

Untraceable Communication Protocols with Different Types of Anonymity between Sender and Receiver

Masahiro MAMBO

Hirotsugu KINOSHITA

Shigeo TSUJII

Department of Electrical and Electronic Engineering
Tokyo Institute of Technology
2-12-1 O-okayama Meguro-ku, Tokyo, 152, Japan

Addressee : Masahiro Mambo
Present affiliation : School of Information Science
Japan Advanced Institute of Science and Technology
Address : 15 Asahidai Tatsunokuchi Nomi, Ishikawa 923-12 JAPAN
Tel. : +81-761-51-1287
Fax. : +81-761-51-1338
Email : mambo@jaist.ac.jp
Written in August 1992
Printed on September 1, 1994

Untraceable Communication Protocols with Different Types of Anonymity between Sender and Receiver

ABSTRACT Research on information security so far has mostly focused on the protection of the contents of messages, but in many cases it is equally important to protect the identities of senders and receivers and also the time of message transmission from eavesdroppers. In this paper, we propose communication protocols which protect both the contents of messages and the sender and receiver identities. We classify the protocols into four types according to one of the communicator's possibility to identify the other communicator. Following this classification, we study the security, the relationship and the implementability of these protocols.

1 Introduction

As communication networks extend and become more sophisticated, information has come under the threat of such crime as manipulation of data and wire-tapping to obtain useful information. On the other hand, concern about privacy has been increasing. For instance, portable telephone and mobile radio have expanded very rapidly in industrialized countries in the last few years and users have begun to recognize the vulnerability of their conversations. Measures to ensure information security are inevitable in modern communication networks.

Research on information security has focused so far on the protection of the contents of messages. The contents of messages are protected with the use of encryption. At the same time, however, it is important to protect the identities of the sender and the receiver and also the time of message transmission. The user-untraceability problem has been previously studied by Chaum [Chaum81, Chaum88] and Pfitzmann [Pfitzmann84]. In Chaum's mix-method [Chaum81], machines called mix are installed in all nodes between the sender and the receiver. These machines execute a procedure to conceal the correspondence between their inputs and outputs, and this procedure makes it impossible for an eavesdropper to trace the messages. Chaum has also made research on another method of user untraceability, called DC-net [Chaum88]. In this method, each user has two keys, his own key and a key shared with one other user. All users take exclusive-or of the possessed keys and their message. By adding all this results modulo 2, each user gets the sum of all users' messages, because all keys are added twice. If collision does not occur (in this case, the users who don't send their own message use 0 sequences as their message), every user can get a message of one of the users. Another method of obtaining user untraceability given by Pfitzmann uses broadcast communication networks. In Pfitzmann's method, rings are used to make messages untraceable. In these rings, the receiver cannot be identified by adversary even from the trace of the

message if the users make randomly access to the ring and only the legitimate receiver can identify his own address from the messages on the ring.

Generally, an adversary can identify the sender and the receiver

- (i) by finding the address of the sender and the receiver from the message on the network, or
- (ii) by tracing the route of the messages from the sender to the receiver.

On the other viewpoint, the person who identifies the communicator could be the third party (adversary), service center, common carrier (network manager or operator), and the communication partner. (Note: We need to consider the existence of the service center only in the communication for the special service organized by the service center, and the service centers don't exist in other communications.)

In mix-method, the route of the message can be determined by corporation among the operators of all the mixes through which the information passes. Hence, the sender and the receiver can be identified by the operators provided these operators cooperate. Moreover, it has been shown[Pfitzmann89] that, when the RSA cryptosystem is used in the mixes in the straightforward way¹, the correspondence between the mix's input and output can be determined by an eavesdropper, and the sender and receiver identities can then be inferred. Therefore, this method is not totally secure against the attack (ii) mentioned above. Concerning disclosure of identities, the third party cannot identify the communicators, but the operators (common carrier) could. In Pfitzmann's method, the sender and the receiver are not identified in the main session, however, the sender in the main session can be identified by an eavesdropper in the preliminary session, because in the preliminary session, the sender must transmit the receiver's address in the clear. Therefore, this method is not totally secure against the attack (i) mentioned above, and the third party could identify the communicators.

Our method prevents the adversary from succeeding in both attacks (i) and (ii). Such a method is shown as the core protocol in section 2. In this protocol, the communicators are not identified by the third party, service center, and even common carrier. Thus, the communicators are untraceable from anyone with possible exception of the communication partner. The construction of the core protocol is one of the two important issues shown in this paper. Particularly, we put emphasis on how to create the address header of the core protocol, viz. the method by which the sender can generate the receiver's address header² by himself, but in a way, no one except the legal receiver can identify the receiver from the address header. It was first admirably introduced by Pfitzmann[Pfitzmann84] to use rings for the untraceable communication. We adopt rings, too, but the address header creation procedure in the core protocol is different

¹When RSA is skillfully used, mix method looks quite secure.

²This header is a invisible implicit header in Pfitzmann's term [Pfitzmann85].

from that in the Pfitzmann's method. In Pfitzmann's method, the user has to be preliminary sent the address header from the communication partner, and a public address is attached to the message in the preliminary session. Hence, the identity could be disclosed. To the contrary, in our method, the user can create it alone in a secure form, and the communication becomes, therefore, invulnerable.

As for the disclosure of the identity of the communicator by the communication partner, there are four different types of communication, i.e.

1. communication where the sender and the receiver can both identify the communication partner.
2. communication where the sender can identify the receiver, but the converse doesn't occur,
3. communication where the receiver can identify the sender, but the converse doesn't occur, and
4. communication where neither the sender nor the receiver can identify the communication partner.

The core protocol preserves the sender's anonymity, and it corresponds to the above second communication. In most cases, users want to communicate with certain users. Hence, both users should identify the communication partner. This protocol, i.e. the above first protocol, is constructed in section 3 by changing inputs of the core protocol. Moreover, the above third communication is constructed by modifying the core protocol. This modified protocol is the component of the above fourth communication. In this way, even the miraculous communication, in which anonymous users communicate each other, is achieved by extending the core protocol. This communication corresponding to the above fourth communication is the other important issue in this paper.

It was first brilliantly introduced by Chaum[Chaum81] to use the return address for the anonymous communications, the above third communication, But the above fourth communication is not clearly described. Although the return address is used in the modified version of the core protocol and the above fourth protocol is constructed by combining the modified versions, no one has shown how to construct the above fourth protocol, nor it has even been shown that it could be constructed with the combination of the modified protocol.

DC-net is the first excellent method to keep the high confidential sender-untraceability. However, because the communication type is the broadcasting, the sender cannot communicate with a specific receiver. Thus, there is a contrast between our method and DC-net concerning the communication type, i.e. one-to-one or one-to-many. In a practical point of view, this method requires very high data transmission rates when many users are in the group protected by this method.

We discuss the security,³ and the realizability of the communications, as well.

2 The core protocol

We need to consider what kind of properties a communication protocol should have in order to protect the communicators' identities from the adversary's attack, especially from the common carrier's tracing the messages. Common carrier can observe all message transit between nodes and the processes at all nodes, and can trace the messages (attack (ii)) in a one-to-one communication.

Therefore,

- the communication channel of the untraceable communication should be the broadcast-type channel.

At the same time, in order to hinder the attack (i),

- only the legitimate receiver should be able to identify the receiver's address from the message on the ring.

In this paper, the communication channels are all types of the broadcast channels where all users or a large number of users receive identical messages. For example, an Ethernet, a token ring system, mobile communications such as portable telephones and automobile telephones using wireless radio, and satellite communications. To implement our method in OSI layer, we consider LAN networks such that ring and Ethernet are connected with the public network. In such networks, user anonymity within each LAN is preserved by embedding the address headers of our protocols in transport layer. We assume that there is no cooperation of the neighbors of the sender in a ring and Ethernet⁴, and no one observes the power and the direction of the radio wave in mobile communications.

Pfitzmann's method is admirable, because of its foresight in realizing that the rings are suitable for untraceable communications, and that the rings will be used here and there in the modern sophisticated communication networks. In Pfitzmann's method, an address header is affixed to each message sent over the ring. This address header is previously sent from the communication partner, and only the legitimate receiver can identify it. As mentioned in introduction, in the preliminary session, the sender must transmit the receiver's address in the clear. Therefore, the anonymity of the sender in the main session is betrayed. In our method, the sender generates the receiver's address

³When we consider security of each communication, we concern on untraceability of the sender and the receiver.

⁴One measure against sending detection by the neighbors of the sender is to send a dummy message in every interval.

header by himself, and he can start communication without the preliminary session. Thus, anonymity of the communicators is surely guaranteed in all sessions.

○ **Core protocol**

The following is the core protocol between a sender X and a receiver Y . This protocol has inputs \mathcal{R}, \mathcal{I} , and \mathcal{M} , where \mathcal{R} is a variable of a random number, \mathcal{I} is a variable of the receiver's name, \mathcal{M} is a variable of the message, \mathcal{AH} is a variable of the address header, \mathcal{SAH} is a variable of the second part of \mathcal{AH} , D_y is the decryption function of Y . R is a random number generated by the sender and m is a message. The precise description of the core protocol is shown in section 5.

We write the message in the form,

$$(\text{Packet}) = ((\text{AH}), (\text{MP})),$$

where AH and MP stand for the address header and the message part, respectively.

Step 1 X executes the packet creation program, by assigning $\mathcal{R} \leftarrow R$, $\mathcal{I} \leftarrow Y$, and $\mathcal{M} \leftarrow m$. Then, X gets $((\text{AH}), (\text{MP}))$.

Step 2 X sends $((\text{AH}), (\text{MP}))$ to many users including Y .

Step 3 Y executes the return address header checking program, by assigning $\mathcal{D} \leftarrow D_y$, and $\mathcal{AH} \leftarrow (\text{AH})$. Then, Y gets $\mathcal{D}(\mathcal{SAH})$ and CHECKER .

If $\text{CHECKER} = \text{Yes}$, go to step 4.

If $\text{CHECKER} = \text{No}$, X throws the message away or give it to the next user, and go to step 5.

Step 4 Y executes the message processing program, by assigning $\mathcal{D} \leftarrow D_y$ and $\mathcal{M} \leftarrow (\text{MP})$. Then, Y gets m .

Step 5 The end of the protocol.

We continue the discussion further assuming that the core protocol is constructed and secure.

3 Non-anonymous UnTraceable Communication

Hereafter, we use the term UTC (UnTraceable Communication) when no user with possible exception of the communicators can identify the communicators, and we use the term NUTC (Non-anonymous UnTraceable Communication) if additionally the communicators can identify their communication partners.

Digital signature scheme is used to convince the receiver of the sender's identity.

Protocol for NUTC

The following is the protocol for NUTC between a sender A and a receiver B .

Step 1 By assigning $\mathcal{R} \leftarrow R\|ID_A$, $\mathcal{I} \leftarrow B$, and $\mathcal{M} \leftarrow D_A(m\|ID_A)$ ⁵, A and B execute the core protocol. Then, B gets $R\|ID_A$ and $D_A(m\|ID_A)$.

Step 2 B certifies the digital signature of A . $E_A(D_A(m\|ID_A)) = m\|ID_A$. Then, B gets m .

4 Anonymous Communication

We use the term AC(Anonymous Communication), when both or one of the communicators cannot identify their partners in a communication. AC is categorized into two types, i.e. AAC and SAC.

4.1 Asymmetric Anonymous Communication

We use the term AAC(Asymmetric Anonymous Communication) when one of the communicators communicates anonymously. Then, AAC is classified by C-AAC and O-AAC.

◇ C-AAC

When users anonymously receive information from a database, we call the communication C-AAC (AAC on which an anonymous user Collects information).

The trick to realize C-AAC is that the user sends a return address header to the database before the database returns required information to the anonymous user.

Before showing protocol for C-AAC, we introduce two modes into the core protocol.

Modified core protocol

The packet creation program and the message processing program are modified so that these programs have two modes (See section 5). The following is the modified core protocol between a sender X and a receiver Y , which has a mode switch. If mode is not specified, the mode switch is set 0. $\mathcal{KE}\mathcal{Y}$ and $\mathcal{KE}\mathcal{Y}'$ are variables of encryption and decryption keys of \mathcal{M} , respectively. In mode 0, the encryption function of \mathcal{M} is publicly known.

Step 1 X inputs $\mathcal{MODE} \leftarrow mode(mode \in \{0, 1\})$.

Step 2 If $\mathcal{MODE} = 0$, by assigning $\mathcal{R} \leftarrow R$, $\mathcal{I} \leftarrow Y$, and $\mathcal{M} \leftarrow m$, X and Y execute the core protocol in section 2 (mode 0). Then, Y gets m . Go to step 4.

Step 3 If $\mathcal{MODE} = 1$, go to step 3-1. Otherwise go to step 4.

Step 3-1 X executes the modified packet program, by assigning $\mathcal{AH} \leftarrow AH$, $\mathcal{KE}\mathcal{Y} \leftarrow K_e$, and $\mathcal{M} \leftarrow m$.

⁵ $A\|B$ means the concatenation of A and B

Step 3-2 X and Y execute step 2 and 3 of the core program.

Step 3-3 Y executes the modified message processing program, by assigning $\mathcal{KEY}' \leftarrow K_d$. Then, Y gets m .

Step 4 The end of the protocol.

Protocol for C-AAC

We suppose that A and B communicate each other and A can identify B but B cannot identify A . K_e and K_d are encryption and decryption keys. m_{ij} ($i, j \in \{A, B\}, i \neq j$) is a message from i to j . R and R' are random numbers generated by the sender. $AH(i, R)$ is an address header to the user i generated from a random number R .

- Sending from an anonymous user A to B

Step 1 A inputs $MODE \leftarrow 0$.

Step 2 By assigning $\mathcal{R} \leftarrow R$, $\mathcal{I} \leftarrow B$, and $\mathcal{M} \leftarrow K_e \| m_{AB} \| AH(A, R')$, A and B execute the modified core protocol. Then, Y gets the message m_{AB} , the return address header $AH(A, R')$, and the encryption key of messages K_e from the outputs of the core protocol, R and $K_e \| m_{AB} \| AH(A, R')$.

Step 3 The end of the sending.

- Returning from B to the anonymous user A

Step 1 B inputs $MODE \leftarrow 1$.

Step 2 By assigning $\mathcal{AH} \leftarrow AH(A, R')$, $\mathcal{KEY}' \leftarrow K_e$, $\mathcal{M} \leftarrow m_{BA}$, and $\mathcal{KEY}' \leftarrow K_d$ B and A execute the modified core protocol. Then, A gets the outputs of the core protocol, m_{AB} .

◇ O-AAC

When users can anonymously offer information to the center, we call the communication O-AAC (AAC on which an anonymous user Offers information). We denote by O-AAC(per) (O-AAC(permitted repetition)) a communication on which the same person is permitted to offer information repeatedly such as in the transfer of accounts etc, in order to indicate the permitted repetition.

○ O-AAC(per)

The core protocol in mode 0 is itself the protocol of O-AAC(per).

4.2 Symmetric Anonymous Communication

We use the term SAC(Symmetric Anonymous Communication) when both the sender and the receiver cannot identify the communication partner.

We consider two cases in which the users do or do not request mediation, and we call these communications SAC(med) (SAC on which anonymous users ask for mediation) and SAC(ind) (SAC on which anonymous users are independent of mediation), respectively. A practical example of the former is marriage counseling, and of the latter is dialing a message.

◊ **SAC(med)**

Anonymous user obtains his partner's information through a service center, and he executes a public-key distribution protocol with the partner to generate a common key. With this key, both users communicate each other.

Protocol of SAC(med)

A user A and the other user B communicate through a service center S . p is a large prime number. α is a primitive root of p . $\beta_k (k \in \{A, B\})$ is a secret key of k . R_1, R_2, R_5, R_7 are random numbers generated by A . R_3, R_4, R_6, R_8 are random numbers generated by B . $m_{ln} (l, n \in \{S, A, B\})$ is a message from l to n . K_{eln} and K_{dln} are a pair of encryption and decryption keys. p, α are public information and $\beta_k, R_1 \cdots R_8, m_{ln}$ are private information.

• Preparation

Step 1 Registration phase

A user A and the service center S execute the protocol of C-AAC.

Step 1-1 A inputs $MODE \leftarrow 0$.

Step 1-2 By assigning $\mathcal{R} \leftarrow R_1$, $\mathcal{A} \leftarrow S$, and $\mathcal{M} \leftarrow K_{eSA} \| m_{AS} \| AH(A, R_2)$, the sender A and the receiver S execute the modified core protocol. Then, S gets m_{AS} , $AH(A, R_2)$, and K_{eSA} .

Similarly, a user B and S execute the protocol of C-AAC.

Step 1-1 B inputs $MODE \leftarrow 0$.

Step 1-2 By assigning $\mathcal{R} \leftarrow R_3$, $\mathcal{A} \leftarrow S$, and $\mathcal{M} \leftarrow K_{eSB} \| m_{BS} \| AH(B, R_4)$, the sender B and the receiver S execute the modified core protocol. Then, S gets m_{BS} , $AH(B, R_4)$, and K_{eSB} .

A and B write the requirements for their communication partner in m_{AS} and m_{BS} , respectively.

Step 2 S chooses a user $B (A \neq B)$ who is suitable for A 's requirements, and the protocol of C-AAC is executed by S and user A , and S and user B . S sends B 's return address header and message to A and sends A 's return address header and message to B .

Step 2-1 S inputs $MODE \leftarrow 1$.

Step 2-2 By assigning $\mathcal{AH} \leftarrow AH(A, R_2)$, $\mathcal{KEY} \leftarrow K_{eSA}$,
 $\mathcal{M} \leftarrow K_{eSB} \| m_{BS} \| AH(B, R_4)$, and $\mathcal{KEY}' \leftarrow K_{dSA}$,
the sender S and the receiver A execute the modified core protocol.
Then, A gets m_{BS} , $AH(B, R_4)$, and K_{eSB} .

Similarly, S and B execute the protocol of C-AAC.

Step 2-1 S inputs $MODE \leftarrow 1$.

Step 2-2 By assigning $\mathcal{AH} \leftarrow AH(B, R_4)$, $\mathcal{KEY} \leftarrow K_{eSB}$,
 $\mathcal{M} \leftarrow K_{eSA} \| m_{AS} \| AH(A, R_2)$, and $\mathcal{KEY}' \leftarrow K_{dSB}$,
the sender S and the receiver B execute the modified core protocol.
Then, B gets m_{AS} , $AH(A, R_2)$, and K_{eSA} .

Step 3 A and B generate a common key with the use of the public key distribution system.
In this step, the returning protocol of C-AAC is used and new return address headers are exchanged. Without loss of generality, we adopt the Diffie-Hellman public key distribution system [DH76]. Before the beginning of SAC(med), by using the secret key β_A (β_B), A (B) generates γ_A (γ_B), where $\gamma_A = \alpha^{\beta_A} \pmod{p}$ ($\gamma_B = \alpha^{\beta_B} \pmod{p}$).

A and B execute the protocol of C-AAC.

Step 3-1 A inputs $MODE \leftarrow 1$.

Step 3-2 By assigning $\mathcal{AH} \leftarrow AH(B, R_4)$, $\mathcal{KEY} \leftarrow K_{eSB}$,
 $\mathcal{M} \leftarrow \gamma_A \| AH(A, R_5)$, and $\mathcal{KEY}' \leftarrow K_{dSB}$,
the sender A and the receiver B execute the modified core protocol.
Then, B gets γ_A and $AH(A, R_5)$.

Similarly, user B and A execute the protocol of C-AAC.

Step 3-1 B inputs $MODE \leftarrow 1$.

Step 3-2 By assigning $\mathcal{AH} \leftarrow AH(A, R_2)$, $\mathcal{KEY} \leftarrow K_{eSA}$,
 $\mathcal{M} \leftarrow \gamma_B \| AH(B, R_6)$, and $\mathcal{KEY}' \leftarrow K_{dSA}$,
the sender B and the receiver A execute the modified core protocol.
Then, A gets γ_B and $AH(B, R_6)$.

A (B) calculates the common key K_{AB} , which satisfies

$$K_{AB} = \gamma_B^{\beta_A} \pmod{p} = \alpha^{\beta_A \times \beta_B} \pmod{p} \quad (K_{AB} = \gamma_A^{\beta_B} \pmod{p} = \alpha^{\beta_A \times \beta_B} \pmod{p}).$$

- Main procedure

Table 1: Relationship among AC, the core protocol and the modified core protocol

Comm. protocols of AC		used protocol	
AC	AAC	C-AAC	modified core protocol
		O-AAC(per)	core protocol
	SAC	SAC(med)	C-AAC
		SAC(ind)	C-AAC

Step 4 Afterwards, A and B do C-AAC using the common key K . The return address headers are renewed in every round.

A and B execute the protocol of C-AAC.

Step 4-1 A inputs $MODE \leftarrow 1$.

Step 4-2 By assigning $\mathcal{AH} \leftarrow AH(B, R_6)$, $\mathcal{KEY} \leftarrow K$,
 $\mathcal{M} \leftarrow m_{AB} \| AH(A, R_7)$, and $\mathcal{KEY}' \leftarrow K$,
the sender A and the receiver B execute the modified core protocol.
Then, B gets m_{AB} and $AH(A, R_7)$.

Similarly, user B and A execute the protocol of C-AAC.

Step 4-1 B inputs $MODE \leftarrow 1$.

Step 4-2 By assigning $\mathcal{AH} \leftarrow AH(A, R_5)$, $\mathcal{KEY} \leftarrow K$,
 $\mathcal{M} \leftarrow m_{BA} \| AH(B, R_8)$, and $\mathcal{KEY}' \leftarrow K$,
the sender B and the receiver A execute the modified core protocol.
Then, A gets m_{BA} and $AH(B, R_8)$.

Step 5 The end of the protocol

The security of SAC(med)

S cannot identify the user, since the users send their information to the service center S by C-AAC. The users cannot identify the communication partner, neither, because the communication between the users is also C-AAC. Moreover, because of the use of the public key distribution system, S cannot know the common key generated by the users. An adversary can get information such that the message from S to A in step 2 and the message from B to A in step 3 (similarly, the message from S to B in step 2 and the message from A to B in step 3) in preparation stage are sent to the same user. But this information is not useful to identify corresponding messages in the main procedure, nor communicators using these pair of messages.

◇ SAC(ind)

Table 2: Identification of the communicators

		user		user (Database)	t.p.	s.c.	c.c.
NUTC		○	⇔	○	×	–	×
C-AAC		○	⇐	×	×	–	×
O-AAC	per	○	⇒	×	×	–	×
SAC	med	×	⇔	×	×	×	×
	ind					–	

- ... able to identify sender and receiver
- ×
- ... included in t.p.
- ⇐ ⇒ ⇔ ... flow of important information
- t.p. ... the third party
- s.c. ... service center
- c.c. ... common carrier

If a user can select a suitable communication partner, the service center is no longer necessary. In SAC(ind), the role of the service center is replaced by broadcast communication. The protocol of SAC(ind) is basically the same as that of SAC(med) with some minor changes. We omit the detail.

The relationship between each communication and identification of communicators are shown in table 1 and table 2, respectively.

5 Components of the core protocol

The following is the precise description of the core protocol. The core protocol consists of the packet creation program, sending part, the address header checking program, and the message processing program.

h is a one-way hash function(a data compression function, for which it is difficult to inverse the function) [Merk89], $E_{\mathcal{I}}$ is an encryption function corresponding to the user \mathcal{I} . $E_{\mathcal{K}\mathcal{E}\mathcal{Y}}$ and $D_{\mathcal{K}\mathcal{E}\mathcal{Y}}$ are encryption and decryption functions corresponding to the key $\mathcal{K}\mathcal{E}\mathcal{Y}$.

- The packet creation program

Step 1 Input \mathcal{R} , \mathcal{I} , and \mathcal{M} .

Step 2 Calculate $h(\mathcal{R})$, $E_{\mathcal{I}}(\mathcal{R})$, and $E_{\mathcal{I}}(\mathcal{M})$

Step 3 Output $(AH(\mathcal{I}, \mathcal{R}), E_{\mathcal{I}}(\mathcal{M}))$, where $AH(\mathcal{I}, \mathcal{R}) = h(\mathcal{R}), E_{\mathcal{I}}(\mathcal{R})$

Step 4 The end of the program.

- The modified packet creation program

Step 1 If $MODE = 0$, execute the above packet creation program. Go to step 2.

If $MODE = 1$, go to step 1-1.

Otherwise, go to step 2.

Step 1-1 Input $\mathcal{AH}, \mathcal{KEY}$, and \mathcal{M} .

Step 1-2 Calculate $E_{\mathcal{KEY}}(\mathcal{M})$. $\mathcal{M}' \leftarrow E_{\mathcal{KEY}}(\mathcal{M})$.

Step 1-3 Output $(\mathcal{AH}, \mathcal{M}')$.

Step 2 The end of the program.

- The address header checking program

We denote by \mathcal{FAH} and \mathcal{SAH} the first and the second part of the address header, respectively.

Step 1 Input \mathcal{D} and \mathcal{AH} .

Step 2 \mathcal{SAH} is decrypted by \mathcal{D} ($\mathcal{T} \equiv \mathcal{D}(\mathcal{SAH})$).

Step 2-1 If $h(\mathcal{T}) = \mathcal{FAH}$, the message is for the user i , and $CHECKER \leftarrow Yes$.

Step 2-2 If $h(\mathcal{T}) \neq \mathcal{FAH}$, the message is not for the user i , and $CHECKER \leftarrow No$.

Step 3 Output \mathcal{T} and $CHECKER$.

Step 4 The end of the protocol.

- The message processing program

Step 1 Input \mathcal{D} and \mathcal{M} .

Step 2 Calculate $\mathcal{D}(\mathcal{M})$. $\mathcal{M}' \leftarrow \mathcal{D}(\mathcal{M})$.

Step 3 Output \mathcal{M}' .

Step 4 The end of the protocol.

- The modified message processing program

Step 1 If $MODE = 0$, execute the message processing program. Go to step 2.

If $MODE = 1$, go to step 1-1.

Otherwise, go to step 2.

Step 1-1 Input \mathcal{KEY}' and \mathcal{M} .

Step 1-2 Calculate $D_{\mathcal{KEY}'}(\mathcal{M})$. $\mathcal{M}' \leftarrow D_{\mathcal{KEY}'}(\mathcal{M})$.

Step 1-3 Output \mathcal{M}' .

Step 2 The end of the program.

The packet to the user i forms $((h(R), E_i(R)), E_i(m))$.

(Note : $\mathcal{FAH} = h(R)$ and $\mathcal{SAH} = E_i(R)$ in the message checking program above.)

The security of the core protocol

The function \mathcal{D} satisfying the equation in step 2 of the address header checking program, i.e. $h(\mathcal{D}(E_i(R))) = h(R)$, is only the legitimate receiver's (i 's) decryption function, and this function is the secret information of i . Therefore, a third party cannot identify the receiver. Moreover, the receiver cannot identify the sender because the address header is composed by substituting a random number into functions h and E_i .

There is other way to form the address header. For the further discussion of the address header composition, see the final paper.

6 Considerations

6.1 Transmission rate

In our method, the processing of the address header restricts the transmission rate⁶. Thus, the header part should be processed independently from the message part by pipeline process.

The transmission rate, R , is obtained from the following equation.

$$R = \frac{H+M}{H} \times \min(V_D, V_h) \text{ (bit/s)}, ^7$$

where M is the length of the message part (bit/s), H is the length of the address header part (bit/s), V_D is the decryption speed of the address header part (bit/s), V_h is the calculation speed of one-way hash function (bit/s).

From data such as

$$\min(V_D, V_h) = V_D \simeq \begin{cases} 500Kbit/s & RSA[THYA88] \\ 93Mbit/s & DES[BS91], \end{cases}$$

the transmission rates are 5.5 M bit/s with RSA and 1.02 G bit/s with DES in the case that $H/M = 1/10$, and 10 M bit/s with RSA and 1.86 G bit/s with DES in the case that $H/M = 1/19$. This means that if RSA cryptosystem is adopted and the address header part is 1/19 of the message part, our method can be used on 10 M bit/s LANs.

⁶For the calculation of the transmission rate, not only the message but also the header part is taken into consideration.

⁷The derivation of this equation is shown in the full paper.

6.2 Security of proposed protocols

All proposed communication protocols are secure because all message transfer are based on the core protocol and the core protocol itself is secure as shown in section 5.

7 Conclusion

In this paper, first, we constructed the core protocol where it is impossible for the adversary to identify neither the sender nor the receiver. Since not only a third party but the common carrier cannot identify the communicators in this protocol, the security against identifying the communicators has increased. Secondly, we have widely studied the anonymity between sender and receiver in this untraceable communication. Studied communications are called NUTC, C-AAC, O-AAC, and SAC. SAC is a remarkable communication such that both of the communicators cannot identify the communication partner. Through our attempt, it becomes possible to realize flexible and diverse communication services.

The proposed method is realizable if the length of the address header part H is set to satisfy the equation $R = (H + M)/H \times \min(V_D, V_h)$, where R is the transmission rate and M is the length of the message part. It is necessary that $H/M = 1/19$, in order to implement the proposed method on the 10M bit-per-second LAN with RSA cryptosystem.

References

- [BS91] A. G. Broscius and J. M. Smith : “*Exploiting Parallelism in Hardware Implementation of the DES* ” Proceedings of Crypto '91, pp. 367-376 (August 1991).
- [Chaum81] D. Chaum : “*Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms,* ” Commun. ACM. ,24, 2, pp.84-88 (Feb. 1981).
- [Chaum88] D. Chaum : “*The Dining Cryptographers Problem : Unconditional Sender and Recipient Untraceability,*” Journal of Cryptology, 1 , 1, pp.65-75 (1988).
- [DH76] W. Diffie and M. Hellman : “*New directions in cryptography,*” IEEE Trans. Inf. Theory, IT-22 , 6, pp.644-645 (Nov. 1976).
- [Merk89] R.C. Merkle : “*One-way hash function and DES,*” Proceedings of Crypto '89, (August 1989).

- [Pfitzmann84] A Pfitzmann : “*A switched/broadcast ISDN to decrease user observability,*” International Zurich Seminar on Digital Communications, Applications of Source Coding, Channel Coding and Secrecy Coding, Proceedings IEEE Catalog, no.84 ch 1988-4, pp.183-190 (March 1984).
- [Pfitzmann85] A Pfitzmann : “*How to implement ISDNs without user observability - some remarks,*” University Karlsruhe inside report 14/85 (1985).
- [Pfitzmann89] B Pfitzmann and A Pfitzmann : “*How to break the Direct RSA Implementation of MIXes,*” Eurocrypt ‘89 (April. 1989).
- [THYA88] N. Torii, T. Hasebe, M. Yamato, and C. Akiyama, “*Hardware for the RSA encryption scheme,*” 1988 Coding Theory and Security Symposium materials.