IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 9, 2017, accepted December 21, 2017, date of publication January 5, 2018,
date of current version February 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2017.2788919

SEGB: Security Enhanced Group Based AKA Protocol for M2M Communication in an IoT Enabled LTE/LTE-A Network

BALU L. PARNE[1], (Student Member, IEEE), SHUBHAM GUPTA[1], (Student Member, IEEE), AND NARENDRA S. CHAUDHARI[1,2], (Senior Member, IEEE)

[1]Department of Computer Science and Engineering, Visvesvaraya National Institute of Technology, Nagpur 440010, India
[2]Department of Computer Science and Engineering, Indian Institute of Technology (IIT), Indore-453552, India

Corresponding author: Balu L. Parne (balu.parne@students.vnit.ac.in)

ABSTRACT Nowadays machine to machine (M2M) communication and its applications are growing tremendously around the globe as millions of devices are communicating with each other in an Internet of Things (IoT)-enabled long term evolution (LTE)/LTE-advanced (LTE-A) network. These applications are effective and secure only after the successful verification of machine type communication devices (MTCDs). Hence, various group-based authentication and key agreement (AKA) protocols were proposed in the literature to achieve the authentication. These protocols fulfill all the security requirements such as privacy preservation, mutual authentication, integrity, and confidentiality. But, none of them have the credential to overcome the single key problem in the communication network. In addition, they do not have the efficacy to maintain the group key unlink-ability and are susceptible to the identified attacks. In some of the protocols, each MTCD needs to authenticate independently to simultaneously access the communication network that generates network congestion overhead. In view of these problems, we propose the security enhanced group-based (SEGB) AKA protocol for M2M communication in an IoT-enabled LTE/LTE-A network. The SEGB-AKA protocol solves the problem of the single key during the authentication process and achieves the key forward/backward secrecy. The protocol overcomes the problem of signaling congestion and high bandwidth consumption. The formal security analysis of the protocol is carried out by the automated Internet security protocols and applications tool. The security analysis shows that the protocol achieves the security goals and is free from various known attacks. Moreover, the performance of the proposed SEGB-AKA protocol is analyzed with the existing group-based AKA protocols. The analysis shows that the protocol has better results in terms of network overheads and fulfills all the security requirements of M2M communication.

INDEX TERMS AVISPA, group authentication, Internet of Things, LTE/LTE-A, M2M communication.

I. INTRODUCTION

There has been a tremendous growth in wireless telecommunication technology during the last two decades. The main objective of the telecommunication technology is to provide ubiquitous connectivity among the people on the planet. The next big achievement for the communication technology would be not only people but also objects that can communicate with each other using the wireless communication network [1]. The Internet of Things (IoT) is one of the most intellectual mechanism where billions of objects are connected by upcoming wired and wireless technologies to control various things in the environment [2]. These objects are the devices and users having the ability to transmit information through the network channel for device to device, person to person and person to device communication in the IoT network. Machine to machine (M2M) communication also known as machine type communication (MTC) is an emerging communication standard that supports the extensive connectivity between MTC Devices (MTCDs) with an ability to communicate independently without human interference [3]. It is an inspiring and innovative feature of the next generation telecommunication networks allowing traffic through any network infrastructure. M2M communications is an important aspect of practical realization of the IoT [4]. It has a number of applications in various fields such as cloud based system, health-care monitoring system, intelligent tracing and tracking system, smart transportation, smart cities, and smart electricity grids [5]–[7]. As per the report of

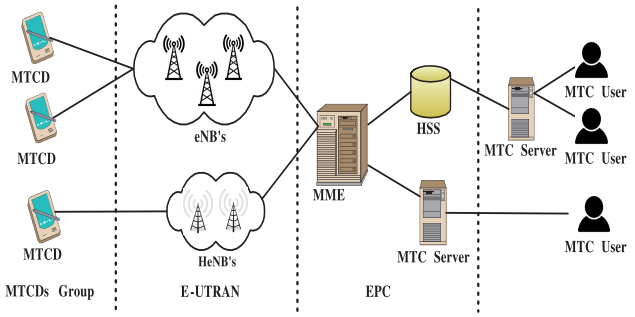

FIGURE 1. The architecture of M2M commuincation in 3GPP network.

ABI researchers, there will be more than 30 billion devices connected through wireless network by the end of 2020 [8].

The Long Term Evolution-Advanced (LTE-A) networks is evolved with several objectives that enable the fourth generation (4G) heterogeneous network with high resource capacity, low latency, flexible bandwidth, low cost at the customer end, good coverage across a wide area and good quality of services [9]. Hence, it serves as the most suitable platform for the M2M communication vis-a-vis other wireless technologies. The communication scenario and security requirements are mentioned by the 3GPP committee for M2M communication in Release-11 [10]. Security and privacy are the main challenges to establish M2M communication for the mass MTCDs in the IoT enabled network [11]. Moreover, the MTCDs such as mobile phones and smart devices have limited computing resources. If MTCDs fail to authenticate and securely access the communication network, the MTC based applications cannot be universally accepted. Hence, it is required to propose a security enhanced authentication and key agreement (AKA) protocol for M2M communication. The protocol aims to provide the mutual authentication to mitigate security attacks by gaining control over MTCDs in an IoT enabled network.

The 3GPP committee introduced the MTC architecture in LTE/LTE-A network to obtain the authentication between communication entities as shown in Fig. 1. The Mobile Management Entity (MME) and Home Subscriber Server (HSS) are communication entities in the network operator domain [12]. The architecture also consists of the MTCDs, MTC users, and servers. The user is a control center unit outside the network domain. To operate one or more MTCDs, the legitimate MTC user can access the service provided by one or more MTC servers. The MTCDs can communicate with the MTC server through the LTE/LTE-A network. The MTC server is connected to the network and it can be placed within the network or outside the network domain. The MTC user can access the MTC server with an application program interface (API). The MTCDs communicates with the MTC server and are controlled by the MTC user via MTC servers. To enable the secure communication between MTCDs and MTC server, it is essential to authenticate the MTCDs by LTE/LTE-A network.

For secure communication between MTCDs and server, the MTCDs follow the evolved packet system (EPS-AKA) protocol for 3GPP network [13] and extensible authentication protocol (EAP-AKA) for non-3GPP network (WLAN/WiMAX) [14]. To ensure the M2M communications, large number of MTCDs are involved in the LTE/LTE-A network. Each device needs to execute the entire AKA process that increases the computation overhead at the HSS and signaling congestion overhead in the communication network [15], [16]. Moreover, the conventional protocols suffer from the security issues such as identity protection, impersonation, and denial of service (DoS) attack [17]. In addition, security of the protocol rely on shared symmetric key between the communication entities. If an adversary compromised this key, he/she can successfully retrieve all other keys and probably an attacker would be authenticated by the network [18]. Further, it is noticed that the existing AKA protocols are not convenient for group based communication and are susceptible to the above mentioned attacks. In addition, the protocols fail to establish the key forward/backward secrecy (KFS/KBS) whenever a new MTCD joins or retires from the group. Hence, it is required to propose a security enhanced group based AKA protocol for M2M communication in an IoT enabled LTE/LTE-A network.

A. CONTRIBUTION AND APPROACH

To overcome the above mentioned problems, our concern is to focus on the existing security and privacy issues present in the group based AKA protocols for M2M communication. In this paper, we propose the security enhanced group based (SEGB-AKA) protocol for M2M communication in an IoT enabled LTE/LTE-A network. The main contributions of the paper are as follows:

1) The proposed protocol follows the 3GPP standard and incorporates the group authentication mechanism that verifies the group of MTCDs simultaneously by using a symmetric key cryptosystem and aggregate message authentication code (MAC).
2) We introduce a mechanism to solve the single key problem by securing the symmetric shared key between the MTCD and HSS and preserves the privacy of the MTCDs during the AKA process.
3) The mechanism is designed to maintain the unlinkability and traceability in MTCDs of a group that ensure the KFS/KBS for mass MTCDs. Also, a session key is implanted between the MTCDs and group leader for secure communication between them.
4) A formal security analysis of the protocol is carried out by the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool. The analysis shows that the protocol accomplishes all the security goals and defeats the identified attacks.
5) The performance of the proposed SEGB-AKA protocol is analyzed with the existing protocols. The protocol incurs less storage and communication overhead compared to the existing group based AKA protocols. The

protocol fulfills all the security requirements with competitive computation and transmission overhead.

B. ORGANIZATION OF THE PAPER

The subsequent sections of the paper deal with the following aspects. Section II describes the related research work of the group based AKA protocols. The proposed SEGB-AKA protocol for M2M communication in an IoT enabled LTE/LTE-A network is presented in section III. Section IV illustrates the formal analysis of the proposed protocol using the AVISPA tool. The security analysis of the protocol in terms of security goals, key privacy properties and resistance against different possible attacks is presented in section V. The performance analysis of the proposed protocol with respect to existing group based AKA protocol is illustrated in section VI. Finally, section VII summarizes the conclusion and future work.

II. RELATED WORK

In view of the above identified problems, several group based AKA protocols were proposed for M2M communication in the LTE/LTE-A network. In order to strengthen the security and reduce the network overhead, several symmetric, asymmetric and hybrid key cryptosystem based AKA protocols have been addressed in the 3GPP network. Therefore, a brief overview of the existing group based AKA protocols is presented in this section.

To avoid the signaling congestion problem from M2M communication, Jung *et al.* [19] proposed a congestion avoidance algorithm. The authors form a group within a local communication area and select a group leader to transmit or receive information from each MTCD. Chen *et al.* [20] followed the grouping method to propose the G-AKA protocol. In this protocol, once the first device is authenticated by the HSS, the same entity authorizes the MME to authenticate the remaining devices of the group. Thus, the authentication process can be simplified for all the remaining devices in the group. However, the protocol generates the high signaling congestion overhead when the mass MTCDs require to access the network simultaneously. It is susceptible to potential attacks such as MiTM, DoS and redirection attack. To enhance the data integrity and confidentiality in AKA protocols, the asymmetric key based SE-AKA protocol is proposed by Lai *et al.* [21]. The authors improved the security of the protocol but it suffers from the network signaling congestion. Similarly, Jiang *et al.* [22] proposed EG-AKA protocol to authenticate the group of MTCDs in non-3GPP network. The protocol measures the high computation overload at network due to asymmetric key operations and suffers from security attacks. Further, Lai *et al.* [23] proposed the symmetric key based NOVEL-AKA protocol in which the first MTCD carries out a full authentication with the HSS. To authenticate the remaining MTCDs, HSS calculates the group temporary key (GTK), index table, authentication data response and transmit to the MME. Hence, the remaining MTCDs of the group are validated by the MME without involving the HSS. Unfortunately, the protocol did not address the group authentication and suffered from the various security attacks.

To overcome the problems of above mentioned schemes, Choi *et al.* [24] recommended the GROUP-AKA protocol that mitigates the problem of signaling congestion as it successfully authenticates a group of devices simultaneously. Moreover, the protocol maintains the unlink-ability in the group key whenever the MTCD joins or retires from the group. However, the protocol does not preserve the privacy of MTCDs and suffers from identity catching attack while authenticating a new MTCD in the group. To mitigate the problems of GROUP-AKA protocol, Cao *et al.* [25] suggested a group signature based GBAAM-AKA protocol. In this protocol, the group leader computes the aggregated signature and sends it to the MME. Then MME verifies the received aggregate signature and mutually authenticates each MTCD by sending access response message. The protocol successfully avoids the DoS attack but suffers from the privacy preservation problem. Moreover, this protocol generates high computation overhead due to asymmetric cryptosystem based operations. To overcome the issue of privacy preservation, Fu *et al.* [26] proposed the privacy preserving group authentication (PRIVACY-AKA) protocol. The protocol generates the pseudo identity by elliptic curve cryptography. Initially, each MTCD transfers its message authentication code to the group leader. Then, group leader compiles each code into aggregate MAC and transfers to the HSS. HSS authenticates the MME and forwards the group authentication vectors (GAVs) to validate the MTCDs in the group. However, the protocol protect the network from all potential attacks but generates high computational overhead due to asymmetric key cryptosystem. As a matter of fact, this protocol does not consider the group key secrecy and KFS/KBS.

To minimize the network signaling overhead, Lai *et al.* [27] proposed a group based lightweight authentication (GLARM-AKA) protocol for resource constrained MTCDs. The protocol follows an aggregate signature based approach on MACs and simultaneously validates a group of MTCDs. The protocol exhibits less communication and computation overhead compared to above explained AKA protocols. It fails to maintain the unlink-ability in a group key when MTCD joins/leaves the group. Moreover, the protocol suffers from privacy preservation and impersonation attack. Li *et al.* [28] proposed the group based (GR-AKA) protocol with dynamic policy updating in LTE-A network. The protocol preserves the identity of MTCDs and generates the authentication message by Lagrange Component (LC). This protocol illustrates a mechanism to update the group key and avoid various attacks. However, the protocol generates the high bandwidth consumption due to complex and time consuming cryptographic operations. Hence, it might be difficult for 3GPP committee to accept this expensive framework for resource constrained devices in group based AKA protocols. Recently, Yao *et al.* [29] proposed the group based secure authentication (GBS-AKA) protocol to overcome the attacks and problem of high bandwidth consumption. The protocol incurs

less communication overhead but fails to preserve the privacy of MTCDs and suffers from the impersonation and DoS attacks. Further, the KFS/KBS is not considered for secure group key management and fails to maintain the unlinkability in the group key.

The security of the above symmetric key cryptosystem based protocol depends on the confidentiality of the shared symmetric key between the communication entities. Once this key is negotiated, each key can be recovered by an intruder. None of the existing protocols attempt to protect the pre-shared secret key. Moreover, many of the aforementioned group AKA protocols suffer from the various security problems. The existing group based AKA protocols for M2M communication generates high network overhead. All these protocols fails to maintain the KFS/KBS whenever a new MTCD joins or leaves the group. In view of these problems, we propose the SEGB-AKA protocol for M2M communication in LTE/LTE-A network. In the proposed protocol, we introduce the mechanism to secure a pre-shared symmetric key between the MTCD and HSS. Similarly, the protocol provides a secure group authentication mechanism that authenticates the group of MTCDs simultaneously. The protocol preserves the privacy of the MTCDs and achieves the secrecy of a group key whenever the MTCD joins or retires from the group. The proposed SEGB-AKA protocol overcome the security problems of the network and generates less overhead compared to the existing group based AKA protocols. It accomplishes all the security requirements for M2M communication with competitive transmission and computational overhead.

III. THE PROPOSED SEGB-AKA PROTOCOL

This section illustrates the proposed SEGB-AKA protocol for M2M communication in an IoT enabled LTE/LTE-A network. The protocol executes a four stage mechanism: i) Group initialization and key establishment stage, ii) Authentication and key agreement stage, iii) Session key compliance stage and iv) MTCD join and leave event stage. The standardized notations and symbols of the group based AKA protocols are shown in Table 1.

A. BASIC ASSUMPTIONS

Before illustrating the proposed protocol, we define the basic assumptions of the protocol. We consider the conventional MTC application scenario such as vehicle to everything (V2X) technology that allows vehicles to communicate with moving parts of the traffic system around them. LTE-V2X linked with multi-access edge computing (MEC) provides a feasible and effective solution for different V2X applications. The V2X based on the LTE/LTE-A technology assists the vehicle to discover what is around the corner; detect a pedestrian or a car approaching to an intersection even when an object is obstructed by buildings or large vehicles. The evolution of V2X technology supports autonomous vehicles, high throughput sensor data/map sharing among vehicles and improves the positioning by maintaining backward compatibility [30], [31]. For supporting the application of V2X technology, there are mass MTCDs that send their information to the server. In this scenario, MTCDs are the reporting devices installed in the smart vehicles that are located in dense areas. Some pre-shared parameters are considered at the time of vehicle subscription by storing at the universal integrated circuit card (UICC) of authentication center. The parameters that are shared at the time of subscription are defined as follows:

- f_1' is a supplementary cryptographic one way key derivation function. It is strictly known to MTCD and HSS only.
- As defined by the 3GPP, $K_{GRP_{1-i}}$ is the shared secret key between MTCDs ($MTCD_{GRP1-i}$) and HSS.
- We introduced a unique key identifier (KID_i) that points to the key $K_{GRP_{1-i}}$. The KID_i is updated after each successful AKA process.
- E_c and X are newly introduced pre-selected encryption functions based on ciphering algorithm supported by the MTCDs and HSS.
- For each group, the key generation center (KGC) establish the secure group key $GRPK_i$ and unique group identity (ID_{GRP_i}) of a group GRP_i.
- To protect the transmitted messages over the communication channel, the path between MME and HSS is assumed to be secure on the basis of diameter protocol [32].

TABLE 1. Notations and their interpretation.

Notation/Acronym	Interpretation	Size (in bits)
$IMSI$	International Mobile Subscriber Identity	128
PID	Permanent Identity of MTCD	128
LAI	Location Area Identity	40
ID_{GRP_i}	Group Identity	128
AMF	Authentication Management Field	48
MAC	Message Authentication Code	64
SN_{ID}	Serving Network identity	128
SQN	Sequence Number	48
TS	Timestamp	64
$AUTN$	Authentication Token	Variable
$AV/GMAV$	Authentication Vector	Variable
$AUTH$	Authentication of individual device	128
$RAND/PR$	Random Number	128
CK/IK	Cipher/Integrity Key	128
K_{ASME}	Access Security Management Entity Key	256
KSI_{ASME}	Key Set Identifier of K_{ASME}	3
LMK	Local Master Key	256
$GRPK_i$	Group key	128
K_{GRP1-i}	Shared secret key	128
KID_i	Key identifier of K_{GRP1-i}	128
$SSDK$	Secure secret dynamic key	128
MSK	Master Session Key	128
LC	Lagrange Components	128
$ECDH$	Elliptic Curve Diffie Hellman Key	192
CV	Confirmation Value	256
IV	Initialization Vector	128
$ECDS$	Elliptic Curve Digital Signature	448
$RES/XRES$	Response/ Expected Response	64

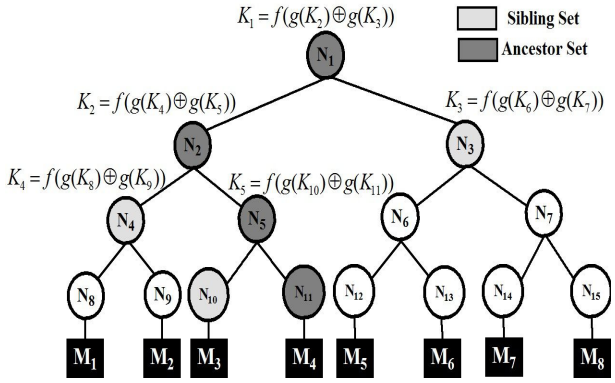

FIGURE 2. Initialization of MTCDs in binary tree.

B. GROUP INITIALIZATION AND KEY ESTABLISHMENT STAGE

In the group initialization phase, the mass MTCDs form a group and ID_{GRP_i} will be encapsulated with each MTCD. The network service provider form a group of the MTCDs involved in the authentication process. The group is formed on the basis of certain perceptions on the MTCDs such as same local communication area, services and features [33]. The device with high communication capability, storage capacity and long last battery backup will be selected as the group leader of the MTCDs in a group [19]. The group membership can be changed at any time when a new MTCD is added or removed by the service provider. The HSS and the MTCDs keeps the same framework of the binary tree for MTCDs as shown in Fig.2. The group initialization executes the three steps such as: i) Assign K_{GRP1-i} to each $MTCD_{GRP1-i}$, ii) The binary tree is created by the service provider and allocate MTCDs to the leaf nodes, and iii) Finally, computes the group key $GRPK_i$ and shares between the MTCDs and the HSS.

In the proposed group key management scheme, there are exactly two children for each interior node of the binary tree. The MTCDs are associated with the leaf node and the key value computed at the root node is the common group key ($GRPK_i$). This group key is used by each MTCD in the group to provide privacy protection and mutual authentication between MTCDs and service provider. All the interior node N_i in the binary tree computes the node secret value K_i as :

$$K_i = f(g(K_{left(i)}) \oplus g(K_{right(i)})) \tag{1}$$

where $left(i)$ and $right(i)$ denote the left and right children of a node N_i respectively. The function g is a one way function, f is a mixing function and $\oplus$ is bitwise exclusive-OR.

The security and privacy of the proposed group key management scheme is established on the fact that ''Each MTCD have knowledge about the node/key secrets on the path from its associated node to the root node (therefore the key values of the nodes along this path) and also the blinded node secrets that are siblings to this path'' [34].

Ancestor of a node are the nodes in the path from its parent to the root. Besides, the ancestor set is the set of ancestor of the node and the sibling set is set of siblings of the nodes in the ancestor set. For instance, Fig.2 shows the ancestor set and sibling set of node N_{11} (MTCD M_4). Each MTCD of a group maintains the key value of the associated leaf node, and a list of blinded node secrets for all the siblings of the nodes along the path from that node to the root. Using this information, the MTCD can compute the node/key secrets along its path to the root node including the root key. For instance, in Fig.2, MTCD M_4 knows key K_{11} and its sibling's blinded key K_{10}^B, K_4^B and K_3^B (blinded keys in M_4's sibling set). Using this, M_4 can compute all keys in its ancestor set (K_5, K_2, and K_1 i.e. group key ($GRPK_i$)). This approach maintains the security of the group key. To secure the $K_{GRP_{1-i}}$ and preserve the privacy of an International Mobile Subscriber Identity (IMSI) of each MTCD of a group GRP_1, we introduce a new key identifier (KID_i) which will be uniquely assigned to $K_{GRP_{1-i}}$ of $MTCD_{GRP_{1-i}}$ and points to the $K_{GRP_{1-i}}$ in the UICC of HSS. The KID_i is used to compute a new secure secret dynamic key ($SSDK$) as follows:

$$SSDK_i = f_1'(KID_i)_{K_{GRP_{1-i}}} \tag{2}$$

C. AUTHENTICATION AND KEY AGREEMENT STAGE

In this phase, the MTCDs and the HSS carry out the mutual authentication through MME. The session keys are established between MTCDs and HSS for secure transmission of messages. In the proposed protocol, the security of the conventional mechanism is enhanced and maintain the security services provided by the system. The messages transmitted in the AKA process of the proposed SEGB-AKA protocol are shown in Fig.3 and details of the steps to be executed is illustrated as follows:

Step-1: The MTCDs, $MTCD_{GRP_{1-1}}, MTCD_{GRP_{1-2}}, \ldots, MTCD_{GRP1-n}$ of the group GRP_1 sends the access request message to the MME through corresponding group leader ($GRP_{1-leader}$).

Step-2: The MME transmits the request identity message to the $GRP_{1-leader}$ to get the identity of $MTCD_{GRP_{1-i}}$.

Step-3: The $GRP_{1-leader}$ generates the identity response message $AUTH_{GRP_1}$ as:

- Each MTCD generates a fresh $SSDK_i$ using (2).
- Each device of a group calculates the $MAC_{MTCD_{GRP_{1-i}}}$, $i = 1, 2, ..., n$. To prevent from replay attack, a time-stamp TS_{GRP_1} is concatenated with it. $MAC_{MTCD_{GRP_{1-i}}} = f_1(TS_{GRP_1}||IMSI_{GRP_{1-i}})_{K_{GRP_{1-i}}}$
- Each device generates its own authentication message $M_{MTCD_{GRP1-i}}$ as $M_{MTCD_{GRP1-i}} = (E(ID_{GRP_1}||IMSI_{GRP_{1-i}} ||LAI)_{SSDK_i}||TS_{GRP_1}||KID_i)$
- Later, the MTCDs of a group GRP_1 forwards their $MAC_{MTCD_{GRP_{1-i}}}$ and $M_{MTCD_{GRP_{1-i}}}$ to the $GRP_{1-leader}$.
- The $GRP_{1-leader}$ calculates the aggregate message authentication code MAC_{GRP_1} and also generates authentication response $AUTH_{GRP_1}$ message as: $MAC_{GRP_1} = f_1(MAC_{MTCD_{GRP_{1-1}}} \oplus MAC_{MTCD_{GRP_{1-2}}} \oplus$

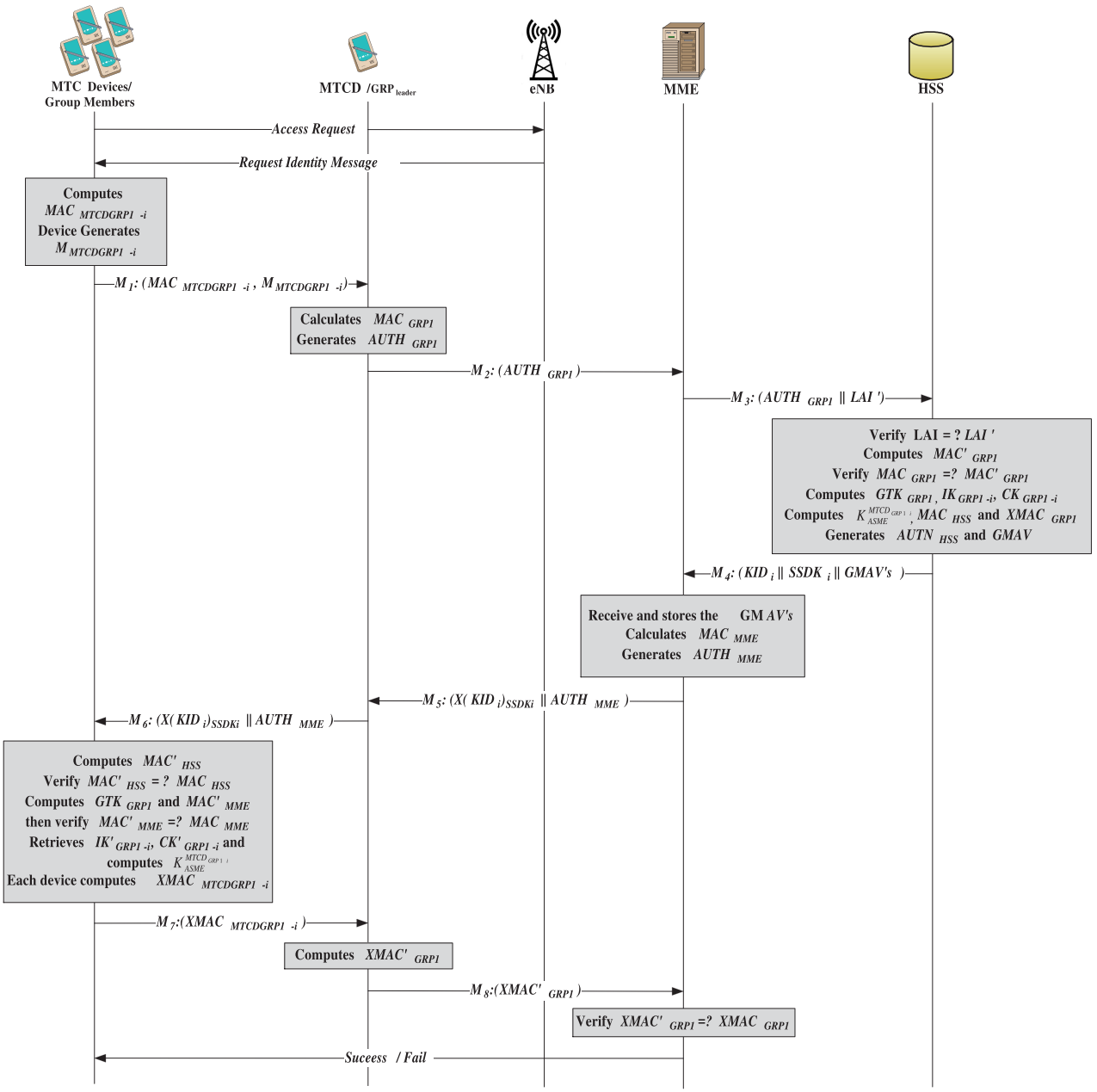

FIGURE 3. The proposed SEGB-AKA protocol.

$$\ldots \oplus MAC_{MTCD_{GRP_{1-n}}})_{GRPK_1}.$$

$$AUTH_{GRP_1} = (M_{MTCD_{GRP_{1-1}}} || M_{MTCD_{GRP_{1-2}}} || \ldots || M_{MTCD_{GRP_{1-n}}} || MAC_{GRP_1} || TS_{GRP_1}).$$

Finally, the $GRP_{1-leader}$ transmits $AUTH_{GRP1}$ to the MME.

Step-4: MME concatenates the LAI' with $AUTH_{GRP_1}$ and forwards the authentication request $(AUTH_{GRP_1}||LAI')$ to the HSS.

Step-5: After receiving the authentication data request from MME, HSS verifies the authentication request message as follows:

- First, it computes the $(TS_{GRP_1} - TS_{HSS})$ and verifies whether it exceeds the threshold $\triangle T$. If it exceeds, there is a possibility that $AUTH_{GRP_1}$ can be a replayed message and HSS declines the authentication request of the MME.
- Using KID_i assigned to $K_{GRP_{1-i}}$, HSS retrieves the respective key $(K_{GRP_{1-i}})$ and computes the $SSDK_i$ as shown in (2).
- Later, HSS decrypts $E(ID_{GRP_1}||IMSI_{GRP_{1-i}}||LAI)_{SSDK_i}$ that provides access to the ID_{GRP_1}, $IMSI_{GRP_{1-i}}$ and LAI.
- HSS verifies whether $LAI' = ?LAI$. If equality holds, HSS authenticates LAI' received from MME. Otherwise,

forwards an authentication failure message to the MME.

- After retrieving ID_{GRP_1} and $IMSI_{GRP_{1-i}}$, HSS computes MAC'_{GRP_1} using $GRPK_1$ and verifies whether computed MAC'_{GRP1} matches with the received MAC_{GRP1} or not. If they are found to be equal, it validates the MTCDs in a group; otherwise an authentication decline message is transmitted from HSS to MTCDs.

Step-6: After successful verification of MAC_{GRP1} and $LA1$, HSS computes the authentication response as:

- HSS retrieves the corresponding group key $GRPK_1$ and generates the random number $RAND_{HSS}$ and computes the group temporary key GTK_{GRP_1} as $GTK_{GRP_1} = f_3(ID_{GRP_1}||RAND_{HSS})_{GRPK_1}$
- Later, HSS computes the integrity key $IK_{GRP_{1-i}}$ and the cipher key $CK_{GRP_{1-i}}$ of each MTCD.

$$IK_{GRP_{1-i}} = f_4(ID_{GRP_1}||RAND_{HSS})_{K_{GRP1-i}}$$
$$CK_{GRP_{1-i}} = f_5(ID_{GRP_1}||RAND_{HSS})_{K_{GRP1-i}}$$

- HSS computes the session key K_{ASME} (Key for Access Security Management Entity) by using KDF (Key Derivation Function) for each MTCD. $K_{ASME}^{MTCD_{GRP_{1-i}}} = KDF(GTK_{GRP_1}||IK_{GRP_{1-i}}||CK_{GRP_{1-i}}||ID_{GRP_1}||IMSI_{GRP_{1-i}})$
- HSS computes the MAC_{HSS} using GTK_{GRP_1} as $MAC_{HSS} = f_1(RAND_{HSS}||AMF)_{GTK_{GRP_1}}$
- HSS generates $AUTH_{HSS}$ as $AUTH_{HSS} = (MAC_{HSS}||RAND_{HSS}||AMF)$
- The respective response value of each MTCD is computed using GTK_{GRP_1} as $XMAC_{MTCD_{GRP_{1-i}}} = f_1(ID_{GRP_1}||RAND_{HSS}||IMSI_{GRP_{1-i}})_{GTK_{GRP_1}}$
- HSS generates the response value for group as

$$XMAC_{GRP_1} = f_1(XMAC_{MTCD_{GRP_{1-1}}} \oplus XMAC_{MTCD_{GRP_{1-2}}} \oplus .. \oplus XMAC_{MTCD_{GRP_{1-n}}})_{GRPK_1}$$

- Finally, the HSS generates the group member authentication vectors (GMAVs) from the above computed parameters.

$$GMAV = (K_{ASME}^{MTCD_{GRP_{1-i}}}||AUTH_{HSS}||XMAC_{GRP_1}||GTK_{GRP_1})$$

- HSS assigns a new KID_i to the key $K_{GRP_{1-i}}$ and transmits the $(KID_i||SSDK_i||GMAVs)$ to MME.

Step-7: After acquiring $(KID_i||SSDK_i||GMAVs)$ from HSS, MME generates the $RAND_{MME}$ and calculates the $MAC_{MME} = f_1(MAC_{HSS}||RAND_{MME})_{GTK_{GRP_1}}$ and generates it's authentication token as $AUTH_{MME} = (MAC_{MME}||RAND_{MME}||MAC_{HSS}||RAND_{HSS}||AMF)$

Step-8: After computing the authentication token, MME sends $AUTH_{MME}$ to the GRP_{leader} concatenating with newly assigned KID_i encrypted under $SSDK_i$ by encryption algorithm X i.e $(AUTH_{MME}||X(KID_i)_{SSDK_i})$.

Step-9: After acquiring $AUTH_{MME}$ and encrypted KID_i from MME, GRP_{leader} broadcasts them to all the MTCDs in group GRP_1. Each MTCD will perform the following operations:

- Each MTCD generates the GTK_{GRP_1} and MAC'_{HSS} as $GTK_{GRP_1} = f_3(ID_{GRP_1}||RAND_{HSS})_{GRPK_1}$
  $MAC'_{HSS} = f_1(RAND_{HSS}||AMF)_{GTK_{GRP_1}}$
  Then, it verifies whether MAC'_{HSS} equals MAC_{HSS} or not. If they match, MTCDs authenticate the HSS; otherwise they fail to do the same.
- Each MTCD computes the $MAC'_{MME} = f_1(MAC_{HSS}||RAND_{MME})_{GTK_{GRP_1}}$ and verifies whether $MAC'_{MME} = ?MAC_{MME}$. If equality holds, the MME is validated by each MTCD; otherwise declines the authentication process.
- Each MTCD computes the integrity key and cipher key

$$IK'_{GRP_{1-i}} = f_4(ID_{GRP_1}||RAND_{HSS})_{K_{GRP1-i}}$$
$$CK'_{GRP_{1-i}} = f_5(ID_{GRP_1}||RAND_{HSS})_{K_{GRP1-i}}$$

  From $IK'_{GRP_{1-i}}$ and $CK'_{GRP_{1-i}}$, each MTCD generates the $K_{ASME}^{MTCD_{GRP_{1-i}}}$ to prevent the modification and eavesdropping of messages transmitted in the authentication process.

$$K_{ASME}^{MTCD_{GRP_{1-i}}} = KDF(GTK_{GRP_1}||IK'_{GRP_{1-i}}||CK'_{GRP_{1-i}}||ID_{GRP_1}||IMSI_{GRP_{1-i}})$$

- Each MTCD generates its response value as $XMAC_{MTCD_{GRP_{1-i}}} = f_1(ID_{GRP_1}||RAND_{HSS}||IMSI_{GRP_{1-i}})_{GTK_{GRP_1}}$ and sends its corresponding $XMAC_{MTCD_{GRP_{1-i}}}$ to the GRP_{leader}.

Step-10: GRP_{leader} calculates the respective response value as $XMAC'_{GRP_1} = f_1(XMAC_{MTCD_{GRP_{1-1}}} \oplus XMAC_{MTCD_{GRP_{1-2}}} \oplus \ldots \oplus XMAC_{MTCD_{GRP_{1-n}}})_{GRPK_1}$.

The GRP_{leader} transmits the computed response value $XMAC'_{GRP_1}$ to the MME for mutual authentication of each MTCD with the MME.

Step-11: After receiving the $XMAC'_{GRP_1}$ from GRP_{leader}, MME verifies whether $XMAC'_{GRP_1}$ matches with the $XMAC_{GRP_1}$ or not. If they are equal, MME broadcasts the authentication success message to each MTCD in the group. Otherwise, MME broadcasts the authentication failure message to the MTCDs. Each MTCD decrypts the KID_i and retains it for future authentication purpose. Hence, the authentication and key compliance procedure is executed.

D. SESSION KEY COMPLIANCE STAGE

For secure message transmission between the MTCDs and group leader, each MTCD establishes a session key between them. The message traffic between the group leader and the MTCDs is encrypted using the shared secret session key between them. The group leader and the MTCD perform the hash operation on their common key values and the random number (RAND) of a group that generates the unique shared secret session key. The session key between the MTCD and the group leader is generated as follows:

$$SSK_{i,j} = H((K_{\lfloor \frac{i}{2} \rfloor}||K_{\lfloor \frac{j}{2} \rfloor}||K_j)||RAND) \quad (3)$$

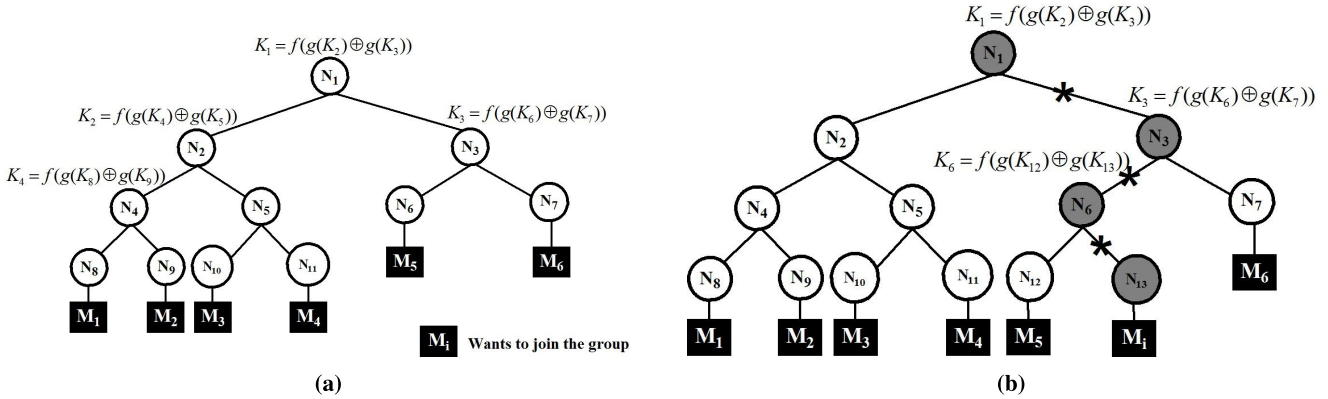

FIGURE 4. MTCD joining event scheme. (a) Before MTCD joins the group. (b) After MTCD joins the group.

For instance, as shown in Fig. 2, let us consider the MTCD M_1 (associated with node N_8 and having key value K_8) as a group leader and the MTCD M_7 (associated with node N_{14} and having key value K_{14}) want to communicate with each other. The session key $SSK_{1,7}$ between M_1 and M_7 is computed as follows using (3).

$$SSK_{1,7} = H((K_{\lfloor \frac{8}{2} \rfloor}||K_{\lfloor \frac{14}{2} \rfloor}||K_{14})||RAND)$$
$$SSK_{1,7} = H((K_4||K_7||K_{14})||RAND)$$

The unique secret session key is generated between these two devices. It is merely impossible to generate the same key between the group leader and any other MTCD of the group.

E. MTCD JOIN AND LEAVE EVENT STAGE

MTCD join and leave event stage illustrates the addition and removal of MTCDs from the group. For each operation, a node key value of some leaf node of the binary tree is updated that affects all the nodes keys along a path from that leaf node to the root node. The KGC needs to communicate the updated information along this path to the MTCDs. The KGC and all the MTCDs individually computes the new group key. For instance, if x is any non-root node along the updated path and, if y is the sibling of x, the KGC broadcasts the blinded new node secret $g(K_x)$ of x encrypted with the node key K_y of y. It allows all descendants of y to learn the new $g(K_x)$ but not K_x. In this phase, we consider the dynamic binary tree which grows and shrinks in size while addition and removal of MTCDs in the group. The details of join and leave operation of MTCDs are as follows:

1) MTCD JOIN EVENT SCHEME

Whenever a new MTCD joins a group, the new device is designated to the leaf node of a binary tree. When the leaf node splits, the member associated with leaf node associates with the left child of leaf node and the new member associates with the right child. Both members have the new keys. There is a possibility that the former sibling of the old member have the knowledge of the old blinded key and it use this information to gain an unblinded key with another group member. So, it is important to assign a new key to an old member too. The old member gets the new key after each successful authentication and the KID_i assigned to each key $K_{GRP_{1-i}}$ is updated. As described, the new values of the blinded node keys are updated and broadcasted secretly to the respective subgroups. The MTCD joining event scheme is illustrated in Fig. 4. To maintain the height of a tree, the closest leaf of the root is splitted when a new member is added.

For example, consider the MTCD join event scheme as shown in Fig. (4a). A new MTCD M_i wants to join the group. The leaf node N_6 splits and the MTCD linked with N_6 now links to the left child of node N_6. The new member associates with the right child of node N_6. Both nodes (N_{12} and N_{13}) have their updated key values that affects all secret nodes along their path to the root node. After MTCD joins the group, the updated key value path is shown in Fig. (4b).

2) MTCD LEAVE EVENT SCHEME

The MTCD can leave the group on completion of their functionality or due to battery exhaustion. Once a device retires from the group, it must not be capable to generate the group key and all the remaining previously known keys to that device should be updated. The MTCD leaving event scheme is illustrated in Fig. 5. When a member associated with the leaf node leaves the group, the member allocated to the sibling of the leaf node is reallocated to the parent of leaf node and provided a new key value. If the sibling of the leaf node is the root of a subtree then the parent node becomes the root of that subtree, moves the subtree closer to the root and assign a new key to one of the leaves of this subtree. The new values of the blinded node keys are updated and broadcast secretly to the respective nodes.

For instance, consider the MTCD leave event scheme as shown in Fig. (5a). The MTCD M_3 wants to leave the group. The node N_{11} is the sibling of node N_{10} (where M_3 is associated) and let N_5 is the parent of the N_{11}. Node N_{11} is a leaf node so the MTCD assigned to N_{11} is reassigned to N_5 and updates the key value. After MTCD leaves the group, the updated key path is shown in Fig.(5b). The group key is

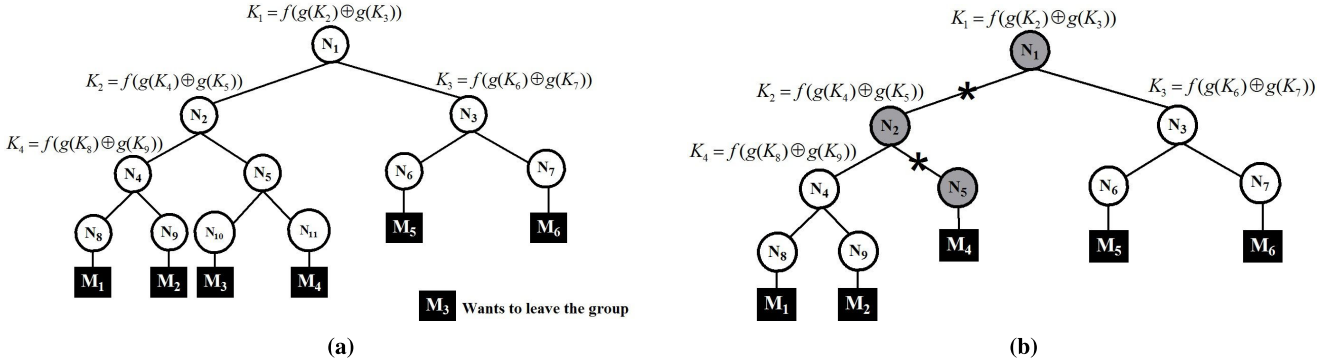

FIGURE 5. MTCD leaving event scheme. (a) Before MTCD leaves the group. (b) After MTCD leaves the group.

```
goal
secrecy_of sec_skey, sec_tkey
authentication_on mobile_hss
authentication_on mobile_mme
authentication_on hss_mme
end goal
```

FIGURE 6. Goals of the proposed protocol.

updated whenever a new MTCD is added or removed from the group. As illustrated in this phase, the proposed scheme maintains the unlink-ability in the group key whenever the MTCD joins or leaves the group. Hence, an adversary will neither violate the group key nor perform security attacks in the proposed protocol.

IV. FORMAL VERIFICATION USING AVISPA TOOL

The proposed protocol is formally verified using AVISPA tool [23], [35]. AVISPA supports various security analysis and verification models such as SATMC (SAT-based Model-Checker), OFMC (On-the-Fly-Model-checker) and Cl-AtSe (Constraint Logic-Based Attack Searcher) [36]. The protocol is coded in High Level Protocol Specifications Language (HLPSL) to verify security properties of the protocol. The main objectives of the proposed protocol are to provide mutual authentication and key agreement between the communicating entities. Moreover, it is required to achieve the secrecy of pre-shared secret keys ($K_{GRP_{1-i}}$, $SSDK_i$ and GTK_{GRP_i}) in the authentication process. The goals of the proposed SEGB-AKA protocol are shown in Fig. 6. In the proposed protocol, there are three participants: MTCD, MME and HSS. The basic role of these participants is described in HLPSL code in appendix VII. We verify the proposed protocol using OFMC and CL-AtSe model checker and the results are shown in the Fig. 7 and Fig. 8 respectively. The SAFE keyword in Fig. 7 and 8 prove that the proposed protocol achieves the specified goals and avoids all the identified attacks.

V. SECURITY ANALYSIS

The proposed SEGB-AKA protocol maintains the same architecture as EPS-AKA protocol. Hence, there is a

```
% OFMC
% Version of 2006/02/13
SUMMARY
SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
/home/span/span/testsuite/results/SEGB-AKA.if
GOAL
as_specified
BACKEND
OFMC
COMMENTS
STATISTICS
parseTime: 0.00s
searchTime: 0.09s
visitedNodes: 66 nodes
depth: 6 plies
```

FIGURE 7. Result summarized by OFMC backend.

```
SUMMARY
SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
TYPED_MODEL
PROTOCOL
/home/span/span/testsuite/results/SEGB-AKA.if
GOAL
As Specified
BACKEND
CL-AtSe
STATISTICS
Analysed : 3 states
Reachable : 2 states
Translation: 0.02 seconds
Computation: 0.00 seconds
```

FIGURE 8. Result summarized by CL-AtSe backend.

possibility of similar security issues in the proposed protocol. We present the security analysis of the proposed protocol in terms of the security goals, key privacy properties and resistance against different possible attacks.

- *Property 1: Mutual authentication and key agreement:* The proposed protocol achieves the mutual authentication between each MTCD, the MME and HSS by generating the aggregate MAC. In the proposed protocol, the HSS authenticates each MTCD by

verifying the MAC_{GRP_1} received from $GRP_{1-leader}$. If the HSS fails to verify the MAC_{GRP_1}, HSS determines the malicious MTCD in the group. The HSS computes MAC_{HSS}, $XMAC_{GRP_1}$ and generates GMAV for each MTCD. HSS sends the GMAVs to the MME. The MME computes MAC_{MME} and transfers it to $GRP_{1-leader}$. Simultaneously, $GRP_{1-leader}$ transmits MAC_{MME} to each MTCD. Further, each device authenticates the MME and HSS by verifying the received MAC_{MME} and MAC_{HSS} respectively. Moreover, each MTCD generates their $XMAC_{MTCD_{GRP_{1-i}}}$ using GTK_{GRP_i} and transfers to $GRP_{1-leader}$. Then, $GRP_{1-leader}$ generates the $XMAC'_{GRP_1}$ by aggregating each received $XMAC_{MTCD_{GRP_{1-i}}}$ from MTCDs. Each MTCD is authenticated at MME by verifying the received $XMAC'_{GRP_1}$. Hence, the communication entities obtain the mutual authentication and key agreement.

- *Property 2: Solution to the single key problem:*
  The security of the existing symmetric key based group AKA protocols completely depends upon the confidentiality of the K_{GRP1-i}. Once, K_{GRP1-i} is compromised, each key can be recovered and there is a possibility that an adversary can be verified by the network. To avoid this problem, we introduced KID_i that points to the K_{GRP1-i} in the proposed protocol. The KID_i is reallocated after each successful authentication and key agreement process. Moreover, the dynamic $SSDK_i$ is used to generate the authentication request and response messages in the communication network. The $SSDK_i$ is generated using the function f_1' and the pre-shared secret key. The adversary never succeeds in deriving the valid $SSDK_i$ as the f_1' is irreversible and KID_i is untraceable. Hence, an adversary will never compromise the pre-shared secret key in the communication network. Therefore, the proposed protocol avoids the problem of single key in the authentication process.
- *Property 3: Privacy preservation and identity theft:*
  An adversary cannot trace the original identity of the MTCDs. The privacy of each MTCD ($IMSI_{GRP_{1-i}}$) is well protected during the authentication over the network. To establish privacy preservation in the proposed protocol, each MTCD generates the $SSDK_i$. The $SSDK_i$ is generated using the KID_i and pre-shared cryptographic function f_1'. The identity of each MTCD is encrypted by respective $SSDK_i$ ($E(ID_{GRP_1}||IMSI_{GRP_{1-i}}||LAI)_{SSDK_i}$) and transmitted over the network. Hence, an adversary will never succeed in retrieving the identity of MTCDs. For each authentication request, a unique KID_i and $SSDK_i$ is generated whenever a device connects to the visiting MME. Hence, an adversary cannot generate the encryption keys (KID_i and $SSDK_i$). Only HSS can retrieve these shared secret keys. Hence, the proposed protocol preserves the privacy of each MTCD.
- *Property 4: Resistance to signaling congestion overload:*
  To overcome the problem of signaling congestion overload during the authentication process of the proposed protocol, each MTCD transmits its $MAC_{MTCD_{GRP_{1-i}}}$ to the $GRP_{1-leader}$. The $GRP_{1-leader}$ aggregates the received $MAC_{MTCD_{GRP_{1-i}}}$ into a single authentication request message MAC_{GRP_1} and sends it to communication entities in the network. Thus, only a single message is transmitted by the $GRP_{1-leader}$ instead of transmitting n authentication vectors in the proposed protocol. Similarly, the $GRP_{1-leader}$ sends an aggregated response value $XMAC'_{GRP_1}$ to the MME. Moreover, MME can simultaneously authenticate a group of MTCDs by message aggregation. Hence, the proposed protocol does not generate the simultaneous authentication request for each MTCD and avoids the network signaling congestion during the authentication process.
- *Property 5: Key secrecy, key identity theft, and attempt to derive keys:*
  To maintain the secrecy of transmitted messages, the communication entities compute the session keys in the proposed protocol. Each session key is generated using the $IK'_{GRP_{1-i}}$ and $CK'_{GRP_{1-i}}$ at the communication entities without being transmitted over the network. Moreover, the key identifier is encrypted by using encryption function and the respective $SSDK_i$. Hence, an adversary can never succeed in extracting the key identifier and the shared secret key. In addition, a unique session key is generated at either end for secure message transmission between MTCDs and $GRP_{1-leader}$. Hence, an adversary will never succeed in extracting the K_{GRP1-i}, $K_{ASME}^{MTCD_{GRP_{1-i}}}$ and KID_i over the network.
- *Property 6: Session unlink-ability and maintenance of KFS/KBS:*
  In the proposed protocol, the $GRPK_i$ is shared between the MTCDs and the HSS. Only the MTCDs present in the group GRP_i have the knowledge of a $GRPK_i$. Whenever any new MTCD wants to join the group and performs the access authentication process, it never gets access to the group key before it joins the group. Similarly, whenever the MTCD leaves the group, it fails to access the group communication as the group key is updated as soon as the device leaves the group. An adversary cannot link the current session group key with the previous or next sessions. Hence, the SEGB-AKA maintains the group key unlink-ability and KFS/KBS.
- *Property 7: Attack resistance:* The proposed protocol can successfully defeat all the identified attacks such as redirection, MiTM, replay, impersonation and DoS attack. Moreover, an adversary can neither compromise the signaling messages nor retrieve any information by delaying the messages. We justify that the proposed protocol is secure against various attacks.
  - *Resistance to redirection attack:* An adversary may impersonate as the MTCDs and establish a false base station to access the user information. If an

TABLE 2. Security analysis of various group based AKA protocols for M2M communication.

Security Properties	AKA Protocols										
	G-AKA [20]	SE-AKA [21]	EG-AKA [22]	NOVEL-AKA [23]	GBAAM-AKA [25]	GROUP-AKA [24]	GLARM-AKA [27]	PRIVACY -AKA [26]	GR-AKA [28]	GBS-AKA [29]	SEGB -AKA
SP_1	Symmetric	Hybrid	Hybrid	Symmetric	Asymmetric	Symmetric	Symmetric	Hybrid	Asymmetric	Symmetric	Symmetric
SP_2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SP_3	✓	✗	✗	✓	✗	✗	✓	✗	✗	✓	✓
SP_4	✗	✓	✓	✓	✗	✗	✗	✓	✓	✗	✓
SP_5	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓	✓
SP_6	✗	✓	✗	✗	✗	✓	✓	✓	✓	✓	✓
SP_7	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SP_8	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SP_9	✓	✓	✓	✓	✓	✓	✗	✓	✓	✗	✓
SP_{10}	✗	✓	✗	✗	✓	✗	✓	✓	✓	✗	✓
SP_{11}	✗	✓	✗	✓	✓	✓	✓	✗	✓	✗	✓
SP_{12}	✗	✗	✗	✗	✗	✓	✗	✗	✓	✗	✓
SP_{13}	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓

SP_1: Type of cryptosystem; ✓: Achieved; ✗: Not Achieved; SP_2: Basic security requirements; SP_3: Follow the 3GPP standard; SP_4: Privacy preservation and protection; SP_5: Network signaling congestion avoidance; SP_6: Protection from redirection attack; SP_7: Protection from MiTM attack; SP_8: Protection from replay attack; SP_9: Protection from impersonation attack; SP_{10}: Protection from DoS attack; SP_{11}: Key forward/backward secrecy; SP_{12}: Maintains dynamic group membership management scheme; SP_{13}: Avoids the single key problem.

intruder fails to obtain the user information, it is not possible to perform the redirection attack on the network. In the proposed protocol, real identity of each MTCD is encrypted using $SSDK_i$ and transferred to the HSS. Therefore, an adversary can never catch the identity of the MTCDs and fails to impersonate them. Moreover, to maintain the integrity, the LAI is embedded into $M_{MTCD_{GRP_{1-i}}}$ and transferred to the MME. The HSS compares the embedded LAI with the received one from the MME. The authentication request is discarded if the LAI is not verified by the HSS.

– *Resistance to MiTM attack:* The session key and secure identity of devices protect the LTE network from the MiTM atttack. The secret session key $K_{ASME}^{MTCD_{GRP_{1-i}}}$ is generated between communication entities to prevent the modification and eavesdropping of messages transmitted in authentication process. An adversary can never compute the legitimate GTK_{GRP1}. Therefore, it is impossible for him/her to generate the valid authentication request and response messages in the communication network. Hence, it is merely impossible to launch the MiTM attack at the network.
– *Resistance to replay attack:* In the proposed protocol, each MTCD uses the time stamp TS_{GRP_1} to generate the authentication request and response messages over the network. The TS_{GRP_1} establishes a concurrent communication between the authentication entities in the network. The synchronization failure in the authentication process declines the authentication request. Secondly, the distinct $RAND_{HSS}$ and $RAND_{MME}$ are used to compute authentication challenge for MTCDs. An adversary cannot generate a false authentication challenge even if he/she obtains these random numbers. This is the reason why an adversary can never perform the replay attack.
– *Resistance to impersonation attack:* In the proposed AKA protocol, $GRP_{1-leader}$ generates MAC_{GRP_1} and transfers to the HSS. For instance, an adversary attempts to generate the legitimate MAC_{GRP_1} by masquerading MTCD. The HSS computes the MAC'_{GRP_1} and verifies with MAC_{GRP_1}. If the verification fails, a malicious MTCD is recognized by the HSS. In addition, the malicious MTCDs cannot generate $K_{ASME}^{MTCD_{GRP_{1-i}}}$ and communication between the entities remain persistent. An adversary can never modify the communication between $GRP_{1-leader}$ and $MTCD_{GRP_{1-i}}$ because the computation of distinct session key is a tedious task. Hence, it is not possible to perform

TABLE 3. Communication overhead of group based AKA protocols for M2M communication.

AKA Protocols	Communication Overhead Per Message												
	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	Total
G-AKA [20]	448*m	448*m	432*m	496*m	64*m	320*n	496*n	64*n	-	-	-	-	880*(n-m)+1888*m
SE-AKA [21]	512*m	552*m	432*m	1072*m	256*m	384*n	1072*n	256*n	-	-	-	-	1712*(n-m)+2824*m
EG-AKA [22]	576*m	704*m	768*m	512*m	256*m	256*m	128*m	576*n	512*n	256*n	256*n	128*n	1728*(n-m)+3200*m
NOVEL-AKA [23]	544*m	544*m	816*m	496*m	64*m	480*n	496*n	64*n	-	-	-	-	1040*(n-m)+2464*m
GBAAM-AKA [25]	128*n + 128	128*n +128*m +128	128*n +736*m	419*n	64*n	128*n	960*n	384*n + 576	1024*m	1024*n	-	-	3363*n + 1888*m + 832
GROUP-AKA [24]	128*n +128	128*n + 128*m +128	1200*m	624*m	624*n	64*m	752*n	448*n	384*m	-	-	-	2080*n + 2400*m + 256
GLARM-AKA [27]	448*n	384*n+64*m	384*n + 104*m	1072*m	688*m	688*n	64*n	64*m	-	-	-	-	1968*n + 1992*m
PRIVACY-AKA [26]	448*n	384*n+64*m	384*n + 192*m	992*m	864*m	864*n	64*n	64*m	-	-	-	-	2144*n + 2176*m
GR-AKA [28]	320*m	360*m	256*m	448*m	192*n +192	320*n	384*n	-	-	-	-	-	896*n + 1384*m +192
GBS-AKA [29]	512*n	448*n + 64*m	448*n + 64*m	432*m	512*m	512*n	-	-	-	-	-	-	1920*n + 1072*m
SEGB-AKA	384*n	320*n+128*m	320*n +168*m	256*n +688*m	128*n +432*m	128*n +432*m	64*n	64*m	-	-	-	-	1600*n+1912*m

the impersonation attack in the communication network.

– *Resistance to DoS attack:* In the DoS attack, an adversary can impersonate as the legitimate MTCD and constantly transmit the bogus authentication requests to gain the access to the network. In the SEGB-AKA protocol, $GRP_{1-leader}$ computes the MAC_{GRP_1} and sends to HSS. Then, HSS generates the MAC'_{GRP_1} and compares it with received MAC_{GRP_1}. If HSS finds the mismatch between them, it determines the malicious MTCDs in the network and transmits an authentication decline message to the group. On the other side, each MTCD computes the MAC_{MME}/MAC_{HSS} and verifies the authenticity of them. If the verification fails, an authentication declined message is transmitted to the MME and HSS. Hence, it is impossible to launch DoS attack.

The comparative study of the different security properties identified for M2M communication in the LTE/LTE-A network is shown in Table 2. It is observed that the existing AKA protocols for M2M communication fail to fulfill all the goals and also suffer from the single key problem during authentication process. Moreover, the existing protocol does not maintain the unlink-ability between the group key for different sessions. The proposed protocol follows the symmetric key cryptosystem approach and avoid the single key problem. In addition, the protocol preserves the privacy of MTCDs in the communication network. The protocol successfully avoids all the identified attacks in the communication network and maintains KFS/KBS. Hence, the SEGB-AKA protocol is comparatively superior among all the existing group based AKA protocols for M2M communication.

VI. THE PERFORMANCE ANALYSIS OF THE PROPOSED PROTOCOL

This section illustrates the performance analysis of the proposed protocol in terms of the communication overhead, computational overhead, message transmission overhead and the storage overhead with respect to existing group AKA protocol for M2M communication in the LTE/LTE-A network.

A. COMPARATIVE ANALYSIS OF THE COMMUNICATION OVERHEAD

In the ensuing section, an attempt has been made to evaluate the communication overhead of the proposed protocol with respect to the existing group based AKA protocols in the LTE/LTE-A networks. To compute the communication overhead, we considered there are n number of MTCDs that formed m group. The total number of bits required for each group AKA protocol are the total bits transmitted by each message during the authentication process. The list of the standard parameters and their sizes to evaluate the communication overhead is presented in Table 1. The comparative analysis for the communication overhead of the proposed and existing group AKA protocols for M2M communication is shown in Table 3. The communication overhead of the proposed SEGB-AKA protocol is computed as:

$$M_1 = (MAC_{MTCDG1-i}, M_{MTCDG1-i}) = 64 * \mathrm{n} + 320 * \mathrm{n} = 384 * \mathrm{n}$$

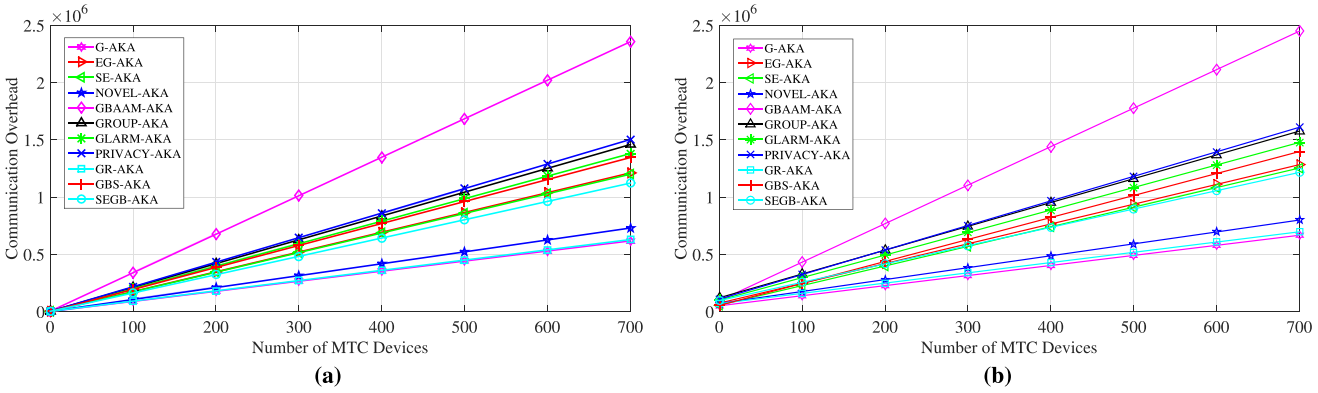

FIGURE 9. Comparison of the communication overhead. (a) m=1. (b) m=50.

$$M_2 = (M_{MTCDG1-i}, MAC_{MTCDG1-i}, TS_{GRP1}) = 320 * \mathrm{n} + 64 * \mathrm{m} + 64 * \mathrm{m} = 320 * \mathrm{n} + 128 * \mathrm{m}$$

$$M_3 = (M_{MTCDG1-i}, MAC_{MTCDG1-i}, TS_{GRP1}, LAI') = 320 * \mathrm{n} + 64 * \mathrm{m} + 64 * \mathrm{m} + 40 * \mathrm{m} = 320 * \mathrm{n} + 168 * \mathrm{m}$$

$$M_4 = (KID_i, SSDK_i, GMAV's) = 128 * \mathrm{n} + 128 * \mathrm{n} + 688 * \mathrm{m} = 256 * \mathrm{n} + 688 * \mathrm{m}$$

$$M_5 = (X(KID_i)_{SSDK_i}, AUTH_{MME}) = 128 * \mathrm{n} + 432 * \mathrm{m}$$

$$M_6 = (X(KID_i)_{SSDK_i}, AUTH_{MME}) = 128 * \mathrm{n} + 432 * \mathrm{m}$$

$$M_7 = (XMAC_{MTCDG1-i}) = 64 * \mathrm{n}$$

$$M_8 = (XMAC'_{GRP1}) = 64 * \mathrm{m}$$

The total communication overhead of the SEGB-AKA protocol for the M2M communication in the LTE/LTE-A network is $1600 * n + 1912 * m$.

Fig. (9a) and (9b) illustrates the comparative study of the communication overhead that incurs in several group based AKA protocols for varying number of MTCDs in the group. The communication overhead incurred by the GR-AKA protocol is comparatively less than the proposed SEGB-AKA protocol. But, the GR-AKA follows the asymmetric key cryptosystem based scheme that does not suit to the resource constrained MTCDs. Moreover, the G-AKA and Novel-AKA also incur a less communication overhead. But, these protocols fail to avoid the network signaling congestion when a group of MTCDs simultaneously request for the authentication. Therefore, these protocols are not suitable for group authentication and do not maintain the KFS/KBS. Hence, it is observed that the SEGB-AKA protocol achieves all the security properties of the M2M communication in LTE/LTE-A network with lesser communication overhead as compared to other existing group based AKA protocols.

B. COMPARATIVE ANALYSIS OF THE COMPUTATION OVERHEAD

To evaluate the total computation overhead generated by each protocol, the computation time of the cryptographic functions is defined as [24], [28]: Lagrange component time at MTCD (T_{L-MTCD}) = 0.0572 ms; Lagrange component time at HSS (T_{L-HSS}) = 0.0351 ms; multiplication over an elliptic curve (T_{mul})=0.612 ms; pairing (T_{pair})=4.51 ms; Hash operation (T_{hash})=0.067 ms; symmetric encryption/decryption (T_{aes})=0.161 ms; modulus (T_{mod})=0.124 ms; maptopoint hash operation (T_{mtp})= 0.525 ms. It is considered that the computation time of X-OR operation (T_{X-OR}) is negligible. In addition, there are n number of MTCDs forming m group. The comparative analysis of the computation overhead of the proposed SEBG-AKA protocol with the existing group based AKA protocols is presented in Table 4. Moreover, Fig. (10a) and (10b) illustrates the comparative analysis of the computation overhead of these protocols with varying number of MTCDs and groups.

It is observed that the GBAAM-AKA protocol has the highest computation overhead compared to all other group AKA protocols as it executes the time consuming maptopoint hash, pairing and multiplication functions. Although, some of the existing protocols has better computation overhead than the proposed protocol, many of them do not provide the privacy protection, suffer from authentication signaling overload and different possible attacks on the communication network. The computation overhead incurred by the proposed SEGB-AKA protocol is competitive with respect to existing group AKA protocols. Different from the prior protocols, the SEGB-AKA protocol follows the symmetric key based approach and avoids the single pre-shared key problem. Moreover, the protocol avoids all the identified attacks and fulfills the security requirements of the M2M communication network. Hence, the proposed SEGB-AKA protocol provides the improved security compared to an existing group based AKA protocols with competitive computation overhead.

C. COMPARATIVE ANALYSIS OF THE TRANSMISSION OVERHEAD

It is worthwhile to demonstrate the comparative analysis of message transmission overhead of the existing and proposed

TABLE 4. Computation overhead of group based AKA protocols for M2M communication.

AKA Protocols	Computation Overhead		
	MTC Devices	Network	Total (in ms)
G-AKA [20]	$(4T_{hash}) + (4T_{hash})(n-1)$	$(3T_{hash}) * n + (2T_{hash}) * m$	$(7T_{hash}) * n + (2T_{hash}) * m$
SE-AKA [21]	$(2T_{mul} + 4T_{hash})$ $+(2T_{mul} + 3T_{hash})(n-1)$	$(2T_{mul} + 3T_{hash}) * n$ $+(2T_{hash}) * m$	$(4T_{mul} + 6T_{hash}) * n$ $+(2T_{hash}) * m + T_{hash}$
EG-AKA [22]	$(2T_{mul} + 3T_{hash} + 2T_{aes})$ $+(2T_{mul} + 3T_{hash} + T_{aes}) * (n-1)$	$(2T_{mul} + 2T_{hash} + T_{aes}) * n$ $+(T_{hash} + T_{aes}) * m$	$(4T_{mul} + 5T_{hash} + 3T_{aes}) * n$ $+(T_{hash} + T_{aes}) * m$
NOVEL-AKA [23]	$4T_{hash} + (3T_{hash}) * (n-1)$	$(3T_{hash}) * n + (2T_{hash}) * m$	$(6T_{hash}) * n + (2T_{hash}) * m + T_{hash}$
GBAAM-AKA [25]	$(4T_{mul} + 2T_{hash}) * n$	$(3T_{mul} + 2T_{hash} + T_{mtp}) * n$ $+(T_{mul} + 2T_{pair} + T_{hash}) * m$	$(7T_{mul} + 4T_{hash} + T_{mtp}) * n$ $+(T_{mul} + 2T_{pair} + T_{hash}) * m$
GROUP-AKA [24]	$(T_{mod} + 2T_{hash}) * n$ $+(2T_{hash}) * m$	$(T_{mod} + 2T_{hash}) * n$ $+(6T_{hash} + T_{aes}) * m$	$(2T_{mod} + 4T_{hash}) * n$ $+(8T_{hash} + T_{aes}) * m$
GLARM-AKA [27]	$(4T_{hash}) * n + (2T_{hash}) * m$	$(3T_{hash}) * n + (2T_{hash}) * m$	$(7T_{hash}) * n + (4T_{hash}) * m$
PRIVACY-AKA [26]	$(6T_{hash} + 2T_{mul}) * n$ $+(2T_{hash}) * m$	$(2T_{hash} + T_{mul}) * n$ $+(4T_{hash} + T_{mul}) * m$	$(8T_{hash} + 3T_{mul}) * n$ $+(6T_{hash} + T_{mul}) * m$
GR-AKA [28]	$(T_{LC} + 2T_{mul} + T_{hash}) * n$ $+(4T_{hash}) * m$	$(T_{hash} + T_{LC}$ $+2T_{mul}) * m$	$(T_{LC} + 2T_{mul} + T_{hash}) * n$ $+(5T_{hash} + T_{LC} + 2T_{mul}) * m$
GBS-AKA [29]	$(3T_{hash}) * n + (T_{hash}) * m$	$(2T_{hash}) * n + (T_{hash}) * m$	$(5T_{hash}) * n + (2T_{hash}) * m$
SEGB-AKA	$(4T_{hash} + 2T_{aes}) * n$ $+(2T_{hash}) * m$	$(3T_{hash} + 2T_{aes}) * n$ $+(2T_{hash}) * m$	$(7T_{hash} + 4T_{aes}) * n$ $+(4T_{hash}) * m$

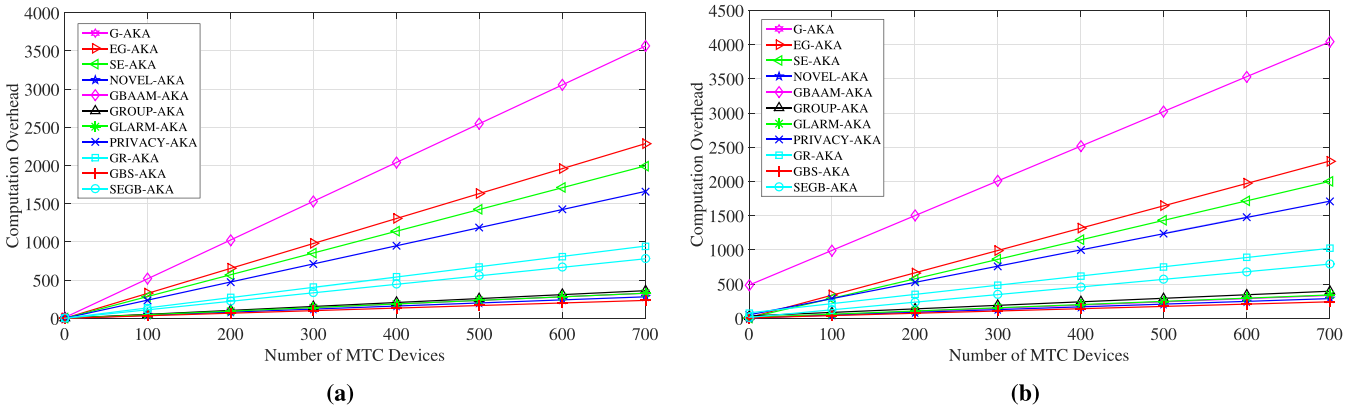

FIGURE 10. Comparison of the computational overhead. (a) m=1. (b) m=50.

group based AKA protocols. Let us consider the authentication overhead to deliver the authentication message between i) MTCDs and MME : x unit; ii) the MME and HSS : y unit respectively. As the MME is far away from the HSS, $y \gg x$. We also consider that there are n MTCDs forming m groups ($n > m$). The comparative analysis of the transmission overhead of the proposed protocol with the existing protocols is presented in Table 5. From Table 5, we can observe that the transmission overhead of the proposed AKA protocol is similar to other existing AKA protocols. It proves that the proposed protocol achieves all the objectives of the M2M communication in an IoT enabled LTE/LTE-A network without compromising the authentication message transmission overhead.

D. COMPARATIVE ANALYSIS OF THE STORAGE OVERHEAD AT MME

The storage overhead of different group AKA protocols at MME of communication networks is analyzed. All the group based AKA protocols need to store authentication vector generated by HSS on MME for further authentication process. For the comparative study of storage overhead, we consider m group of n MTCDs. The comparative study of the storage overhead incurred in different group based AKA protocol is shown in Table 6. The GBS-AKA protocol requires the less storage overhead but is vulnerable to various security threats in the communication network. Moreover, the GR-AKA protocol incurs less storage overhead compared to the proposed protocol. But, the asymmetric cryp-

TABLE 5. Message transmission overhead in group based AKA protocols for M2M communication.

AKA Protocols	Message Transmission Overhead
G-AKA [20]	$(7x+2y)m+(7x)(n-m) = 7nx+2ny$
SE-AKA [21]	$(6x+2y)m+(6x)(n-m) = 6nx+2my$
EG-AKA [22]	$(10x+2y)m+(10x)(n-m) = 10nx+2my$
NOVEL-AKA [23]	$(6x+2y)m+(6x)(n-m) = 6nx+2my$
GBAAM-AKA [25]	$(6x+4y)m+(6x)(n-m) = 6nx+4my$
GROUP-AKA [24]	$(6x+3y)m+(6x)(n-m) = 6nx+3my$
GLARM-AKA [27]	$(6x+2y)m+(6x)(n-m) = 6nx+2my$
PRIVACY-AKA [26]	$(6x+2y)m+(6x)(n-m) = 6nx+2my$
GR-AKA [28]	$(6x+2y)m+(6x)(n-m) = 6nx+2my$
GBS-AKA [29]	$(6x+2y)m+(6x)(n-m) = 6nx+2my$
SEGB-AKA	$(6x+2y)m+(6x)(n-m) = 6nx+2my$

TABLE 6. Storage overhead at MME in group based AKA protocols for M2M communication.

AKA Protocols	Storage Overhead (bits)
G-AKA [20]	$432m$
SE-AKA [21]	$432m$
EG-AKA [22]	$512m$
NOVEL-AKA [23]	$816m$
GBAAM-AKA [25]	$128n+736m$
GROUP-AKA [24]	$1200m$
GLARM-AKA [27]	$1072m$
PRIVACY-AKA [26]	$992m$
GR-AKA [28]	$448m$
GBS-AKA [29]	$432m$
SEGB-AKA	$688m$

tosystem based scheme is not suitable for the resource constrained MTCDs in the M2M communication. The G-AKA, EG-AKA, and SE-AKA protocols also incur the less storage overhead compared to the proposed protocol but, all these protocols are not suitable for the group authentication and fail to maintain the KFS/KBS. The SEGB-AKA protocol achieves all the security requirements of the M2M communication in the LTE/LTE-A network with competitive storage overhead in the network.

```
role device(D, M, H:agent,
            SND, RCV: channel(dy),
            Ssdki, Grpk1, KG1: symmetric_key,
            TSgrp1, Kidi, Rmme, Idg1, Amf:text,
            F1,F2,F3,F4,F5,KDF,Ec:function)
played_by D
def=
  local
       State :nat,
       NAI, Mtcd_gi, Rhss:text
       const sec_ssdki, sec_grpk1, mtcd_mme,
       hss_mme, hss_mtcd :protocol_id,
       success: text

  init State := 0

  transition
  1. State = 0 /\ RCV(start) =|>
     State'= 1 /\ SND ((Ec(Ssdki.Idg1.Mtcd_gi).
                    TSgrp1.Kidi).F1(KG1.TSgrp1.
                    Mtcd_gi).TSgrp1.NAI)
                  /\ secret(Idg1,TSgrp1,sec_ssdki,
                    {D,M,H})
  2. State = 1 /\ RCV ((F1(F3(Grpk1.Idg1.Rhss')).
                    Rmme.(F1(F3(Grpk1.Idg1.Rhss')).
                    Rhss'.Amf)).(F1(F3(Grpk1.Idg1.
                    Rhss')).Rhss'.Amf).Rmme.Rhss'.
                    Amf.Kidi) =|>
     State'= 2 /\ SND (F1({(F1(F3(Grpk1.Idg1.
                     Rhss')).Idg1.Rhss'.Mtcd_gi)}_F1
                    (F3(Grpk1.Idg1.Rhss')).Idg1.Rhss'
                    .Mtcd_gi))
                  /\ secret(Rhss, Idg1,sec_grpk1,{D,
                    M,H})
                  /\ witness(D,M,mtcd_mme,Rhss')
                  /\ request(D,M,H, hss_mtcd,Rhss')
  3. State = 4 /\ RCV(success) =|>
     State'= 5
 end role
```

Role 1. MTCD.

VII. CONCLUSION AND FUTURE WORK

In this paper, the security enhanced group based AKA protocol for M2M communication in an IoT enabled LTE/LTE-A networks is proposed. Compared with the prior work, the SEGB-AKA improves the security, preserves the privacy of MTCDs and avoids the single key problem from the communication network. It can simultaneously authenticate the group of MTCDs and maintain the unlink-ability in the group key for the dynamic access policy scenario. The protocol resolves the problem of network signaling congestion and avoids the known attacks from the communication network. Moreover, the proposed SEGB-AKA protocol is formally analyzed using the AVISPA tool. The security analysis proves that the protocol fulfills all the security requirements and validates the security against various known attacks. The performance analysis illustrates that the protocol incurs less storage and communication overhead as compared to other existing AKA protocols for M2M communication. The proposed protocol provides the improved security with competitive transmission and computation overhead. To the best of our knowledge, it is the first attempt to solve the single key problem with privacy preservation of the MTCD in the group based AKA protocol for M2M communication.

```
role mme(M,H,D:agent,
          SND, RCV: channel(dy),
          Ssdki, Grpk1, KG1: symmetric_key,
          TSgrp1, Kidi,Rmme, Idg1, Amf :text,
          F1,F2,F3, F4, F5,KDF, Ec: function)
played_by M
def=
  local
       State :nat,
       NAI,Mtcd_gi, Rhss:text
       const sec_ssdki, sec_grpk1,sec_kg1,mtcd_mme,
       hss_mme, hss_mtcd:protocol_id,
       success: text

  init State := 0

  transition
  1. State = 0 /\ RCV ((Ec(Ssdki.Idg1.Mtcd_gi).
                  TSgrp1.Kidi).F1(KG1.TSgrp1.
                  Mtcd_gi).TSgrp1.NAI) =|>
     State':= 1 /\ NAI'= new()
                /\ SND ((Ec(Ssdki.Idg1.Mtcd_gi).
                   TSgrp1.Kidi).(F1({F1(KG1.TSgrp1
                   .Mtcd_gi)_F1(KG1.TSgrp1.Mtcd_gi)
                   ).NAI'.Grpk1).TSgrp1.NAI')
                /\ secret(Idg1,Kidi, sec_ssdki,{M,
                   H,D})
                /\ secret(TSgrp1,NAI,sec_kg1,{H,M
                   ,D})
                /\ witness(M,H, hss_mme, NAI')
                /\ request (M,H,D, hss_mtcd, NAI')
  2. State = 1 /\ RCV ((KDF(F3(Grpk1.Idg1.Rhss'))
                  .(F4(Grpk1.Idg1.Rhss')).(F5(Grpk1
                  .Idg1.Rhss')).Idg1.Mtcd_gi).((F1
                  (F3(Grpk1.Idg1.Rhss')).Rhss'.Amf)
                  .Rhss'.Amf).(F1({(F1(F3(Grpk1.Idg1
                  .Rhss')).Idg1.Rhss'.Mtcd_gi)}_F1
                  (F3(Grpk1.Idg1.Rhss')).Idg1.Rhss'
                  .Mtcd_gi)).(F3(Grpk1.Idg1.Rhss'))
                  .Kidi.Ssdki)  =|>
     State':= 2 /\ SND ((F1(F3(Grpk1.Idg1.Rhss'))
                   .Rmme(F1(F3(Grpk1.Idg1.Rhss')).
                   Rhss'.Amf)).(F1(F3(Grpk1.Idg1.
                   Rhss')).Rhss'.Amf).Rmme.Rhss'.
                   Amf.Kidi)
                /\ secret(Rmme, Idg1,sec_grpk1,{M,
                   H,D})
                /\ witness(M,H,hss_mme,Rhss')
                /\ request (M,H,D, mtcd_mme, NAI')
  3. State = 2 /\ RCV (F1({(F1(F3(Grpk1.Idg1.Rhss')
                  ).Idg1.Rhss'.Mtcd_gi)}_F1(F3(Grpk1
                  .Idg1.Rhss')).Idg1.Rhss'.Mtcd_gi))
                  =|>
     State':= 3 /\ SND(success)
                /\ request(M,H,D, hss_mme,Rhss')
 end role
```

Role 2. **MME.**

In the group based communication, when a group of MTCDs are moving then they face a new problem in the authentication process. There is a possibility that a long delay and huge computational overhead may incur during the handover scenario. Therefore, the group based AKA protocol for these scenarios in the M2M communication will be further analyzed.

APPENDIX
HLPSL CODE DEFINING THE ROLE OF MTCD, MME AND HSS

See Role1, Role2, and Role3

```
role hss(H,M,D:agent,
         SND, RCV: channel(dy),
         Ssdki,Grpk1, KG1: symmetric_key,
         TSgrp1, Kidi,Rmme, Idg1, Amf :text,
         F1,F2,F3, F4, F5,KDF, Ec: function)

played_by H
  def=
    local
        State :nat,
        NAI, Mtcd_gi, Rhss:text
        const sec_ssdki, sec_grpk1, mtcd_mme,
        hss_mme, hss_mtcd :protocol_id,
        success: text
  init  State := 0
  transition
  1. State = 0 /\ RCV ((Ec(Ssdki.Idg1.Mtcd_gi).
                   TSgrp1.Kidi).(F1({F1(KG1.TSgrp1
                   .Mtcd_gi)}_F1(KG1.TSgrp1.Mtcd_gi)
                   ).NAI'.Grpk1).TSgrp1.NAI')=|>
     State':= 1 /\ Rhss' := new()
                /\ SND ((KDF(F3(Grpk1.Idg1.Rhss'))
                   (F4(Grpk1.Idg1.Rhss')).(F5(Grpk1
                   .Idg1.Rhss')).Idg1.Mtcd_gi).((F1
                   (F3(Grpk1.Idg1.Rhss')).Rhss'.Amf
                   ).Rhss'.Amf). (F1({(F1(F3(Grpk1.
                   Idg1.Rhss')).Idg1.Rhss'.Mtcd_gi)
                   }_F1(F3(Grpk1.Idg1.Rhss')).Idg1
                   .Rhss'.Mtcd_gi)).(F3(Grpk1.Idg1.
                   Rhss')).Kidi.Ssdki)
                /\ secret(Amf,Idg1,sec_ssdki,{H,
                   M,D})
                /\ secret(Rmme,Amf,sec_grpk1,{D,M
                   ,H})
                /\ witness (H, M, hss_mtcd, Rhss')
                /\ request(H,M,D, hss_mme,Rhss')
 end role
```

Role 3. **HSS.**

REFERENCES

[1] N. Saxena, S. Grijalva, and N. S. Chaudhari, "Authentication protocol for an IoT-enabled LTE network," *ACM Trans. Internet Technol.*, vol. 16, no. 4, p. 25, 2016.

[2] A. Aijaz and A. H. Aghvami, "Cognitive machine-to-machine communications for Internet-of-Things: A protocol stack perspective," *IEEE Internet Things J.*, vol. 2, no. 2, pp. 103–112, Apr. 2015.

[3] C. Lai, R. Lu, H. Li, D. Zheng, and X. S. Shen, "Secure machine-type communications in LTE networks," *Wireless Commun. Mobile Comput.*, vol. 16, no. 12, pp. 1495–1509, 2016.

[4] S. Gupta, B. L. Parne, and N. S. Chaudhari, "DGBES: Dynamic group based efficient and secure authentication and key agreement protocol for MTC in LTE/LTE-A networks," *Wireless Pers. Commun.*, pp. 1–33, Oct. 2017, doi: 10.1007/s11277-017-5005-6.

[5] F. Ghavimi and H.-H. Chen, "M2M communications in 3GPP LTE/LTE-A networks: Architectures, service requirements, challenges, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 525–549, 2nd Quart., 2015.

[6] S. M. R. Islam, D. Kwak, M. Humaun Kabir, M. Hossain, and K.-S. Kwak, "The Internet of Things for health care: A comprehensive survey," *IEEE Access*, vol. 3, pp. 678–708, Jun. 2015.

[7] R. P. Jover, "Security and impact of the IoT on LTE mobile networks," in *Security and Privacy in the Internet of Things (IoT): Models, Algorithms, and Implementations*, vol. 6. Boca Raton, FL, USA: CRC Press, 2015.

[8] ABIresearch. (2013). *Internet of Everything Market Tracker*. [Online]. Available: https://www.abiresearch.com/market-research/product/1017642-internet-of-everything-market-tracker/

[9] H. A. H. Hassan, A. Pelov, and L. Nuaymi, "Integrating cellular networks, smart grid, and renewable energy: Analysis, architecture, and challenges," *IEEE Access*, vol. 3, pp. 2755–2770, 2015.

[10] *Technical Specification Group Services and System Aspects; Security Aspects of Machine-Type Communications (MTC) (Release 11)*, document 3GPP TR 33.868 VO.7.0, 3GPP, Valbonne, France, 2012.

[11] C. Lai, R. Lu, D. Zheng, H. Li, and X. Shen, "Toward secure large-scale machine-to-machine comm unications in 3GPP networks: Challenges and solutions," *IEEE Commun. Mag.*, vol. 53, no. 12, pp. 12–19, Dec. 2015.
[12] P. Roychoudhury, B. Roychoudhury, and D. K. Saikia, "Hierarchical group based mutual authentication and key agreement for machine type communication in LTE and future 5G networks," *Security Commun. Netw.*, vol. 2017, Jan. 2017, Art. no. 1701243.
[13] *Technical Specification Group Services and System Aspects; Service Requirements for the Evolved Packet System (EPS); (Release 13)*, document 3GPP TS 22.278 V13.2.0, 3GPP, Aug. 2014.
[14] *Technical Specification Group Services and System Aspects; 3GPP System Architecture Evolution (SAE); Security Aspects of Non-3GPP Accesses (Release 11)*, document 3GPP TS 33.402 V11.4.0, 3GPP, Jun. 2012.
[15] T. Taleb and A. Kunz, "Machine type communications in 3GPP networks: Potential, challenges, and solutions," *IEEE Commun. Mag.*, vol. 50, no. 3, pp. 178–184, Mar. 2012.
[16] J. Mišić, V. B. Mišić, and N. Khan, "Sharing it my way: Efficient M2M access in LTE/LTE-A networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 696–709, Jan. 2017.
[17] S. Mavoungou, G. Kaddoum, M. Taha, and G. Matar, "Survey on threats and attacks on mobile networks," *IEEE Access*, vol. 4, pp. 4543–4572, 2016.
[18] F. B. Degefa, D. Lee, J. Kim, Y. Choi, and D. Won, "Performance and security enhanced authentication and key agreement protocol for SAE/LTE network," *Comput. Netw.*, vol. 94, pp. 145–163, Jan. 2016.
[19] K.-R. Jung, A. Park, and S. Lee, "Machine-type-communication (MTC) device grouping algorithm for congestion avoidance of MTC oriented LTE network," in *Security-Enriched Urban Computing and Smart Grid*. Berlin, Germany: Springer, 2010, pp. 167–178.
[20] 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.
[21] C. Lai, H. Li, R. Lu, and X. S. Shen, "SE-AKA: A secure and efficient group authentication and key agreement protocol for LTE networks," *Comput. Netw.*, vol. 57, no. 17, pp. 3492–3510, 2013.
[22] R. Jiang, C. Lai, J. Luo, X. Wang, and H. Wang, "EAP-based group authentication and key agreement protocol for machine-type communications," *Int. J. Distrib. Sensor Netw.*, vol. 9, no. 11, p. 304601, 2013.
[23] C. Lai, H. Li, X. Li, and J. Cao, "A novel group access authentication and key agreement protocol for machine-type communication," *Trans. Emerg. Telecommun. Technol.*, vol. 26, no. 3, pp. 414–431, 2015.
[24] D. Choi, H.-K. Choi, and S.-Y. Lee, "A group-based security protocol for machine-type communications in LTE-advanced," *Wireless Netw.*, vol. 21, no. 2, pp. 405–419, 2015.
[25] J. Cao, M. Ma, and H. Li, "GBAAM: Group-based access authentication for MTC in LTE networks," *Secur. Commun. Netw.*, vol. 8, no. 17, pp. 3282–3299, 2015.
[26] A. Fu, J. Song, S. Li, G. Zhang, and Y. Zhang, "A privacy-preserving group authentication protocol for machine-type communication in LTE/LTE-A networks," *Secur. Commun. Netw.*, vol. 9, no. 13, pp. 2002–2014, 2016.
[27] C. Lai, R. Lu, D. Zheng, H. Li, and X. S. Shen, "GLARM: Group-based lightweight authentication scheme for resource-constrained machine to machine communications," *Comput. Netw.*, vol. 99, pp. 66–81, Apr. 2016.
[28] J. Li, M. Wen, and T. Zhang, "Group-based authentication and key agreement with dynamic policy updating for MTC in LTE-A networks," *IEEE Internet Things J.*, vol. 3, no. 3, pp. 408–417, Jun. 2016.
[29] J. Yao, T. Wang, M. Chen, L. Wang, and G. Chen, "GBS-AKA: Group-based secure authentication and key agreement for M2M in 4G network," in *Proc. IEEE Int. Conf. Cloud Comput. Res. Innov. (ICCCRI)*, May 2016, pp. 42–48.
[30] J. Lee *et al.*, "LTE-advanced in 3GPP Rel-13/14: An evolution toward 5G," *IEEE Commun. Mag.*, vol. 54, no. 3, pp. 36–42, Mar. 2016.
[31] M. Villarreal-Vasquez, B. Bhargava, and P. Angin, "Adaptable safety and security in V2X systems," in *Proc. IEEE Int. Congr. Internet Things (ICIOT)*, Jun. 2017, pp. 17–24.
[32] V. Fajardo, J. Arkko, J. Loughney, and G. Zorn, *Diameter Base Protocol*, document RFC 6733, Oct. 2012. [Online]. Available: https://tools.ietf.org/html/rfc6733
[33] *Technical Specification Group Services and System Aspects; Service Requirements for Machine-Type Communication (MTC); (Release 13)*, document 3GPP TS 22.368 V13.1.0, 3GPP, Dec. 2014.
[34] A. T. Sherman and D. A. McGrew, "Key establishment in large dynamic groups using one-way function trees," *IEEE Trans. Softw. Eng.*, vol. 29, no. 5, pp. 444–458, May 2003.
[35] AVISPA. (2003). *AVISPA Automated Validation of Internet Security Protocols*. [Online]. Available: http://www.avispa-project.org
[36] D. Shin, V. Sharma, J. Kim, S. Kwon, and I. You, "Secure and efficient protocol for route optimization in PMIPv6-based smart home IoT networks," *IEEE Access*, vol. 5, pp. 11100–11117, 2017.

BALU L. PARNE (S'17) received the bachelor's degree from the Shri Sant Gajanan Maharaj College of Engineering, Shegaon, that is affiliated to Sant Gadge Baba Amravati University, Amravati, India, and the master's degree from the National Institute of Technology, Rourkela, India. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Visvesvaraya National Institute of Technology, Nagpur, India. His current areas of research are wireless communication, network security, Internet of Things, and mobile computing and its applications.

SHUBHAM GUPTA (S'17) received the B.Tech. degree in information technology from Uttar Pradesh Technical University, Lucknow, and the M.Tech. degree in computer science and engineering from the University College of Engineering, RTU, Kota, India. He is currently pursuing the Ph.D. degree in computer science and engineering from the Visvesvaraya National Institute of Technology, Nagpur, India. His research interests include security in cellular networks, machine type communication, wireless communication networks, and mobile computing.

NARENDRA S. CHAUDHARI (M'88–SM'10) received the bachelor's, master's, and Ph.D. degrees from IIT Bombay, Mumbai, India, in 1981, 1983, and 1988, respectively. He has done significant research work on game AI, novel neural network models like binary neural nets and bidirectional nets, graph isomorphism problem, security of the wireless mobile communication, mobile computing, and Internet of Things. He has successfully completed eight Research and Development Projects funded by DST, UGC, AICTE, MHRD, etc. He is a recipient of the Eminent Engineer Award (Computer Engineering) of the Institution of Engineers, India (IE-India) and the Bharat Vidya Shiromani Award (with Gold Medal). He is a Fellow of the Institution of Engineers, India, and the Institution of Electronics and Telecommunication Engineers, India, a Senior Member of the Computer Society of India, and a member of the Indian Mathematical Society, Cryptology Research Society of India, and many other professional societies. He is a Referee and a Reviewer for a number of premier conferences and journals including IEEE Transactions, *Neurocomputing*, etc.