of course not needed when accessing services through the WAP access channel, since there the WTLS handshake already authenticates the user.

The figure 1 illustrates the mechanism. The parties involved, are the user, the service the user wants to be authenticated with and an authentication service, that can be a completely separate entity from the actual service. It is also assumed that a CA exists and is accessible

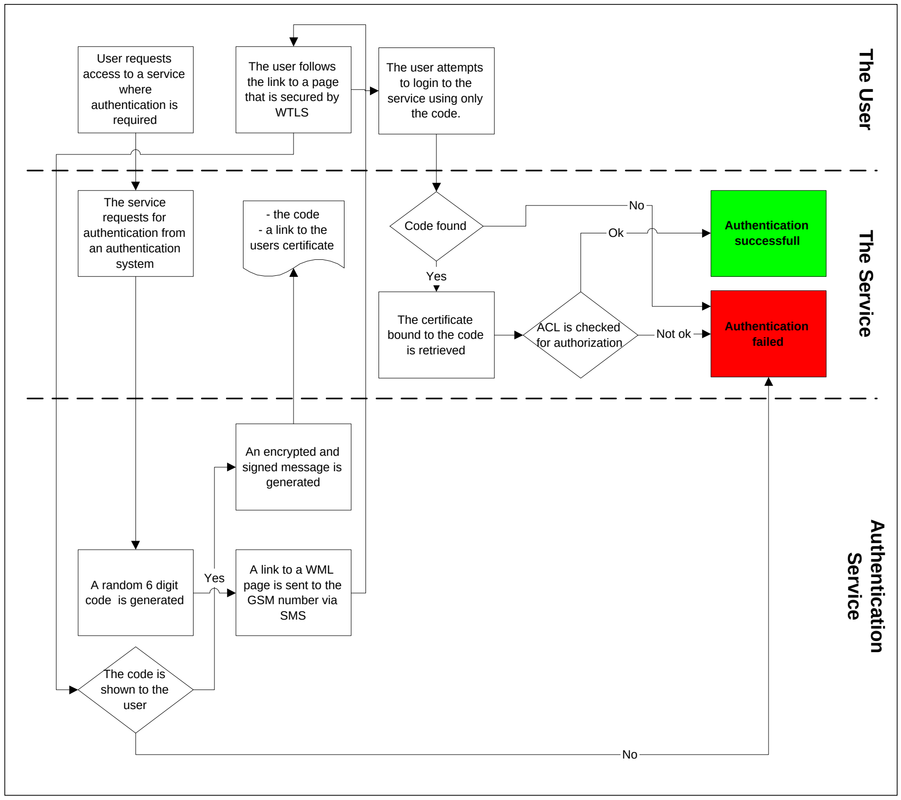

Figure 1: A flow chart describing a universal authentication and authorization mechanism

somewhere in the network.

The sequence of actions illustrated in the figure 1 is explained in more detail below.

1. The user requests for access to the service. The service can be e.g. an Internet Banking service. The user might use e.g. Internet (WWW) or telephone as the access channels.

2. The service requests the authentication service to authenticate the user. The service passes to the authentication service at least a link to its certificate and the MSISDN number of the user.

3. The authentication service generates a random code - say 6 digits long. The code only lives for a short period - say 30-60 seconds, so it doesn't have to be very long.

4. A link that points to a WML page, where the code would be presented is pushed to the users MSISDN number. SMS (Short Message System), OTA messages or WAP push could be used to do this. The page is protected by WTLS so that user authentication is required in the WTLS handshake.

5. The user follows the link. In the WTLS handshake he is automatically authenticated by the WIM facilities in the ME.

6. The authentication service becomes immediately aware of whether or not the WTLS handshake authentication was successful. If it wasn't the service that requested the authentication is notified and it can consider the authentication a failure. If it was successful, the service is sent a message containing at least the code that was presented to the user on the WML page and a link to the users certificate. The message is encrypted using the public key, the service initially indicated to the authentication service by giving it a link to its certificate. This could be just as well a normal X.509 certificate or a SPKI certificate - as long as it contains information based on which the service can authorize the user.

7. Now the user can attempt to log on to the service just by entering the code he was given.

8. When the service receives a login attempt with a code, it looks up the recently received messages from the authentication service, for that code. If it finds it, it checks the identity of the user from the link to his certificate. Then it checks from an ACL, whether or not the user is authorized to access the service. If all checks out ok, the user is authorized and he may proceed. Otherwise the authentication and authorization failed.

The benefits of this mechanism are:

- It is secure - the level of certainty can be adjusted according to the need
- It is simple to use
- It is resistant to attacks - the lifetime and length of the codes dictate the degree of strength
- It relies completely on PKI (X.509), WTLS and WIM, that are standard specifications
- In theory anonymity is possible. The authentication system could only provide the service with an SPKI certificate without the identification information of the user. This would although require slight changes to scheme.
- The user only needs to have a GSM phone, with a WTLS capable WAP browser, a WIM with a certified private key - no extra equipment or investment is needed
- The same method of authentication can be used over different access channels
- The user doesn't have to remember any passwords, other than his PIN to the SIM/WIM

The disadvantages of the mechanism are:

- The WIM and WTLS with full set of specified functionality is not yet available (this will be remedied during the next year)
- The system relies on WAP, that is currently cumbersome to use over GSM data connections (this should be remedied by GPRS)

- The user always needs to have his GSM phone at hand when he wants to authenticate himself with a service
- Usability is not proven - issues may exist

9 Conclusions

Authorizing e.g. payments is one of the central issues faced when doing commerce in open networks. Moreover, when services and resources can be accessed over different kind of information networks, mobile and fixed, and different kind of devices and interfaces, a mechanism that allows the authentication authorization to be done independent of the attributes of the access channel or mode.

PKI provides the basis for such functionality. X.509 certificates are already specified to be used and partly already in use in current GSM phones equipped with WAP browsers. The management of keys is, however, an integral problem faced in this setting. The same private key should be usable in different environments, both when working on the workstation at the office and when surfing the web with a WAP phone.

The WIM may be the solution for this dilemma. Its specification is, nevertheless, still very young and there seems to be some turmoil in the WAP standardization community. The MeT initiative formed recently by industry giants Nokia, Ericsson and Motorola and later joined by Siemens will start tackling the same problematics that the WAP Forum is trying to solve through the WTLS and WIM specifications, even though it doesn't, by its own words compete with any of the existing initiatives. The coming year will show what direction the standardization will take on this matter.

With the current technology, it is already possible to set up systems with authentication and authorization that are based on PKI and provide a relatively high level usability. Only the future will show what the eventual model to be adopted will be.

References

[1] Abelson, H. et al. The Risks of Key Recovery, Key Escrow, and Trusted Third Party Encryption 20.8.1998, [referred 9.11.2000], <http://www.cdt.org/crypto/risks98/>

[2] Branchaud, M. A Survey of Public Key Infrastructures March 1997, [referred 9.11.2000], <http://home.xcert.com/ marcnarc/PKI/thesis/>

[3] Diffie, W. and Hellman, M. New Directions in Cryptography, *IEEE Transactions on Information Theory* November 1976, pp. 644-654

[4] Ellison, C. et al. RFC 2693 - SPKI Certificate Theory September 1999, [referred 9.11.2000], <ftp://ftp.ietf.org/rfc/rfc2693.txt>

[5] Ford, M. Identity Authentication and 'E-Commerce' 30.10.1998, [referred 9.11.2000], <http://www.law.warwick.ac.uk/jilt/98-3/ford.html>

[6] Gerk, E. Overview of Certification Systems: X.509, CA, PGP, and SKIP 17.4.1997, [referred 9.11.2000], <http://www.mcg.org.br/cert.htm>

[7] Gerk, E. Towards Real World Models of Trust: Reliance on Received Information 23.1.1998, [referred 9.11.2000], <http://www.mcg.org.br/trustdef.htm>

[8] The MeT Initiative. MeT Overview White Paper Version 1.0, 2 of October 2000, [referred 9.11.2000], <http://www.mobiletransaction.org/techinfo.html>

[9] Puhakainen, P. Electronic Commerce: Market Estimates and Security Considerations Licentiate's thesis. Helsinki University of Technology. Espoo Finland. July 2000, [referred 9.11.2000], <http://www.certall.fi/finnish/content/businessarea/lic4.pdf>

[10] RSA Security RSA SecurID, Web Portfolio. How RSA SecurID Agents Can Secure Your Website 2000, [referred 9.11.2000], <http://www.rsasecurity.com/products/securid/whitepapers/web/Web_Agent_Solution_WP.pdf>

[11] Sonera SmartTrust Ltd. SmartTrust SIM Security Client 2000, [referred 9.11.2000], <http://www.smarttrust.com/products/sim_security_client.html>

[12] SSH Communications Security Introduction to cryptography 2000, [referred 9.11.2000], <http://www.ssh.com/tech/crypto/intro.html>

[13] Wireless Application Forum, Ltd. Wireless Application Protocol, Identity Module Specification 18 February 2000, [referred 9.11.2000], <http://www.wapforum.org>

[14] Wireless Application Forum, Ltd. Wireless Application Protocol, Wireless Transport Layer Security Specification 18 February 2000, [referred 9.11.2000], <http://www.wapforum.org>