PAPER

Development and Evaluation of FMC Authentication Technology with Cellular Phones

Katsuyuki UMEZAWA[†a)] *and* **Satoru TEZUKA**[††b)], *Members*

SUMMARY The cellular phone ownership rate continues to increase, meaning one person may now own two or more. Meanwhile, a lot of terminals that receive cellular phone services through a mass broadband communication network are being commercialized. When service is received through the cellular phone, the mobile network operator authenticates the subscriber. However, service providers other than the mobile network operators provide communication services and other services through fixed networks. In this situation, if we can use the subscriber authentication that the mobile network operator provide for the fixed network service, fixed mobile convergence (FMC) will be achieved and mobile network operators will be able to better prevent unauthorized users from using their services. In addition, services will become more convenient because users will be authenticated by swiping one cellular phone when switching from using a fixed terminal to another fixed terminal. A mechanism has been developed that allows mobile network operator to authenticate their subscribers' account when using a terminal connected to a fixed network. In addition, services can be easily switched between fixed terminals by using the proposed mechanism. Moreover, a system is constructed on the basis of the proposed mechanism, and its performance is evaluated.

key words: *mobile phone, authentication, federation, FMC, 3GPP, bluetooth*

1. Introduction

The cellular phone ownership rate continues to increase, meaning one person may now own two or more. Meanwhile, a lot of terminals that receive cellular phone services through a mass broadband communication network are being commercialized. When a service is received through a cellular phone, the mobile network operator authenticates the subscriber. Also, service providers other than the mobile network operators provide communication services and other services through fixed networks.

In this situation, if we can use the subscriber authentication that the mobile network operator provides for the fixed network services, fixed mobile convergence (FMC) will be achieved and mobile network operators will be able to better prevent unauthorized users from using their services. In addition, services will become more convenient because users will be authenticated by swiping one cellular phone when switching from using a fixed terminal to another fixed terminal.

Manuscript received November 26, 2010.
Manuscript revised May 16, 2011.

[†]The author is with Hitachi Ltd., Systems Development Laboratory, Yokohama-shi, 244-0817 Japan.
[††]The author is with School of Computer Science, Tokyo University of Technology, Hachioji-shi, 192-0982 Japan.
a) E-mail: katsuyuki.umezawa.ue@hitachi.com
b) E-mail: tezuka@cs.teu.ac.jp

DOI: 10.1587/transcom.E94.B.3009

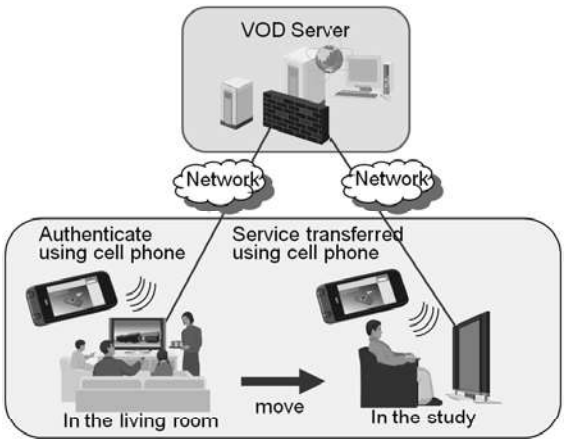

Fig. 1 Concept of proposed system.

Figure 1 shows the concept of our system and outlines how it is used. First the user swipes his/her cellular phone (mobile terminal) across a television in the living room. The video delivery server authenticates the user via the network server when the user swipes the mobile terminal, and the video is delivered. Afterwards, the user moves to the study. It then becomes possible to receive the same video by swiping the cellular phone (mobile terminal) across the television in the study room.

In this paper, we propose a mechanism that enables mobile network operator to authenticate their subscribers' account when using a terminal connected to a fixed network. In addition, services can be easily switched between fixed terminals by using the proposed mechanism. Moreover, a system is constructed on the basis of the proposed mechanism, and its performance is evaluated.

Section 2 describes the generic bootstrapping architecture (GBA), a handover mechanism using near field communication (NFC), a remote-access mechanism using a mobile terminal (mobile terminal and cellular phone refer to the same thing). Section 3 describes an authentication federation mechanism between terminals. Section 4 describes the proposed mechanism. Specifically, we detail the way in which we separate the user-side terminal into mobile and fixed terminals. We also detail the protocol of our proposed mechanism. Section 5 considers the efficiency improvement when we switch a service from one fixed terminal to another and evaluates the system that we developed on the basis of our proposal in Sect. 6. Section 7 considers the safety and convenience of our proposal. Section 8 is the conclusion.

2. Background

This section describes the generic bootstrapping architecture (GBA) specified by the 3rd Generation Partnership Project (3GPP) [1], [2], a handover mechanism using Near Field Communication (NFC handover) [3], a remote access mechanism using the mobile terminal, and an authentication federation mechanism between terminals.

2.1 Outline of GBA Protocol

GBA is an authentication method provided by 3GPP [1], [2] by which a mobile network operator authenticates a subscriber. Figure 2 shows the basic system configuration of the GBA method.

BSF-S refers to the bootstrapping server function server. The BSF-S authenticates the bootstrap and generates the service key. NAF-S refers to the network application function server. NAF-S provides services for mobile terminals. BSF-C and NAF-C refer to the client pairs with BSF-S and NAF-S, respectively.

2.1.1 Bootstrap Authentication

Figure 3 shows and Table 1 explains the processing flow of the bootstrap authentication in the GBA method.

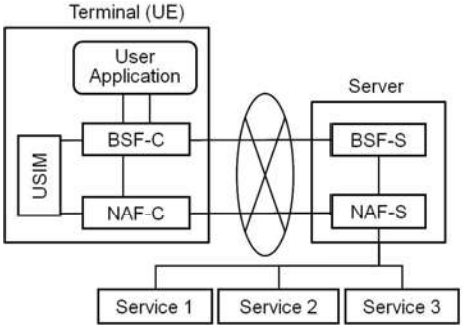

Fig. 2 Basic system configuration of GBA system.

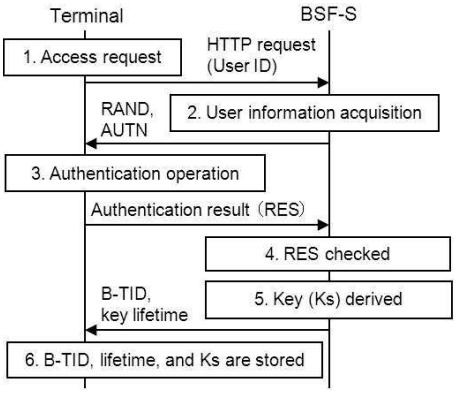

Fig. 3 Basic protocol flow of bootstrap authentication.

Figure 4 shows the details of the authentication operation in the bootstrap authentication flow (Step 3 in Fig. 3).

In the authentication operation, XMAC, RES, CK, and IK are derived using RAND and AUTN received from BSF-S and shared key K, which is previously shared by BSF-S and the terminal. First, the XMAC in the operation result is checked to see whether it is the same as the MAC included in AUTN. If XMAC differs from MAC, the operation fails. Then CK and IK are concatenated and stored in a confidential area. RES is sent to BSF-S. After this operation succeeds, RAND, B-TID, lifetime, and Ks are stored in a terminal. The functions f1 to f5 are described in detail in 3GPP TS 35.206 [4].

2.1.2 Service Authentication

Service authentication is needed to receive the service after the abovementioned bootstrap processing has finished authenticating the user. The service key is derived in the service authentication process. The processing flow of the service authentication is shown in Fig. 5 and explained in Table 2.

The derivation method of Ks_NAF is as follows.

$$Ks\_NAF = \texttt{HMAC-SHA-256}(Key, S) \quad (1)$$

Here, Key is Ks=(CK||IK), and S is S = FC || P0 || L0 || P1 || L1 || P2 || L2 || P3 || L3. When, FC, P0, L0, ..., L3 is as follows.

- FC = 0x01

Table 1 Explanation of Fig. 3.

No.	Explanation
1	The terminal sends the user ID to the BSF-S.
2	The BSF-S acquires subscriber information and then transmits random numbers (RAND) and information necessary for the authentication operation (AUTN) to the terminal.
3	The terminal checks the AUTN and calculates CK, IK (Ks = CK||IK), and RES. The terminal sends the authentication operation result (RES) to the BSF-S.
4	The BSF-S checks the RES.
5	The BSF-S derives a key (Ks). The BSF-S sends B-TID and the lifetime of Ks to the terminal.
6	As a result of the bootstrap authentication, the terminal stores B-TID, lifetime, and Ks until they expire or are updated.

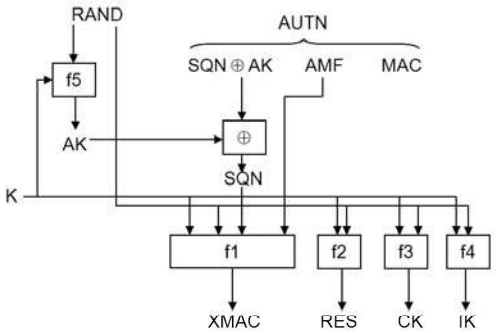

Fig. 4 Details of authentication operation.

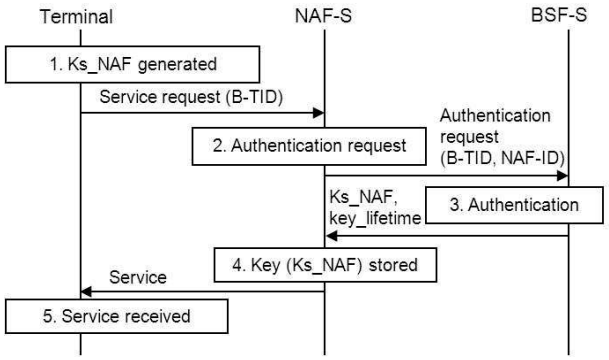

Fig. 5 Basic protocol flow of service authentication.

Table 2 Explanation of Fig. 5.

No.	Explanation
1	The terminal generates the service key (Ks_NAF) by using Ks. The terminal requests service from the NAF-S. At that time, B-TID is notified.
2	The NAF-S sends the authentication request to the BSF-S, and, at the same time, B-TID and NAF-ID are notified.
3	The BSF-S derives Ks_NAF and sends it to NAF-S.
4	The NAF-S stores Ks_NAF. NAF-S then provides the service to the terminal by using Ks_NAF.
5	The terminal receives the service.

- P0 = "gba-me" (i.e. 0x67 0x62 0x61 0x2d 0x6d 0x65)
- L0 = Length of P0 (i.e., 0x00 0x06)
- P1 = RAND
- L1 = Length of RAND (i.e. 0x00 0x10),
- P2 = IMPI encoded by UTF-8
- L2 = Length of IMPI (Variable Length) (MAX 65535)
- P3 = NAF_ID encoded by UTF-8
- L3 = Length of NAF_ID (Variable Length) (MAX 65535)

After service authentication, UE stores Ks_NAF and lifetime.

2.2 Handover Mechanism Using Near Field Communication

The handover mechanism using Near Field Communication (NFC Handover) connects one terminal to another as specified by the NFC Forum. In this mechanism, two terminals are paired by NFC communication (ISO 14443 TypeA/TypeB). Then the data communication is done through fast wireless communication such as Bluetooth, Wi-Fi, etc. [3]. Figure 6 outlines NFC Handover.

2.3 Remote Access Mechanism Using Mobile Terminal

We developed a remote access system using a mobile terminal as a security device [5]–[7]. In this system, the mobile terminal and PC terminal communicate with each other via Bluetooth. The combination of a mobile terminal and a PC terminal is fixed because both terminals are regarded as personal devices. This system cannot be applied to public terminals such as a kiosk terminal. Because a kiosk terminal does not belong to any one individual, we need a mechanism to establish a combination dynamically.

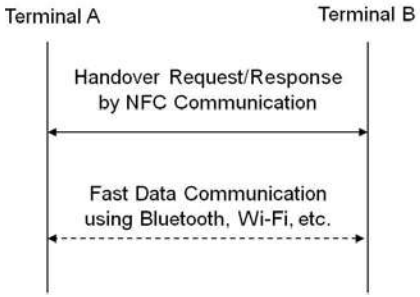

Fig. 6 Outline of NFC Handover.

3. Related Works

This section describes an authentication federation mechanism between terminals.

A lot of federation technologies used for authenticating information on a Web service are known as Single Sign-On (SSO). For instance, OASIS provides a SSO that uses SAML2.0 for Web services between companies [8]. However, this paper describes a federation technology for authenticating information not between the servers but between the terminals at the client side. The following are the federation technologies for authenticating information between terminals at the client side.

However, note that the following related technologies do not target the GBA protocol at which we are aiming.

3.1 Federation Technology for Cookie Information

Umezawa et al. developed technology that makes the mobile terminal cooperate with the PC terminal and simplifies the server authentication by using authenticated information [9], [10]. In this case, the authenticated information used here is cookie information, and the protocol is Hypertext Transfer Protocol (HTTP).

3.2 Federation Technology for ID-WSF Assertion

ID Web Service Framework (ID-WSF) [11], which is the specification for the user to log in to two or more sites with one terminal, is provided for in the Liberty Alliance. Fujii et al. [12] are enhancing ID-WSF so that two or more terminals may federate. Specifically, the authentication information, called an "assertion", is moved between terminals.

3.3 Other Federation Technology

Additionally, Miyakawa et al. are developing crossing device cooperation as an upgrade technology for the IPTV service [13]. This technology shares bookmark information and resume information that achieves continuous watching.

These pieces of information are shared between a mobile terminal and the IPTV terminal not via the local communication between terminals but via the server.

In addition, Nagai is developing federation technology for cellular phone networks [14]. This means enhancing the Digital Living Network Alliance (DLNA) [15] technology, which is the network technology that controls the multimedia contents in order to handle a mobile terminal. This is a coordinating technology for the multimedia contents that the terminal has and is not a coordinating technology for the server to authenticate a user.

4. Proposed System

This section describes the proposed system. The essence of the proposal is to enable the device on the user side to be divided into a mobile terminal and a fixed terminal without changing the mechanism on the server side at all in the GBA protocol for a mobile network operator. As a result, our proposal improves convenience for the user and enables the current impossibility of a mobile network operator authenticating the user in a terminal connected to a fixed network. None of the related works detailed in Chapter 3 proposed making two terminals cooperate without changing the mechanism on the server side in the GBA protocol.

We detail the structure in which we separate the user-side terminal into mobile and fixed terminals in accordance with the concept of 3GPP. We also detail the protocol of our proposed system.

Our proposed system uses the generic bootstrapping architecture (GBA), which is the authentication protocol of the mobile network operator provided by 3GPP. This point is different from those of the related works described in Chapter 3.

4.1 Basic Structure of Proposed System

We divide the concept of the user-side terminal in the GBA system into two components, the mobile terminal and the fixed terminal, as shown in Fig. 7. We propose dividing the user-side terminal of the GBA specification of 3GPP into the mobile terminal and the SIM card (secure device). However, in this proposal, the mobile terminal in Fig. 6 is for the cellular phone terminal with the short distance wireless communication function, and the fixed terminal is for PCs, public displays, etc.

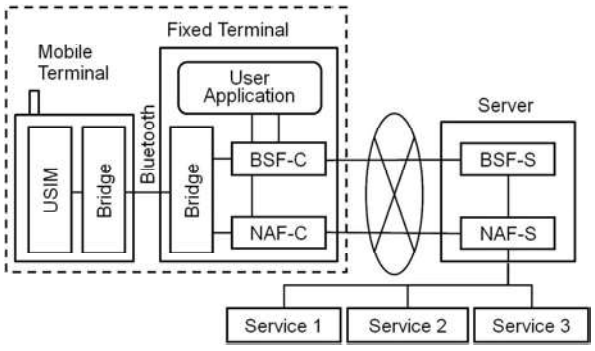

Fig. 7 Basic structure of proposed system.

The proposed system differs from the conventional system as follows.

- We have divided the conventional concept of the terminal, defined by the 3GPP standard, into a fixed and mobile terminal. Specifically, we use the User Application, the BSF-C, the NAF-C implemented in the PC terminal, and the USIM inserted in the mobile terminal.

The above-mentioned composition is achieved by implementing the following new feature.

- The bridge function is implemented at two divided terminals to federate both terminals by using Bluetooth.

Adopting this composition makes it possible to use an individual user's mobile terminal as an authentication device. This will, for instance, make age requirements for certain services easier to enforce.

4.2 Local Communication between Mobile Terminal and Fixed Terminal

In this proposal, we use the handover technology to enable the data communication to be done by swiping the mobile terminal across the fixed terminal.

Specifically, middleware (Bridge) in the fixed terminal makes the reading signal from a contactless IC reader that triggers an operation, and Bluetooth is paired with the mobile terminal. Afterwards, the GBA protocol described later is executed. The following describes each function of Bridge in the proposed system implemented here.

4.2.1 Function on Bridge in Fixed Terminal

Both functions of Bridge on the fixed-terminal side are as follows.

- Reading function of contactless IC chip ID: This operates when a mobile terminal is swiped over a contactless IC reader and acquires a contactless IC chip ID.
- Pairing function of Bluetooth communication: This pairs the Bluetooth equipment of a PC and mobile terminal automatically. In this pairing, a 16-byte value generated from the contactless IC chip ID is used as a PIN code[†].

4.2.2 Function on Bridge in Mobile Terminal

Both functions of Bridge on the mobile-terminal side are as follows.

[†]Ideally, a Bluetooth address that has an ID is acquired by using the above-mentioned reading function of the contactless IC chip ID and the connection destination is decided. However, this was not implemented this time.

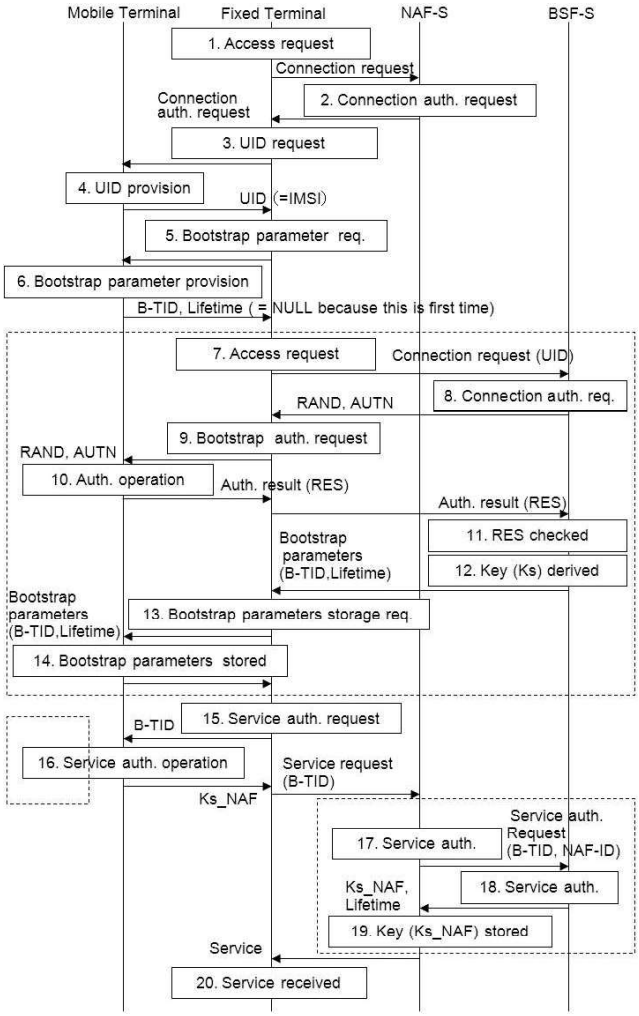

Fig. 8 Protocol of proposed system.

- Acquisition function for contactless IC chip ID: This acquires ID of contactless IC chips built into the terminal[†].
- Pairing function of Bluetooth communication: This pairs the Bluetooth equipment of a PC and mobile terminal automatically. In the pairing, a 16-byte value generated from the ID of a contactless IC chip is used as a PIN code[††].

4.3 Protocol of Proposed System

Figure 8 shows the processing flow of the system. In this processing flow chart, the fixed terminal accesses a NAF-S and is authenticated by BSF-S, and then the NAF-S provides a service for the fixed terminal. At this time, the mobile terminal is used by the back end in the fixed terminal. Table 3 explains the flow.

Table 3 Explanation of Fig. 8.

No.	Explanation
1	First of all, the fixed terminal sends a service request to the NAF-S (same as step 1 in Fig. 3).
2	The NAF-S that receives the service request notifies the fixed terminal of a necessary authentication.
3	The fixed terminal demands the UID be acquired from the mobile terminal.
4	The mobile terminal provides UID (=IMSI) that it maintains itself. However, in actual operation, UID is provided only when the mobile terminal asks the user whether it may provide UID and the user accepts.
5	The fixed terminal confirms whether the mobile terminal has already maintained the bootstrap parameters (B-TID, Lifetime).
6	The mobile terminal notifies the fixed terminal if the bootstrap parameter has already been maintained. NULL is notified so that the mobile terminal does not maintain the bootstrap parameters the first time.
7	The fixed terminal sends connection requests to the BSF-S.
8	The BSF-S that receives a connection request transmits a bootstrap request (RAND, AUTN) to the fixed terminal (same as step 2 in Fig. 3).
9	The fixed terminal receives RAND and AUTN and forwards RAND and AUTN to the mobile terminal.
10	A mobile terminal does the authentication operation by using RAND, AUTN, and private key (K) shared with the BSF-S beforehand. Specifically, XMAC, RES, CK, and IK are derived, and MAC included in AUTN is checked to see whether it is the same as the XMAC of the operation result. If it is not the same, it is an error. The mobile terminal concatenates CK with IK, and it secretly maintains Ks(=CK||IK). The authentication result (RES) is generated and transmitted to the fixed terminal. The fixed terminal forwards RES to the BSF-S (same as step 3 in Fig. 3).
11	The BSF-S checks RES (same as step 4 in Fig. 3).
12	Private key (Ks) is derived at the same time, and B-TID and Lifetime are transmitted to the fixed terminal (same as step 5 in Fig. 3.).
13	The fixed terminal sends the mobile terminal the preservation request of the authentication result. B-TID and Lifetime are transmitted to the mobile terminal.
14	The mobile terminal preserves B-TID and Lifetime (same as step 6 in Fig. 3).
15	The fixed terminal transmits B-TID to the mobile terminal and does the service authentication request (NAF).
16	The mobile terminal derives Ks_NAF by using information such as RAND and B-TID, and transmits Ks_NAF to the fixed terminal. The fixed terminal transmits B-TID to the NAF-S (same as step 1 in Fig. 5).
17	The NAF-S transmits B-TID and NAF-ID to the BSF-S (same as step 2 in Fig. 5).
18	The BSF-S generates Ks_NAF and transmits Ks_NAF to the NAF-S (same as step 3 in Fig. 5).
19	The NAF-S preserves Ks_NAF and provides service by using Ks_NAF (same as step 4 in Fig. 5).
20	The fixed terminal receives service from the NAF-S by using Ks_NAF (same as step 5 in Fig. 5).

[†]Preferably, this function acquires ID from a contactless IC chip built into the terminal. In this implementation, ID to which the registry is set beforehand is acquired.

[††]The terminal on the connected side should also specify the other party's address in the pairing of Bluetooth. In this implementation, the socket connection from the fixed-terminal side is waited for on the mobile-terminal side, the Bluetooth address on the fixed-terminal side is acquired from the socket connection information after the connection is completed, and the pairing processing is executed.

4.4 Interface of Mobile Terminal and Fixed Terminal

This subsection describes the interfaces between the mobile and the fixed terminals. These interfaces are achieved by using the function of Bridge explained in Sect. 4.2. The interface for the connected authentication operation (bootstrap authentication operation) and the service authentication operation (Ks_NAF key deriving operation) is necessary to achieve the flow shown in Fig. 8. This necessary interface is enumerated as follows.

- GET UID
  is a UID acquisition request that acquires International Mobile Subscriber Identity (IMSI)
  [Input]: None, [Output]: IMSI (ex.200150999999999)
- GET BOOTSTRAP Parameter
  reads EF_{GBABP} as Bootstrap parameter
  [Input]: None, [Output]: RAND, B-TID, Lifetime
- BOOTSTRAP
  calculates CK,IK,RES by using RAND and AUTN
  [Input]: RAND, AUTN, [Output]: RES
- STORE TID
  preserves B-TID and Lifetime as authentication results
  [Input]: B-TID, Lifetime, [Output]: None
- NAF Derivation
  Derives Ks_NAF by using Ks
  [Input]: NAF_ID, IMPI, [Output]: Ks_NAF

4.5 Function in Mobile Terminal

This subsection describes the functions needed in the mobile terminal when the proposed method is achieved. These functions are executed in USIM or in the memory of the mobile terminal in Fig. 7. The following are the main functions.

- Maintenance and transmitting function of UID
- Maintenance and transmitting function of bootstrap parameter
- BOOTSTRAP function (CK, IK, and RES are calculated from RAND and AUTN).
- STORE TID function (B-TID and Lifetime are preserved).
- NAF Derivation function (Ks_NAF is made by using Ks).

5. Efficiency Improvement

This section describes a situation in which the user who is receiving a service through a TV in the living room moves to the study with the mobile terminal and receives the same service through the TV there. When a service is switched from one fixed terminal to another by the flow shown in Fig. 8, the authentication processing can be simplified. Figure 9 shows the processing flow chart of the connection of the mobile terminal to a new fixed terminal other than the above-mentioned fixed terminal and the service received in a new fixed terminal from the NAF-S.

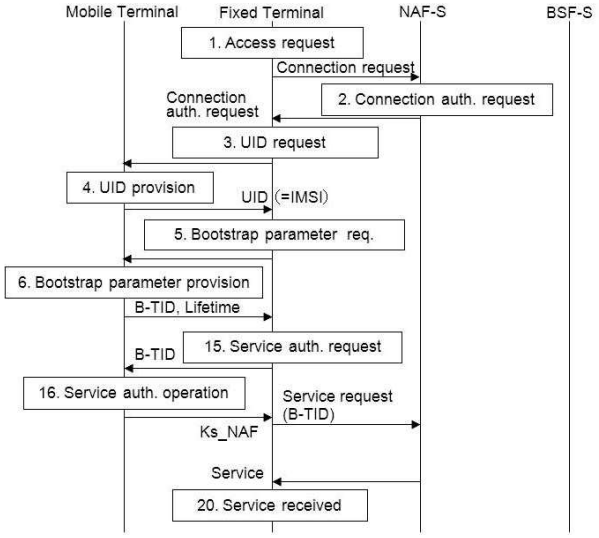

Fig. 9 Processing flow that connects mobile terminal with new fixed terminal and receives service.

When switching to a new fixed terminal, the processing represented by the short dashed line in Fig. 8 is omissible. First of all, processing from step 7 to just before step 15 can be omitted. This is because the mobile terminal has already been authenticated once or more. Specifically, after the previous connection, bootstrap results (B-TID, Lifetime) are maintained in the conserved region in step 14. Therefore, it is possible to provide the bootstrap parameter from the mobile terminal to a new fixed terminal in step 6. Therefore, a new fixed terminal need not acquire B-TID and Lifetime from the BSF-S.

Next, if key (Ks_NAF) at the previous service authentication is maintained, it need not be an operation of the mobile terminal that derives the key. Therefore, a part of step 16 where Ks_NAF is derived is omissible.

Finally, for processing from step 17 to step 19, the NAF-S maintains Ks_NAF relating to the session ID (B-TID) and to the expiration date (Lifetime). Therefore, because NAF-S need not request the generation of Ks_NAF to BSF-S, the processing from step 17 to 19 is omissible.

Figure 9 shows the flow that omits the processing represented by the short dashed line in Fig. 8. Thus, the same service can cooperate with a new fixed terminal by making service authentication information (Ks_NAF) maintained in the mobile terminal cooperate in a new fixed terminal when a service is switched from a certain fixed terminal to a new one. Moreover, new services can be received even if the bootstrap has not been authenticated again because a new service key (Ks_NAF2) can be derived by using bootstrap parameter information (RAND, AUTN, Lifetime, Ks) about when another service (NAF2) will be received.

6. Evaluation of Performance

This section evaluates the performance. Section 6.1 com-

pares the processing speeds of the conventional and proposed methods. Section 6.2 compares the processing speeds the first and second times the proposed method was tested.

The processing of the measuring object is as follows. The following steps correspond to the processing steps in Fig. 8.

- Connection request time (step 1 to 2)
- UID acquisition time (step 3 to 4)
- Bootstrap parameter acquisition time (step 5 to 6)
- Connection request time (step 7 to 8)
- Bootstrap time (step 9 to 10)
- Authentication result transmission time (step 11 to 12)
- Bootstrap parameter preservation time (step 13 to 14)
- Ks_NAF derivation time (step 15 to 16)
- Service receiving time (step 17 to 20)

We implemented the function described in Sect. 4.5 by the following two methods because the interface to an actual USIM was uncertain[†] when a smart phone was used and was measured this time.

1. Implementing an application program in a smart phone. (APP Implementation)
2. Implementing an application program on an integrated circuit chip installed in a smart phone as the second chip other than USIM (IC Implementation)

We used FeliCa for contactless IC communication between terminals. Moreover, we used an adaptor corresponding to Bluetooth Ver.2.1 for data communication after the handover. In addition, we used a LAN for the network between PC and the servers.

6.1 Comparison between Conventional and Proposed Methods

This section evaluates the performances of the conventional and proposed systems. A conventional system is a GBA system that does not communicate between terminals via Bluetooth. (In other words, there is only one terminal on the user side.) Table 4 details the equipment composition of the smart phone and notebook PC used to assess the performance.

Table 5 lists the measurement results of the conventional and proposed methods attained during APP implementation. We measured each measurement item 10 times. The mean value was calculated from eight measurements from which the minimum and the maximum measurements were excluded.

According to Table 5, the overhead of the communication is about two seconds at most between two terminals of the proposed method. In addition, we can say that this difference will become smaller as mobile terminals become more powerful, as shown in the evaluation results in the following section.

Table 4 Specs of equipment used for evaluation.

	Mobile Terminal	Fixed Terminal	NAF-S/BSF-S
CPU	Marvell PXA 270 (520 MHz)	Intel Pentium M (1.2 GHz)	Intel Pentium M (1.2 GHz)
OS	Windows Mobile 6	Windows XP	Windows XP
Memory	112 MB	752 MB	752 MB

6.2 Performance Evaluation of Proposed Method

This section evaluates the performance of the proposed system. Table 6 details the equipment composition of the smart phone and notebook PC used to assess the performance.

Table 7 shows the measurement results of the proposed authentication processing attained during APP implementation. Table 7 shows times taken when the proposed protocol was executed by using one fixed terminal (Terminal 1) the first time and a different fixed terminal (Terminal 2) the second time. We measured each measurement item 12 times. The mean value was calculated from ten measurements from which the minimum and the maximum measurements were excluded.

Next, Table 8 shows the measurement results of the authentication processing attained during IC implementation.

Table 5 Measurement results of Conventional and Proposed methods.

Measured Item	Conventional	Proposed
Connection request	0.94	0.93
UID acquisition & Bootstrap parameter acquisition	0.00	0.75
Connection request	10.04	10.06
Bootstrap & Authentication result transmission & Bootstrap parameter preservation	0.06	1.07
Ks_NAF derivation & Service receiving	0.09	0.63
Total	11.13	13.45

Table 6 Specs of equipment used for evaluation.

	Mobile Terminal	Fixed Terminal	NAF-S/BSF-S
CPU	Qualcomm QSD8250 (1 GHz)	Intel Core 2 Duo T8100 (2.10 GHz)	Intel Core 2 Duo T8100 (2.10 GHz)
OS	Windows Mobile 6.5 Professional	Windows XP SP2	Cent OS 5.3
Memory	512 MB (ROM) 256 MB (RAM)	3 GB	3 GB

Table 7 Measurement result of APP Implementation.

Measured Item	Terminal 1 at first time	Terminal 2 at second time
Connection request	0.83	0.88
UID acquisition	1.43	1.37
Bootstrap parameter acquisition	0.46	0.47
Connection request	0.08	0.00
Bootstrap	0.17	0.00
Authentication result transmission	0.01	0.00
Bootstrap parameter preservation	0.22	0.00
Ks_NAF derivation	0.22	0.22
Service receiving	0.89	0.84
Total	4.31	3.77

[†]However, the IC chip used this time has a Java Card OS, which is adopted for a lot of USIMs. Therefore, implementing functions on USIM is thought to be easy.

Table 8 Measurement results of IC Implementation.

Measured Item	Terminal 1 at first time	Terminal 2 at second time
Connection request	0.73	0.81
UID acquisition	1.78	1.83
Bootstrap parameter acquisition	0.80	0.78
Connection request	0.07	0.00
Bootstrap	52.49	0.00
Authentication result transmission	0.01	0.00
Bootstrap parameter preservation	0.43	0.00
Ks_NAF derivation	2.67	2.64
Service receiving	0.75	0.70
Total	59.73	6.76

The measuring method is similar to that in Table 7.

The difference between the total times of the first and second APP implementations was 0.54 seconds. The difference between the total times of the first and second IC implementations was 52.97 seconds[†].

Because this APP implementation does not use tamper-resistant hardware, it is not very safe. Moreover, because this IC Implementation does not have the coprocessor for AES, it is slow. It is assumed that USIM with AES that is implemented by hardware will be used in the future. Two kinds of these actual measurement values are the fastest value, in which safety is sacrificed for speed, and the slowest value, in which speed is sacrificed for safety. This means the speed assumed at the actual business service obtained the upper and lower bounds. This can be expected to be the value between these actual measurement speeds at the time of actual service. This result for the processing time makes it is possible to believe that simplifying the authentication by federating the terminal is an effective enough method.

7. Safety and Convenience Considerrations

7.1 Safety

This subsection evaluates the safety of the proposed system. Shared key K is shared with the terminal and BSF-S in the GBA Protocol as shown in Fig. 4. Therefore, the key sharing ends beforehand. When the user is authenticated, the bootstrap is done with key K. Afterwards, key for service Ks_NAF is generated by using the bootstrap result. On the server side, BSF-S generates Ks_NAF and it is forwarded to NAF-S. It is protected by the lease line or the virtual private network between NAF-S and BSF-S. It is protected with shared key K between the terminal and BSF-S and with Ks_NAF between the terminal and NAF-S.

The difference between the conventional and proposed methods was to have the concept of the conventional terminal divided into a mobile terminal and fixed terminal. In the proposed method, the mobile terminal and the fixed terminal communicate in accordance with the Bluetooth communication as described in Sect. 4.2. Because the user's passphrases of Bluetooth are exchanged by using the contactless IC communication, and then after communication is encrypted, the communication between the mobile terminal and fixed terminal is protected. In addition, unintended usage can be prevented by the user having to clearly swipe the mobile terminal[††]. In addition, if the user loses the mobile terminal, the use of the mobile terminal can be limited by the remote lock function, which a lot of current mobile careers provide.

7.2 Convenience

This subsection evaluates the convenience of the proposed system. A conventional GBA protocol was the technology for a mobile terminal and was used for the authentication when service for a mobile terminal was used. However, the proposed method enables it to receive services that use the terminals connected to a fixed network rather than connected to a mobile terminal for an easy "touching" operation. The convenient points of the proposed method are as follows.

- The user can easily use it.
- A mobile network operator can provide the authentication function to the terminal connected with a network other than a mobile one.
- The service provider leaves the authentication and the account to a mobile network operator, so the user can concentrate on the service provided.

8. Conclusion and Future Work

In this paper, we propose a mechanism that enables mobile network operator to authenticate their subscribers' account when using a terminal connected to a fixed network. In addition, we showed that fixed terminals are easy to switch between by using the proposed mechanism. Moreover, we constructed a system on the basis of the proposed mechanism and evaluated its performance, safety, and convenience.

We showed that a mobile network operator can use the existing subscriber authentication system for the new service via a fixed terminal that has been connected with a fixed network by using our proposed system. Moreover, the performance assessment showed that the authentication time when switching between fixed terminals was greatly shortened. The application of this system to various services in the future will be promoted.

Acknowledgments

This study is a result of "Research and development of

[†]We implemented the AES code in IC chip processing in software, which took a lot of time. If a coprocessor is used for processing the AES code in the future, it should be able to be implemented faster.

[††]For instance, when a safe execution environment for a service is demanded, like when a user confirms his/her validity when a video is delivered, the safety of the fixed terminal must be secured separately.

the terminal platform technology" sponsored by National Institute of Information and Communications Technology (NICT).

About Trademarks

- Windows, Windows Mobile, and Internet Explorer are registered trademarks of the Microsoft Corporation.
- Intel, Pentium and Intel Core™ are registered trademarks of the Intel Corporation and its subsidiary companies.
- Qualcomm is a registered trademark of QUALCOMM Incorporated.
- Bluetooth is a registered trademark of Bluetooth-SIG Inc.
- FeliCa is a registered trademark of the Sony Corporation.
- Wi-Fi is a registered trademark of Wi-Fi Alliance.
- ARM is a registered trademark of ARM Limited.
- Firefox is a registered trademark of the Mozilla Foundation.
- DLNA is a registered trademark of the Digital Living Network Alliance.
- OASIS and SAML are trademarks of OASIS.
- Marvell is a registered trademark of the Marvell Technology Group and its subsidiary companies.

References

[1] "Generic authentication architecture (GAA) generic bootstrapping architecture," 3GPP TS 33.220 3rd Generation Partnership Project.
[2] "Technical specification group core network and terminals bootstrapping interface (Ub) and network application function interface (Ua) protocol details," 3GPP 3GPP TS 24.109, 3rd Generation Partnership Project.
[3] "Connection handover technical specification," NFC Forum, Nov. 2008.
[4] "Specification of the MILENAGE algorithm set," 3GPP 3GPP TS 35.206, 3rd Generation Partnership Project.
[5] K. Umezawa and S. Susaki, "Development of an remote access system with smart phone," 31st Society of Information Theory and its Applications Symposium (SITA2008) Proceeding, pp.971–974, Oct. 2008.
[6] K. Umezawa, T. Kato, and S. Tezuka, "Development of FMC authentication technique with mobile terminal," IEICE Technical Report, ISEC2009-36, SITE2009-28, ICSS2009-50, July 2009.
[7] K. Umezawa, T. Kato, and S. Tezuka, "Development and evaluation of a remote access system with smart phone," 8th Forum on Information Technology (FIT2009) Proceeding, vol.4, pp.67–73, Sept. 2009.
[8] S. Cantor, J. Kemp, R. Philpott, and E. Maler, "Assertions and protocols for the OASIS security assertion markup language (SAML) V2.0," OASIS Standard, March 2005.
[9] K. Umezawa, T. Kato, and T. Tashiro, "Development and evaluation of coordination technology using cookie as authenticated information between terminals," Computer Security Symposium (CSS2009) Proceeding, pp.81–86, Oct. 2009.
[10] K. Umezawa, T. Tashiro, and S. Tezuka, "A proposal for federation technology for authenticated information between terminals," International Conference on Mobile, Ubiquitous and Pervasive Computing (ICMUPC 2010), World Academy of Science, Engineering and Technology, vol.63, pp.277–284, March 2010.
[11] Liberty Alliance, "Liberty Alliance ID-WSF 1.1 specifications," http://www.projectliberty.org/resource_center/specifications/liberty_alliance_id_wsf_1_1_1_specifications.
[12] A. Fujii, K. Ishikawa, T. Morizumi, Y. Kikuchi, T. Yamada, M. Kawamori, and K. Kawazoe, "Seamless viewing service for multi-device users by accession of authentication information," The Institute of Image Information and Television Engineers Technical Report, vol.32, no.37, pp.21–26, 2008-09-25.
[13] K. Miyakawa, S. Hibino, K. Horiguchi, H. Seshimo, S. Fukada, T. Takahashi, and T. Yamada, NTT Technical Journal, pp.12–17, Oct. 2009. (In Japanese)
[14] T. Nagai, "Network connectivity technologies for mobile devices," Toshiba Review, vol.64, no.12, pp.37–40, 2009.
[15] DLNA Networked Device Interoperability Guidelines v1.5.

Katsuyuki Umezawa received a B.E., M.E., and Dr.E. in Industrial and Management Systems Engineering from Waseda University, Tokyo, Japan, in 1994, 1996 and 2007, respectively. In 1996, he joined Hitachi, Ltd., Systems Development Laboratory, Kanagawa, Japan, which was changed to Yokohama Laboratory in 2011. His research interests are distributed object systems, mobile security systems, and smart card security systems. He is a member of the Information Processing Society of Japan and the Institute of Electrical Engineers of Japan.

Satoru Tezuka has been a Professor in the School of Computer Science, Tokyo University of Technology since 2009. He received a B.E. and Dr.E. in Science and Technology from Keio University, Kanagawa, Japan, in 1984 and 2000, respectively. In 1984, he joined Hitachi, Ltd., Micro-Electronics Development Laboratory, Kanagawa, Japan. After that, he went to Hitachi, Ltd., Systems Development Laboratory, Kanagawa, Japan. His research interest is security systems, especially the development of Electronic Certification Systems. He is a member of the Information Processing Society of Japan. He received the 2004 and 2008 IPSJ Best Paper Awards.