Anonymous access scheme for electronic-services

Hua Wang [1] **Lili Sun** [1] **Yanchun Zhang** [2] **Jinli Cao** [3]

[1] Department of Maths & Computing, University of Southern Queensland
Toowoomba QLD 4350 Australia
Email: (wang, sun)@usq.edu.au
[2] School of Computer Science and Mathematics
Victoria University of Technology, Melbourne City, MC8001, Australia.
Email: yzhang@csm.vu.edu.au
[3] Department of Computer Science & Computer Engineering
La Trobe University, Melbourne, VIC 3086, Australia
Email: jinli@cs.latrobe.edu.au

Abstract

This paper presents an anonymous access scheme for electronic-services. The scheme based on tickets supports efficient authentications of users, services and service providers over different domains. Tickets are issued by a Credential Centre through a signature protocol and are used to verify correctness of the requested service as well as to direct billing information to the appropriate user. The service providers can avoid roaming to multiple service domains, only contacting the Credential Centre to certify 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: Electronic-service, signature, ticket, anonymity.

1 Introduction

With recent advances in the Internet and mobile technology, electronic-service (E-service) is becoming an important factor in business. Vendors and customers can provide and obtain services without the limitation of locations. At the same time, the security and privacy issues in E-service systems are more critical, especially for mobile consumers (e.g. moving from one place to another or using wireless mobile systems). The static security access control is incompatible with dynamic mobile environments. Consumers may access service across multiple service domains, and the traditional access mechanisms rely on cross-domain authentications using roaming agreements starting at the home location. The cross-domain authentications will involve many complicated authentication activities when the roam path is long. This limits future E-service applications.

Furthermore, there can be different types of E-services. Some services such as flight services bind users and service providers as well as services , and some services do not bind any participants, for instance shopping services by using cash, that means that everyone can use cash to buy anything in shops. Hence, depending on which parts are bound, there are different kinds of E-services. However, there is no scheme to provide a solution for all kinds of E-services. Users have to change E-service systems if they want to do different kinds of E-services on the Internet. There are several proposals related to E-service systems (Excellent E-service 2002, Paul C. 2002, Mehrotra A. 1997, Mehrotra A. and Golding L. 1998). Probably it is accurate to say that most of them lack the required flexibility in security management. The Excellent E-service (Excellent E-service 2002), for example, provides service via different channels and uses Internet technology to provide their customers with service in a cost-effective and professional manner. It manages customer communications via e-mail, text chat, fax in the same system and can develop their customer service so that the E-service system is perceived as being efficient. However, customers have to trust the system since credit card numbers are needed when they join in. Another E-service system Red Hat is designed to provide enterprise-class Linux for enterprise-class servers and applications (Paul C. 2002). It supplies source codes of some productions and requires similar private information of customers for payment. The Global system for mobile communications (Mehrotra A. 1997) provides mechanisms for user authentication as well as integrity and confidentiality, including protection of information exchanged between mobile terminals and fixed networks. It provides only limited privacy protection for users by hiding their real identities from eavesdroppers on the radio interface (Mehrotra A. and Golding L. 1998). These works are very useful for customers to access E-services, but there are some other important issues in E-service systems:

Global solution. A global solution can be used for all kinds of E-services. Users can access different services through this global solution. Current solutions can only solve particular service problems for E-services. Users have to change E-service systems if they want to do different kind of services on the Internet. This is not convenient for users.

Trust. Users in current E-service systems have to trust service providers to bill their service usage correctly and not to mishandle the related information of users and services. This kind of trusted model is not reasonable for future E-service systems. With the fast growing number of service providers and services, most of which are new on the market, and unknown to the users, such a trusted model is no longer justified. New mechanisms are needed to guarantee correct and indisputable billing and to ensure anonymous service usage.

Scalability. The basic requirements of E-service systems are to offer access to any service, anywhere, at

any time. The mechanisms of present E-service systems are not adequate to fulfill these requirements. Current solutions for users rely on cross-domain authentication and roaming agreements. A user, when applying services in a foreign domain, has to authenticate himself to the foreign service provider. This may increase a potentially time wasting authentication protocol over long distances. In addition, the foreign service provider has to trust the home domain agent of the user when cross-domain authentications between the service provider and the user. The trust is based on roaming agreements between various service providers. With the rapidly growing number of service providers, roaming agreements are becoming inadequate and no longer practical. This requires mechanisms that do not have to contact with the home domain of the user when applying a service in foreign domains, nor accessing agreements between domains.

Clear charging. In most cases, users in current E-service systems receive a charging bill monthly or bimonthly. Users do not know how much they have to pay and how many services they can access before getting the next bill. Users prefer a clear and continuously updated account statement which can be checked at anytime.

In the future, E-service systems should provide a global solution for all kinds of services and guarantee higher levels of security than current systems. It means that as well as being a global solution, 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. Furthermore, clear billing has to be ensured.

In this paper, a new approach to address the above-mentioned issues 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.

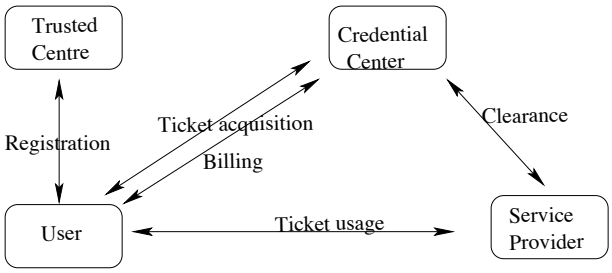

Figure 1: E-service Model

In this model, users, service providers and services are registered with the Trusted Centre. The Credential Centre issues tickets to its users. A ticket is a piece of information that represents the rights of a user to access a service provided a service provider. 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 checking the ticket, the service provider provides the requested service to the user and reports to the Credential Centre. Later, the user can see a clear charging bill in the Credential Centre.

In summary, the model has the following features:

1. It is a anonymous and trusted model. Users and service providers can trust each other. Users do not need to provide private information to service providers when they request E- services.
2. It provides a global scheme for all types of service. Users do not have to change E-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. According to these two ticket groups, a global access solution has two sub-schemes. A single signature scheme for ticket group_1 is presented in section 3 while how to extend the single signature scheme to a multi-signature scheme for ticket group_2 is discussed in section 4. The security of the global solution and its usages for different tickets are analyzed in section 5. Related works are compared 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 E-service 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 charging 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 H., Zhang Y. 2001, Wang H., Cao J., and Zhang Y. 2003) for details.

There are several advantages in using tickets for accessing services (Buttyan L. and Hubaux J. 1999):

Flexibility. Tickets may include all required information about services and service providers etc. Users can buy and use the tickets to obtain appropriate services provided by service providers. There is no contractual relationships between users and 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 perform cross-domain authentications and roaming agreements.

Anonymity. 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 (Pratel B. and Crowcroft J. 1997).

Duplication. There are two kinds of duplications needed to be considered. The first one is that users either use or sell 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 modify tickets. This is to prevent users from accessing resources for which they have not been permitted in tickets, e.g. a ticket allows travelling by a bus, should not be modifiable to allow travelling 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. We will design different mechanisms relating 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 (multi-signature). 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 cooperative 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.

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 their legitimate users used them. If necessary, anonymity can be revoked and the Trusted Centre can trace users who behave in a malicious way.

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 present a global solution for various kinds of tickets and discuss how the Credential Centre issues a continuously updated account statement for users. The global solution includes two sub-schemes for two different ticket groups. One is single signature scheme and its extension; multi-signature scheme is the other one. We are not interested in ticket t_0 since it does not bind any entities and electronic cash can be used instead of it.

3 Single signature scheme for ticket group_1

To facilitate discussions, some well-known primitive cryptographic terminologies, which will be used in the remaining 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 M., Canetti R., and Krawczyk H. 1996), digital signatures, and message authentication (Waleffe D. d. and Quisquater J. J. October, 1990).

RSA, is a public key cryptosystem that offers both encryptions and digital signatures (authentication) (Rivest R. L., Shamir A., and Adleman L. M. 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 D. 1981), especially on the Internet, since an individual need 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 the message is signed simultaneously (Stinson D. R. 1995). A multi-signature is the enhancement of a single signature.

Now we introduce a single signature scheme for tickets t_1, t_2, t_4. There are four particpants 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

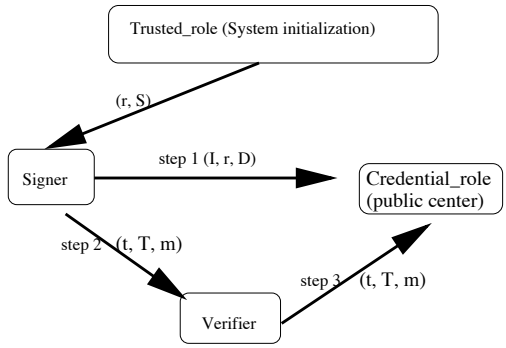

Figure 2: Single signature scheme for ticket group_1

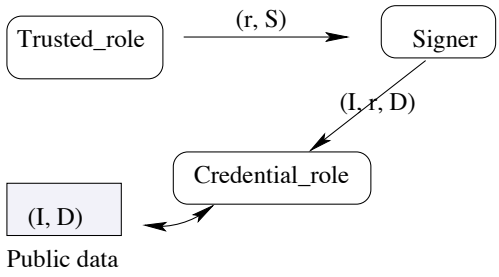

Figure 3: Initialization for group_1

Centre and does not include any private message of the Signer. 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 outline of the process in the scheme is shown in Figure 2. In 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 Signer. 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 service information and conditions 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 control the usage of the ticket, and even find who the Signer is if it contacts the Trusted_role. In the final step the Verifier sends a message which includes the ticket to the Credential Centre when the ticket is true. The latter will update the data (I, D) that is used to 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.

3.1 System initialization

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. The initialization processes of the system are shown in Figure 3.

3.2 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 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 for the next service 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 that are shown in Figure 4..

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 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

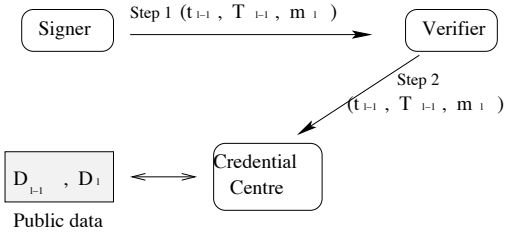

Figure 4: Single signature scheme

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 sends 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. ◇

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.

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.

4 Multi-signature scheme for ticket group_2

We will extend the single signature scheme to a multi-signature scheme for tickets t_3, t_5, t_6, t_7. The number of signers is not limited to two or three, but v signers. This means that the scheme can also be used when some services are provided by many cooperative providers.

This is, in brief, the process of the multi-signature scheme. Instead of the public key I of a signer in the last section, we use ID_i $(i = 1, 2, ...v)$ as a public keys for signers U_i since there are more than one signers in a multi-signature. In system initialization, the

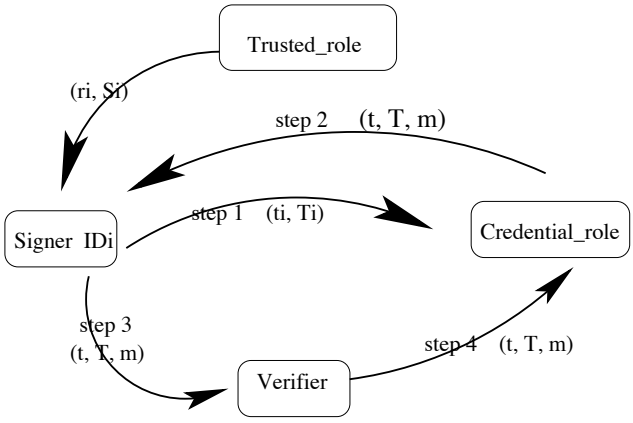

Figure 5: Multi-signature scheme for ticket group_2

Trusted_role computes and secretly sends the messages (r_i, S_i) to signers U_i in the group when the Signers are set up. This step is the same as the first step in the last section. 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 Centre, where g_1 is computed by the Credential_role 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 cannot 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 need to change in order to suit different kinds of tickets.

4.1 System initialization

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:

$$D_i = r_i * ID_i^e \ (mod \ n) \quad (1)$$

The data (ID_i, r_i, D_i) are valid when 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 is used and changed when a valid signature has been signed. The processes are shown in Figure 6.

4.2 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),
Signer U_i:
Step 1.
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 can now produce a ticket but it is not able to get the secret key $S_{i\,l-1}$ from the data (t_{il}, T_{il}).

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 required to issue another ticket. (t_l, T_l, m_l) is a ticket which will be used for asking services.

It should be noted, for instance that for 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 by the Credential_role 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:

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

It is easy to see that if the Signer and the Credential_role follow the steps, 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 Security analysis and the usage of tickets

The multi-signature scheme is an extension of the single signature since they use the same system key d. Therefore, the global solution has two sub-schemes. We analyze its security and usage for various tickets.

5.1 Threat 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. Recall that 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) and $(ID_1, ..., ID_v, g_l)$. It is hard to compute the secret key S_l from D and S_{il} from g_l without system key d under the RSA assumption.

Verifier: knows (I, D_l) and ticket (t_{l-1}, T_{l-1}, m_l) in the first sub-scheme and $(ID_1, ..., ID_v, g_l)$ and ticket (t_l, T_l, m_l) in the second sub-scheme. But no useful message can be obtained from these public data. The Verifier knows no more information about the key than the outside.

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.

Signer: knows the secret key S_l of the ticket in the sub-scheme for group_1, 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. There is a similar cases for the Signers in the second sub-scheme.

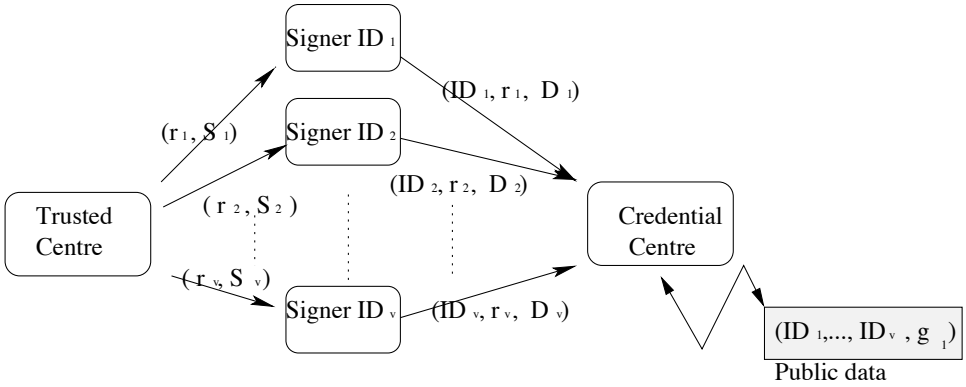

Figure 6: Initialization of Multi-signature scheme

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

Duplication is prevented since using a ticket twice needs twice verifications, the second verification cannot succeed as the data in Credential Centre are changed after the first verification. In the multi-signature scheme, for instance, 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. The chance of regenerating an old data value increases when the number of signers of a ticket increases. Whether or not a ticket is available depends on the regenerated old data and the data in the public directory. Therefore, a regenerated old data has no effects with our system because the data in the public directory is changed once a ticket is used.

5.2 The usage of tickets

Tickets are pieces of messages, which can be signatures, and the Credential_role can remember them. Ticket t_4, for instance, is a signature of a user and can be bought by the user. We first discuss the usage of the tickets in group_1. 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. 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 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 at any time. 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 time ordering 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 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 E-service system.

We now analyze 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 in ticket group_2 are similar to that of ticket t_6. So only ticket t_6 is analyzed and the other tickets are omitted.

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.

As the tickets in the group_1, tickets in group_2 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.

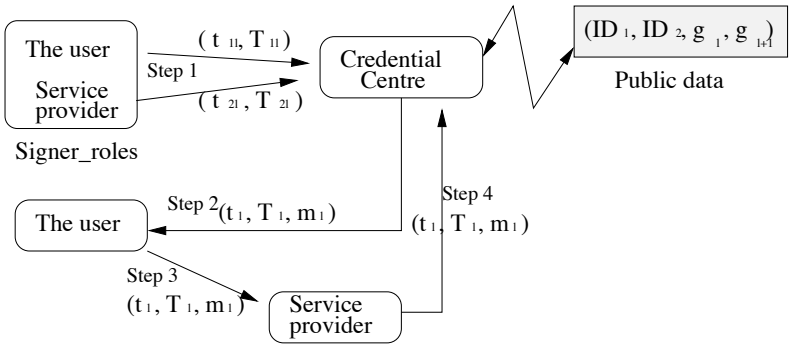

Figure 7: Usage of ticket t_6

In addition, 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.

Baesd on the two sub-schemes, the global solution 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 keys S_l and S_{il} are used only once.

6 Related work

There are some related works on ticket-based authentication Kerberos system (Neuman B.C. and Ts'o T. 1994), secure billing for E-services (Martin K., Preneel B., Mitchell C., Hitz H., Poliakova A., and Howard P. 1998), and accountable anonymous access to services(Buttyan L. and Hubaux J. 1999).

The ticket-based authentication Kerberos system (Neuman B.C. and Ts'o T. 1994) was introduced by MIT to satisfy the requirements of Project Athena. Kerberos is a distributed authentication service that allows a process running on behalf of a principal to prove its identity to a verifier (an application server, or just server) through tickets without sending data across the network that might allow an attacker or the verifier to subsequently impersonate the principal. The use of timestamps to reduce the number of messages needed for basic authentication and the addition of a "ticket-granting" service to support subsequent authentication without re-entry of a principal's password are applied. The client and server do not initially share an encryption key. A client authenticates itself to a new verifier it relies on the authentication server to generate a new encryption key and distribute it securely to both parties. The format of a Kerberos ticket for "c" to use service "s" is

$$\{T_{c,s}\}K_s = \{s, c, IP, timestamp, lifetime, K_{c,s}\}K_s$$

where $\{T_{c,s}\}K_s$ is an encrypted ticket, s is the name of a service, c, IP are the the name and IP address of a client, lifetime is the life time of a ticket. K_s and $K_{c,s}$ are the private key of a service and the session key for "c" and "s" respectively. However, our work is distinct from Kerberos system due to several limitations of the Kerberos authentication system that exist (Bellovin S. and Merritt M. 1991, Fox A. and Gribble S. 1996). For example, Kerberos tickets are limited in both space and time because tickets are usable only within the realm of the ticket-granting server, and for a period of time. The longer a ticket is in use, the greater the risk of it being stolen or compromised. By contrast, in our work, service providers can join the system without any limitation with services and a ticket can only be used once. The IP address of a client is required in Kerberos system. Such usage is problematic on multi-homed hosts (i.e., hosts with more than one IP address). This requirement limits the application of Kerberos system in local networks where ranges of IP addresses are pretended by firewalls. Furthermore, tickets of Kerberos cannot be used by mobile users since the IP address is often changed. Comparing with Kerberos system, we do not bind tickets to IP addresses. Clients can use tickets in our system wherever they are. Our tickets can be applied in local networks and mobile service systems.

A secure billing scheme for E-service has been proposed in (Martin K., Preneel B., Mitchell C., Hitz H., Poliakova A., and Howard P. 1998). It demonstrates how a micro payment scheme can be integrated into a pre-paid charging protocol and users obtain tickets from the Universal Mobile Telecommunications Systems (UMTS) service providers, who act as brokers. When requiring services from service providers, the tickets are then sent by the users to the service providers. The settlements between the service providers and the brokers are then accomplished off-line. The UMTS service providers will collect the billing information from all the service providers accessed by given users and integrate them in a single bill addressed to the users. The proposal is different from ours in two aspects. First, it focuses on authentications between users and service providers to billings by using smart card technology and elliptic curve cryptography. Therefore, there is no discussion for various services and no protocols for different kinds of tickets. By contrast, our work provides rich variety of options that can deal with all varieties of services. Second, users in the secure billing scheme have to send their identities to service providers. The identities are encrypted on the way to the service providers and are protected from eavesdroppers. However, the service providers know the identities. Hence, it has the weakness of not providing anonymity for the users with respect to service providers. In our work, users are anonymous with respect to service providers since tickets sent by the users to the service providers include all required message for services.

Finally, Buttyan and Hubaux offer anonymous access to services in mobile environments (Buttyan L. and Hubaux J. 1999). It has illustrated a ticket based mechanism for service access and proposed how agencies and tickets work together by a ticket based protocol between users, customer care agencies and service providers. The protocol accomplishes authentication of service providers to mobile users, establishment of a shared session key between users and service providers, and correct and undeniable charging. However, our work substantially differs from that proposal. Differences are due to the following three aspects. First, their protocol does not provide a global solution for various services but only a special mobile service, type t_4. By comparison, we have analyzed the

characters of various services and presented a scheme in details for all kinds of services. Second, the protocol focuses on the problems of lack of trust and scalability in mobile systems. We have discussed not only the problems of lack of trust and scalability, but also clear charging and global solution. We have designed a scheme to overcome the four important issues in E-service systems. Finally, The tickets in their work have to follow some models such as Outlet model, Kiosk model or Agency model. Therefore, the main processing in the protocol is authentications between users, service providers and customer care agencies. By contrast, users in our scheme just follow the steps to obtain tickets and use them when requiring services, and in the same time, authentications between users, service providers and Credential Centre are implemented.

7 Conclusion

E-service systems are becoming extremely popular, which makes the provision of services to 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 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. It is an anonymous system for users since tickets provide a flexible and scalable mechanism for service access. New users and service providers can trust each other and join the system at anytime. Furthermore users can obtain a continuously updated statement.

References

Beimel A., Ishai Y., Kushilevitz E., and Malkin T. (1999), One-way functions are essential for single server private information retrieval, *in* 'STOC'99'.

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.

Bellovin S. and Merritt M. (1991), Limitations of the Kerberos authentication system, *in* 'USENIX Conference Proceedings', USENIX, Dallas, TX, pp. 253–267.

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

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

Excellent E-service (2002), Excellent e-service, http://www.excellenteservice.com/.

Fox A. and Gribble S. (1996), Security on the move: indirect authentication using kerberos, *in* 'Proceedings of the second annual international conference on Mobile computing and networking', ACM Press, pp. 155–164.

Frankel Y., Herzberg A., Karger P., Krawczyk H., Kunzinger C. and Yung M. (August 1995), Security issues in a cdpd wireless network, *in* 'IEEE personal communications'.

Horn G. and Preneel B. (1998), Authentication and payment in future mobile systems, *in* 'ESORICS'.

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

Lubinski A. (August 2000), Database security meets mobile requirements, *in* 'Proceeding International symposium on database technology software engineering, WEB and Cooperative systems', Baden.

Lubinski A. and Heuer A. (2000), Configured replication for mobile applications, *in* 'Rostocker informatik berichte', Vol. 24, pp. 101–112.

Martin K., Preneel B., Mitchell C., Hitz H., Poliakova A., and Howard P. (1998), Secure Billing for Mobile Information Services in UMTS, *in* 'IS & N'.

Mehrotra A. (1997), GSM system engineering, *in* 'Norwood', Artech House.

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

Neuman B.C. and Ts'o T. (1994), 'Kerberos: An Authentication Service for Computer Networks', *IEEE Communications* **32**(9), 33–38.

Paul C. (2002), Migrate with red hat linux advanced server, http://www.redhat.com/solutions/migration/.

Pratel B. and Crowcroft J. (1997), Ticket based service access for the mobile user, *in* 'Proceedings of MobiCom: International Conference on Mobile Computing and Networking', Budapest, Hungary, pp. 223–232.

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.

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

Waleffe D. d. and Quisquater J. J. (October, 1990), Better login protocols for computer networks, *in* 'ESORICS'90', pp. 163 – 172.

Wang H., Cao J., and Zhang Y. (2003), A flexible payment scheme and its permission-role assignment, *in* 'Proceedings of the Twenty-Sixth Australasian Computer Science Conference (ACSC2003)', Adelaide, Australia, pp. 189–198.

Wang H., Cao J., Kambayashi Y. (Feb. 25-26, 2002), Building a consumer anonymity scalable payment protocol for the internet purchases, *in* '12th International Workshop on Research Issues on Data Engineering: Engineering E-Commerce/E-Business Systems', San Jose, USA.

Wang H., Zhang Y. (2001), Untraceable Off-line Electronic Cash Flow in E-Commerce, *in* 'Proceedings of the 24th Australian computer science conference ACSC2001', IEEE computer society, GoldCoast, Australia, pp. 191–198.

Wang H., Zhang Y., Cao J., Kambayahsi Y. (2003), 'A global ticket-based access scheme for mobile users', *Special Issue on Object-Oriented Client/Server Internet Environments, Information Systems Frontiers* **5**(3).

Wilhelm U. Staamann S. and Buttyan L. (1998), On the problem of trust in mobile agent systems, *in* 'IEEE network and distributed systems security symposium', San Diego, CA, USA, pp. 11–13.