A Provably Secure Group Authentication Protocol for Various LTE Networks

Boriphat Kijjabuncha[1] and Pipat Hiranvanichakorn[2]
(Corresponding author: Boriphat Kijjabuncha)

Graduate school of Applied Statistics, National Institute of Development Administration[1]
118 Moo 3, Serithai Road, Klong-Chan, Bangkapi, Bangkok 10240, Thailand
Suksoomboon Laboratory[2]
242 Tanalai Road, Amphoe Muang Chiang Rai, Chiang Rai 57000, Thailand
(Email: boriphat.k@gmail.com)

(Received Mar. 22, 2018; revised and accepted Sep. 2, 2018)

Abstract

Group authentication is beneficial for group work in the Long Term Evolution (LTE) networks because it reduces the traffic of networks. For practical use, members of a group should be able to come from different network providers. In addition, while some group members use a network service, others may use other network services. Although the group members are in different networks, they should be able to work together. To fulfill these needs, we propose a secure group authentication protocol (SE-GA) in which each group member uses his/her long term private key and public key to create shared secret (keys) with network devices, such as Home and mobile management entity (MME). These shared keys are computed by using the Diffie-Hellman key exchange and are utilized in the authentication process. By using this technique instead of pre-shared keys between mobile devices and network devices, SE-GA is flexible and scalable. In SE-GA, only the first member in a MME's area has to authenticate himself/herself with the Home, while the remaining members in the area can authenticate directly with the MME. This reduces the network traffic. In this paper, authentication proof is also given using the well-known BAN logic, and the security of the protocol is analyzed and compared with some protocols.

Keywords: BAN Authentication Logic; Diffie-Hellman Key Exchange; Group Authentication; LTE Network

1 Introduction

The research group model helps users to work together with their group even though they live in different LTE networks. However, group communication needs security management to control any risks occurred in the system and protect against unauthorized users causing a system failure. Thus, network applications need privacy, confidentiality, integrity, authentication methods to protect their information from unauthorized access.

In the mobile environment, in order to use services of a network, mobile equipment (smart phones, smart watches, laptops, *etc.*) have to authenticate themselves with their home networks (HNs). However, if several mobile equipment in the same group authenticate with their HNs at the same time the traffic of the network will be crowded. This can reduce the stability of the system, and the performance of the network decreases. Therefore, an efficient group authentication protocol is needed in the group model.

Recently, several research works have been studied on group communication and authentication [1, 6, 7, 10–15, 17, 19]. In 2009, Ou *et al.* [16] proposed a Cocktail protocol with authentication and key agreement (Cocktail-AKA) on the Universal Mobile Telecommunications System (UMTS). The protocol allows a service network (SN) to calculate the medicated authentication vectors (MAV) in advance. MAV is calculated only once and can be reused. The MAV is used with prescription authentication vector (PAV) to produce many effective authentication vectors (AVs) for mutual authentication with the mobile stations (MSs). PAV is calculated from home environment (HE). Even though the protocol can reduce computational overhead on the HE and communication overhead for delivering the AVs, the protocol has some weakness which cannot resist denial-of-service attack (DoS attack) as described by Wu *et al.* [20]. In 2012, Cao *et al.* [3] proposed a group-based authentication scheme and key agreement for Machine Type Communication (MTC) in LTE network. In the protocol, the traffic of authentication is crowded and the cost of cryptographic computing is high because MTC devices may be simultaneously authenticated by the network. Then this protocol may not be suitable for mobile devices as discussed by Lai *et al.* [8]. In the same year, Chen *et al.* [4] proposed a group-based authentication and key agreement (G-AKA) protocol for

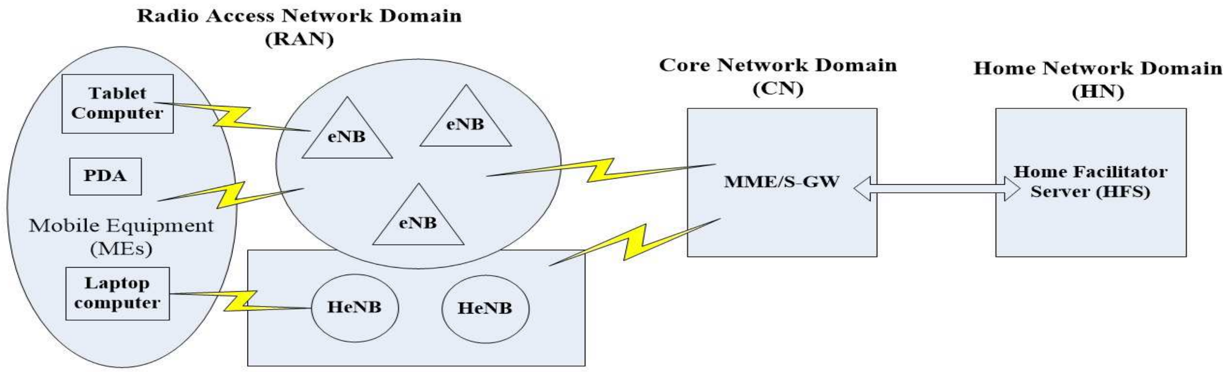

Figure 1: LTE network architecture

mobile stations (MSs) roaming from the same home network to a serving network. However, the protocol has some vulnerability such as man-in-the-middle attack as discussed by Lai *et al.* [9]. In 2013, Lai *et al.* [8] have introduced a secure and efficient group authentication and key agreement protocol (SE-AKA) which was supposed to be more secure than the evolved packet system authentication and key agreement (EPS-AKA) protocol proposed in the LTE project. In the protocol, the first mobile equipment (ME) uses its secret key to authenticate itself with its Home. Each remaining ME uses a group key and a synchronization value (SV) to authenticate itself with the service MME. However, this protocol has some weakness because a group member can be disguised by other members in the group as discussed in Section 3. In 2016, Lai *et al.* [9] proposed the group-based lightweight authentication scheme for resource-constrained machine to machine communication (GLARM). The protocol can reduce the MME overhead because the group leader collects all authentication messages from the group's members and communicates with the MME. However, as the protocol needs a group leader to send and response messages with the MME, if the group leader has some problems then the authentication process fails. Furthermore, the scope of this work is limited that all members of the group need to be in the same service network. In real work, there may be some situation that some members of the group are in different service networks.

In this paper, we propose a secure group authentication protocol (SE-GA) which makes use of users' long-term public and private keys to create secret keys with network nodes such as Home and MME. The shared keys are computed by using the Diffie-Hellman key exchange protocol based on ECC. By this way, the authentication process is flexible and scalable, and it makes group authentication easy even though group members are on different networks. In the protocol, only the first member in an MME's area has to authenticate himself/herself with the Home, while the remaining members in the area can authenticate directly with the MME. Thus SE-GA protocol can reduce network traffic. In addition, we introduce a proof for group authentication by using the well-known BAN authentication logic [2]. We have also analyzed the security of SE-GA and compared the features of the protocol with other works. From the analysis, we found SE-GA outperforms many of the past.

The rest of this paper is organized as follows. Section 2 provides some preliminaries for LTE network and elliptic curve cryptography. In Section 3, we discuss the security analysis of some previous work. SE-GA protocol is described in Section 4, and authentication proof by using BAN logic is shown in Section 5. Section 6 provides the security analysis of the protocol against some well-known attacks. The conclusion is drawn in the last section, Section 7.

2 Preliminaries

2.1 LTE Network

The LTE network architecture can be classified into 3 domains, including radio access network (RAN) domain, core network (CN) domain, and home network (HN) domain, respectively. As demonstrated in Figure 1, the network includes entities as shown in Table 1. The network is described according to 3GPP (Third Generation Partnership Project) standard as follows.

1) RAN domain includes mobile equipment (MEs), base stations (BSs) (*i.e.* eNodeB for outdoor, HeNodeB for indoor) where MEs are mobile equipment of 3GPP standard mobile devices and BSs forward messages from MEs to the serving network domain.

2) CN domain includes mobile management entities (MMEs) or serving gateways (S-GWs). An MME prepares services for the MEs's requests and S-GW forwards messages to another machines.

3) HN domain includes the Home facilitator server (HFS) which provides services for authentication process with MEs.

In the LTE networks, we assume that the network of service providers is secure. Data transmission between service providers' devices such as Home and MME is protected.

Table 1: The notations of entities in the network architecture

Notations	Definition
ME	Mobile Equipment (machine)
eNB	Type of base station (BS) called evolved Node B (eNodeB)
HeNB	Type of base station (BS) called Home evolved Node B (HeNodeB)
MME	Mobile Management Entity
S-GW	Serving Gateway
HFS	Home Facilitator Server

2.2 Elliptic Curve Cryptography

For the Elliptic Curve Cryptography (ECC), we describe the situation of Alice and Bob which they have a pair of keys (public key and private key) [18]. Public keys can be published. Alice and Bob can create a shared key for sending data in secure communication by using the Diffie-Hellman key exchange. The principle is as follows. In a finite field (Fq), an elliptic curve E is defined over Fq and P is a point on E (*i.e.* $P \in E$). Alice chooses a random secret a in Fq (*i.e.* $a \in Fq$) and computes her public key aP on E (*i.e.* $aP \in E$) and sends the key to Bob. In the same way, Bob chooses a random secret b and calculates bP on E and sends it to Alice. The secret common key between Alice and Bob is abP on E.

3 Security Analysis of SE-AKA Protocol

In this section, we give some security analysis of the SE-AKA for the LTE network. The SE-AKA protocol is used to facilitate mobile equipment (MEs) that have been subscribed to the home network (HN) to roam into a serving network (SN) which is far from HN. The SE-AKA protocol can be divided into 2 protocols:

1) Protocol execution for the first equipment;

2) Protocol execution for the remaining equipment of the same group. Because the supplier provides a group key (GK) to each group for secure communication, then all MEs of the group can know the group key. Table 2 shows the notations used in the SE-AKA protocol illustrated in Figure 2.

In the first device authentication process, the ME_1 uses a secret key which known only between it and the Home to generate a message authentication code (MAC) to authenticate itself with the Home via MME. Home verifies ME_1 by using the same secret key. If the verification is successful, the Home sends the group information management list (GIML), including group name, group ID, MEs' IDs and synchronization values (SVs) to MME/SN.

Table 2: Notations use in the SE-AKA protocol [8]

Notations	Definition
R_{G1-j}	The random number generated by ME_j in group G1
R_{MME}	The random number generated by MME
ID_{G1}	The identity of group G1
ID_{MME}	The identity of MME
$TID_{ME_{G1-j}}$	The temporary identity of ME_j in group G1
MAC_{MME}	The message authentication code computes by MME
$MAC_{ME_{G1-j}}$	The message authentication code computes ME_j in group G1
AMF	Authentication management field
LAI	Location Area Identification
$KGK_{ME_{G1-j}}$	The key generation key between ME_{G1-j} and MME
$f_{GTK_{G1}}$	A key generation function of group G1
aP, bP	A device's public key
abP	A shared key between two parties
ME	Mobile Equipment
MME	Mobile Management Entity
HSS	Home Subscriber Server

In self-confirmation of each remaining ME of the group, the GK and SV are mainly utilized in the authentication process. For GK, every ME knows this value and SV is not a key, so the security of this verification is reduced, and the authentication process can be easily attacked. Then, a group member can impersonate other ones who have not yet confirmed themselves.

As shown in Figure 2, an ME wants to disguise to be another one by sending the identity information ($AUTH_{ME_{G1-j}} = ID_{G1}||TID_{ME_{G1-j}}||R_{G1-j}$) of target member to the service MME. The MME uses a group temporary key (GTK) which got from Home (HSS) to perform mutual authentication with the ME without HSS's assistance. The GTK is generated from Home by using group key (GK). This key makes the MME to believe an ME.

In the protocol, the MME sends authentication request $AUTH_{MME} = (ID_{MME} \,||\, ID_{G1}||\, TID_{ME_{G1-j}}||\, MAC_{MME}||\, R_{HSS} \,||\, R_{MME} \,||\, R_{G1-j}||\, AMF \,||\, aP)$ where $MAC_{MME} = f_{GTK_{G1}}(ID_{MME}||\, ID_{G1}||\, TID_{ME_{G1-j}}||R_{HSS}||R_{MME}||R_{G1-j} ||AMF||aP||SV_{G1-j} + i)$ to the ME. The value i is the sequence of the mutual authentication with ME_{G1-j}. If the fake ME could ever attack the synchronization value (SV_{G1-j}), it selects a random number b and can computes bP, and computes $KGK_{ME_{G1-j}} = f_{GTK_{G1}}$ $(ID_{MME} \,||\, TID_{ME_{G1-j}} \,||\, R_{MME} \,||\, R_{G1-j} \,||\, abP)$ and

$\mathrm{MAC}_{\mathrm{ME}_{\mathrm{G1-j}}} = f_{\mathrm{KGK}_{\mathrm{ME}_{\mathrm{G1-j}}}}(\mathrm{ID_{MME}} \,||\mathrm{ID_{G1}}\, ||\mathrm{TID}_{\mathrm{ME}_{\mathrm{G1-j}}}|| \mathrm{R_{MME}} \,||\, \mathrm{LAI} \,||\, bp \,||\, abP \,||\, \mathrm{SV_{G1-j}} + i)$. It then sends $(\mathrm{MAC}_{\mathrm{ME}_{\mathrm{G1-j}}}||bp)$ to the MME.

Upon receiving the response, MME verifies $\mathrm{MAC}_{\mathrm{ME}_{\mathrm{G1-j}}}$ by using the received information to compute $\mathrm{MAC}_{\mathrm{ME}_{\mathrm{G1-j}}}$ by itself. It then compares the computed $\mathrm{MAC}_{\mathrm{ME}_{\mathrm{G1-j}}}$ with the received $\mathrm{MAC}_{\mathrm{ME}_{\mathrm{G1-j}}}$. If they are the same then MME believes that ME.

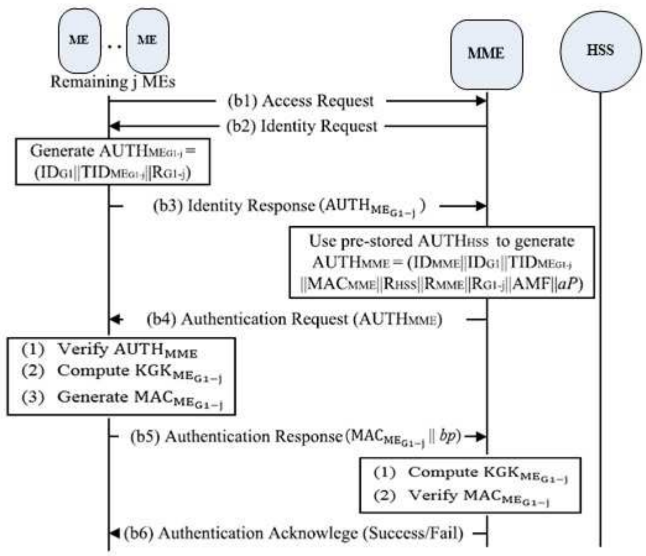

Figure 2: The authentication procedure of remaining MEs [8]

In this way, a member of the group will be able to disguise itself as other exist members. Although this protocol has some vulnerable points and is designed to work in the only one LTE network, the idea to seperate the authentication process into the authentication of the first device and the remaining devices can reduce the network traffic. According to this idea, we then apply it to create a new protocol.

4 The Proposed SE-GA Protocol

In this section, we propose SE-GA protocol for ME/MEs in a group to access into serving network domains. The design goals of SE-GA protocol are:

1) Members of the group must be independent.

2) The protocol allows the group in which members can come from different home networks and they can work on different networks at the same time as shown in Figure 3;

3) Each member cannot impersonate another member within the group;

4) Protocol must be able to prevent attacks such as secure key derivation, man-in-the-middle attacks, and so on. In addition, identity verification should be secure to ensure accuracy and to minimize interaction time.

4.1 Initialization

In the initial stage, each ME creates a pair of long-term private key and public key, and it sends the public key to its Home. Then the HN and ME can create a shared secret key by using a Diffie-Hellman key exchange. It is noted that a long-term public key of the Home is well-known. When several MEs form a group $\mathrm{G_n}$, they create a session group key.

Each group member then sends the group's information, *i.e.* Group ID, number of members, Temporary identity numbers (TID) and all long-term public keys of the group members to his/her Home. This data is sent with integrity control by utilizing the shared key between the group member and the Home. The data does not need to be secret. However, if we need secrecy the information can be covered by using the shared key. On receiving the messages, each Home keeps the group's information in GDL as shown in Table 3.

Table 3: Group detail list (GDL)

Group number	Group ID	$\mathrm{TID_{ME_i}}$	$\mathrm{ID_{HFS_k}}$	Public $\mathrm{key_{ME_i}}$
$\mathrm{G_1}$	$\mathrm{ID_{G_1}}$	$\mathrm{TID_{ME_1}}$	$\mathrm{ID_{HFS_1}}$	$\mathrm{Pub_{ME_1}}$
.	.	$\mathrm{TID_{ME_2}}$	$\mathrm{ID_{HFS_2}}$	$\mathrm{Pub_{ME_2}}$
.	.	.	.	.
.	.	.	.	.
.	.	$\mathrm{TID_{ME_i}}$	$\mathrm{ID_{HFS_k}}$	$\mathrm{Pub_{ME_i}}$
$\mathrm{G_2}$	$\mathrm{ID_{G_2}}$	$\mathrm{TID_{ME_1}}$	$\mathrm{ID_{HFS_1}}$	$\mathrm{Pub_{ME_1}}$
.	.	$\mathrm{TID_{ME_3}}$	$\mathrm{ID_{HFS_2}}$	$\mathrm{Pub_{ME_3}}$
.	.	.	.	.
.	.	.	.	.
.	.	$\mathrm{TID_{ME_m}}$	$\mathrm{ID_{HFS_l}}$	$\mathrm{Pub_{ME_m}}$

4.2 SE-GA Protocol for Each $\mathrm{ME_i}$ in a Group $\mathrm{G_n}$

When an $\mathrm{ME_i}$ connects to a wireless point, it authenticates itself with that network in order to use network services.

In the authentication process, an $\mathrm{ME_i}$ device in a group $\mathrm{G_n}$, connects to the wireless point in any area mobile management entity ($\mathrm{MME_j}$). The $\mathrm{ME_i}$ then sends an access request $\mathrm{AUTH_i}$ to the $\mathrm{MME_j}$. When the $\mathrm{MME_j}$ receives a request, it checks whether the $\mathrm{ME_i}$ is a member in the previously requested group by using $\mathrm{HFS_k}$ and $\mathrm{ID_{G_n}}$ in the $\mathrm{AUTH_i}$ to determine if a group detail list (GDL) exists in the $\mathrm{MME_j}$'s database. If not, $\mathrm{ME_i}$ is the first machine in the group that requests the connection with $\mathrm{MME_j}$. $\mathrm{MME_j}$ then performs the authentication process for the first ME device (*i.e.* using case 1) and gets a GDL from $\mathrm{ME_i}$'s Home. Otherwise, if there is the GDL of that $\mathrm{ME_i}$, then $\mathrm{MME_j}$ performs an authentication process as if the

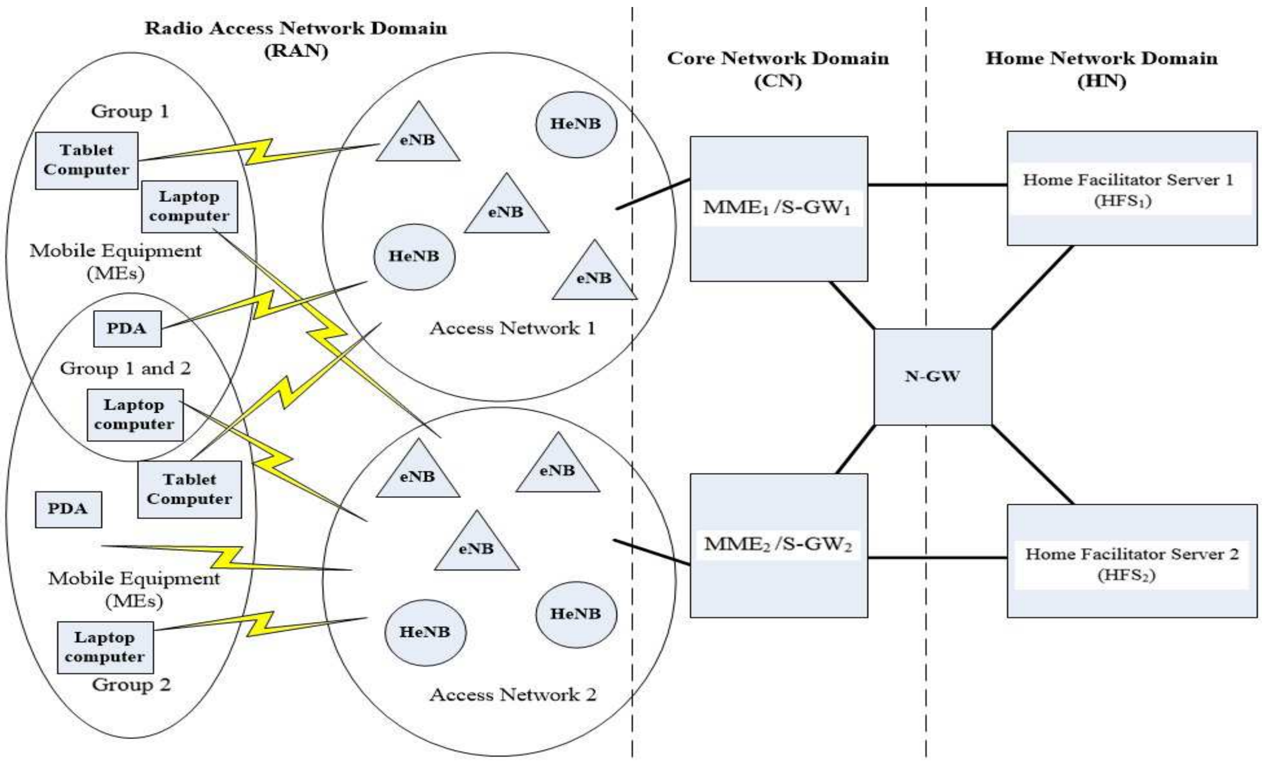

Figure 3: Network Architecture based on 3GPP standard in SE-GA protocol

ME_i is a remaining ME device (*i.e.* using case 2). Table 4 shows the notations used in the SE-GA protocol. The machine x or y can be an MME, HFS or ME. When x or y is represent by G_n- i, it means an ME_i of a group n.

The steps of the SE-GA protocol are as the following.

Case 1: Authentication for the first ME

If ME_i is the first member of a group G_n that want to authenticate with MME_j, then MME_j does not have a GDL of the ME_i's group in MME_j's database. Therefore, MME_j looks for the ME_i's home network (HFS_k) in the authentication request and then forwards the authentication data request, local area identification of MME_j, identity of MME_j and MAC_{MME_j} (*i.e.* $AUTH_i$, LAI_{MME_j}, ID_{MME_j}, MAC_{MME_j}) to HFS_k of ME_i through N-GW. If the authentication data request passes the network gateway (N-GW), the N-GW only forwards the authentication request to the destination (HFS_k). This case is composed of Steps 1 – 5 as shown in Figure 4.

Step 1. $ME_i \rightarrow MME_j$: **Access Request** ($AUTH_i$).

The ME_i generates $AUTH_i = (ID_{G_n} \,||\, TID_{ME_i} \,||\, R_{G_n-i} \,||\, TS_{G_n-i}||HFS_k||LAI_{ME_i}||bP||MAC_q||MAC_i)$ and sends it to MME_j. $MAC_q = f^1_{SK_{MME_j-ME_i}}$ $(ID_{G_n} \,||\, TID_{ME_i} \,||\, R_{G_n-i} \,||\, TS_{G_n-i}||HFS_k||LAI_{ME_i}||bP)$ and it is used by MME_j to verify whether it is the correct ME_i. While $MAC_i = f^5_{SK_{ME_i-HFS_k}}$ $(ID_{G_n} \,||\, TID_{ME_i} \,||\, R_{G_n-i} \,||\, TS_{G_n-i} \,||\, HFS_k \,||\, LAI_{ME_i} \,||\, bP)$ and it is used by HFS_k to verify whether it is the correct ME_i. The function $f^1_{SK_{MME_j-ME_i}}$ and $f^5_{SK_{ME_i-HFS_k}}$ are used for generating message authentication codes MAC_q and MAC_i respectively. $SK_{MME_j-ME_i}$ is a shared secret key between MME_j and ME_i, and is computed from ME_i's private key and MME_j's public key by using the Diffie-Hellman key exchange. It is noted that MME_j's public key is well-known on the internet. In part of $SK_{ME_i-HFS_k}$, it is a shared secret key between ME_i and its home network (HN) which is computed by performing the Diffie-Hellman key exchange in the initialization state. The value bP is a session public key of ME_i. It is created by selecting a random number b and computing bP on Elliptic Curve. TID_{ME_i} is a temporary identity of ME_i in HFS_k and is used for registration in 3GPP/LTE networks. The value is installed in ME_i by the supplier of ME_i.

Step 2. $MME_j \rightarrow HFS_k$: **Authentication Data Request** ($AUTH_i$, TS_{MME_j}, LAI_{MME_j}, ID_{MME_j}, MAC''_{MME_j}).

When the MME_j receives the authentication data request from ME_i, it uses HFS_k and ID_{G_n} in the $AUTH_i$ to find out whether this request is the first request of group, by searching for ID_{G_n} in the Group Detail List (GDL) of MME_j's database. If it cannot find the information in MME_j's database, then MME_j forwards $AUTH_i$, TS_{MME_j}, ID_{MME_j},

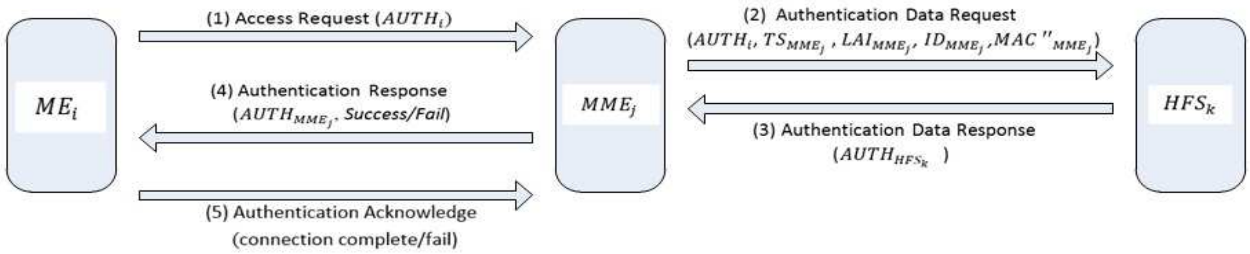

Figure 4: The SE-GA protocol for the first ME

$\mathrm{LAI_{MME_j}}$, $\mathrm{MAC''_{MME_j}}$ to the $\mathrm{HFS_k}$. The $\mathrm{LAI_{MME_j}}$ reports the location of the wireless point which $\mathrm{ME_i}$ connects to, and $\mathrm{MAC''_{MME_j}} = f^3_{\mathrm{SK_{MME_j-HFS_k}}}$ $(\mathrm{AUTH_i}||\ \mathrm{TS_{MME_j}}||\ \mathrm{ID_{MME_j}}||\ \mathrm{LAI_{MME_j}})$. The long-term secret key ($\mathrm{SK_{MME_j-HFS_k}}$) between $\mathrm{MME_j}$ and $\mathrm{HFS_k}$ is computed by using the $\mathrm{HFS_k}$'s public key and $\mathrm{MME_j}$'s private key in the Diffie-Hellman key exchange. It is noted that $\mathrm{HFS_k}$'s public key is well-known on the internet.

Table 4: Notations used in the SE-GA protocol

Notations	Definition
$\mathrm{R_x}$	The random number generated by machine x
$\mathrm{TS_x}$	The time stamp generated by machine x
$\mathrm{ID_x}$	The identity of machine x
$\mathrm{PID_x}$	The permanent identity of machine x
$\mathrm{TID_x}$	The temporary identity of machine x
$\mathrm{SK_{x-y}}$	The shared secret key between machine x and y
$\mathrm{SSK_{x-y}}$	The shared session key between machine x and y
$\mathrm{MAC_x}$	The message authentication code computed by machine x
$\mathrm{LAI_x}$	Location Area Identification of machine x
$f^1_{\mathrm{SK_{MME_j-ME_i}}}$	MAC generating function using $\mathrm{SK_{MME_j-ME_i}}$
$f^2_{\mathrm{SK_{MME_j-ME_i}}}$	SSK generating function using $\mathrm{SK_{MME_j-ME_i}}$
$f^3_{\mathrm{SK_{MME_j-HFS_k}}}$	MAC generating function using $\mathrm{SK_{MME_j-HFS_k}}$
$f^4_{\mathrm{SK_{MME_j-HFS_k}}}$	MAC generating function using $\mathrm{SK_{MME_j-HFS_k}}$
$f^5_{\mathrm{SK_{ME_i-HFS_k}}}$	MAC generating function using $\mathrm{SK_{ME_i-HFS_k}}$
aP, bP	A device's public key
abP	A shared key between two parties

$\mathrm{MME_j}$ also keeps bP and $\mathrm{MAC_q}$ in order to use them afterward.

Step 3. $\mathrm{HFS_k} \rightarrow \mathrm{MME_j}$: **Authentication Data Response** ($\mathrm{AUTH_{HFS_k}}$).

Upon receiving authentication data request ($\mathrm{AUTH_i}$, $\mathrm{TS_{MME_j}}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}}, \mathrm{MAC''_{MME_j}}$) from $\mathrm{MME_j}$, the $\mathrm{HFS_k}$ verifies $\mathrm{MME_j}$ by computing $\mathrm{MAC'''_{MME_j}} = f^3_{\mathrm{SK_{MME_j-HFS_k}}}$ $(\mathrm{AUTH_i}||\mathrm{TS_{MME_j}}||\mathrm{ID_{MME_j}}||\mathrm{LAI_{MME_j}})$ and compares it with $\mathrm{MAC''_{MME_j}}$. Here, $\mathrm{SK_{MME_j-HFS_k}}$ is computed by using $\mathrm{HFS_k}$'s private key and $\mathrm{MME_j}$'s public key. If it is the same MAC value then $\mathrm{HFS_k}$ believes that the message is sent from $\mathrm{MME_j}$.

Before $\mathrm{HFS_k}$ verifies $\mathrm{MAC_i}$ which is in $\mathrm{AUTH_i}$, the $\mathrm{HFS_k}$ compares $\mathrm{LAI_{MME_j}}$ with $\mathrm{LAI_{ME_i}}$ to check whether they are the same. If they have the same value, $\mathrm{HFS_k}$ verifies $\mathrm{MAC_i}$ by computing $\mathrm{MAC'_i} = f^5_{\mathrm{SK_{ME_i-HFS_k}}}$ $(\mathrm{ID_{G_n}}||\mathrm{TID_{ME_i}}||\mathrm{R_{G_n-i}}||\mathrm{TS_{G_n-i}}||\mathrm{HFS_k}||\mathrm{LAI_{ME_i}}||bP)$ from data in $\mathrm{AUTH_i}$. Then $\mathrm{HFS_k}$ compares $\mathrm{MAC'_i}$ with the $\mathrm{MAC_i}$. If these values are the same, the $\mathrm{HFS_k}$ can believe that the message is sent from $\mathrm{ME_i}$.

The $\mathrm{HFS_k}$ then generates $\mathrm{AUTH_{HFS_k}} = (\mathrm{R_{G_n-i}}||$ $\mathrm{ID_{HFS_k}}$ || $\mathrm{HFS_k}$ || GDL || $\mathrm{TS_{HFS_k}}$ || $\mathrm{MAC_{HFS_k}}$), where $\mathrm{MAC_{HFS_k}} = f^4_{\mathrm{SK_{MME_j-HFS_k}}}$ ($\mathrm{R_{G_n-i}}$ || $\mathrm{ID_{HFS_k}}$ || $\mathrm{HFS_k}$ || GDL || $\mathrm{TS_{HFS_k}}$) and it sends $\mathrm{AUTH_{HFS_k}}$ to the $\mathrm{MME_j}$. GDL is composed of group number, group identity, temporary identity of every $\mathrm{ME_i}$, identity of $\mathrm{HFS_k}$ and public keys of all MEs in this group as shown in Table 2.

Step 4. $\mathrm{MME_j} \rightarrow \mathrm{ME_i}$: **Authentication Response** ($\mathrm{AUTH_{MME_j}}$, Success/Fail).

After $\mathrm{MME_j}$ receives $\mathrm{AUTH_{HFS_k}}$ from $\mathrm{HFS_k}$, $\mathrm{MME_j}$ computes $\mathrm{MAC'_{HFS_k}} = f^4_{\mathrm{SK_{MME_j-HFS_k}}}$ ($\mathrm{R_{G_n-i}}$ || $\mathrm{ID_{HFS_k}}$ || $\mathrm{HFS_k}$||GDL||$\mathrm{TS_{HFS_k}}$) to verify the message from $\mathrm{HFS_k}$. If the verification passes, $\mathrm{MME_j}$ computes $\mathrm{MAC'_q} = f^1_{\mathrm{SK_{MME_j-ME_i}}}$ ($\mathrm{ID_{G_n}}$ || $\mathrm{TID_{ME_i}}$ || $\mathrm{R_{G_n-i}}$ || $\mathrm{TS_{G_n-i}}$|| $\mathrm{HFS_k}$|| $\mathrm{LAI_{ME_i}}||bP$) and compares it with $\mathrm{MAC_q}$ from Step 1. The $\mathrm{SK_{MME_j-ME_i}}$ is computed by $\mathrm{MME_j}$'s

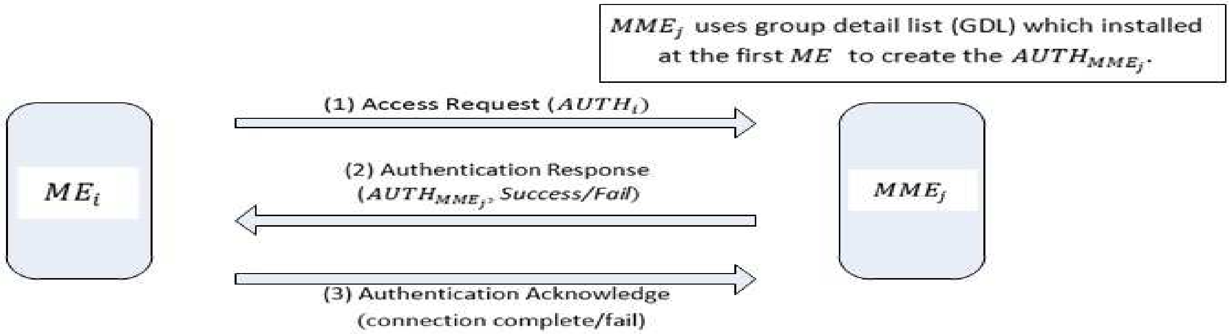

Figure 5: The SE-GA protocol for remaining ME devices

private key and $\mathrm{ME_i}$'s long-term public key got from GDL. If $\mathrm{MAC'_q} = \mathrm{MAC_q}$, $\mathrm{MME_j}$ installs GDL of $\mathrm{G_n}$ into $\mathrm{MME_j}$'s database. The GDL facilitates the $\mathrm{MME_j}$ to check the remaining $\mathrm{ME_i}$'s authentication information. Then, $\mathrm{MME_j}$ can trust the message $\mathrm{AUTH_i}$ which is sent by $\mathrm{ME_i}$, because $\mathrm{MME_j}$ got correct response from $\mathrm{ME_i}$'s Home.

$\mathrm{MME_j}$ then randomizes a number a to compute a session public key aP and a secret value abP on Elliptic Curve. Note that bP is obtained from Step 1. $\mathrm{MME_j}$ also generates $\mathrm{AUTH_{MME_j}} = (\mathrm{ID_{MME_j}}||\mathrm{ID_{G_n}}||\mathrm{TID_{ME_i}}||\mathrm{R_{MME_j}} ||\mathrm{R_{G_n-i}}||\ \mathrm{TS'_{MME_j}}||\ aP||\mathrm{MAC_{MME_j}})$, where $\mathrm{MAC_{MME_j}} = f^1_{\mathrm{SK_{MME_j-ME_i}}}\ (\mathrm{ID_{MME_j}}\ ||\ \mathrm{ID_{G_n}}\ ||\ \mathrm{TID_{ME_i}}\ ||\ \mathrm{R_{MME_j}}\ ||\ \mathrm{R_{G_n-i}}\ ||\ \mathrm{TS'_{MME_j}}||aP)$. It then sends $\mathrm{AUTH_{MME_j}}$ and a response 'success' to $\mathrm{ME_i}$. $\mathrm{MME_j}$ can now compute session key between it and $\mathrm{ME_i}$ by $\mathrm{SSK_{MME_j-ME_i}} = f^2_{\mathrm{SK_{MME_j-ME_i}}}$ $(\mathrm{ID_{MME_j}}||\mathrm{TID_{ME_i}}||\mathrm{R_{MME_j}}||\mathrm{R_{G_n-i}}||abP)$.

Step 5. $\mathrm{ME_i} \rightarrow \mathrm{MME_j}$: **Authentication Acknowledge** (connection complete/fail).

When the $\mathrm{ME_i}$ gets the authentication data response from $\mathrm{MME_j}$, it verifies $\mathrm{MME_j}$ by computing $\mathrm{MAC'_{MME_j}} = f^1_{\mathrm{SK_{MME_j-ME_i}}}\ (\mathrm{ID_{MME_j}}\ ||\ \mathrm{ID_{G_n}}\ ||\ \mathrm{TID_{ME_i}}\ ||\ \mathrm{R_{MME_j}}\ ||\ \mathrm{R_{G_n}} - \mathrm{i}\ ||\ \mathrm{TS'_{MME_j}}\ ||\ aP)$ and compares $\mathrm{MAC_{MME_j}}$ with $\mathrm{MAC'_{MME_j}}$. The $\mathrm{SK_{MME_j-ME_i}}$ is computed from $\mathrm{ME_i}$'s private key and $\mathrm{MME_j}$'s public key by using the Diffie-Hellman key exchange. $\mathrm{MME_j}$'s long-term public key is well-known on the internet.

If $\mathrm{MAC_{MME_j}}$ and $\mathrm{MAC'_{MME_j}}$ are the same then it is the correct $\mathrm{MME_j}$. $\mathrm{ME_i}$ then computes abP by using aP from $\mathrm{AUTH_{MME_j}}$ and creates a session key between $\mathrm{ME_i}$ and $\mathrm{MME_j}$ by $\mathrm{SSK_{MME_j-ME_i}} = f^2_{\mathrm{SK_{MME_j-ME_i}}}$ $(\mathrm{ID_{MME_j}}||\mathrm{TID_{ME_i}}||\mathrm{R_{MME_j}}||\mathrm{R_{G_n-i}}||abP)$. Now, the $\mathrm{ME_i}$ has a shared session key $\mathrm{SSK_{MME_j-ME_i}}$ with $\mathrm{MME_j}$ and sends connection complete to $\mathrm{MME_j}$. Otherwise, $\mathrm{ME_i}$ sends a response, 'connection failure' to $\mathrm{MME_j}$.

Case 2: Authentication for the remaining MEs

If $\mathrm{ME_i}$ is a remaining member of the group $\mathrm{G_n}$ that has a member authenticated with $\mathrm{MME_j}$, then $\mathrm{MME_j}$ has the group detail list (GDL) of group $\mathrm{G_n}$ in the $\mathrm{MME_j}$'s database. The $\mathrm{MME_j}$ can use the $\mathrm{ME_i}$'s public key in GDL to create a shared secret key $(\mathrm{SK_{MME_j-ME_i}})$ between $\mathrm{MME_j}$ and $\mathrm{ME_i}$. This case is composed of Steps 1 – 3 as shown in Figure 5.

Step 1. $\mathrm{ME_i} \rightarrow \mathrm{MME_j}$: **Access Request** $(\mathrm{AUTH_i}$).

The $\mathrm{ME_i}$ generates $\mathrm{AUTH_i} = (\mathrm{ID_{G_n}}\ ||\ \mathrm{TID_{ME_i}}\ ||\ \mathrm{R_{G_n}} - \mathrm{i}\ ||\ \mathrm{TS_{G_n-i}}\ ||\ \mathrm{HFS_k}\ ||\ \mathrm{LAI_{ME_i}}\ ||\ bP\ ||\ \mathrm{MAC_q}\ ||\ \mathrm{MAC_i})$, $\mathrm{MAC_q} = f^1_{\mathrm{SK_{MME_j-ME_i}}}\ (\mathrm{ID_{G_n}}\ ||\ \mathrm{TID_{ME_i}}\ ||\ R_{G_n-i}\ ||\ \mathrm{TS_{G_n-i}}\ ||\ \mathrm{HFS_k}\ ||\ \mathrm{LAI_{ME_i}}\ ||\ bP)$ and $\mathrm{MAC_i} = f^5_{\mathrm{SK_{ME_i-HFS_k}}}\ (\mathrm{ID_{G_n}}\ ||\ \mathrm{TID_{ME_i}}\ ||\ \mathrm{R_{G_n-i}}\ ||\ \mathrm{TS_{G_n-i}}\ ||\ \mathrm{HFS_k}\ ||\ \mathrm{LAI_{ME_i}}\ ||\ bP)$ and sends $\mathrm{AUTH_i}$ to $\mathrm{MME_j}$.

Step 2. $\mathrm{MME_j} \rightarrow \mathrm{ME_i}$: **Authentication Response** $(\mathrm{AUTH_{MME_j}}$, Success/Fail).

When the $\mathrm{MME_j}$ receives an authentication data request from $\mathrm{ME_i}$, it checks the request of $\mathrm{ME_i}$ by using $\mathrm{HFS_k}$ and $\mathrm{ID_{G_n}}$ in the $\mathrm{AUTH_i}$ to find out whether this request is the first request of group, by searching for $\mathrm{ID_{G_n}}$ in the Group Detail List (GDL) of $\mathrm{MME_j}$'s database. If it can find $\mathrm{ID_{G_n}}$, then $\mathrm{MME_j}$ computes a long-term secret key $(\mathrm{SK_{MME_j}} - \mathrm{ME_i})$ between $\mathrm{MME_j}$ and $\mathrm{ME_i}$ by using $\mathrm{ME_i}$'s public key in GDL and $\mathrm{MME_j}$'s private key.

Before $\mathrm{MME_j}$ verifies $\mathrm{MAC_q}$ which is in $\mathrm{AUTH_i}$, the $\mathrm{MME_j}$ compares $\mathrm{LAI_{ME_i}}$ with $\mathrm{LAI_{MME_j}}$ to check whether they are the same. If they have the same value, the $\mathrm{MME_j}$ computes $\mathrm{MAC'_q} = f^1_{\mathrm{SK_{MME_j-ME_i}}}\ (\mathrm{ID_{G_n}}||\mathrm{TID_{ME_i}}||\mathrm{R_{G_n-i}}||\mathrm{TS_{G_n-i}} ||\mathrm{HFS_k}||\mathrm{LAI_{ME_i}}||bP)$. It then compares $\mathrm{MAC'_q}$ with $\mathrm{MAC_q}$ from Step (1). If $\mathrm{MAC'_q} = \mathrm{MAC_q}$ then $\mathrm{MME_j}$ trusts $\mathrm{ME_i}$ and messages are sent by $\mathrm{ME_i}$.

$\mathrm{MME_j}$ then randomizes a number a to compute a session public key aP and a secret value abP on Elliptic Curve. Further, $\mathrm{MME_j}$ generates $\mathrm{AUTH_{MME_j}}$

$= (ID_{MME_j} \| ID_{G_n} \| TID_{ME_i} \| R_{MME_j} \| R_{G_n-i} \| TS'_{MME_j} \| aP \| MAC_{MME_j})$, where $MAC_{MME_j} = f^1_{SK_{MME_j-ME_i}} (ID_{MME_j} \| ID_{G_n} \| TID_{ME_i} \| R_{MME_j} \| R_{G_n-i} \| TS'_{MME_j} \| aP)$. It then sends $AUTH_{MME_j}$ and a response, 'success' to ME_i.

Now, MME_j can compute a session key between it and ME_i by $SSK_{MME_j-ME_i} = f^2_{SK_{MME_j-ME_i}} (ID_{MME_j}\|TID_{ME_i}\|R_{MME_j}\|R_{G_n-i}\|abP)$.

Step 3. $ME_i \rightarrow MME_j$: **Authentication Acknowledge** (connection complete/fail).

When the ME_i gets the authentication response from MME_j, it verifies the message by computing $MAC'_{MME_j} = f^1_{SK_{MME_j-ME_i}} (ID_{MME_j}\|ID_{G_n}\|TID_{ME_i}\|R_{MME_j}\|R_{G_n-i}\| TS'_{MME_j}\|aP)$ and compares MAC_{MME_j} with MAC'_{MME_j}. The $SK_{MME_j-ME_i}$ is computed from ME_i's private key and MME_j's public key.

If MAC_{MME_j} and MAC'_{MME_j} have the same value then ME_i believes that the message is sent from MME_j. ME_i then computes abP by using aP from $AUTH_{MME_j}$ and creates session key between ME_i and MME_j by $SSK_{MME_j-ME_i} = f^2_{SK_{MME_j-ME_i}} (ID_{MME_j}\|TID_{ME_i}\|R_{MME_j}\|R_{G_n-i}\|abP)$. Now, the ME_i has a shared session key $SSK_{MME_j-ME_i}$ with MME_j and sends connection complete to MME_j. Otherwise, if MAC_{MME_j} and MAC'_{MME_j} are not the same then ME_i sends a response, 'connection failure' to MME_j.

5 Authentication Proof by using BAN Logic

In this section, we give a proof of the SE-GA protocol by using the well-known BAN Logic. The notations used in SE-GA protocol are listed in Table 5.

Table 5: Notations used in the proof

Notations	Definition
bP	A session public key of ME_i
aP	A session public key of MME_j
$SK_{ME_i-HFS_k}$	A long-term secret shared between ME_i and HFS_k
$SK_{ME_i-MME_j}$	A long-term secret shared between ME_i and MME_j
$SK_{MME_j-HFS_k}$	A long-term secret shared between MME_j and HFS_k
$SSK_{MME_j-ME_i}$	A shared session key between MME_j and ME_i

We will prove the authentication of the mobile equipment in both cases: the case of the first ME device and the case of the remaining ME devices.

5.1 Authentication Proof for the First ME

The communicating messages used in the case of the first ME device are as follows:

(a) $ME_i \rightarrow MME_j$:
$AUTH_i = (ID_{G_n}, TID_{ME_i}, R_{G_n-i}, TS_{G_n-i}, HFS_k, LAI_{ME_i}, bP, MAC_q((ID_{G_n}, TID_{ME_i}, R_{G_n-i}, TS_{G_n-i}, HFS_k, LAI_{ME_i}, bP), SK_{ME_i-MME_j}), MAC_i((ID_{G_n}, TID_{ME_i}, R_{G_n-i}, TS_{G_n-i}, HFS_k, LAI_{ME_i}, bP), SK_{ME_i-HFS_k}))$.

(b) $MME_j \rightarrow HFS_k$:
$(AUTH_i, TS_{MME_j}, LAI_{MME_j}, ID_{MME_j}, MAC''_{MME_j}((AUTH_i, TS_{MME_j}, LAI_{MME_j}, ID_{MME_j}), SK_{MME_j-HFS_k}))$.

(c) $HFS_k \rightarrow MME_j$:
$(R_{G_n-i}, ID_{HFS_k}, HFS_k, GDL, TS_{HFS_k}), MAC_{HFS_k}((R_{G_n-i}, ID_{HFS_k}, HFS_k, GDL, TS_{HFS_k}), SK_{MME_j-HFS_k})$.

(d) $MME_j \rightarrow ME_i$:
$(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP), MAC_{MME_j}((ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP), SK_{ME_i-MME_j})$.

The messages can be transformed into the idealized forms as

(a) $ME_i \rightarrow MME_j$:
$AUTH_i = < ID_{G_n}, TID_{ME_i}, R_{G_n-i}, TS_{G_n-i}, HFS_k, LAI_{ME_i}, bP >_{SK_{ME_i-MME_j}} < ID_{G_n}, TID_{ME_i}, R_{G_n-i}, TS_{G_n-i}, HFS_k, LAI_{ME_i}, bP >_{SK_{ME_i-HFS_k}}$

(b) $MME_j \rightarrow HFS_k$:
$< AUTH_i, LAI_{MME_j}, TS_{MME_j}, ID_{MME_j} >_{SK_{MME_j-HFS_k}}$

(c) $HFS_k \rightarrow MME_j$:
$< R_{G_n-i}, ID_{HFS_k}, HFS_k, GDL, TS_{HFS_k} >_{SK_{MME_j-HFS_k}}$

(d) $MME_j \rightarrow ME_i$:
$< ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP >_{SK_{ME_i-MME_j}}$

In this form $TS_{G_n-i}, TS_{MME_j}, TS'_{MME_j}, TS_{HFS_k}$ **are nonces.**

We need to prove that MME_j believes ME_i's long term public key in GDL which it has received from HFS_k and uses the key to compute a long-term secret key ($SK_{ME_i-MME_j}$) between MME_j and ME_i. MME_j uses $SK_{ME_i-MME_j}$ to verify ME_i's message. It then can believe ME_i's session public key, bP. Further, it needs to prove that ME_i can believe MME_j's session public key, aP. Both MME_j and ME_i can use aP and bP to compute a shared session secret, abP. To analyze this protocol, the following assumptions are made.

(1) HFS_k believes $MME_j \overset{SK_{MME_j-HFS_k}}{\longleftrightarrow} HFS_k$.

(2) HFS_k believes $ME_i \overset{SK_{ME_i-HFS_k}}{\longleftrightarrow} HFS_k$.

(3) $\mathrm{MME_j}$ believes $\mathrm{HFS_k} \xleftrightarrow{\mathrm{SK_{MME_j-HFS_k}}} \mathrm{MME_j}$.
(4) $\mathrm{ME_i}$ believes $\mathrm{MME_j} \xleftrightarrow{\mathrm{SK_{ME_i-MME_j}}} \mathrm{ME_i}$.
(5) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{G_n-i}})$.
(6) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{HFS_k}})$.
(7) $\mathrm{HFS_k}$ believes fresh $(\mathrm{TS_{G_n-i}})$.
(8) $\mathrm{HFS_k}$ believes fresh $(\mathrm{TS_{MME_j}})$.
(9) $\mathrm{ME_i}$ believes fresh $(\mathrm{TS'_{MME_j}})$.
(10) $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ control $(\mathrm{AUTH_i}, \mathrm{TS_{MME_j}}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$.
(11) $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ control $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.
(12) $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ controls $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$.
(13) $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ controls $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.
(14) $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ controls $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$.

The steps of the proof are as follows:

a) $\mathrm{HFS_k}$ believes $\mathrm{MME_j} \xleftrightarrow{\mathrm{SK_{MME_j-HFS_k}}} \mathrm{HFS_k}$
and $\mathrm{HFS_k}$ sees $< \mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}} >_{\mathrm{SK_{MME_j-HFS_k}}}$,
then $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ said $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}})$.

b) $\mathrm{HFS_k}$ believes fresh $(\mathrm{TS_{MME_j}})$
and $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ said $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}})$,
then $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ believes $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}})$.

The conjunction can be broken and the result is $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ believes $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$.

c) $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ control $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$
and $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ believes $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$,
then $\mathrm{HFS_k}$ believes $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$.

In steps *a) - c)*, $\mathrm{HFS_k}$ uses a long-term secret key between $\mathrm{MME_j}$ and $\mathrm{HFS_k}$ (*i.e.* $\mathrm{SK_{MME_j-HFS_k}}$) to verify the message $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}})$ received from $\mathrm{MME_j}$. If the verification passes, $\mathrm{HFS_k}$ believes that the message is sent from $\mathrm{MME_j}$.

After $\mathrm{HFS_k}$ believes the message is sent from $\mathrm{MME_j}$, it verifies the authentication message $(< \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP >_{\mathrm{SK_{ME_i-HFS_k}}})$ which is in $\mathrm{AUTH_i}$. If the verification passes, $\mathrm{HFS_k}$ believes that the message is from $\mathrm{ME_i}$. The proof is as follows.

d) $\mathrm{HFS_k}$ believes $\mathrm{ME_i} \xleftrightarrow{\mathrm{SK_{ME_i-HFS_k}}} \mathrm{HFS_k}$ **and** $\mathrm{HFS_k}$ sees $< \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP >_{\mathrm{SK_{ME_i-HFS_k}}})$,
then $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ said $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

e) $\mathrm{HFS_k}$ believes fresh $(\mathrm{TS_{G_n-i}})$ **and** $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ said $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

The conjunction can be broken and the result is $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

f) $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ control $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$
and $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{HFS_k}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

In steps *d) - f)*, $\mathrm{HFS_k}$ verifies message $\mathrm{MAC_i}$ $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$ by computing $\mathrm{MAC'_i}$. The $\mathrm{HFS_k}$ then compares $\mathrm{MAC'_i}$ with the $\mathrm{MAC_i}$. If the verification passes, it is the correct $\mathrm{ME_i}$. Then $\mathrm{HFS_k}$ believes authentication message from $\mathrm{ME_i}$.

After that, $\mathrm{HFS_k}$ sends the authentication message $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}})$ to $\mathrm{MME_j}$.

g) $\mathrm{MME_j}$ believes $\mathrm{HFS_k} \xleftrightarrow{\mathrm{SK_{MME_j-HFS_k}}} \mathrm{MME_j}$
and $\mathrm{MME_j}$ sees $< \mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}} >_{\mathrm{SK_{MME_j-HFS_k}}}$,
then $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ said $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}})$.

h) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{HFS_k}})$
and $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ said $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}})$,
then $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ believes $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}})$.

The conjunction can be broken and the result is $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ believes $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$.

i) $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ controls $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$
and $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ believes $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$,
then $\mathrm{MME_j}$ believes $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$.

In steps *g) – i)*, $\mathrm{MME_j}$ gets message $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, < \mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL} >_{\mathrm{SK_{MME_j-HFS_k}}})$ from $\mathrm{HFS_k}$, and uses a long-term secret key $(\mathrm{SK_{MME_j-HFS_k}})$ between $\mathrm{MME_j}$ and $\mathrm{HFS_k}$ to verify message from $\mathrm{HFS_k}$. If the verification passes, $\mathrm{MME_j}$ believes that the message is from $\mathrm{HFS_k}$.

After that, $\mathrm{MME_j}$ verifies the authentication message $\mathrm{MAC_q}$ from $\mathrm{ME_i}$ as follows.

j) $\mathrm{MME_j}$ believes $\mathrm{ME_i} \overset{\mathrm{SK_{ME_i-MME_j}}}{\longleftrightarrow} \mathrm{MME_j}$
and $\mathrm{MME_j}$ sees $<\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP>_{\mathrm{SK_{ME_i-MME_j}}}$,
then $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ said
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

k) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{G_n-i}})$ **and** $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ said $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

The conjunction can be broken and the result is $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

l) $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ controls
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}), bP)$
and $\mathrm{MME_j}$ believe $\mathrm{ME_i}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{MME_j}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

In steps *j) - l)*, $\mathrm{MME_j}$ verifies message $\mathrm{MAC_q}$ $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$ from $\mathrm{ME_i}$ by using $\mathrm{SK_{ME_i-MME_j}}$ to compute $\mathrm{MAC'_q}$. If the verification passes, it is the correct $\mathrm{ME_i}$. Then $\mathrm{MME_j}$ believes authentication message from $\mathrm{ME_i}$.

After that, $\mathrm{MME_j}$ selects random number a and computes aP and uses bP in $\mathrm{ME_i}$'s message to compute abP. $\mathrm{MME_j}$ now can compute a shared session key $\mathrm{SSK_{MME_j-ME_i}}$ between $\mathrm{MME_j}$ and $\mathrm{ME_i}$. $\mathrm{MME_j}$ then sends the authentication message $\mathrm{MAC_{MME_j}}$ $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP)$ to $\mathrm{ME_i}$.

m) $\mathrm{ME_i}$ believes $\mathrm{MME_j} \overset{\mathrm{SK_{ME_i-MME_j}}}{\longleftrightarrow} \mathrm{ME_i}$ **and** $\mathrm{ME_i}$ sees $<\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP>_{\mathrm{SK_{ME_i-MME_j}}}$,
then $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ said
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP)$.

n) $\mathrm{ME_i}$ believes fresh $(\mathrm{TS'_{MME_j}})$ **and** $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ said $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP)$,
then $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ believes
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP)$.

The conjunction can be broken and the result is $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ believes $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$.

o) $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ controls
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$
and $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ believes
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$,
then $\mathrm{ME_i}$ believes
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$.

In steps *m) - o)*, $\mathrm{ME_i}$ verifies message from $\mathrm{MME_j}$ by using $\mathrm{SK_{ME_i-MME_j}}$ and believes that the message is from $\mathrm{MME_j}$.

$\mathrm{ME_i}$ uses aP in a message to compute abP. $\mathrm{ME_i}$ now can compute a shared session key $\mathrm{SSK_{MME_j-ME_i}}$ between $\mathrm{ME_i}$ and $\mathrm{MME_j}$.

5.2 Authentication Proof for the Remaining MEs

We need to prove that the $\mathrm{MME_j}$ which has believed $\mathrm{ME_i}$'s long-term public key in GDL uses the key to compute a long-term secret key $(\mathrm{SK_{ME_i-MME_j}})$ between $\mathrm{ME_i}$ and $\mathrm{MME_j}$. $\mathrm{MME_j}$ uses $\mathrm{SK_{ME_i-MME_j}}$ to verify $\mathrm{ME_i}$'s message. It then can believe $\mathrm{ME_i}$'s session public key, bP. Further, the proof is that $\mathrm{ME_i}$ can believe $\mathrm{MME_j}$'s session public key, aP. Both $\mathrm{MME_j}$ and $\mathrm{ME_i}$ can use aP and bP to compute a shared session key, abP. To analyze this protocol, the following assumptions are made.

(1) $\mathrm{ME_i}$ believes $\mathrm{MME_j} \overset{\mathrm{SK_{ME_i-MME_j}}}{\longleftrightarrow} \mathrm{ME_i}$.
(2) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{G_n-i}})$.
(3) $\mathrm{ME_i}$ believes fresh $(\mathrm{TS'_{MME_j}})$.
(4) $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ controls $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.
(5) $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ controls $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$.

The steps of the proof are as follows:

a) $\mathrm{MME_j}$ believes $\mathrm{ME_i} \overset{\mathrm{SK_{ME_i-MME_j}}}{\longleftrightarrow} \mathrm{MME_j}$
and $\mathrm{MME_j}$ sees $<\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP>_{\mathrm{SK_{ME_i-MME_j}}}$,
then $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ said
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

b) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{G_n-i}})$ **and** $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ said $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

The conjunction can be broken and the result is $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

c) $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ controls
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}), bP)$
and $\mathrm{MME_j}$ believe $\mathrm{ME_i}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{MME_j}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

In steps *a) – c)*, $\mathrm{MME_j}$ verifies message from $\mathrm{ME_i}$ by using $\mathrm{SK_{ME_i-MME_j}}$.

After that, $\mathrm{MME_j}$ selects random number a and computes aP. It then uses bP in $\mathrm{ME_i}$'s message to compute abP. $\mathrm{MME_j}$ now can compute a shared session key $SSK_{MME_j-ME_i}$ between MME_j and ME_i. MME_j then sends the authentication message MAC_{MME_j} $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP)$ to ME_i.

d) ME_i believes $MME_j \overset{SK_{ME_i-MME_j}}{\longleftrightarrow} ME_i$ **and** ME_i sees $< ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP >_{SK_{ME_i-MME_j}}$,
then ME_i believes MME_j said
$(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP)$.

e) ME_i believes fresh (TS'_{MME_j}) **and** ME_i believes MME_j said $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP)$,
then ME_i believes MME_j believes
$(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP)$.

The conjunction can be broken and the result is ME_i believes MME_j believes $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, aP)$.

f) ME_i believes MME_j controls
$(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, aP)$
and ME_i believes MME_j believes
$(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, aP)$,
then ME_i believes
$(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, aP)$.

In steps *d) - f)*, ME_i verifies message from MME_j by using $SK_{ME_i-MME_j}$ and believes that the message is from MME_j.

ME_i uses aP in a message to compute abP. ME_i now can compute a shared session key $SSK_{MME_j-ME_i}$ between ME_i and MME_j.

6 Security Analysis

In this section, we have analyzed the security of SE-GA as follows.

6.1 Entity Mutual Authentication:

The main goal is to have an authentication between MME and ME in order to create a secure channel for sending data. For the first ME, it will authenticate itself with the home facilitator server (HFS) because the information of ME and the group is at the Home of ME. After ME has confirmed its success, the Home will send ME's group detail list (GDL) to MME. MME trusts ME and the authentication message from ME because MME gets a correct response from ME's Home.

The rest of the group members can authenticate directly with MME because the information of MEs and the group has been sent to MME after the first ME has finished its authentication process.

For example, ME and HFS have a shared key (SK_{ME-HFS}) generated from Diffie-Hellman key exchange in the initialization stage. For authentication of the first ME, ME generates $AUTH_i$ and sends it to the MME. The MME verifies Home of ME from $AUTH_i$ and then forwards $AUTH_i$ to the Home. Home verifies the first ME by function MAC_i which is computed by using a shared key (SK_{ME-HFS}) between ME and Home. For authentication between ME and MME, MME uses the information obtained from ME's Home to generate a key (SK_{MME-ME}) between ME and MME to validate MAC_q. If it is valid, MME trusts ME and sends $AUTH_{MME_j}$ to ME. ME checks the MME by verifying MAC_{MME_j} in $AUTH_{MME_j}$ using the key (SK_{MME-ME}) between ME and MME. If the verification passes, ME believes MME.

For the rest of the group, the mutual authentication between ME and MME is made by using function MAC_q and MAC_{MME_j} which are computed by using a long-term secret key (SK_{MME-ME}).

6.2 Confidentiality

After the authentication process, the key data used for generating the session key (SSK/KGK) between MME and ME is abP computed by using the Diffie-Hellman key exchange. The session key (SSK/KGK) is utilized to encrypt data between ME and MME. Thus, SSK/KGK can provide the data confidentiality.

6.3 Data Integrity

The integrity of messages between ME and MME, and between ME and Home are controlled by MAC function calculated from key SK_{MME-ME}, SK_{HFS-ME}, respectively. These keys are computed by using the Diffie-Hellman key exchange and known only between the two parties. Then every message sent in the protocol has a MAC function to achieve integrity control.

6.4 Enhanced Privacy-Preservation

For the first time when ME registers with the HFS, the ME gets a pair of permanent/temporary identity (PID_{ME}/TID_{ME}) to register in 3GPP networks. In the real case, ME does not send PID_{ME} into the communication network without protection because PID_{ME} is ME's privacy which may cause harm if it is sniffed. In SE-GA protocol, ME can send TID_{ME} into the communication network to the other party with MAC and the party can verify TID_{ME} by MAC function. In addition, in the case that the network needs ME to send PID_{ME} to the home network, the PID_{ME} may be encrypted with the long-term secret key between ME and HFS.

6.5 Secure Key Derivation

In the SE-GA protocol, SSK_{MME-ME} is created from a function which uses a shared secret between MME and

ME. As described in Section 5, MME and ME send a session public key of their own (aP/bP) to compute a shared secret abP between them. This abP is computed by making use of Diffie-Hellman key exchange which is secure. After that, both MME and ME use abP to generate SSK_{MME-ME}.

6.6 Key Forward/Backward Secrecy (KFS/KBS)

In the SE-GA protocol, the session public keys (aP/bP), which are used to compute session key, are sent between MME and ME, while the long-term secret SK_{ME-MME} is calculated from a long-term public/private keys of MME and ME respectively. Then, the session public keys are not related to the calculation of the SK_{ME-MME}. In addition, the SSK key value between ME and MME is very difficult to attack. Because this value is based on abP and known only between ME and MME, then the KFS/KBS can be achieved.

6.7 Group Key Forward/Backward Secrecy (GKFS/GKBS)

When group members join or leave the group, the group key needs to update in order to preserve backward and forward secrecy. Up to now, several protocols have been proposed for dynamic group key agreement, such as Pipat [5] and Zhu [21]. After updating the group key, the group will send a group's information such as the public keys of new members/leaving members, group members' numbers to each member's Home. Then the member who has joined or left cannot know any information before joining or after leaving.

6.8 Resistance to Replay Attack

While MME and ME are communicating, authentication messages are sent with timestamps and random numbers, thus preventing replay attacks. For example of case 1, between MME and ME, there is a chance of replay attack, so while ME sending a message to MME in Step 1 to request services, a timestamp (TS_{G_n-i}) is included into the message. Similarly, when MME responds to ME in Step 4, a timestamp (TS'_{MME}) is attached to the message to prevent replay attack.

6.9 Resistance to Redirection Attack

Because the authentication message ($AUTH_i$) from ME included with LAI_{ME}, MAC_q and MAC_i. The LAI_{ME} indicates the BS which ME contacts at that time. If the MME forwards $AUTH_i$ to HFS, then the HFS uses LAI_{MME_j} to compare with LAI_{ME}. In the case $LAI_{ME} = LAI_{MME_j}$, the HFS computes MAC'_i and compares with MAC_i in Step (3) of authentication for the first ME. If $MAC'_i = MAC_i$ then HFS accepts the authentication. It rejects the authentication if the verification of MAC_i fails. For the remaining ME, the MME uses LAI_{MME_j} getting from the BS to compare with LAI_{ME} embedded in $AUTH_i$. If LAI_{ME} has the same value as LAI_{MME_j} then MME verify MAC_q with MAC'_q. Thus, SE-GA protocol can prevent the redirection attack.

6.10 Resistance to Man-in-the-Middle Attack

During the first confirmation of ME, an attacker may disguise as MME to sniff the information. Then the attacker disguises as the ME and sends the information to the real MME. As the attacker does not know the value b, he/she may try to perform man-in-the-middle attack by replacing bP with b_1P. However, it cannot fool the Home because the attacker does not know the secret key SK_{ME-HFS} which is utilized to compute MAC between ME and its Home.

In the case of the remaining ME, the secret key (SK_{ME-MME}) is utilized to protect messages between ME and MME. If an attacker changes messages, the MME can know messages which are not sent from the real ME. Thus, the protocol can prevent a man-in-the-middle attack.

6.11 Resistance to DoS Attack

While performing the authentication process, a malicious ME can run DoS attack either on HFS or MME. If a malicious ME forges the message, HFS or MME can detect the forged message by checking TS and comparing LAI in the message from the ME with LAI from MME.

6.12 Resistance to Impersonate Attack

The SE-GA protocol makes use of each ME's long-term private and public keys to achieve secure authentication between ME and MME. It is very difficult for an ME to disguise itself as another ME.

Table 6 shows the comparison of security and flexibility based on an actual usage in some group authentication protocols. By the comparison, we see that SE-GA is better than other protocols.
AK: Authentication key; RMA: resistance to man-in-the-middle-attack; RRA: resistance to redirection attack; GMD: group members can come from the different home networks; GMS: Group members can use different networks simultaneously; GDO: group members disguised as others.

* The first ME uses a pre-shared key which it got from the Home in the initial stage to authenticate with the Home in order to use the network service, while the remaining MEs use mainly the group key to authenticate with the MME.

Table 6: Comparisons of the proposed protocol with some schemes

Protocol / Features	SE-AKA	GLARM	SE-GA
AK	Symmetric Keys & Group Key*	Symmetric Keys**	Diffie - Hellman***
RMA	Yes	Yes	Yes
RRA	Yes	Yes	Yes
GMD	No	No	Yes
GMS	No	No	Yes
GDO	Yes	No	No

** Each ME uses the symmetric key defined by its Home when it first registered with the Home in order to authenticate itself with the service network.

*** The key used in the authentication process can be created on the fly between the two parties by making use of the Diffie-Hellman key exchange.

7 Conclusions

In this work, we have developed the SE-GA protocol that assists group authentication on LTE networks. The authentication protocol uses the long-term private keys and public keys between parties to create shared secret keys used in the authentication process. By using this technique, SE-GA can be flexible and scalable. It helps the group members to be able to work simultaneously on different LTE networks. In addition, group members can be from different Homes. In the protocol, the authentication process is divided into two steps, the authentication of the first machine which tries to connect to a service network and that of the remaining machines. The first machine needs to authenticate itself with its Home, while the remaining machines can authenticate with the service MME. This reduces the providers' network traffic as well as network delays.

During the initialization of SE-GA, the network will be a little crowded because each group member has to send group information to its Home. However, during the authentication of the group members excluding the first one, SE-GA needs only three steps for the authentication of each member while the former SE-AKA needs at least four steps.

In this paper, we provided an authentication proof by using the well-known BAN logic. Security analysis of the proposed protocol is also given and a comparison of our protocol with SE-AKA and GLARM was demonstrated. According to the comparison, we can see that the proposed protocol outperforms the former ones.

References

[1] S. B. Babu and P. Venkataram, "A dynamic authentication scheme for mobile transactions," *International Journal of Network Security*, vol. 8, no. 1, pp. 59–74, 2009.

[2] M. Burrows, M. Abadi, and R. Need ham, "A logic of authentication," *ACM Transactions on Computer Systems*, vol. 8, no. 1, pp. 18–36, 1990.

[3] J. Cao, M. Ma, and H. Li, "A group-based authentication and key agreement for mtc in lte networks," *IEEE Globecom (Globecom'12)*, pp. 1017–1022, 2012.

[4] Y. W. Chen, J. T. Wang, K. H. Chi, and C.-C. Tseng, "Group-based authentication and key agreement," *Wireless Pers Commun*, vol. 62, no. 4, pp. 965–979, 2012.

[5] P. Hiranvanichakorn, "Provably authenticated group key agreement based on braid groups - The dynamic case," *International Journal of Network Security*, vol. 19, no. 4, pp. 517–527, 2017.

[6] M. S. Hwang, S. K. Chong, and H. H. Ou, "On the security of an enhanced umts authentication and key agreement protocol," *European Transactions on Telecommunications*, vol. 22, no. 3, pp. 99–112, 2011.

[7] M. S. Hwang, C. C. Lee, and W. P. Yang, "An improvement of mobile users authentication in the integration environments," *International Journal of Electronics and Communications*, vol. 56, no. 5, pp. 293–297, 2002.

[8] C. Lai, H. Li, R. Lu, and X. Shen, "Se-aka: A secure and efficient group authentication and key agreement protocol for lte networks," *Computer Networks*, vol. 57, pp. 3492–3510, 2013.

[9] C. Lai, R. Lu, D. Zheng, H. Li, and X. Shen, "Glarm: Group-based light weight authentication scheme for resource-constrained machine to machine communications," *Computer Networks*, vol. 99, pp. 66–81, 2016.

[10] C. C. Lee, M. S. Hwang, and L. H. Li, "A new key authentication scheme based on discrete logarithms," *Applied Mathematics and Computation*, vol. 139, no. 2, pp. 343–349, 2003.

[11] C. C. Lee, M. S. Hwang, and I. E. Liao, "A new authentication protocol based on pointer forwarding for mobile communications," *Wireless Communications and Mobile Computing*, vol. 8, no. 5, pp. 661–672, 2008.

[12] C. C. Lee, M. S. Hwang, and W. P. Yang, "Extension of authentication protocol for gsm," *IEEE Proceedings – Communications*, vol. 150, no. 2, pp. 91–95, 2003.

[13] C. C. Lee, I. E. Liao, and M. S. Hwang, "An efficient authentication protocol for mobile communications," *Telecommunication Systems*, vol. 46, no. 1, pp. 31–41, 2011.

[14] Y. Liu, C. Cheng, J. Cao, and T. Jiang, "An improved authenticated group key transfer protocol based on secret sharing," *IEEE Transactions on Computers*, vol. 62, no. 11, pp. 2335–2336, 2013.

[15] Y. Liu, C. Cheng, T. Gu, T. Jiang, and X. Li, "A lightweight authenticated communication scheme for smart grid," *IEEE Sensors Journal*, vol. 16, no. 3, pp. 836–842, 2016.

[16] H. H. Ou, M. S. Hwang, and J. K. Jan, "A cocktail protocol with the authentication and key agreement on the umts," *Journal of Systems and Software*, vol. 83, no. 2, pp. 316–325, 2010.

[17] H. H. Ou, I. C. Lin, M. S. Hwang, and J. K. Jan, "Tk-aka: Using temporary key on authentication and key agreement protocol on umts," *International Journal of Network Management*, vol. 19, no. 4, pp. 291–303, 2009.

[18] S. F. Tzeng, M. S. Hwang, "Digital signature with message recovery and its variants based on elliptic curve discrete logarithm problem", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 61–71, Mar. 2004.

[19] F. Wang, C. Chang, and Y. Chou, "Group authentication and group key distribution for ad hoc networks," *International Journal of Network Security*, vol. 17, no. 2, pp. 199–207, 2015.

[20] S. Wu, Y. Zhu, and Q. Pu, "Security analysis of a cocktail protocol with the authentication and key agreement on the umts," *IEEE Communications Letters*, vol. 14, no. 4, pp. 366–368, 2010.

[21] H. F. Zhu, "Secure chaotic maps-based group key agreement scheme with privacy preserving," *International Journal of Network Security*, vol. 18, no. 6, pp. 1001–1009, 2016.

Biography

Boriphat Kijjabuncha received the B.S. degree from Silpakorn University, Thailand, in 2000 and M.S. degree in Applied Information System at School of Applied Statistics, National Institute of Development Administration (NIDA), Thailand. He has worked as a lecturer of Information and Communication Technology Department at Silpakorn University, Thailand since 2006. His research interest is in information security.

Pipat Hiranvanichakorn received the B.E. degree in Electrical Engineering from Chulalongkorn University, Thailand, in 1977 and the M.E. and D.E. degrees in Information Processing from Tokyo Institute of Technology, Japan, in 1982 and 1985, respectively. He has worked as an associate professor of Computer Science at School of Applied Statistics, National Institute of Development Administration, Thailand until 2016. His current research interests include natural language processing, computer networks, cryptography and information security.