Joint Conference of 1996 International Computer Symposium
December 19~21, Kaohsiung, Taiwan, R.O.C.

Anonymous Channel and Authentication in Wireless Communications*

Wen-Shenq Juang, Chin-Laung Lei and Chun-I Fan

Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan, R. O. C.
lei@cc.ee.ntu.edu.tw

Abstract

In this paper, we propose a scheme for providing anonymous channel service in wireless communications. By this service, many interesting applications, such as electronic elections, anonymous discussion groups, with user identification confidential can be easily realized. No one can trace the sender's identification and no one but the authority centre can distinguish anonymous messages from normal messages when a user uses the anonymous channel. In our proposed scheme, we also consider the key distribution problem. Our scheme can be easily applied to existing wireless systems, such as GSM and CDPD.

1 Introduction

Many applications, such as electronic voting schemes [16, 17, 22], anonymous group discussions, can be easily realized using anonymous channels [3, 5]. In wireline networks, several anonymous channel protocols [3, 5] have been proposed. The *mix-net* approach is used in [3] to realize a sender untraceable e-mail system. In the *mix-net* approach, the encrypted messages are sent to a *mix* agent who will disarrange all received messages and send them to the next agent. Finally, the last agent will send the encrypted messages to their destinations. The basic assumption of the *mix-net* approach is at least one mix agent is honest. The *dc-net* method based on the Dining Cryptographers Problem is used in [5] to achieve a sender untraceable e-mail system which is unconditionally or cryptographically secure depending on whether it is based on one-time keys or on keys generated by public key distribution systems or pseudo random number generators. In a *dc-net* scheme without a trusted authority, every pair of potential senders must share a secret key. To send an anonymous message, all potential senders must transmit the message bit by bit. In the *ith* bit transmission, each potential sender outputs the sum (modulo two) of all the *ith* bits of the keys he shares. If a sender wishes to transmit the bit '1', he inverts his output. Since every secret bit contributes exactly twice, if only one participant transmits the bit '1', the total sum (modulo two) of all participants' outputs must be one. If the message includes some redundancy information, other potential senders can detect collision of messages. When the sender detects a collision, he can retransmit his message after a period of time. In the *dc-net* method, it does not need any trusted mix-agent, but all potential senders must participant in the mail system when someone is delivering a message.

The most important and popular wireless systems are digital cellular systems, such as AMPS and GSM [14]. The first generation cellular systems, such as AMPS, are primarily aimed at voice communications. There is a growing need for wireless communication systems to provide data services, such as e-mail, fax etc., for mobile units. The *Cellular Digital Packet Data* (CDPD) system is designed to provide data services in an overlap to the existing analog cellular telephone network used in the North America. It is designed to make use of cellular channels that are not being used for voice traffic. Another cellular system, the *global systems for mobile communication systems* (GSMs) used in European and some Asian countries, is designed to provide secure digital services such as user authentication, traffic confidential and key distribution.

In wireless communications, due to the lack of association between a user and a particular physically secure location, it can make an illegal user more easy to attempt at fraudulent acquisition of service.

*This research is supported in part by the National Science Council of the Republic of China under grant NSC-86-2221-E-002-014.

Thus, user authentication and the anonymous channel service must be addressed simultaneously. In wireless channels, user anonymity and user authentication have not been addressed simultaneously. Many schemes [1, 15] have considered user identification confidential against outsiders but not against the service provider. For accounting purpose, before the service provider provides an anonymous channel to a user, user authentication must be considered in advance.

In wireless communications, it is easier to realize anonymous channels due to roaming, dynamic channel assignment and broadcasting. In this paper, we will design an efficient anonymous channel in wireless environments, such that, it can be easily applied to the existing wireless systems. The user anonymity in our scheme is neither based on any trusted authority like [3] nor on the cooperation of all potential senders [5]. In our scheme, message transmissions are transparent to the *visiting service domain* (VSD), and no one but the *home service domain* (HSD) can distinguish general messages from anonymous messages. In the downlink channels (base station to mobile station), our scheme uses secret key cryptosystems to encipher the transmission message but in the uplink channels (mobile station to base station), it uses public key cryptosystems to preserve the message privacy. The reason for our scheme to use public key encipher functions instead of secret key encipher functions in the downlink channel is to preserve the privacy of the subscriber's identification and alleviate the key management problem introduced by secret key cryptosystems.

The remainder of the paper is organized as follows: In Section 2, we describe a high-level system architecture for GSM and CDPD-like wireless communication systems. In Section 3, we briefly describe blind signatures and low-cost public key message encryption algorithms used in our protocol. Based on the high-level system architecture, we propose our anonymous channel and the authentication scheme in Section 4. Then we discuss the implementation issues in Section 5. Finally, a concluding remark is given in Section 6.

2 System Architecture

In this section, we describe a high-level system architecture for GSM and CDPD-like wireless communication systems. In this architecture, a *mobile station* (MS) will communicate with a *base station* (BS), which comprising the radio equipment and small switch functions. The BS links the *mobile service switching centre* (MSC) with the MSs. The MSC, which performs the switching functions for MSs and allocates radio resources, can connect to other MSCs or the existing wireline networks, such as PSTN, B-ISDN, etc., via wireline communications. The MSC also connects to a dedicate *authentication centre* (AUC) which performs the authentication for each call attempt made by an MS.

BSs, MSCs and the AUC collectively form a *service domain* (SD). For simplicity, we will treat MSCs and the AUC as an integrated logical entity. Any data manipulation in an SD must be done in the logical entity, and all BSs will not keep any important secret data. The functions of BSs are just data receiving and transmission. Any user who plans to acquire a wireless service has to register himself with a SD and becomes a subscriber to this SD. The SD that a user registered with is referred to as his *home SD* (HSD), and other SDs that the user visits are his *visiting SDs* (VSDs). The SDs do not have to trust each other, so they do not have to share any private information of their own subscribers. In each SD, the network topology of BSs and the MSC is a star network. Each BS only connects to its MSC. An MS can communicate with the current MSC via the nearest BS by radio. The MSCs can communicate with other MSCs or existing B-ISDN, Internet by the wireline networks.

3 Blind signature and low-cost public key message encryption

3.1 Secure blind signature schemes

The concept of blind signatures was proposed by Chaum [4]. It allows to realize secure voting schemes [16, 17, 22] protecting the voters' privacy and secure electronic payment systems [6, 12] protecting customers' anonymity. Such systems have a party called the signer who is able to make certain digital signatures. The other parties called requesters would like to obtain such signatures on messages they provide to the signer. In a distributed environment, the signed blind messages can be thought as tickets in some applications, such as secret voting schemes, and as fixed amount of e-cashes in secure electronic payment systems. A distinguishing property required by a typical blind signature scheme [2, 4, 11, 18] is so-called the "unlinkability", which ensures that the requesters can prevent the signer from knowing the exact correspondence between the actual signing process performed by the signer and the signature which later made public. The security of the schemes proposed in [4, 11] is based on the hardness of factorization [21] while the scheme proposed in [2, 18] is based on the hardness of computing discrete logarithm [10, 19].

Any secure blind signature scheme can be applied to our proposed scheme. For simplicity, we adopt the RSA blind signature scheme [4] as an example. The

RSA blind signature scheme is illustrated as follows: Let m be a message to be signed and s be the signature of m.

1. The requester sends to the signer a message $m' \equiv_n mR^e$, where (e, n) is the public key of the signer and R is a random number chosen by the requester such that $\gcd(R, n) = 1$.
2. Upon receiving the message m', the signer generates its signature $s' \equiv_n (m')^d$ with his secret key d. Then he sends the message s' back to the requester.
3. Upon receiving the message s', the requester can obtain signature s for m by computing $s \equiv_n s'R^{-1} \equiv_n (mR^e)^d R^{-1} \equiv_n m^d$.

The signer can not derive m from m' since m' is transformed by the unknown random number R. On the other hand the requester, knowing the value R, can compute the signature s of the message m from the message s'.

3.2 Low-cost public key encryption algorithm

For achieving the low cost computations in mobile units, the subscriber encrypts his messages by modified RSA encryption schemes [13, 20, 24]. For simplicity, we use the Rabin's scheme [13, 20] in our protocol to encrypt subscriber's messages. The Rabin's scheme is illustrated as follows. Every user i randomly chooses his secret key (p_i, q_i), where p_i and q_i are two large strong primes, and publishes his public key n_i, where $n_i = p_i q_i$. For sending a secret message m to user i, anyone can encrypt the message by compute $c = m^2 \bmod n_i$ and send the ciphertext c to user i. With the information of the secret key (p_i, q_i), user i can efficiently decrypt the ciphertext as $m = \sqrt{c} \bmod n_i$. Rabin proved that computing m given c and n_i is as difficult as factoring n_i. Although the Rabin's encryption function is not one-to-one (it is four to one), if we add some redundancy information to the message, the user with the secret key can decrypt the ciphertext and choose the correct plaintext. Since Rabin's scheme only needs one modulo multiplication to encrypt a message, it is especially suitable for mobile units with low-computation capability.

4 The proposed scheme

In this section, an authentication scheme for anonymous channel is presented. A typical session of the scheme involves a subscriber, his HSD and the VSD from where the subscriber requests the service. The communication between the subscriber and his VSD is via wireless communications. The VSD can communicate with the HSD via a high-speed wireline network. The scheme consists of three protocols: the ticket issuing protocol, the authentication protocol and the ticket revivification protocol. In our scheme, if a subscriber plans to send an anonymous message, he first requests a blind ticket from his HSD using the ticket issuing protocol. Then he can use the ticket in the authentication protocol. If the lifetime of the ticket expires, the subscriber can revive the lifetime of the ticket via the ticket revivification protocol. For accounting purpose, the HSD keeps a ticket database to check if the requested ticket is out of money.

The underlying assumptions of these protocols are that: (a) There exists a secure blind signature scheme [2, 4, 18]; (b) There exists a secure asymmetric cryptosystem [20, 24]; (c) There exist a secure symmetric cryptosystem [7] and a secure one-way hash function [23]; (d) No one can derive the origin of any message in the underlying mobile communication systems [14].

In our protocol, for simplicity, we use RSA blind signature schemes [4] as an example to generate blind tickets in the ticket issuing protocol. For achieving the low cost computations in mobile units, the subscriber encrypts his messages by modified RSA encryption schemes [20, 24].

Let S denote a subscriber, V denote the current VSD of S, H denote the HSD of S and "$X \rightarrow Y : Z$" denote that sender X sends message Z to receiver Y. Also, let K_{vh} be the secret key shared by H and V, HID be H's identification number, $\{m\}_{e_r}$ denote the ciphertext of m encrypted using Rabin's public key e_r, $(m)_k$ denote the ciphertext of m encrypted using the secret key k of some secure symmetric cryptosystem and " $\cdot$ " denote the conventional string concatenation operator. Let f be a secret one-way function known only by H and h be a public one-way function. H has RSA keys n_h, e_h and d_h, where n_h and e_h are the public keys and d_h is the corresponding secret key, and the Rabin's public key e_r and the corresponding secret keys p_r and q_r,where p_r and q_r are two large strong primes and $e_r = p_r * q_r$. Let ID_i be the identification of subscriber i. Upon registration, every subscriber i shares a unique identification (ID_i) and a secret key $f(ID_i)$ with his HSD and keeps $(ID_i, f(ID_i), e_r, n_h, e_h, h())$ in his handset (mobile unit).

4.1 The ticket issuing protocol

Before S can send an anonymous message via the wireless channel, he must purchase a blind ticket from H. This ticket will be used as the authentication ticket and the hash value of the ticket will be used as the secret key shared with H when S uses the anonymous channel. The protocol is as follows.

1. $S \rightarrow V : HID, N_1, \{ID_i, \Psi, Cert_i, T_1\}_{e_r}$
2. $V \rightarrow H : \{ID_i, \Psi, Cert_i, T_1\}_{e_r}, N_2$

3. $H \rightarrow V : (\Gamma, N_2)_{K_{vh}}$
4. $V \rightarrow S : N_1, \Gamma$

In step 1, S sends his HID, a nonce N_1, his ID_i, a blind message Ψ, his authentication information $Cert_i$ and a timestamp T_1 to V, where

$$\Psi = r^{e_h}(Tkt) \bmod n_h,\ gcd(r,n) = 1 \text{ and } 1 < r < n,$$

$$\text{and } Cert_i = (T_1, \gamma)_{f(ID_i)}, \qquad (1)$$

where γ is a random number. The timestamp T_1 will be used for accounting check such that any malicious person can not replay the message $\{ID_i, \Psi, Cert_i, T_1\}_{e_r}$ to fool H. The blind ticket information $Tkt = RD \cdot \delta \cdot lifetime$ contains a redundancy string RD, a ticket lifetime and a random number δ for increasing entropy of the message Tkt. All messages, except HID and N_1, will be encrypt by the H's Rabin's public key for privacy. In step 2, V simply passes a nonce N_2 and the received encrypted message to H.

Upon receiving the message in step 2, H first decrypts the message and then checks if S's identification is valid by verifying if the certificate $Cert_i$ is valid and T_1 is fresh. If yes, H signs the blind ticket by computing

$$\Gamma = (\Psi)^{d_h} \ mod\ n_h \qquad (2)$$

and deducts a fix amount of money from S's account. Then he sends the signed ticket $(\Gamma, N_2)_{K_{vh}}$ back to V. For simplicity we will take the size of $|n_h|$ to be 512 bits in our discussion. For verifying the signature signed by H, we can define a valid signature space as

$$\begin{aligned} \Re &= \{RD \cdot x \cdot y | RD = 0^{64}, x \in \{0,1\}^{416}, |y| = 32, \\ y &\geq Current - T\}, \end{aligned} \qquad (3)$$

where $Current$ is the current time when the verifier receives the ticket and T is the length of the duration that the ticket is valid.

Upon receiving the message in step 3, V checks if the nonce N_2 is in the encrypted message. If yes, V then simply broadcasts the received blind ticket Γ and N_1 to S via the wireless channel. The nonce N_1 will be used as the indicator of blind ticket Γ so that S can seize Γ from the downlink channel. Upon receiving the blind ticket, S can obtain the real ticket by computing

$$K_{sh} = r^{-1}\Gamma \ mod\ n_h = (Tkt)^{d_h} \ mod\ n_h \qquad (4)$$

and verify the validity of the ticket by checking if $(K_{sh})^{e_h} \bmod n_h = Tkt$.

4.2 The authentication protocol

After receiving the ticket from V, S can use it as an authentication ticket when he requests an anonymous message service. When the first anonymous call is made, H will assign a pseudo account (PA) to this ticket. This ticket can be used until the volume of its associated PA becomes empty. The following protocol is the ith anonymous call with respect to this ticket.

1. $S \rightarrow V : HID, N_3, \{K_{sh}, r_i\}_{e_r}$
2. $V \rightarrow H : \{K_{sh}, r_i\}_{e_r}, N_4$
3. $H \rightarrow V : (K_i, r_i, PA, lifetime, (r_i)_{h(K_{sh})}, N_4)_{K_{vh}}$
4. $V \rightarrow S : N_3, (I_i, r_i)_{K_i}, (r_i)_{h(K_{sh})}$

In step 1, S sends his HID, N_3 and the encrypted message $\{K_{sh}, r_i\}_{e_r}$ to V. The encrypted message includes the authentication ticket K_{sh} and the ith random challenge r_i. The challenge r_i is used for computing the ith session key K_i and checking freshness. In step 2, V sends the received message $\{K_{sh}, r_i\}_{e_r}$ and a nonce N_4 to H.

Upon receiving the message in step 2, H first decrypts the message, and then checks if $((Tkt)^{d_h})^{e_h}$ $mod\ n_h = (Tkt) \in \Re$, where the domain $\Re$ is defined in (3) and r_i is fresh. H rejects the ticket if it is not valid. If it is valid and the ticket is fresh, H assigns a new PA to this ticket. Otherwise, H retrieves the PA corresponding to this ticket from the ticket database. If the PA is not empty, then he computes the session key $K_i = h(K_{sh} \cdot r_i)$, and sends the message $(K_i, r_i, PA, lifetime, (r_i)_{h(K_{sh})}, N_4)_{K_{vh}}$ back to V.

Upon receiving the message in step 3, V decrypts the message and checks if the nonce N_4 is in it for freshness checking. If yes, V generates a pseudo identification number I_i for this call and encrypts I_i and the challenge r_i with the session key K_i. Then he sends the message $N_3, (I_i, r_i)_{K_i}, (r_i)_{h(K_{sh})}$ back to S. The nonce N_3 will be used as the indicator of this call response so that S can seize the message $(I_i, r_i)_{K_i}, (r_i)_{h(K_{sh})}$ from the downlink channel.

After receiving the encrypted message, S then obtains I_i by the session key K_i, which can be computed by $K_i = h(K_{sh} \cdot r_i)$, and verifies the freshness of the message from the challenge r_i. Then he can use the pseudo identification number I_i and the session key K_i to send anonymous messages.

4.3 The ticket revivification protocol

If the lifetime of the requested ticket expires, S can ask H to revalidate the ticket lifetime. The protocol is similar to the ticket issuing protocol except the authentication message $ID_i, Cert_i$ is replaced by the expired ticket K_{sh}. Note that S does not have to send another stamp T_2 to H since if any malicious person replays the encrypted message to fool H, he can neither derive the expired ticket nor any new ticket without knowing the expired ticket, and H will not deduct any money from S. The protocol is described as follows.

1. $S \to V : HID, N_5, \{K_{sh}, Tkt'\}_{e_r}$
2. $V \to H : \{K_{sh}, Tkt'\}_{e_r}, N_6$
3. $H \to V : ((K'_{sh})_{h(K_{sh})}, N_6)_{K_{vh}}$
4. $V \to S : N_5, (K'_{sh})_{h(K_{sh})}$

In step 1, S sends his HID, a nonce N_5 and the encrypted message $\{K_{sh}, Tkt'\}_{e_r}$ to V. The encrypted message includes the expired ticket K_{sh} and the new ticket contents $Tkt' = RD \cdot \delta' \cdot lifetime'$, where $lifetime'$ is the new ticket lifetime and δ' is another random number. In step 2, V simply passes the received encrypted message and a nonce N_6 to H.

Upon receiving the message in step 2, H first decrypts the message, and then computes $(K_{sh})^{e_h}$ $mod\ n_h = Tkt$ and checks if the redundancy information RD is in the message Tkt. If yes and the expired ticket has been used, he signs the new ticket $K'_{sh} = (Tkt')^{d_h} \bmod n_h$, and places the new ticket into the entry of the expired ticket in the ticket database. If yes and the expired ticket has not been used, he also signs the new ticket K'_{sh} and adds a new entry contained the expired ticket and the new ticket to the ticket database. Then he sends the encrypted message $((K'_{sh})_{h(K_{sh})}, N_6)_{K_{vh}}$ back to V.

Upon receiving the message in step 3, V checks if N_6 is in the encrypted message. If yes, he broadcasts the message $N_5, (K'_{sh})_{h(K_{sh})}$ via the wireless channel. The nonce N_5 will be used as the indicator of this call response so that S can seize the encrypted ticket $(K'_{sh})_{h(K_{sh})}$ from the downlink channel. Upon receiving the encrypted message, S can obtain the new ticket K'_{sh} by decrypting the message $(K'_{sh})_{h(K_{sh})}$.

5 Discussions

5.1 Untraceability and Accountability

The most important feature of our proposed protocol is the untraceability property. Moreover, the subscriber and the HSD must authenticate each other. We now describe that our proposed scheme satisfies the above properties.

Based on the technique of blind signatures, we first describe that no one can derive the subscriber's identification when he uses the anonymous channel.

Definition 1 (Un-traceability) A channel is said to be *untraceable* if no one can derive the identification of the sender of a message transmitted through this channel.

There are two possible ways that the identification of a subscriber may be deduced by his HSD: (1) the HSD and the VSD cooperate to derive the link between the plaintext message $(ID_i, \Psi, Cert_i, T_1)$ which is sent to V in step 1 of the ticket issuing protocol and the plaintext message (K_{sh}, r_i), which is sent to V in step 1 of the authentication protocol. (2) Acquire the identification of S when he sends the message $\{K_{sh}, r_i\}_{e_r}$ to V in step 1 of the authentication protocol or sends any message to V in the subsequent communication.

To derive the link between the string (ID_i, Ψ, $Cert_i$, T_1) and (K_{sh}, r_i), is computational infeasible since it clearly contradicts to the assumption (a) mentioned in Section IV. To acquire the identification of S when he sends the message $\{K_{sh}, r_i\}_{e_r}$ to V or sends any message to V in subsequent communication is impossible since it contradicts the assumption (d) mentioned in Section IV.

Thus, we claim that the provided channel is untraceable.

Definition 2 (Accountability) A channel is said to be *accountable* if no subscriber can use it without being charged.

In our protocol, the accounting of using wireless channel is achieved by PAs. When a subscriber requests a blind ticket from the HSD, the HSD will withdraw a fix amount of money from the subscriber's account. Upon receiving a ticket from an anonymous subscriber, the HSD will assign a PA to this ticket. All costs of using the wireless channel will be deducted from its PA until the volume of PA is empty. If some subscriber plans to use the anonymous channel without paying the cost, he must forge a legal ticket $(Tkt'')^{d_h}$ $mod\ n_h$ or impersonate a legal subscriber i in the ticket issuing protocol. However, this will contradict to the assumption that RSA blind signature scheme is secure or the assumption that there exists a secure symmetric cryptosystem. From the above, we claim that the provided channel is accountable.

For manipulating the database of PA, a ticket can only be used during its life time. If the life time of the ticket expires, the subscriber can revive the ticket by the ticket revivification protocol.

5.2 Authentication in the same VSD

In GSM [8], the secret key cryptosystem is used to implement the authentication protocol. It uses the challenge and response method to simplify the authentication process when requesting calls are in the same VSD. In the authentication protocol, the subscriber first sends his international identity number to the VSD, then the VSD sends the request to his HSD. The HSD then sends a set of 3-component tuple ($Rand_i$, $SRES_i$, K_{c_i}), $1 \le i \le j$, back to the VSD. Upon receiving the set of 3-component tuple ($Rand_i$, $SRES_i$, K_{c_i}), $1 \le i \le j$, the VSD chooses a 3-component tuple ($Rand_i$, $SRES_i$, K_{c_i}) and sends

$Rand_i$ to challenge if the subscriber can reply with $SRES_i = A3(Rand_i, K_i)$, where K_i is the secret key shared between the subscriber and the HSD and $A3$ is a public one-way function. If yes, then the subscriber can communicate with the VSD using the session key $K_{c_i} = A8(Rand_i, K_i)$, where $A8$ is another public known one-way function.

In the authentication protocol of our proposed scheme, if the next call is also in the same VSD, the authentication protocol can be simplified as follows.

1. $S \rightarrow V : I_i, r_{i+1}$
2. $V \rightarrow S : r_{i+1}, (I_{i+1}, r_{i+1})_{K_{i+1}}$

In step 1, the subscriber sends the previous pseudo identification number I_i and the challenge $r_{i+1} = h(r_i \cdot K_{sh})$, depending on the previous challenge r_i and the secret ticket K_{sh}, to the current VSD . Upon receiving the message in step 2, the VSD first computes the current session key $K_{i+1} = h(K_i \cdot r_{i+1})$ and chooses a new pseudo identification number I_{i+1}. Then he simply broadcasts the challenge value r_{i+1} and the encrypted message $(I_{i+1}, r_{i+1})_{K_{i+1}}$, which is encrypted by the current session key K_{i+1} via the wireless channel. The challenge r_{i+1} will be used as the indicator of the encrypted message $(I_{i+1}, r_{i+1})_{K_{i+1}}$ so that S can seize $(I_{i+1}, r_{i+1})_{K_{i+1}}$ from the uplink channel. Upon receiving the encrypted message, the subscriber can retrieve the current pseudo identification number I_{i+1} by the session key $K_{i+1} = h(K_i \cdot r_{i+1})$ and verify the freshness of the message from the challenge r_{i+1}. Then he can use the pseudo identification number I_{i+1} and the session key K_{i+1} to send anonymous messages.

In stead of randomly choosing the challenge value r_{i+1}, the challenge values r_{i+1} of the $(i+1)th$ call must be modified to $r_{i+1} = h(r_i \cdot K_{sh})$, which can only be computed by the subscriber and the HSD. If the VSD is not honest, he can only attempt at fraudulent acquisition of this call. When the next call is initialized, the next challenge value must be recomputed.

5.3 Comparison of protocols

We summarize the functionality and complexity of related wireless authentication protocols in Table 1. The most important feature of our protocol is its untraceability. Generally, the adversaries can be classified into "outsiders" and "insiders". An "outsider" is someone who can only ascertain what can be intercepted via radio waves. While an "insider" is someone who can obtain information by theft, conspiracy or computer system intrusion. An "insider" can be either a VSD or the HSD. Generally, the HSD will keep some secret information of the subscriber, such as the subscriber's secret key. The only secret information shared between the subscriber and the VSD is the session key for some session call. From Table 1, we know that no one can derive sender's identification in our protocol. Lin *et al.*'s scheme can hide a sender's identification from outsiders and the VSD but not the HSD. For GSM and Beller *et al.*'s scheme, the VSD and the HSD both know a sender's identification when the sender sends messages. Beller *et al.* proposed a session key protection scheme that if some subscriber's secret key has been compromised, the old session keys will not been compromised. In our authentication protocol, the ith session key K_i is the hash value of the blind ticket K_{sh} and the ith challenge value r_i. All the challenge values are encrypted by the HSD's Rabin's public key. If the HSD's Rabin's secret key has not been compromised, even the intruder knows the subscriber's secret key, he still can not know the conversation of the ith session. In our authentication protocol, the number of multiplications is 1 for the subscriber by using the Rabin's encryption function.

	Untraceability			Session Key	mod
	Outsider	VSD	HSD	Protection	multiplies
Our scheme	Yes	Yes	Yes	Yes	1
Lin [15]	Yes	Yes	No	Yes	1
Beller [1]	Yes	No	No	Yes	2
GSM [8]	Yes	No	No	No	0

Table 1: The functionality and complexity of related wireless authentication protocols.

Assume that n is the number of potential anonymous senders in the dc-net method, t is the number of mix-agents in the mix-net approach and x is the number of bit operations of a DES encryption, r is the number of bit operations of an RSA encryption, with a 512-bit modulo, and y is the computation cost of the subscriber in our authentication protocol, which includes one modulo multiplication and several DES decryptions. Table 2 illustrates the comparison of related anonymous channels. Mix-net needs at least a trustworthy mix-agent to preserve the sender's identification privacy. Dc-net needs all possible senders to cooperate when someone sends an anonymous message. Our protocol needs neither a trustworthy agent nor all potential senders cooperating during an anonymous message transmission. User authentication is not considered in the dc-net method since all potential senders need to cooperate when some one is delivering an anonymous message and all potential senders is the candidate of the current sender. If the anonymous message is m bits, the communication complexity of our protocol is $2m$. In dc-net method, it need nm bits to send an m bits anonymous message which is very impractical in a large network. In [9], a fast RSA implementation on the DSP56000 achieves 11.6K bits/s for 512-bit exponentiation with Chinese remainder theorem and the DES implementation with the optional DES chip runs at 3.8M bits/s in CBC mode. In the above implementations, the ratio of r/x is about $\frac{3.8*10^6*8}{11.6*10^3} = 2.62*10^3$. The computation cost for the subscriber in our scheme is only $y + \lceil \frac{m}{64} \rceil x$ bit operations. But the computation cost of the mix-net method for transmitting m bits anonymous message is $\lceil \frac{m}{512} \rceil tr$ bit operations. In the dc-net method, each sender must share a secret key with every other potential sender. But in our scheme, every subscriber only need to share a secret key with his HSD. It is more suitable for the dc-net method to use the broadcast transmission mode to transmit messages since all users need to compute the sum of all potential senders' outputs. But in the mix-net method, since the sender only needs to transmit the anonymous message to the first trusted agent, it does not need broadcast channel to transmit anonymous messages.

	Mix-net	Dc-net	Our protocol
Require a trusted authority	Yes	No	No
Cooperation of all paticipants	No	Yes	No
Free of charge	No	Yes	No
Communication complexity	tm $(bits)$	nm $(bits)$	$2m$ $(bits)$
Computation cost (senders)	$\lceil \frac{m}{512} \rceil tr$	mn^2	$y + \lceil \frac{m}{64} \rceil x$
No. of secret key (each sender)	0	$n-1$	1
Network infrastructure	Wireline	Wireline	Wireless
Transmission mode	Multicast	Broadcast	Broadcast

Table 2: The comparison of related anonymous channels.

5.4 Implementation considerations

Different from Kerberos [23] which the ticket lifetime is chosen by the key server, the ticket lifetime is chosen by the subscriber in our protocol. This approach let the subscriber hide the ticket lifetime information when he buys a ticket. The service provider can provide several values of the durations that the ticket is valid according the response time the subscriber can tolerate. For example, if there are three kinds of durations: one, three and five years. If the subscriber chooses the duration time is five years, he can use this ticket longer but has a longer waiting time for the service provider to check the validation of this ticket.

We assume that a portable unit contain a low-power microcontroller in order to perform the various tasks associates data manipulation and user interface. A typical 8-bit microcontroller dissipates 75-150 MW when operating at 6-MHz. Beller *et al.* [1] implemented modular multiplication on a typical microcontroller, such that, the implementation completes a single 512-bit modulo multiplication in 180 ms. The network server (HSD) can be done using a special-purpose processor. Dussé and Kaliski [9] have published an algorithm and claimed performance results that correspond to performing a single 512-bit modular multiplications in around 145 μs on a general-purpose Digital Signal Processor (DSP). A single such processor could perform the decryption of the modified RSA cryptosystems [13, 20, 24] for 10-20 calls/s. Assume 3 calls per user per hour, thus processor can support of 12000-24000 customers.

When e_h=3, our protocol requires a precomputation on the order of 200 modular multiplications (20 seconds on an 8-bit microcontroller) in the portable unit for the ticket issuing protocol because the subscriber must compute the inverse of r to extract the real ticket from the blind ticket. The ticket issuing protocol can be performed in advance, and the ticket can be preserved for future authentication. For the ticket revivification protocol, it only needs 3 modular multiplications: one for encrypting the old ticket message and two for verifying the new ticket when receiving the ticket from the HSD.

6 Conclusion

In this paper, we propose an efficient scheme for providing anonymous channel service in mobile communications. By this service, many interesting applications with user identification confidential can be easily realized. In our proposed scheme, we also consider the key distribution problem. Our scheme can be easily applied to existing mobile communication systems, such as GSM, CDPD, without affecting the underlying structure of VSDs. The user anonymity in our scheme is neither based on any trusted authority nor on the cooperation of all potential senders. Nobody can trace the identification of any subscriber when he uses the anonymous channel. Furthermore, no one but the HSD authority can distinguish anonymous messages from normal messages.

References

[1] M. J. Beller, Li-Fung Chang and Yacov Yacobi, "Privacy and authentication on a portable communication system," IEEE J. on Selected Areas in Commu, Vol. 11, No. 6, pp. 821-829, 1993.

[2] J. L. Camenisch, J. M. Pivereau and M. A. Stadler, " Blind signatures based on the discrete logarithm problem," Advances in Cryptol-

ogy: Proc. of EuroCrypt'94, LNCS 950, pp. 428-432, Springer-Verlag, 1995.

[3] D. Chaum. "Untraceable electronic mail, return addresses, and digital pseudonyms," Communi. of the ACM, Vol. 24, No. 2, pp. 84-88, 1981.

[4] D. Chaum, "Blind signatures systems," Advances in Cryptology: Proc. of CRYPT'83, Plenum, pp. 153.

[5] D. Chaum. "The dining cryptographers problem: unconditional sender and recipient untraceability," J. of Cryptology, No. 1 : pp. 65-75, 1988.

[6] D. Chaum, A. Fiat and M. Naor, "Untraceable electronic cash," Advances in Cryptology: Proc. of CRYPT'88, LNCS 403, Springer-Verlag, pp. 319-327, 1989.

[7] National Bureau of Standard, "Data encryption standard," FIPS, NBS, 1977.

[8] C. Duraiappan, Y. Zheng, "Enhancing security in GSM," International Computer Symposium, Taiwan, R.O.C, pp. 297-302, 1994.

[9] S. R. Dusse and B.S. Kaliski, Jr., "A cryptographic library for the Motrola DSP56000, Advances in Cryptology: Proc. of EuroCrypt'90, LNCS 473, pp. 230-244, Springer-Verlag, 1991.

[10] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithm," IEEE Trans. Inform. Theory, Vol. 31, pp. 469-472, 1985.

[11] Chun-I Fan and Chin-Laung Lei, "Efficient blind signature scheme based on quadratic residues," Electronic Letters, Vol.32, No. 9, pp. 811-813, 1996.

[12] N. Ferguson, 'Single term off-line coins,' Advances in Cryptology: Proc. of EuroCrypt'93, LNCS 765, pp. 318-328, Springer-Verlag, 1993.

[13] L. Harn and T. Kiesler, "Improved Rabin's scheme with high efficiency," Electronic Letters, Vol.25, No. 11, pp. 726-728, 1989.

[14] Thomas Haug, "Overview of GSM: philosophy and results," Int. J. of Wireless Information Networks, Vol. 1, pp. 7-16, 1994.

[15] Hung-Yu Lin, Lein Harn, "Authentication in wireless communications," IEEE Global Com., pp. 550-554, 1993.

[16] W. Juang and C. Lei, "A collision free secret ballot protocol for computerized general elections," Computers & Security Vol. 15, No. 4, pp. 339-348, 1996.

[17] W. Juang and C. Lei, "A secure and practical electronic voting scheme for real world environments," to appear in IEICE Trans. on Fundamentals (A preliminary version was presented at Proc. 6th National Conf. on Informa. Security, Taiwan, pp. 153-160, 1996).

[18] W. Juang and C. Lei, "Blind threshold signatures based on discrete logarithm," to appear in Asian Computing Science Conference, Sigapore, 1996.

[19] S. Pohlig, M. E. Hellman, "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance," IEEE Trans. on Inform. Theory, Vol. IT-24, pp. 106-110, 1978.

[20] M. O. Rabin, " Digitalized signatures and public key functions as intractable as factorization," MIT Lab. Computer Sci., TR 212, Jan. 1979.

[21] R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," Comm. ACM, No. 21, pp. 120-126, 1978.

[22] K. Sako, "Electronic voting scheme allowing open objection to the tally," IEICE Trans. fundamentals, Vol.E77-A, No.1, pp. 24-30. 1994.

[23] W. Stallings, "Network and internetwork security," Prentice hall international, 1995.

[24] H. C. Williams, "A Modification of RSA public-key encryption," IEEE Trans. Inform. Theory, Vol. IT-26. No. 6, Nov. 1980.