A global ticket-based service scheme for mobile users

Hua Wang [1] Yanchun Zhang [1] Jinli Cao [1] Yahiko Kambayahsi [2]
[1] Department of Maths & Computing, University of Southern Queensland
Toowoomba QLD 4350 Australia
Email: (wang, zhang, cao)@usq.edu.au
[2] Graduate School of Informatics, Kyoto University, Kyoto, Japan
Email:yahiko@db.soc.i.kyoto.ac.jp

Abstract

This paper presents a ticket-based access control model for mobile services. The model supports efficient authentications of users, services and service providers over different domains. Tickets are used to verify correctness of the requested service as well as to direct billing information to the appropriate user. The service provider can avoid roaming to multiple service domains, only contacting a Credential Centre to check the user's ticket since tickets carry all authorization information needed for the requested services. The user can preserve anonymity and read a clear record of charges in the Credential Centre at anytime. Furthermore, the identity of misbehaving users can be revealed by a Trusted Centre.

Keywords: Mobile service, Signature, Ticket, E-commerce.

1 Introduction

With recent advances in wireless computing and communication, mobile services are becoming an important factor in business. At the same time, the security and privacy issues in mobile systems are more critical. The static security access control is incompatible with dynamic mobile environments. Mobile service access across multiple service domains, and the traditional access mechanisms rely on cross-domain authentications using roaming agreements starting home location. However, the cross-domain authentications will involve many complicated authentication activities when the roam path is long. This limits the future mobile applications. Normally, there can be three participants in a mobile service. These are users, service providers and services. Some services bind users and service providers as well as services, and some services do not bind any participants. However, there is no scheme to provide a solution for all kinds of mobile services. Users have to change mobile service systems if they want to do different kind of mobile services on the Internet. From the consumer's point of view, there is often a preference for a total solution for all kinds of mobile services, some degrees of anonymity such as no more cross authentications, and a clear statement of account when shopping over the Internet. There are several proposals for mobile systems [Mehrotra, 1997, Mehrotra and Golding, 1998, Frankel et al., 1995]. All of them lack some flexibility in security management. The Global system for mobile

communications [Mehrotra, 1997], for example, provides mechanisms for user authentication as well as integrity and confidentiality, including protection of information exchanged between the mobile terminal and the fixed network. It provides only limited privacy protection for users by hiding their real identities from eavesdroppers on the radio interface [Mehrotra and Golding, 1998]. Another contemporary mobile communication system CDPD [Frankel et al., 1995] provides similar security services. These works are very useful for mobile service systems, but there are some other issues which need to be addressed:

Global solution. A global solution is a mobile service system that can provide all kinds of mobile services. Current solutions can only solve particular service problems for mobile users. Users have to change mobile service systems if they want to do different kind of mobile services on the Internet. This is not convenient for users.

Trust. Mobile users in current mobile service systems have to trust service providers to bill their service usage correctly and not to misuse either users or service usage related information. This kind of trusted model is not adequate for future mobile communication systems. With the rapidly growing number of service providers, most of which are new on the market, and unknown to the users, such a trusted model is no longer justified. This requires mechanisms that guarantee correct and indisputable billing and ensure anonymous service usage.

Scalability. The basic requirements of mobile communication systems are to offer access to any service, anywhere, at any time. The mechanisms of current mobile communication systems are not sufficiently scalable to be able to fulfill these requirements. Traditional solutions for implementing user mobility rely on cross-domain authentication and roaming agreements. A user, when visiting a foreign domain and accessing a service there, has to authenticate himself to the foreign service provider with the help of his home domain agent. This may involve a potentially time consuming authentication protocol over long distances. Furthermore, cross-domain authentication requires the foreign service provider to trust the home domain agent of the user. Today, this trust is based on roaming agreements between various service providers. With the rapidly growing number of service providers, however, roaming agreements are becoming inefficient and no longer feasible. New mechanisms are needed that do not require contact with the home domain of the user when accessing services in a foreign domain, nor business agreements between domains.

Clear charging. Mobile users wish to see a clear and continuously updated account statement. Users do not like receiving a charging bill only monthly or bi-monthly, but like to be able to check it at anytime.

In the future, mobile service systems should provide a global solution for all kinds of mobile services and guarantee higher levels of security than current systems. This means that, as well as requiring confidentiality and the protection of the integrity of the message exchanged between the user and the service provider, and authentication of the user to the service provider, future systems should also require authentication of the service provider to the user and guarantee high levels of privacy. Furthermore, clear billing has to be ensured.

In this paper, a new approach to address the above-mentioned problems is proposed. This approach is based on a Trusted Centre, a Credential Centre and a ticket-based mechanism for service access. The main idea is illustrated in Figure 1.

In this model, each user is registered with the Trusted Centre. The Credential Centre issues tickets to its users. The users can use the tickets to access services anonymously. When requesting a service, the user is required to hand over an appropriate ticket. After

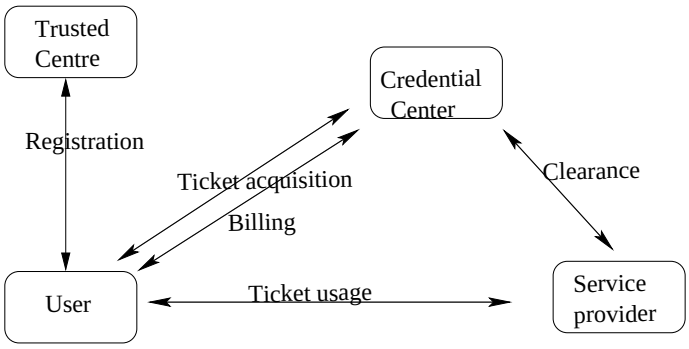

Figure 1: **Ticket Model**

checking the ticket, the service provider provides the requested service to the user. Later, the user can see a clear charging bill in the Credential Centre.

The model has the following features:

1. It is a trusted model. Users and service providers can trust each other.
2. It provides a global scheme for all types of service. Users do not have to change mobile service systems when they do different kinds of business on the Internet.
3. It is scalable and there is a clear charging bill in the Credential Centre.

This paper is organized as follows: in section 2, the basic ticket model and ticket types are introduced. There are eight different kinds of tickets that are divided into two groups, group_1 and group_2. Some basic definitions and their examples are reviewed in section 3. The single signature scheme for ticket group_1, its security analysis and its ticket usage are presented in section 4 while a multi-signature scheme for ticket group_2 and its usage are discussed in section 5. Related works are given in section 6. Finally the conclusions are presented in section 7.

2 Basic ticket model

There are four participants (the user, the service provider, the Trusted Centre and the Credential Centre) and a protocol with several sub-protocols (ticket acquisition, ticket usage, clearance, and billing) in the ticket model. The user obtains tickets by running the ticket acquisition protocol. These tickets can be used to access services. The user presents an appropriate ticket to the service provider, which can verify the validity of the ticket. If the verification of the ticket is successful, then the service provider provides the service to the user according to the conditions on the ticket. Based on the received tickets, the Credential Centre prepares a clear bill for each user. The exact forms of the clearance (payment to the service provider) and billing (payment to the Credential Centre) protocols are not specified in our model. Readers may refer to [Wang and Zhang, 2001] for details.

There are several advantages in using tickets for accessing services [Buttyan and Hubaux, 1999]:

Flexibility. Users can choose services as they need and buy an appropriate ticket that matches their personal requirements. They do not have to enter into long term contractual relationships with service providers.

Scalability. The information in tickets are used for a service provider to decide whether the service should be provided or not. Therefore, it is not necessary to run long distance protocols to perform authentication.

Privacy. Users only have to show tickets, they do not need to reveal their real identities. No private information is available to service providers.

Transfer. In real life, not all tickets can be transferred. It is not convenient for users to limit the wide use of tickets. In our ticket-based service access mechanism, a ticket can be lent to other users even though it is bound with the user. This means the ticket buyer and the ticket user do not have to be the same.

In addition to the advantageous issues, some security problems such as duplication, forgery and modification must be solved in order to implement a ticket system [Patel and Crowcroft, 1997].

Duplication. There are two kinds of duplications needed to be considered. The first one is that users either use or sold a ticket many times (similar to double spending in electronic cash systems). The second one is an eavesdropper who listens to someone else acquiring a ticket and makes a copy for itself.

Forgery. Forgery refers to the illegal construct of a valid ticket, which can be used for accessing to resources.

Modification. Users must not modified tickets. This is to prevent users from accessing resources for which they have not been permitted in tickets, e.g. a ticket allows traveling by a bus, should not be modifiable to allow traveling by a flight.

A ticket may bind a given user, a given service, and a given service provider together. For example, a movie ticket, which usually does not specify who can use it (i.e., the user) or a travel card, which may not restrict the means of transport (i.e., the service). Based on this observation, there are eight types of tickets. These are illustrated in Table 1, where $'\Theta'$ means that the corresponding entity, user, service provider or service is bound by the ticket, while $'-'$ means that it is not.

A ticket of type t_0, for instance, does not restrict the service for which it can be used, the service provider which accepts it, and the user who can use it. This is much like cash in real life. The other extreme is a ticket of type t_7, which can only be used by a given user, for a given service, provided by a given service provider. An example of this type is a flight ticket.

Types	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7
user	-	-	-	-	Θ	Θ	Θ	Θ
provider	-	-	Θ	Θ	-	-	Θ	Θ
service	-	Θ	-	Θ	-	Θ	-	Θ

Table 1: Ticket types

As mentioned in Table 1, tickets t_1, t_2 and t_4 have only one entity bounded and tickets t_3, t_5, t_6 and t_7 have two or three entities bounded. The tickets can be divided into two groups, one is ticket group_1 including tickets t_1, t_2, t_4, and another one is ticket group_2 including t_3, t_5, t_6, t_7. There are different mechanisms relate to each ticket group. Users are anonymous in purchasing since no private message needs to be shown to service providers. Use of a ticket-based system can avoid roaming multiple service domains. A simple case is a single signature. This case can be used in tickets with only one bound entity (users,

service providers or services). As a signer, the bound entity uses a signature to authenticate a ticket. To cope with the cases of two or more bound entities, it is extended to $v(v = 2, 3)$ Signers. This means that a user can get a service if all v entities agree. The v Signers case can also associate with the other services provided by many providers since the number v is not limited to 2 or 3. A Credential_role in the Credential Centre is set up to issue tickets and control the user's charging bill, and a Trusted_role in the Trusted Centre is also set up to judge conflicts. Each user's statement of account can be seen clearly in the Credential Centre. Furthermore, it is dynamic, since new users and providers can join the ticket-based system at anytime.

Through the usage of tickets, the problems of lack of Trust and scalability are also addressed as follows:

Trust. Users can anonymously access services by using tickets. They neither need to reveal their identities nor need to fully trust service providers to handle user and service usage related information. On the other hand, the information of service providers are bound in tickets, thus, the user can assure that the service is provided by the selected service provider. Therefore, users and service providers can trust each other. Service providers can verify the validity of the tickets and check if they were used by their legitimate users. If necessary, anonymity can be revoked and users who behave in a malicious way can be traced by the Trusted Centre.

Scalability. The service providers only need to verify the ticket. Users do not require long distance protocols but connect to the Credential Centre. They will acquire the ticket from the Credential Centre before roaming into the foreign domain.

In the remaining sections, we will discuss how the protocol works for various kinds of tickets. We are not interested in ticket t_0 since it does not bind any entities and electronic cash can be instead of it.

3 Some basic definitions

To facilitate discussions, some well-known primitive cryptographic terminologies, which will be used in the remaining sections of the paper, are reviewed.

Hash function, $h(x)$ is a hash function. For a given Y it is computationally hard to find a x such that $h(x) = Y$, where x might be a vector.

Hash functions have been used in computer science for a long time. They are major building blocks for several cryptographic protocols, including pseudo-random generators [Bellare et al., 1996], digital signatures, and message authentication [Waleffe and Quisquater, 1990].

RSA, is a public key cryptosystem that offers both encryptions and digital signatures (authentication) [Rivest et al., 1978]. RSA works as follows: taking two large primes p and q, and computing their product $n = pq$; n is called the modulus. Choosing a number e, less than n and relatively prime to $(p-1)(q-1)$. Finding another number d such that $(ed-1)$ is divisible by $(p-1)(q-1)$. The public key is the pair (n, e), the private key is d. The factors p and q may be kept with the private key or destroyed.

It is currently difficult to obtain the private key d from the public key (n, e). RSA is often used in modern environments [Chaum, 1981], especially on the Internet, since an individual needs not send any private secret key to others when they want to contact him.

Multi-signatures, are multiple signatures signed on the same document. There are two ways to implement multi-signature. One is that each person signs separately, the other is that

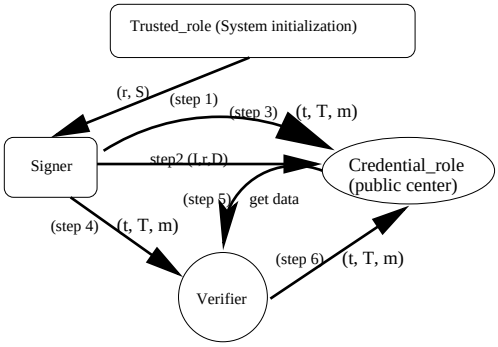

Figure 2: **Single signature scheme for ticket group_1**

the message is signed simultaneously [Stinson, 1995]. A multi-signature is the enhancement of a single signature.

4 Single signature scheme for ticket group_1

This section introduces a single signature scheme for tickets t_1, t_2, t_4. The single signature scheme is analysed for how it works for a ticket. There are four roles in the single signature scheme, Signer, Verifier, Credential_role and Trusted_role. Depending on tickets, the Signer can be a user, service or service provider that signs a signature as a ticket. The Verifier might be a user or service provider that verifies the signature of the Signer. The Credential_role in the Credential Centre will issue tickets. It provides information for the Verifier to check the signature. Whether the signature is valid or not depends on the information. The Trusted_role is a judge to solve the conflict between users, service providers and services. This is because only the Trusted_role has the secret key of the system and can trace users and service providers. Each Signer has a different but fixed identity I, which is validated once the Signer is registered in the Trusted Centre. Ticket t_4, for instance, is bound to a user only. A user can follow this scheme to sign a signature as a ticket, the service provider verifies it and then sends some information to the Credential_role and asks for payment. Tickets t_1, t_2 are similar to ticket t_4, the signers are service provider and service separately but not users.

The following is an outline of the process in the scheme. In the system initialization, the Trusted_role sends the private messages (r, S) to the Signer when the Signer I is set up, where r, S are computed by the Trusted_role, r will be used in the first verification by the Credential_role and S will be used as the first signature key by the user. In the second step, the Credential_role verifies if the data (I, r, D) sent by the Signer are valid or not, where D is used in the ticket verification. The data (I, D) will be put on a public directory in the Credential Centre if the data are valid. At this time, the Signer can do a signing message job.

While the Signer signs a message m, the Signer will send the signed message (t, T, m) as a ticket to the Verifier, and the latter checks if it is true or not, where t and T are computed by the Signer and m may include some service information etc. The data (I, D) in the Credential Centre are needed. The Verifier cannot verify the message when the data (I, D) in the Credential Centre are not correct. Then the Credential_role can revoke the anonymity of the Signer, and even find who the user is if it contacts the Trusted_role. In the final step, the Verifier sends a message which including the ticket to the Credential Centre while the

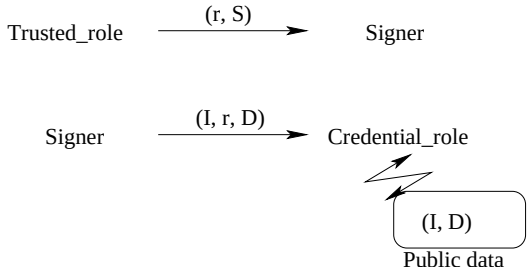

Figure 3: **Initialization for group_1**

ticket is true. The latter will update the data (I, D) and issue a charging bill. The data (I, D) is changed while the ticket is used and the ticket is invalid if the verifier cannot get the correct data (I, D). Thus, the ticket cannot be used twice and the user can see a clear statement.

This can be expressed as in Figure 2.

4.1 Initialization of the system

Usually, there are two components in a signature scheme, one is the Signer played by consumers (users), service providers, or services; the other is the Verifier played by consumers or service providers. As a ticket, a signature is valid only if its verification is correct.

The Trusted_role computes a public composite modulus $n = pq$ where factors are strong primes. The Trusted_role chooses also prime exponents e and d such that:

$$e * d = 1 \ (mod\, \phi(n)).$$

Where $\phi(n) = (p-1)(q-1)$. The pair (n, e) are made public, and d is kept secret by the Trusted Centre as the system key. The Trusted_role computes when the Signer with identity I signs up:

$$r = k^e \ (mod\, n), \quad S = k * I \ (mod\, n)$$

where $k \in_R Z_n$ ($a \in_R A$ means that the element a is selected randomly from the set A with uniform distribution). Then

$$S^e = r * I^e \ (mod\, n).$$

Let $D = S^e \ (mod\, n)$. The Trusted_role secretly sends (r, S) to the Signer whose public identity is I. S will be used as the first signature key to issue a ticket. Obviously, it is hard to compute S from D without system key d under the RSA assumption.

The Signer with the public key I sends (I, r, D) to the Credential_role, and the latter verifies the following equation:

$$D = r * I^e \ (mod\ \ n).$$

The data (I, r, D) are valid when the equation is successful, in which r and D are computed by the Trusted_role; otherwise the (I, r, D) is invalid. The Credential_role publishes in a public directory the pair (I, D) for the Signer with the public key I when the Signer is set up. The initialization processes of the system are shown in Figure 3.

4.2 The single signature scheme

The Verifier can access the public values n, e and the public pair (I, D) registered in the Credential Centre. The data D in the Credential Centre must be right, otherwise the signed message (the ticket) cannot be verified by the Verifier.

To express the general process of the single signature scheme, it is assumed that messages $m_1, m_2, ..., m_{l-1}$ $(l \geq 1)$ have already been signed by the Signer with identity I. The messages $m_1, m_2, ..., m_{l-1}$ $(l \geq 1)$ can indicate different service requirements that are included in tickets. A user can get a valid ticket if the signature is right. The corresponding public key (I, D_{l-1}) $(D_0 = D)$ of the Signer is now registered in the public directory of the Credential Centre. The message m_l will be signed by the Signer using the secret key $S_{l-1}(S_0 = S)$. The Signer and the Verifier perform the following steps.

Input: (I, D_{l-1}, e, n),

Signer:

1. Picks $r_{l-1} \in_R Z_n$ and computes: $T_{l-1} = r_{l-1}^e (mod\, n)$.
2. Computes: $S_l = S_{l-1} * m_l \ (mod\ \ n)$, S_l will be used as the secret key by the Signer with public key I in the next signing operation.
3. Computes the Hashing value $d_{l-1} = h(T_{l-1}, m_l) \ (mod\ \ n)$.
4. Computes the final witness $t_{l-1} = r_{l-1} * (S_{l-1} * m_l)^{-d_{l-1}} \ (mod\ \ n)$.

Note: A ticket is the signature (t_{l-1}, T_{l-1}, m_l). The ticket will be sent to the Credential Centre to make a record, it also needs to be sent to a service provider when the user wants to go shopping.

Credential_role:

The Credential_role computes D_l for the ticket, where

$$D_l = D_{l-1} * m_l^e \ (mod\ \ n) = S_l^e \ (mod\ \ n).$$

D_l is published in the Credential Centre. It will be used to verify the ticket by the Verifier and used to issue another ticket.

Verifier:

5. The Verifier gets (t_{l-1}, T_{l-1}, m_l) and knows (I, D_{l-1}), then checks that:

$$d_{l-1} = h(t_{l-1}^e * D_{l-1}^{d_{l-1}} * m_l^{ed_{l-1}} \ (mod\ \ n), m_l) \ (mod\ \ n).$$

It is easy to see that if the Signer follows the protocol, the equation will be valid. Indeed:

$$\begin{aligned} d_{l-1} &= h(T_{l-1}, m_l) \ (mod\ \ n). \\ T_{l-1} &= r_{l-1}^e \ (mod\ \ n) \\ &= (t_{l-1} * (S_{l-1} * m_l)^{d_{l-1}})^e \ (mod\ \ n) \\ &= (t_{l-1}^e * D_{l-1}^{d_{l-1}} * m_l^{ed_{l-1}}) \ (mod\ \ n). \end{aligned}$$

Using this protocol the Verifier is convinced with overwhelming probability that the Signer knows the secret key S_{l-1}. This S_{l-1} is used but not revealed at the end of the protocol.

6. The Verifier sents the ticket to the Credential_role. The latter updates (I, D_{l-1}) in the public director and takes a record. The ticket (t_{l-1}, T_{l-1}, m_l) cannot be used twice since it has been marked by the Credential_role. ◇

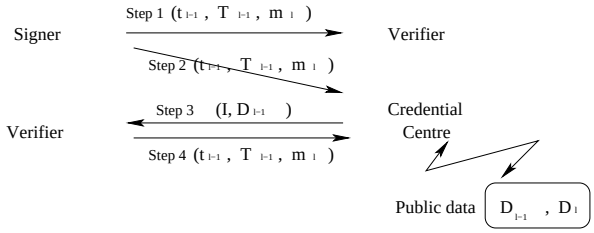

Figure 4: **Single signature scheme**

These steps are shown in Figure 4.

Remark: The Verifier must use the public data D_{l-1} in the Credential Centre when it checks whether the signed message is true or not. The signed message will be unavailable if the data D_{l-1} are changed, then the Credential_role can revoke the anonymity of the Signer.

4.3 Security analysis

This subsection first analyses threats to the system from all parts, including the outside part, which is the people who do not join the system, then shows how to solve the security problems of duplication, forgery and modification. There are four roles in the scheme. They are the Signer, the Verifier, the Credential_role and the Trusted_role.

Outside: knows the public data (I, D_l). It is hard to compute the secret key S_l from D without system key d under the RSA assumption.

Verifier: knows (I, D_l) and (t_{l-1}, T_{l-1}, m_l). But no useful message can be obtained from (t_{l-1}, T_{l-1}, m_l) to identify the secret key, the Verifier knows no more information about the key than the outside.

Credential_role: can control the ability of the Signer to sign messages. It knows only (I, D_l), so it too cannot get the secret key.

Signer: knows the secret key and the ticket, but cannot use the secret key S_l and the ticket twice. Use, for a second time, of the same secret key S_l to produce another ticket implies a second verification. If the previous verifier was honest, the public data in the Credential Centre would be updated and the second ticket would be rejected.

Trusted_role: knows the system key d, and can get the signer's key S_l. So the Trusted Centre must be trusted. Here the Trusted_role can be a judge.

The secret key S_l is not revealed at the end of the process and no secret information is revealed during the running of the system. The secret S_l is only dependent on the Trusted_role, and does not depend on the Credential_role. The security is also improved since the secret key S_l is changed once a message is signed.

The security problems of duplication, forgery and modification are addressed as follows. The double using duplication problem is prevented as the secret key S_l of a ticket is not revealed, and a copy ticket is not valid since the ticket (t_{l-1}, T_{l-1}, m_l) can only be used once. The double uses of the ticket will need two verifications; the second one will not be successful since the data for the ticket in the Credential Centre are changed after the first verification. The second duplication, eavesdropper problem can be solved by public cryptographic algorithms. Forgery is clearly prevented even the service provider and the Credential Centre do cooperation because neither the service provider nor the Credential Centre knows the secret key. Modification is prevented because the information of service

is also used in the verification. The ticket will be invalid while the service information is modified.

4.4 The usage of tickets in ticket group_1

Tickets in group_1 are records, which can be signatures, and the Credential_role can remember the records. Ticket t_4, for instance, is a signature of a user and can be bought by the user. The following analysis is only of ticket t_4 since the signature for tickets t_1, t_2 are similar to that of t_4.

We suppose that users, service providers and services are registered in the Trusted Centre. A ticket will be obtained by a user who requests the service in the ticket. When requiring a service, the user goes to the Credential Centre for a ticket, which includes the service information. The Credential_role will send a message m_l including the service information, current time, user's requirement etc to the user. As a Signer, the user signs the message and makes a ticket (t_{l-1}, T_{l-1}, m_l). The ticket (t_{l-1}, T_{l-1}, m_l) can be used to obtain a service from a service provider. As a Verifier, the service provider verifies if the ticket is valid or not, using the data (I, D_{l-1}) in the Credential Centre. Neither the service provider nor the Credential_role knows who the user is. Only the Trusted_role can trace the user's identity from the public key I. When the ticket (t_{l-1}, T_{l-1}, m_l) is used the Credential_role will make a record for the data D_{l-1}, the record will be used to prevent from duplication of the ticket and to issue a charging bill. Then users can see the charging bill at any time. This is what consumers expect when they do business on the Internet. Finally, the Credential_role can send a bill to the user.

In this mechanism presented here, a user can issue many tickets which can be used whenever, this is because whether a ticket is valid or not depends on the data in Credential Centre only. The data $D_0, D_1, ..., D_{l-1}, D_l, ...$ are published in the public directory. Thus there is no order of tickets. The user can also lend the ticket to others. He/She gives only the ticket (t_{l-1}, T_{l-1}, m_l) to others. This is very convenient for the mobile users. Furthermore, most computing in this scheme is done by the terminal side (the user or the service provider); this can reduce the resource of the mobile system.

The new single signature scheme has the following properties:

1. It is anonymous for the user.
2. The ticket can be transferred and its security is improved by the once-a-time key S_l.

However, this scheme only suits the ticket in ticket group_1. The problems of tickets t_3, t_5, t_6, t_7 cannot be solved in the scheme of this section. A multi-signature scheme to solve these problems is explained in the next section.

5 Multi-signature scheme for ticket group_2

A multi-signature scheme will be described in this section for tickets t_3, t_5, t_6, t_7. The number of signers is not limited to two or three, but v signers. Then the scheme can also be used when some services are provided by many providers.

This is, in brief, the process of the multi-signature scheme. In the system initialization, the Trusted_role computes and secretly sends the messages (r_i, S_i) to the Signers with public key ID_i (suppose v Signers) in the group when the Signers are set up. The public key ID_i is similar to the public key I in the last section, and only the Trusted_role can trace whose

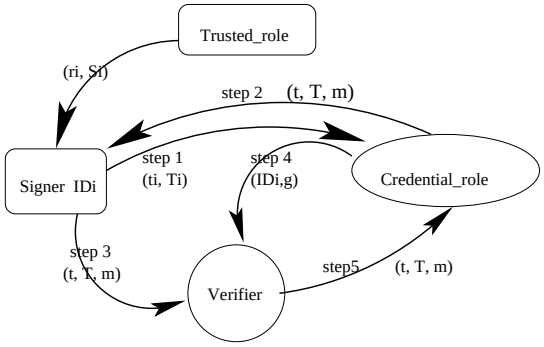

Figure 5: **Multi-signature scheme for ticket group_2**

public key is ID_i. In the second step, the Credential_role verifies if the data (ID_i, r_i, D_i) sent by the Signers are valid or not. A vector $(ID_1, ID_2, ..., ID_v, g_1)$, as the group public key, will be put in the Credential_role, where g_1 is computed by the Credential Centre and will be used in the first ticket verification, then the group can sign.

In the signature process, the Credential_role gets v pairs of data (t_{il}, T_{il}) from the Signers with identity $ID_i (1 <= i <= v)$ when a message m is signed, where (t_{il}, T_{il}) are computed by the Signer ID_i. In the next step, the Credential_role sends the signed message $(t_l = \prod_{il=1}^{v} t_{il} \ (mod \ n), T_l = \prod_{il=1}^{v} T_{il} \ (mod \ n), m)$ to the Signer as a ticket, where n is a public integer defined in the system initialization. The ticket will be sent to the Verifier and the Verifier checks if it is true or not. The Verifier may not verify if the data g_1 in the Credential Centre is not correct, and the signed message is invalid. Therefore the Credential Centre can revoke the anonymity of the Signers. In the final step, the Verifier sends the ticket to the Credential Centre and then the Credential_role can make a record for the ticket. This process is shown in Figure 5.

Suppose there are v Signers $U_1, U_2, ..., U_v$ in the signature system to sign a message simultaneously, for tickets t_3, t_5, t_6, t_7, two or three signers are enough. The scheme can also cope with some other cases for example some services provided by many providers. Ticket t_6, for instance, is bound to the user and the service provider. Then the ticket will include the agreement between these two components. Signers are needed to change in order to suit different kinds of tickets.

5.1 Initialization of the system

Similar to the previous section, the pair (n, e) are made public, and d is kept secret by the Trusted Centre as the system key. The Signer U_i of the system has a public key ID_i which is produced by the Trusted Centre when the signer joins the system. The Trusted_role computes:

$$r_i = k_i^e \ (mod\, n), \quad S_i = k_i * ID_i \ (mod\, n)$$

$k_i \in_R Z_n$, then $S_i^e = r_i * ID_i^e \ (mod\, n)$. Let $D_i = S_i^e \ (mod\, n)$, the Trusted_role secretly sends (r_i, S_i) to the Signer with the public key ID_i. S_i will be used by U_i as the first signature key. It is hard to compute S_i from ID_i without the system key d under the RSA assumption.

The Signer U_i sends (ID_i, r_i, D_i) to the Credential_role, and the latter verifies the following equation:

$$D_i = r_i * ID_i^e \ (mod \ n) \tag{1}$$

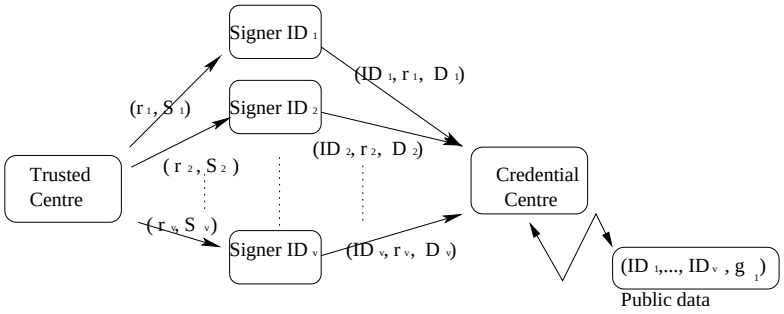

Figure 6: **Initialization of Multi-signature scheme**

The data (ID_i, r_i, D_i) are valid if the equation (1) is successful, which means all v Signers agree to issue a ticket. Otherwise the data (ID_i, r_i, D_i) are invalid. While the equation is successful for $i = 1, 2, ..., v$, the Credential_role computes a system public key:

$$g_1 = \prod_{i=1}^{v} D_i \ (mod \ \ n) = \prod_{i=1}^{v} S_i^e \ (mod \ \ n).$$

The Credential_role registers in a public directory a vector $(ID_1, ID_2, ..., ID_v, g_1)$ for Signers $U_1, U_2, ..., U_v$. The data g_1 will be used and changed when a valid signature is done. The processes are shown in Figure 6.

5.2 The Multi-signature scheme

When the Verifier accesses the system public key n, e and the public vector $(ID_1, ID_2, ..., ID_v, g_1)$ in the Credential Centre, the data g_1 must be correct, otherwise the signature is unavailable since the Verifier cannot verify the signed message.

Assuming that a message $m_l (l = 1, 2, 3, ...)$ including service information, users requirements etc will be signed by the Signers $U_1, U_2, ..., U_v$. $S_{i\,l-1}$, the secret key of Signer U_i is changed when the message m_l has been signed ($(i = 1, 2, ..., v)$ and $S_{i0} = S_i$). This means $S_{i\,l-1}$ is a once-a-time secret key and it will improve the security of the system. z is a public prime number which is known to v Signers and it will be used in the new multi-signature scheme. The processes of the multi-signature scheme are below.

Input:(ID_i, D_i, e, n),

Step 1. Signer U_i:
1.1 Picks $r_{il} \in_R Z_n$ and computes: $T_{il} = r_{il}^e (mod\, n)$.
1.2 Computes: $S_{il} = S_{i\,l-1} * m_l \ (mod\, n)$.

S_{il} will be used as the secret key by U_i in the next signing operation.
1.3 Computes: $t_{il} = r_{il} * (S_{i\,l-1} * m_l)^z \ (mod \ \ n)$.
1.4 Sends the pair (t_{il}, T_{il}) to the Credential_role.
The Credential_role is not able to get the secret key $S_{i\,l-1}$ from the data (t_{il}, T_{il}).

The Credential_role can now produce a ticket.

Credential_role:
Step 2. The Credential_role computes:

$$g_{l+1} = g_l * m_l^{ve} \ (mod \ \ n).$$

and

$$t_l = \prod_{il=1}^{v} t_{il} \ (mod \ \ n), \quad T_l = \prod_{il=1}^{v} T_{il} \ (mod \ \ n)$$

g_{l+1} is published in the public directory, it will be used to issue another ticket. (t_l, T_l, m_l) is a ticket which will be sent to the User.

It should be noted, for instance a ticket t_6, both the user and the service provider are Signers, however, the ticket (t_l, T_l, m_l) is only sent to the user. The user will send the ticket to a service provider to ask for a purchase. The service provider, as a verifier, will verify the ticket. The verifier will follow the next steps when the ticket is received.

Verifier:
Step 3. The Verifier knows the public data $(ID_1, ID_2, ..., ID_v, g_l)$ in the Credential Centre and data (t_l, T_l, m_l), checks that:

$$T_l = t_l^e * g_l^{-z} * m_l^{-zve} \ (mod \ \ n) \tag{2}$$

It is easy to see that if the Signer and the Credential_role follow the steps, the equation (2) will be valid. Indeed,

$$\begin{aligned} T_l &= \prod_{il=1}^{v} T_{il} \ (mod \ \ n) \\ &= \prod_{il=1}^{v} t_{il}^e * (S_{i\ l-1} * m_l)^{-ze} \ (mod \ \ n) \\ &= t_1^e * g_1^{-z} * m_l^{-zve} \ (mod \ \ n). \end{aligned}$$

Step 4. The Verifier sends the ticket to the Credential Centre. The latter will update the data g_l and prepare a charging bill for the user.

Remark: The signed message in the multi-signature scheme will be invalid if the data g_l is changed. Then the Credential_role can revoke the ability to sign messages of the Signers.

5.3 Security analysis

The security analysis is similar to that in the last section. This is the analysis of the attacks from the five parts.

Outside: knows the public data $(ID_1, ..., ID_v, g_l)$. They are not able to get any information about the secret key S_{il} from g_l since there is no relation between these two data.

Verifier: knows $(ID_1, ..., ID_v, g_l)$ and (t_l, T_l, m_l). But no useful message can be gathered from (t_l, T_l, m_l) to obtain the secret key S_{il}. The Verifier, likes the Outside, cannot get the secret key.

Credential_role: can revoke the anonymity of the users since it can control the ability to sign messages by the Signers. It knows only as much as the Outside does, it cannot get the secret key either.

Signers: whether the signed message is valid or not depends on the equation (2). If some Signers use their keys twice, the equation (2) cannot succeed because the data in the Credential Centre have been changed after the last signature. This means the ticket cannot be used twice. Furthermore, if a signer misbehaves many times, the Credential Centre can contact the Trusted Centre to find who the signer is.

Trusted_role: knows the system secret key. It has to be trusted and can be a judge.

The secret key S_{il} is not revealed at the end of the scheme and no secret information is revealed during the running of the system. The security of the system is also enhanced by the secret key S_{il} being changed once a message is signed. Duplication is prevented since using a ticket twice needs twice verifications, the second verification cannot success as the data in Credential Centre are changed after the first verification. In the multi-signature scheme, the Credential Centre issues tickets and sends them to users. The other four, even the Trusted_role, cannot forgery a ticket because the messages of (t_{il}, T_{il}) are only sent to the Credential Centre that is not able to get the secret key S_{il-1} from the data. To protect eavesdroppers or the ticket is sent to other users, the cryptographic technology like PGP (http://www.pgp.com) can be used between users and the Credential Centre. The user cannot modify the service information since it is needed in the ticket verification.

5.4 Usage of tickets in ticket group_2

The usage of tickets in ticket group_2, ticket t_6, for instance, binds a user and service providers and it should be an agreement between the user and the service providers. The usages of other tickets are similar to that of the ticket t_6. So only the ticket t_6 is analysed and the other tickets are omitted.

Similar to the last section, we suppose users, service providers and services are registered in the Trusted Centre and each of them has an identity. When a user requires a ticket t_6 from the Credential Centre, the Credential_role will send the user's requirement to the service providers. The Credential_role will issue a public key for the user and the service providers if the service providers agree to provide the service. The Credential_role sends a message including the service information, current time, requirement and agreements of the service providers and so on to the user and the service providers. As Signers, the user and the service providers use their secret key to sign this message, and then return the data (t_{il}, T_{il}) to the Credential Centre. The Credential_role makes a ticket (t_l, T_l, m_l) and sends it to the user. The ticket (t_l, T_l, m_l) can be used to the service provider. As a Verifier, the service provider uses the public data $(ID_1, ..., ID_v, g_l)$ in the Credential Centre to verify if the ticket is valid or not. Neither the service provider nor the Credential_role knows who the user is. Only the Trusted Centre can trace the user's identity from the public key ID_i. After the data g_l is updated, the user can see a clear charging bill in the Credential Centre. Finally, the Credential_role can send a bill to the user. This can be shown in Figure 7.

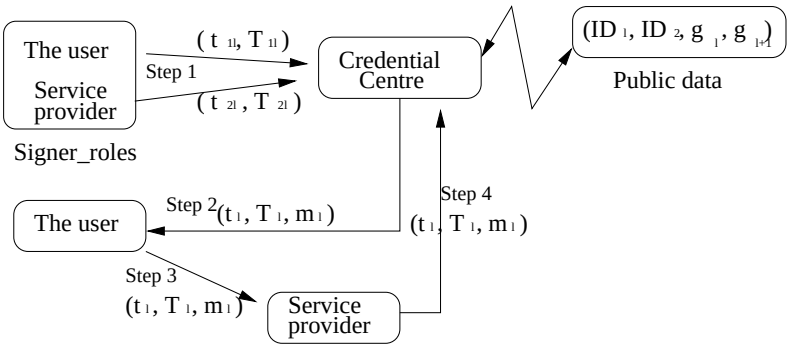

Figure 7: **Usage of ticket** t_6

Tickets have no fixed order, this means no ticket should be used early or late. This is because the data for a ticket verification are $g_1, ..., g_l, g_{l+1}$ in the public directory. The data

g_l is changed and marked while the ticket (t_l, T_l, m_l) is used. Therefore, a ticket cannot be used twice. The new multi-signature scheme has the following features:

1. It is anonymous for the user.
2. The ticket can be lent to others.
3. The security of the system is improved very much since the secret key S_{il} is used only once.

Remark: Not only mobile users but also many other Internet users can use the scheme. The Credential Centre can be decentralized for increased numbers of users. Furthermore, the Trusted Centre can be a judge when users misbehave.

6 Related work

There are some related works on this topic of mobile communication security such as [Horn and Preneel, 1998, Martin et al., 1998, Lubinski and Heuer, 2000, Wilhelm et al., 1998]. Two similar approaches, using ticket access for the third generation mobile system (UMTS) were presented by Horn and Preneel in 1998, and Martin etc in 1998 [Horn and Preneel, 1998, Martin et al., 1998]. In these solutions, the users obtain tokens from the UMTS service providers, who act as brokers. The tokens are then handed by the users to the value-added service providers as a proof of their credit worthiness. The settlements between the value-added service providers and the brokers are then accomplished off-line. The UMTS service providers will collect the billing information from all the value-added service providers accessed by given users and integrate them in a single bill addressed to the users. These mechanisms are a very significant improvement over the ones prevailing in the second generation mobile systems. However, they have the weakness of not providing anonymity to the users.

Other similar approaches for ticket-based service access are described by Patel and Crowcroft in 1997 [Patel and Crowcroft, 1997], and Buttyan and Hubaux in 1999 [Buttyan and Hubaux, 1999]. In [Patel and Crowcroft, 1997], tickets are prepaid and can only be used with the service provider that issued them (according to the categorisation described here, tickets are type t_7 and require a special model). Anonymity can be provided for all services for which it is deemed appropriate. In [Buttyan and Hubaux, 1999], tickets are issued by customer care agents and cannot be transferred to others. This approach only solves the case of ticket t_4. These two methods only solve the particular mobile access problems.

In the proposed ticket-based service access scheme, the users are anonymous since their private information is not revealed to service providers and the Credential Centre. It is a global solution for all kinds of mobile services and the tickets can be lent to others, which will be very convenient and useful for mobile environment users. The users can see a clear record of charges in the Credential Centre and identify any problems in the bill. Furthermore, the scheme can save mobile system resources, since most computing is done by users or service providers.

7 Conclusion

Mobile communication systems are becoming extremely popular, which makes the provision of services to mobile users an attractive business area. This can be regarded as a special form

of electronic commerce, where users buy services instead of products from service providers via the network. Some users prefer a global scheme and clear bill charging.

In this paper, a global ticket-based service access scheme for mobile users is proposed. First, the Credential Centre issues tickets for the users. Second, a ticket-based mechanism is implemented allowing the user to remunerate the service providers. Tickets provide a flexible and scalable mechanism for mobile access. It is an anonymous and dynamic system, and new users and new service providers can join at anytime. Furthermore users can check charges at anytime.

References

[Beimel et al., 1999] Beimel A., Ishai Y., Kushilevitz E., and Malkin T.(1999). One-way functions are essential for single server private information retrieval. In *Proc. of the 31st Annu. ACM Symp. on the Theory of Computing (STOC)*, pages 89–98.

[Bellare et al., 1996] Bellare M., Canetti R., and Krawczyk H. (1996). Pseudorandom functions revisited: The cascade construction and its concrete security. Extended abstract. In *37th Annual Symposium on the Foundations of Computer Science*, IEEE.

[Buttyan and Hubaux, 1999] Buttyan L. and Hubaux J.(1999). Accountable anonymous access to services in mobile communication systems. *Symposium on Reliable Distributed Systems*, pages 384–389.

[Chaum, 1981] Chaum D. (1981). Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM* , 24(2): 84–88,

[Frankel et al., 1995] Frankel Y., Herzberg A., Karger P., Krawczyk H., Kunzinger C. and Yung M.(1995). Security issues in a CDPD wireless network. *IEEE Personal Communications.*

[Horn and Preneel, 1998] Horn G. and Preneel B.(1998). Authentication and payment in future mobile systems. In Quisquater J., etc, editors, *Proceedings European Symposium on Research in Computer Security*, Lecture Notes in Computer Science, volume 1485, pages 277–293, Springer-Verlag.

[Housley et al., 1999] Housley R., Ford W., Polk W. and Solo D.(1999). Internet X.509 Public Key Infrastructure Certificate and CRL Profile. http://www.ietf.org/rfc/rfc2459.txt.

[Lubinski, 1998] Lubinski A.(1998). Security issues in mobile database access. In *Proceedings of the IFIP WG 11.3 Twelfth Int. Conf. on Database Security.*

[Lubinski, 2000] Lubinski A.(2000). Database security meets mobile requirements. In *Proceedings International Symposium on Database Technology Software Engineering, WEB and Cooperative Systems*, Baden.

[Lubinski and Heuer, 2000] Lubinski A. and Heuer A.(2000). Configured replication for mobile applications. *Rostocker Informatik Berichte*, volume 24, pages 101–112.

[Martin et al., 1998] Martin K., Preneel B., Mitchell C., Hitz H., Poliakova A., and Howard P.(1998). Secure billing for mobile information services in UMTS. In *Proceedings 5th International Conference on Intelligence in Services and Networks'98* Lecture Notes in Computer Science, volume 1430, pages 535–548, Springer-Verlag.

[Mehrotra, 1997] Mehrotra A.(1997). *GSM System Engineering*. Norwood, Artech House.

[Mehrotra and Golding, 1998] Mehrotra A. and Golding L.(1998). Mobility and security management in the GSM system and some proposed future improvements. In *Proceedings of IEEE*, 86(7).

[Park and Sandhu, 1999] Park J. and Sandhu S. (1999). RBAC on the Web by Smart Certificates. In *ACM Workshop on Role-Based Access Control 1999*, pages 1–9, Fairfax, VA, USA.

[Patel and Crowcroft, 1997] Patel B. and Crowcroft J.(1997). Ticket based service access for the mobile user. In *Proceedings of MobiCom: International Conference on Mobile Computing and Networking*, pages 223–232, Budapest, Hungary.

[Rivest et al., 1978] Rivest R. L., Shamir A., and Adleman L. M.(1978). A method for obtaining digital signatures and public-Key cryptosystems. *Communications of the ACM*, 21(2):120–126.

[Schechter et al. 1999] Schechter S., Parnell T. and Hartemink A.(1999). Anonymous Authentication of Membership in Dynamic Groups. In *Financial Cryptography '99*, Anguilla, British West Indies, February 1999.

[Stinson, 1995] Stinson D. R.(1995). *Cryptography: Theory and Practice*. Boca Raton, CRC Press.

[Waleffe and Quisquater, 1990] Waleffe D. D. and Quisquater J. J.(1990). Better login protocols for computer networks. In Vandewalle J. edtor, *Proceedings European Symposium on Research in Computer Security*, Lecture Notes in Computer Science, Toulouse, France, Springer-Verlag.

[Wang, Cao and Zhang, 2002] Wang H. Cao J. and Zhang Y.(2002). Ticket-Based Service Access Scheme for Mobile Users. In Oudshoorn M. editor, *Proceedings of Twenty-Fifth Australian Computer Science Conference (ACSC2002)*, Jan. 28-Feb. 2, Monash University, Melbourne, Victoria.

[Wang and Zhang, 2001] Wang H. and Zhang Y.(2001). Untraceable off-line electronic cash flow in e-Commerce. In *Proceedings of the 24th Australian computer science conference ACSC2001*, pages 191–198, GoldCoast, Australia, IEEE Computer Society.

[Wilhelm et al., 1998] Wilhelm U. Staamann S. and Buttyan L.(1999). On the problem of trust in mobile agent systems. In *IEEE Network and Distributed Systems Security Symposium*, pages 11–13, San Diego, CA.