The ISC Int'l Journal of Information Security

July 2020, Volume 12, Number 2 (pp. 101–111)

http://www.isecure-journal.org

GSLHA: Group-based Secure Lightweight Handover Authentication Protocol for M2M Communication **

Mohammad Mahdi Modiri [1,*], Javad Mohajeri [2], and Mahmoud Salmasizadeh [2]

[1] *Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran*

[2] *Electronics Research Institute, Sharif University of Technology, Tehran, Iran*

ARTICLE INFO.

Article history:

Received: 27 December 2019

Revised: 31 December 2019

Accepted: 27 May 2020

Published Online: 31 May 2020

Keywords:

IoT, Network Security, M2M Communication, Group-based Handover Authentication, AVISPA.

Abstract

Machine to machine (M2M) communication, which is also known as machine type communication (MTC), is one of the fascinating parts of mobile communication technology and also an important practical application of the Internet of Things. The main objective of this type of communication is handling massive heterogeneous devices with low network overheads and high security guarantees. Hence, various protocols and schemes were proposed to achieve security requirements in M2M communication and reduce computational and communication costs. In this paper, we propose the group-based secure, lightweight handover authentication (GSLHA) protocol for M2M communication in LTE and future 5G networks. The proposed protocol mutually authenticates a group of MTC devices (MTCDs) and a new eNodeB (eNB) when these simultaneously enter the coverage of the eNB with considering all the cellular network requirements. The security analysis and formal verification by using the AVISPA tool show that the proposed protocol has been able to achieve all the security goals and overcome various attacks. In addition, the comparative performance analysis of the handover authentication protocols shows that the proposed GSLHA protocol has the best computational and communication overheads.

1 Introduction

Machine type communication (MTC) also known as machine to machine (M2M) communication is as one of the most important technologies for future wireless communication. This type of communication has attracted a large amount of attentions and has been a tremendous growth during the last years. It is developed to supply secure and wide communications between MTC devices (MTCDs) with improving efficiency and reducing costs. Moreover, due to its feature of no human intervention and lower network overheads, it will play a key role in designing of next generations of mobile cellular technology. There are many applications for M2M communication such as smart cities, health-care services, industrial automation, smart electricity grids, fleet management and so on [1–3].

In the cellular networks, when a device moves away from current eNodeB (eNB) to a new eNB, it is necessary to re-authenticate it with considering all the security requirements and low network overheads. When

* Corresponding author.

**This article is an extended/revised version of an ISCISC'16 paper.

Email address: m.modiri96@student.sharif.edu

ISSN: 2008-2045

a large number of MTCDs simultaneously roam from an eNB to a new eNB, the current handover mechanisms [4, 5] authenticate them separately which in this case the network is faced to high computational and communication overheads. Hence, it is necessary to propose group-based handover authentication protocols to authenticate simultaneously a group of MTCDs for reducing network overheads.

The main challenges to design the group-based handover authentication protocols in M2M communication are providing the MTCDâĂŹs security and privacy [6]. In addition, the MTCDs such as smart cards and chips, have limited communication and computing capabilities [7]. Hence, the protocols designed in this field should be able to achieve all the security requirements with low computational and communication overheads.

Until now, there is only a little research on the group-based handover authentication in cellular networks. In this paper, we present the group-based handover authentication GSLHA protocol that has been able to achieve all the security goals and overcome known attacks. Moreover, the protocol uses only hash functions during the authentication process, and for this reason, it has very low communication and computational overheads compared with other handover authentication protocols.

The subsequent sections of this paper are the following. Section 2 discusses the related works of the group-based handover authentication protocols. In Section 3, we present the proposed GSLHA protocol for M2M communication. Section 4 formally verifies the proposed protocol using the AVISPA tool, and Section 5 analyzes the security of the proposed protocol. The performance analysis of the protocol and other group-based handover authentication protocols is presented in Section 6. At the end, Section 7 presents the conclusion.

2 Related Work

In the group-based handover authentication protocols, the new eNB authenticates a group of MTCDs when these roam to the coverage of the eNB. In this section, we present an overview of the research works on group-based handover authentication provided for LTE-A and 5G networks.

The first group-based handover authentication scheme for M2M communication in LTE-A networks was presented by Cao *et al.* [8] with the name of GAHAP. In this scheme, the new eNB uses the first authenticated MTCDâĂŹs data to authenticate all other the MTCDs of the group. Hence, the handover authentication process is reduced for other MTCDs. However, this scheme suffers from the network signal congestion problem when a mass of MTCDs simultaneously require handover authentication, and also it cannot achieve mutual authentication. Similar to the GAHAP scheme, the same authors presented UGHA [9] scheme. This scheme authenticates mutually and simultaneously a group of MTCDs along with reducing network signaling congestion compared with [8]. However, both of the schemes have high network overheads.

Afterward, Kong *et al.* [10] presented the secure handover session key management protocol in LTE-A networks. This protocol achieves key forward and backward secrecy of eNB session keys. However, it suffers from various security attacks and brings high computational overhead due to the use of pairing operations.

Recently, Cao *et al.* [11] proposed the UPPGHA protocol that authenticates a group of MTCDs simultaneously and can be applied to all of the scenarios for mobility between eNBs in LTE-A networks. This scheme has been able to achieve all the security requirements and resist against most of the known attacks, but it cannot reduce communication and computational overheads largely .

According to the above protocols problems, we present the group-based secure, lightweight handover authentication GSLHA proto-col for M2M communication in LTE and 5G networks. The proposed protocol provides the privacy of each MTCD and security key derivation in all of the handover scenarios. The protocol successfully achieves all security goals and overcomes known attacks. In addition, due to the use of only hash functions during the authentication process, the proposed protocol has very low network overheads in comparison with all other group-based handover authentication protocols.

3 The Proposed GSLHA Protocol

In this section, we describe the proposed GSLHA protocol for handover authentication in LTE and 5G networks. In this protocol, MTCDs on the same local area set up a group, and then one of these MTCDs with high computational and communication resources such as smartphones or wireless modems is chosen as the group leader. The duty of the group leader is transferring and receiving messages to/from the MTCDs and calculating and verifying the group message authentication codes (MACs).

The proposed protocol consists of the initial group-based authentication and key agreement phase and the group-based handover authentication phase. The used symbols and their interpretation in the proposed protocol and their size are presented in Table 1. Note that, in these symbols, the index G_1 represents the

Table 1. Notations and Symbols

Notation	Description	Size(bits)
$IMSI_{G_{1-i}}$	International Mobile Subscriber ID	128
$TMSI_{G_{1-i}}$	Temporary Mobile Subscriber ID	128
ID_{G_1}	Group Identity	128
TID_{G_1}	Group Identity	128
ID_{SeNB}	SeNB identity	128
ID_{TeNB}	TeNB Identity	128
K_{ASME}	Access Security Management Key	256
K_{eNB}	eNB Key	128
TK_{G_1}	Group Temporary Key	128
$MAC/XMAC$	Message Authentication Code	64
$RAND$	Random Number	128
$KDF(.)$	Key Derivation Function	128
$H(.)$	Hash Function	128

group G_1 and the index G_{1-i} represents the i'th member of the group G_1.

3.1 Initial Authentication and Key Agreement Phase

In the initial phase, the group of MTCDs implements the group-based authentication and key agreement GSL-AKA [12] protocol to achieve initial access to the network. After the successful GSL-AKA protocol process, the network assigns to the group a group temporary key (TK_{G_1}) and a group temporary ID (TID_{G_1}). Moreover, it assigns to each MTCD a secret session key ($K_{ASME}^{MTCD_{G_{1-i}}}$) and a temporary IMSI ($TMSI_{G_{1-i}}$).

In the proposed protocol, to achieve a secure communication between the group and the eNB, the MME and the group shall derive an eNB group key ($K_{eNB}^{G_1}$) from the group temporary key (TK_{G_1}). Moreover, the MME and each MTCD derive an eNB key ($K_{eNB}^{MTCD_{G_{1-i}}}$) from the shared secret session key ($K_{ASME}^{MTCD_{G_{1-i}}}$).

In handovers, when the group roams from the source eNB (SeNB) to the target eNB (TeNB), the SeNB derives a new group temporary key ($K_{eNB}^{G_1}{}^{*}$) for the group and a new eNB key ($K_{eNB}^{MTCD_{G_{1-i}}{}^{*}}$) for each MTCD as follows and sends these keys to the TeNB network.

$$K_{eNB}^{MTCD_{G_{1-i}}{}^{*}} = KDF(K_{eNB}^{G_1}||K_{eNB}^{MTCD_{G_{1-i}}}|| ID_{TeNB}||TMSI_{G_{1-i}}) \quad (1)$$

$$K_{eNB}^{G_1}{}^{*} = (K_{eNB}^{G_1}||ID_{TeNB}||TID_{G_1}) \quad (2)$$

Afterwards, the TeNB assigns to the group new group temporary key ($K_{eNB}^{G_1}{}^{**}$) from the $K_{eNB}^{G_1}{}^{*}$ and assigns to each MTCD new eNB key ($K_{eNB}^{MTCD_{G_{1-i}}{}^{**}}$) from the $K_{eNB}^{MTCD_{G_{1-i}}{}^{*}}$ as describes in the group-based handover authentication phase.

3.2 Group-based Handover Authentication Phase

This phase includes the GSLHA protocol, which mutually and simultaneously authenticates a group of MTCDs and the TeNB with achieving all the security requirements and reducing costs. According to the 3GPP committee technical specifications, there are various scenarios for mobility between eNBs, which are divided into three categories: X2-based handovers, intra-MME handovers, and inter-MME handovers [4]-[5]. The proposed protocol is fit for all the mobility scenarios in the cellular networks. In Figure 1, we illustrate the structure of the GSLHA protocol, and the details are as follows:

Step-1: When a group of MTCDs moves into the coverage of the TeNB, these send a request to the SeNB for handovering to the TeNB.

Step-2: The SeNB performs the following steps according to the handover scenario. Note that, the communication path between eNBs and MMEs is assumed secure.

Step-2-1: X2-based Handover: When the MTCDs roam between eNBs with an X2 interface.

Step-2-1-1: The SeNB generates the $K_{eNB}^{G_1}{}^{*}$ and $K_{eNB}^{MTCD_{G_{1-i}}{}^{*}}$ from Equation 1 and Equation 2 and sends (TID_{G_1}, $K_{eNB}^{G_1}{}^{*}$, ($TMSI_{G_{1-i}}$, $K_{eNB}^{MTCD_{G_{1-i}}{}^{*}}$)$_{i=1,...,n}$) to the TeNB.

Step-2-1-2: Then, the TeNB derives a new group temporary key ($K_{eNB}^{G_1}{}^{**}$) and a new eNB key ($K_{eNB}^{MTCD_{G_{1-i}}{}^{**}}$) as follows:

$$K_{eNB}^{MTCD_{G_{1-i}}{}^{**}} = KDF(K_{eNB}^{G_1}{}^{*} ||K_{eNB}^{MTCD_{G_{1-i}}{}^{*}}||ID_{TeNB}||TMSI_{G_{1-i}}) \quad (3)$$

$$K_{eNB}^{G_1}{}^{**} = (K_{eNB}^{G_1}{}^{*}||ID_{TeNB}||TID_{G_1}) \quad (4)$$

Step-2-2: Intra-MME Handover: When the MTCDs roam between eNBs managed by same MME without an X2-based interface.

Step-2-2-1: The SeNB generates the $K_{eNB}^{G_1}{}^{*}$ and $K_{eNB}^{MTCD_{G_{1-i}}{}^{*}}$ from (Equation 1) and

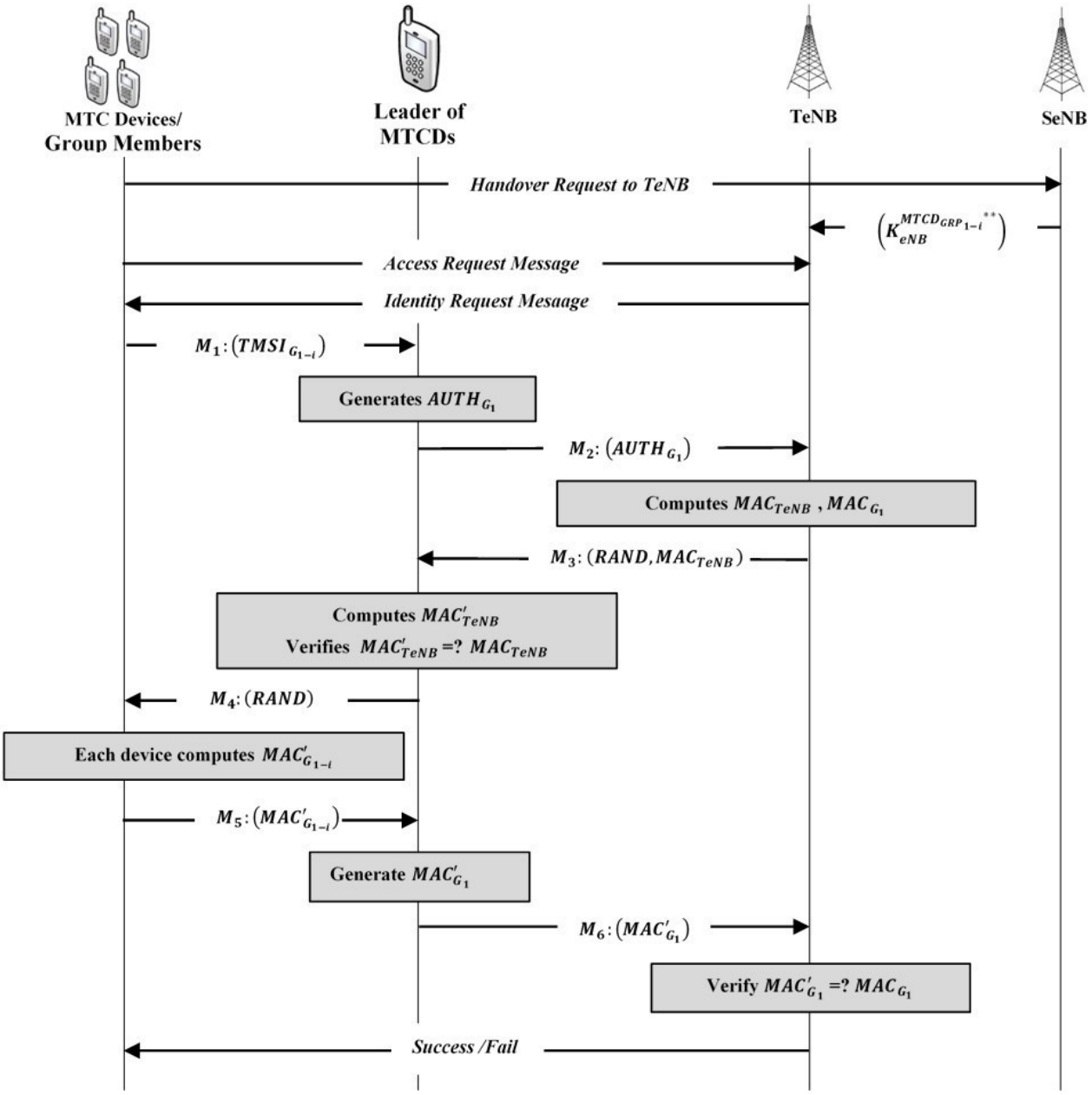


Figure 1. The GSLHA protocol

(2) and sends $(TID_{G_1}, K_{eNB}^{G_1*}, (TMSI_{G_{1-i}}, K_{eNB}^{MTCD_{G_{1-i}}*})_{i=1,\ldots,n})$ to the MME.

Step-2-2-2: The MME computes the $K_{eNB}^{MTCD_{G_{1-i}}*+}$ and $K_{eNB}^{G_1*+}$ as shown in (5) and (6) and sends $(TID_{G_1}, K_{eNB}^{G_1*+}, (TMSI_{G_{1-i}}, K_{eNB}^{MTCD_{G_{1-i}}*+})_{i=1,\ldots,n})$ to the TeNB.

$$K_{eNB}^{MTCD_{G_{1-i}}*+} = KDF(TK_{G_1} ||K_{ASME}^{MTCD_{G_{1-i}}}||K_{eNB}^{MTCD_{G_{1-i}}*}) \quad (5)$$

$$K_{eNB}^{G_1*+} = (TK_{G_1}||K_{eNB}^{G_1*}) \quad (6)$$

Step-2-2-3: Finally, the TeNB derives a new group temporary key $(K_{eNB}^{G_1**})$ and a new eNB key $(K_{eNB}^{MTCD_{G_{1-i}}**})$ of each MTCD as follows:

$$K_{eNB}^{MTCD_{G_{1-i}}**} = KDF(K_{eNB}^{G_1*+} ||K_{eNB}^{MTCD_{G_{1-i}}*+}||ID_{TeNB}||TMSI_{G_{1-i}}) \quad (7)$$

$$K_{eNB}^{G_1**} = (K_{eNB}^{G_1*+}||ID_{TeNB}||TID_{G_1}) \quad (8)$$

Step-2-3: Inter-MME Handover: When the MTCDs roam between eNBs managed by different MME.

Step-2-3-1: same as ***Step-2-2-1***.

Step-2-3-2: The SMME (source MME) sends $(TID_{G_1}, TK_{G_1}, {K_{eNB}^{G_1}}^{*}, (TMSI_{G_{1-i}}, K_{ASME}^{MTCD_{G_{1-i}}}, {K_{eNB}^{MTCD_{G_{1-i}}}}^{*})_{i=1,...,n})$ to the TMME (target MME).

Step-2-3-3: The TMME computes the ${K_{eNB}^{MTCD_{G_{1-i}}}}^{*+}$ and ${K_{eNB}^{G_1}}^{*+}$ as shown in (5) and (6) and sends $(TID_{G_1}, {K_{eNB}^{G_1}}^{*+}, (TMSI_{G_{1-i}}, {K_{eNB}^{MTCD_{G_{1-i}}}}^{*+})_{i=1,...,n})$ to the TeNB.

Step-2-3-4: same as ***Step-2-2-3***.

Step-3: The MTCDs request access to the TeNB through the group leader ($MTCD_{G_{1-leader}}$).

Step-4: Then, the TeNB requests the identities of each MTCD and the group from the $MTCD_{G_{1-leader}}$.

Step-5: The $MTCD_{G_{1-leader}}$ generates the identity response message ($AUTH_{G_1}$) as:

- Each $MTCD_{G_{1-i}}$ forwards its temporary IMSI ($TMSI_{G_{1-i}}$) to the $MTCD_{G_{1-leader}}$.
- The $MTCD_{G_{1-leader}}$ generates $AUTH_{G_1}$ as follow and transmits it to the TeNB.

$$AUTH_{G_1} = (TID_{G_1}||(TMSI_{G_{1-i}})_{i=1,...,n} ||ID_{SeNB}) \quad (9)$$

Step-6: After acquiring $AUTH_{G_1}$, TeNB performs the following:

- TeNB generates a random number ($RAND$) and calculates the MAC_{TeNB} as:

$$MAC_{TeNB} = ({K_{eNB}^{G_1}}^{**}||ID_{TeNB} ||RAND) \quad (10)$$

- TeNB generates the authentication code for each $MTCD_{G_{1-i}}$ as:

$$MAC_{G_{1-i}} = ({K_{eNB}^{MTCD_{G_{1-i}}}}^{**} ||ID_{TeNB}||RAND) \quad (11)$$

- TeNB generates the aggregated authentication code for the group as:

$$MAC_{G_1} = ({K_{eNB}^{G_1}}^{**}||ID_{TeNB} ||MAC_{G_{1-1}}||...||MAC_{G_{1-n}}) \quad (12)$$

- Finally, the TeNB transmits ($RAND$, MAC_{TeNB}) to the $MTCD_{G_{1-leader}}$.

Step-7: After acquiring ($RAND$, MAC_{TeNB}), the $MTCD_{G_{1-leader}}$ performs as follows:

- $MTCD_{G_{1-leader}}$ computes the new group temporary key (${K_{eNB}^{G_1}}^{**}$) using (4) or (8).
- The $MTCD_{G_{1-leader}}$ generates MAC'_{TeNB} using (10) and compares the computed MAC'_{TeNB} and the received MAC_{TeNB}. If these are equal, the group authenticates the TeNB; otherwise, the authentication process aborts.
- The $MTCD_{G_{1-leader}}$ sends $RAND$ and the successful TeNB authentication message to each $MTCD_{G_{1-i}}$.

Step-8: Now, each $MTCD_{G_{1-i}}$ performs as follows:

- Each $MTCD_{G_{1-i}}$ computes the new eNB key (${K_{eNB}^{MTCD_{G_{1-i}}}}^{**}$) using (3) or (7).
- Each $MTCD_{G_{1-i}}$ generates $MAC'_{G_{1-i}}$ using (11) and sends it to the $MTCD_{G_{1-leader}}$.

Step-9: The $MTCD_{G_{1-leader}}$ generates MAC'_{G_1} using (12) and sends it to the TeNB.

Step-10: Finally, TeNB compares MAC'_{G_1} sent from the $MTCD_{G_{1-leader}}$ with the computed MAC_{G_1}. If these are matched, the TeNB sends the successful authentication message to each $MTCD_{G_{1-leader}}$. If not, the TeNB sends the failed authentication message.

4 Formal Verification of The Proposed Protocol

The proposed GSLHA protocol is formally verified using the AVISPA tool and coded in High-Level Protocol Specifications Language (HLPSL) to examine various security properties. The mutual authentication between the TeNB and each MTCD is the main objective of the protocol. Moreover, it is required to preserve the secret eNB key of each MTCD ($K_{eNB}^{MTCD_{G_{1-i}}}$), and the group temporary key (TK_{G_1}) during the authentication process. The defined goals of the proposed protocol are shown in Figure 2. In this protocol, there are three parties: MTCD, TeNB, and SeNB. The description of these parties in the HLPSL code is in appendix I. Moreover, the path between the eNBs and between the eNB to MME is assumed secure, and an attacker can only dominate the path between the group members to the group leader and between the group leader to eNBs.

We analyzed the proposed protocol using OFMC and CL-AtSe backend embedded in the AVISPA tool, and the results of them are shown in Figure 3 and Figure 4, respectively. The results verify that the

```
goal
  secrecy_of sec_kenbi,sec_tkg1
  authentication_on mtcd_tenb
end goal
```

Figure 2. The goals of the proposed GSLHA protocol

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/GSLHA.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.01s
  visitedNodes: 9 nodes
  depth: 8 plies
```

Figure 3. Result summarized by OFMC backend.

```
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
PROTOCOL
  /home/span/span/testsuite/results/GSLHA.if
GOAL
  As Specified
BACKEND
  CL-AtSe
STATISTICS
  Analysed   : 5 states
  Reachable  : 3 states
  Translation: 0.00 seconds
  Computation: 0.00 seconds
```

Figure 4. Result summarized by CL-AtSe backend.

proposed GSLHA protocol achieves the defined goals and resists against all the known attacks.

5 Security Analysis

In this section, we present the security analysis of the proposed GSLHA protocol in terms of achieving security goals, preserving the privacy of each MTCD and resisting against known attacks. In these security analysis, it is assumed that the channel between the eNBs and between the eNB to MME are secure and an adversary only dominates the channel between the MTCDs to group leader and between the group leader to eNBs. So, the adversary can eavesdrop and modify the messages transmitted in these insecure channels and attacks to the protocol. The security analysis of the proposed protocol are discussed as follows:

- ***Mutual authentication***: The proposed GSLHA protocol uses message authentication codes (MAC_{G_1} and MAC_{TeNB}, Eq.12, and Eq.10 respectively) to obtain mutual authentication between each MTCD and the TeNB. As it can be seen, these two MACs are generated by using pre-shared secret keys ($K_{eNB}^{MTCD_{G_{1-i}}{}^{**}}$ and $K_{eNB}^{G_1{}^{**}}$) and then only corresponding MTCD and TeNB could generate these MACs. So, an adversary can never generate them and authenticate his/herself to the network entities. Moreover, some random numbers ($RAND$) are embedded in the MAC generator functions, and an adversary can never reuse previous MACs for him/herself authentication. Then, the proposed protocol could successfully achieve mutual authentication between each MTCD and the TeNB.
- ***Key agreement***: In the proposed protocol, in each authentication process, according to the handover type, the session key between each MTCD and the TeNB ($K_{eNB}^{MTCD_{G_{1-i}}{}^{**}}$) is dynamically re-generated (Eq.3 or Eq.7). Moreover, the group temporary key ($K_{eNB}^{G_1{}^{**}}$), according to the Eq.4 or Eq.8, is also re-generated. These keys are generated using one-way hash functions and these are as a function of top-level pre-shared session keys ($K_{eNB}^{MTCD_{G_{1-i}}}$ and $K_{eNB}^{G_1}$). Then, the MTCDs and the eNBs can successfully agree with the new session keys, and there is no way for an adversary to find out these session keys. Moreover, since the session keys are as a function of the secret keys and these are generated using one-way hash functions, then there is no way to link the current session keys with previous and next keys.
- ***Privacy preserving***: In this protocol, each MTCD and the group, use temporary IMSI ($TMSI_{G_{1-i}}$) and group temporary ID (TID_{G_1}), instead of their original values. These two values are assigned to each MTCD and the group after implementing the initial phase, implementing the GSL-AKA protocol [10]. How these two temporary values are produced and assigned to the group, and each MTCD, are fully explained in section 3.2 of the GSL-AKA protocol paper. As it can be seen, the $TMSI_{G_{1-i}}$ and TID_{G_1} are generated using one-way hash functions, and an adversary never able to trace the original values. Hence, the privacy is well protected during the authentication process of the protocol.
- ***Network signaling congestion prevention***: In the proposed protocol, each MTCD generates its unique message authentication code ($MAC_{G_{1_i}}$, Eq.11) and sends it to the group leader. To prevent the problem of network signaling congestion, the group leader aggregates these MACs into one MAC (MAC_{G_1}, Eq.12) and then sends it to the network channel. In addition, the proposed

protocol uses only one MAC (MAC_{TeNB}, Eq.10) to authenticate the TeNB to all of the group members. Then, the protocol successfully could prevent the network from signaling congestion.

- ***Resistance against MiTM attack***: In the man-in-the-middle (MiTM) attacks, an adversary tries to generate valid message authentication codes (MACs) of a user to authenticate him/herself to the network, instead of the user. The proposed protocol generates the authentication codes (MAC_{TeNB}, MAC_{G1_i}, MAC_{G_1}, Eq.10, Eq.11 and Eq.12 respectively) using secret session keys ($K_{eNB}^{G_1}{}^{**}$ and $K_{eNB}^{MTCD_{G_{1-i}}}{}^{**}$) and an adversary can never generate these authentication codes without knowing them. Then, an adversary cannot implement a MiTM attack and authenticate him/herself as a valid user.
- ***Resistance against impersonation attack***: In the proposed protocol, the new session keys between the group members and the network ($K_{eNB}^{MTCD_{G_{1-i}}}{}^{**}$ and $K_{eNB}^{G_1}{}^{**}$) are generated using pre-shared secret keys (TK_{G_1}, $K_{ASME}^{MTCD_{G_{1-i}}}$) and other secret data. Eq.3, Eq.7, Eq.4, and Eq.8 explain how these session keys are generated. For this reason, an adversary cannot generate these keys and modify and change the secret messages transmitted between the group members and the eNB. Moreover, an adversary can never generate the aggregated message authentication code (MAC_{G1_i}, Eq.12) and impersonate him/herself to the network as a legal group.
- ***Resistance against redirection attack***: In the redirection attacks, an adversary establishes an illegal base station near a user and impersonates as a valid base station to access user secret data. To avoid from redirection attacks, in the proposed protocol, the ID of the connected eNB (ID_{TeNB}) are embedded in the authentication codes (MAC_{G_1} and MAC_{TeNB}, Eq.12 and Eq.10 respectively). Then, whenever a redirection attack occurs, the protocol entities, by checking these authentication codes, can find out this malicious act and abort the authentication process. So, the proposed protocol resists against redirection attacks.
- ***Resistance against replay attack***: The proposed protocol uses random numbers ($RAND$) for generating authentication codes (MAC_{G_1} and MAC_{TeNB}, Eq.12 and Eq.10 respectively). Then, an adversary, by eavesdropping and catching these authentication codes, can never reuse these codes to authenticate him/herself to other entities.
- ***Resistance against DoS attack***: In the denial-of-service (DoS) attacks, an adversary sends invalid data to the victim to disrupt its actions. In all of the AKA protocols, such as EPS-AKA [13], 5G-AKA [14], GSL-AKA [12] and so on, the usual method for preventing from DoS attacks is generating the message authentication codes as a function of all the transmitted messages. In the proposed GSLHA protocol, for avoiding from DoS attacks, each MTCD and the group generate their authentication codes ($MAC_{G_{1-i}}$, MAC_{G_1}, Eq.11, and Eq.12 respectively) as a function of all the sent messages and then send them to the TeNB. That way, the TeNB can verify the received data and determine whether a DoS attack occurs or not. Moreover, the TeNB generates its authentication codes (MAC_{TeNB}, Eq.10) as a function of all the sent data and then sends them to the group. Then, according to these reasons, there is no way to implement a DoS attack on the proposed protocol. Moreover, In the proposed protocol, there is no trust between the MTCD and the group leader. The pre-mentioned message authentication codes are generated in such a way that whenever a malicious MTCD sends invalid data to the others, the protocol entities can find out this malicious act and prevent it. For instance, in Step 8 of the protocol procedure, whenever a malicious MTCD sends an invalid message authentication code (MAC_{G1_i}, Eq.11) to the group leader, the leader, and the TeNB, can find out this malicious act by calculating MAC'_{G_1} (Eq.12) and expel the malicious MTCD from the group.

The security analysis of the existing group-based handover authentication protocols for M2M communication in cellular networks is shown in Table 2. As it can be seen, the proposed GSLHA protocol has been able to achieve all the defined security properties and resist against known attacks.

6 Performance Analysis

This section analyses comparatively the communication and computational overheads of the proposed GSLHA protocol and the other LTE-A and 5G handover authentication protocols. To compute these overheads, let there are n devises aggregated in m groups.

6.1 Communication Overhead

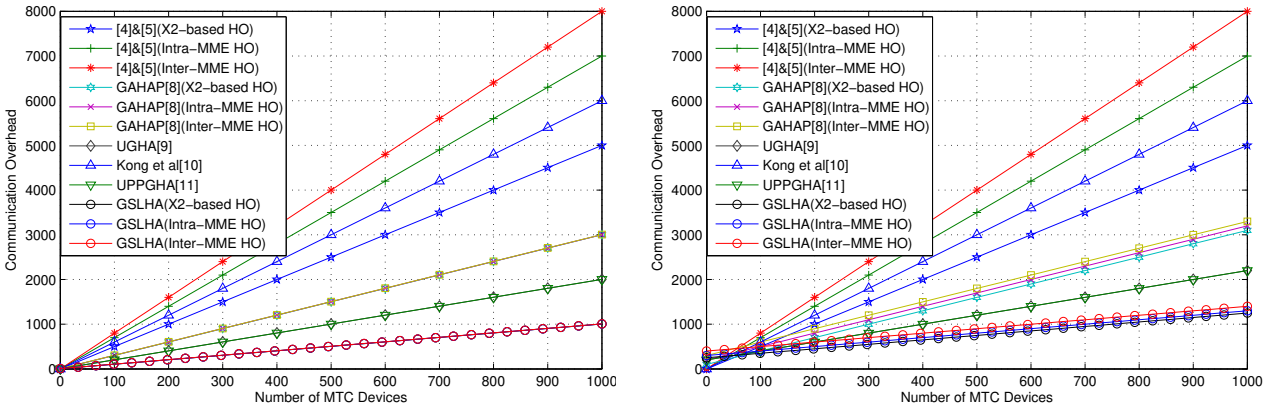
The total messages transmitted by each protocol during its process is the communication overhead of the protocol. The communication overhead of the proposed protocol in each handover scenario are as follows:

In X2-based handover scenario $= n + 5m$,

Table 2. Security Analysis of the Group-based Handover Authentication Protocols

Security properties	Group-based handover authentication protocols					
	[4] & [5]	GAHAP[8]	UGHA [9]	Kong *et al.*[10]	UPPGHA[11]	GSLHA
SP1	✗	✗	✓	✗	✓	✓
SP2	✓	✓	✓	✓	✓	✓
SP3	✗	✓	✗	✗	✓	✓
SP4	✗	✗	✓	✗	✓	✓
SP5	✗	✓	✓	✗	✓	✓
SP6	✗	✓	✓	✗	✓	✓
SP7	✗	✓	✓	✓	✓	✓
SP8	-	✓	✓	✓	✗	✓

SP1: Mutual authentication; ***SP2***: Key agreement; ***SP3***: Privacy preservation; ***SP4***: Prevention from signaling congestion; ***SP5***: Overcome MiTM attack; ***SP6***: Overcome impersonation attack; ***SP7***: Overcome replay attack; ***SP8***: Overcome DoS and insider attacks.



(a) (b)

Figure 5. The communication overheads of the group-based handover authentication protocols for which (a) m=1. (b) m=50.

In Intra-MME handover scenario $= n + 6m$,

And in Inter-MME handover scenario, the communication overhead is equal to $n + 8m$.

Moreover, according to Figure 1 and Table 1, the total bits transmitted in the wireless channel of the proposed protocol is equal to $192n + 640m$ bits.

The communication overheads of the other handover authentication protocols are calculated and illustrated in Table 4. In addition, the comparative analysis of them is shown in Figure 5. As it can be seen, the proposed GSLHA protocol has the lowest communication overhead compared with the all other handover authentication protocols.

6.2 Computational Overhead

The computational overhead generated by each protocol can be calculated by summing the computation time of each cryptographic functions used in the protocol in term of n and m. The computation time of the cryptographic functions used in the protocols are calculated and tested on an Celeron 1.1 GHz processor as an UE and Dual-Core 2.6 GHz as an eNB. These are mentioned in [15] and some of these functions operation time are shown in Table 3. According to the contents of Section 3, the computational overhead of the proposed protocol at the MTC devices is calculated as:

$(T_{hash}) * n + (2T_{hash}) * m$

And at the network is:

$(T_{hash}) * n + (2T_{hash}) * m.$

Thus, the total computational overhead is equal to:

$(2T_{hash}) * n + (4T_{hash}) * m.$

The communication overheads of the other handover authentication protocols are shown in Table 5, and the comparative analysis of them is illustrated in Figure 6. It is observed that the proposed GSLHA protocol has very low computational overhead because it uses only hash-functions during the authentication process.

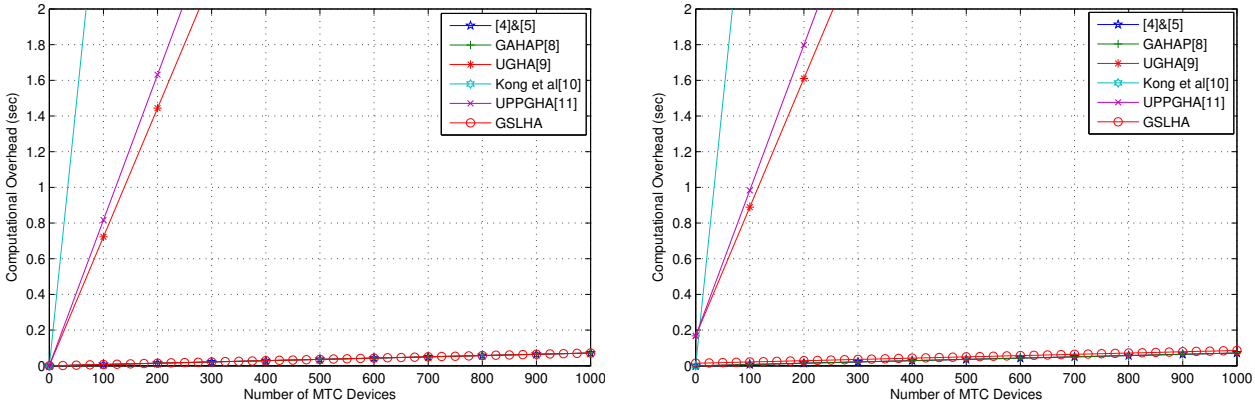


Figure 6. The computational overheads of the group-based handover authentication protocols for which (a) m=1. (b) m=50.

Table 3. The Computation Time of Some of the Cryptographic Functions Calculated in [15]

Notation	Description	Time Costs (ms)
$T_{pair-MTCD}$	pairing at MTCD	38.37
$T_{pair-eNB}$	pairing at eNB	16.32
$T_{hash-MTCD}$	Hash operation at MTCD	0.035
$T_{hash-eNB}$	Hash operation at eNB	0.012
$T_{mod-MTCD}$	Modulus at MTCD	1.698
$T_{mod-eNB}$	modulus at eNB	0.525
$T_{mul-MTCD}$	Point multiplication at MTCD	1.537
$T_{mul-eNB}$	Point multiplication at eNB	0.535
$T_{SYM-MTCD}$	Symmetric encryption at MTCD	0.085
$T_{SYM-eNB}$	Symmetric encryption at eNB	0.029
T_{X-OR}	XOR operation	negligible

Moreover, it is worth noting that the computational overhead of the 3GPP handover mechanism [4]-[5] and the GAHAP scheme [8] are a little less than the computational overhead of our proposed protocol. However, According to Table 2, these schemes could not achieve all of the defined security requirements, but the proposed protocol achieves all of these goals. Then, the proposed GSLHA protocol can achieve all the defined security requirements and also has very low computational overhead.

7 Conclusion and Future Work

In this paper, we propose the GSLHA protocol for Mi2M communication in LTE and 5iG networks. In comparison with other handover authentication protocols, the GSLHA achieves all security requirements, resists against all the known attacks, and preserves the privacy of each MTCD. In addition, the performance analysis of the existing handover authentication protocols shows that the proposed GSLHA protocol has the best computational and communication overheads.

```
role device(
  D,T,S                                   :agent,
  SND,RCV                                 :channel(dy),
  TKG1,KeNBi                              :symmetric_key,
  Handover_Request,Identity_Request,
  TIDG1,TIDi,Rand,IDSeNB,IDTeNB           :text,
  H                                       :hash_func)
played_by D
def=
  local
    State                                 :nat
  const
    sec_kenbi,sec_tkg1,mtcd_tenb          :protocol_id,
    success                               :text
  init State := 0
  transition
    1. State  = 0 /\  RCV(start)=|>
       State':= 1 /\  SND(Handover_Request)
    2. State  = 1 /\  RCV(Identity_Request)=|>
       State':= 2 /\  SND(TIDG1.TIDi.IDSeNB)
    3. State  = 2 /\  RCV(H(TKG1.Rand').Rand')=|>
       State':= 3 /\  SND(H(TKG1.H(KeNBi.Rand')))
                  /\  witness(D,T,mtcd_tenb,Rand')
                  /\  request(D,T,mtcd_tenb,Rand')
    4. State  = 3 /\  RCV(success)=|>
       State':= 4
end role
```

Figure A.1. The role of the MTCDs.

A HLPSL Codes of The Proposed GSLHA Protocol

The basic roles of the MTCDs, SeNB and TeNB of the proposed protocol are shown in Figure A.1, Figure A.2 and Figure A.3, respectively.

References

[1] Nancy L. Russo and Jeanette Eriksson. The Internet of Things and People in Health Care. *Internet of Things A to Z*, page 447âĂŞ474, 2018.

Table 4. The Communication Overhead of the Group-based Handover Authentication Protocols

Communication Overheads	Group-based handover authentication protocols					
	[4] & [5]	GAHAP[8]	UGHA[9]	Kong *et al.*[10]	UPPGHA[11]	GSLHA
X2-based Scenario	$5n$	$3n+2m$	$2n+4m$	$6n$	$2n+4m$	$n+5m$
Intra-MME Scenario	$7n$	$3n+4m$	"	"	"	$n+6m$
Inter-MME Scenario	$8n$	$3n+5m$	"	"	"	$n+8m$

Table 5. The Computational Overhead of the Group-based Handover Authentication Protocols

Computational Overheads	Group-based handover authentication protocols					
	[4] & [5]	GAHAP[8]	UGHA[9]	Kong *et. al.*[10]	UPPGHA[11]	GSLHA
MTC Devices	$(T_{hash})n$	$(T_{hash})n$	$3(T_{mod})n$ $+2(T_{mod})m$	$(4T_{mod}$ $+4T_{pair})n$	$(5T_{mod})n$	$(T_{hash})n$ $+(2T_{hash})m$
Network	$(T_{hash})n$	$(T_{hash})n$	$3(T_{mod})n$ $+2(T_{mod})m$	$(3T_{mod}$ $+2T_{pair})n$	$(T_{mod})n$ $+(4T_{mod})m$	$(T_{hash})n$ $+(2T_{hash})m$
Total (in ms) According to the Table 3	$0.047n$	$0.047n$	$6.67n+4.47m$	$194.49n$	$9.015n+2.1m$	$0.047n+0.094m$

```
role senb(
  D,T,S                                  :agent,
  SND,RCV                                :channel(dy),
  TKG1,KeNBi,SymmetricKey                :symmetric_key,
  Handover_Request,Identity_Request,
  TIDG1,TIDi,Rand,IDSeNB,IDTeNB          :text,
  H                                      :hash_func)
played_by S
def=
  local
    State                                :nat
  const
    sec_kenbi,sec_tkg1,mtcd_tenb         :protocol_id,
    success                              :text
  init State := 0
  transition
    1. State  = 0 /\  RCV(Handover_Request)=|>
       State':= 1 /\  SND({TKG1.KeNBi}_SymmetricKey)
                  /\  secret(KeNBi,sec_kenbi,{T,S})
                  /\  secret(TKG1,sec_tkg1,{T,S})
end role
```

Figure A.2. The role of the SeNB.

```
role tenb(
  D,T,S                                  :agent,
  SND,RCV                                :channel(dy),
  TKG1,KeNBi,SymmetricKey                :symmetric_key,
  Handover_Request,Identity_Request,
  TIDG1,TIDi,Rand,IDSeNB,IDTeNB          :text,
  H                                      :hash_func)
played_by T
def=
  local
    State                                :nat
  const
    sec_kenbi,sec_tkg1,mtcd_tenb         :protocol_id,
    success                              :text
  init State := 0
  transition
    1. State  = 0 /\  RCV({TKG1.KeNBi}_SymmetricKey)=|>
       State':= 1 /\  SND(Identity_Request)
    2. State  = 1 /\  RCV(TIDG1.TIDi.IDSeNB)=|>
       State':= 2 /\  Rand':=new()
                  /\  SND(H(TKG1.Rand').Rand')
                  /\  witness(T,D,mtcd_tenb,Rand')
    3. State  = 2 /\  RCV(H(TKG1.H(KeNBi.Rand')))=|>
       State':= 3 /\  request(T,D,mtcd_tenb,Rand')
                  /\  SND(success)
end role
```

Figure A.3. The role of the TeNB.

[2] SÅĆawomir ÅżÃşÅĆkiewski and Krzysztof Galuszka. Remote Control of Industry Robots Using Mobile Devices. *New Contributions in Information Systems and Technologies Advances in Intelligent Systems and Computing*, page 323âĂŞ332, 2015.

[3] Balu L. Parne, Shubham Gupta, and Narendra S. Chaudhari. SEGB: Security Enhanced Group Based AKA Protocol for M2M Communication in an IoT Enabled LTE/LTE-A Network. *IEEE Access*, 6:3668âĂŞ3684, 2018.

[4] 3rd Generation Partnership Project, âĂIJTechnical Specification Group Services and System Aspects; 3GPP System Architecture Evolution (SAE)âĂİ 3GPP TS 33.401 V15.2.0 , Jan. 2018.

[5] 3rd Generation Partnership Project, âĂIJTechnical Specification Group Radio Access Network,âĂİ Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); (Rel 13), 2016, 3GPP TS 36.300 V13.4.0.

[6] 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.

[7] Muhammad Burhan, Rana Rehman, Bilal Khan, and Byung-Seo Kim. IoT Elements, Layered Architectures and Security Issues: A Comprehensive Survey. *Sensors*, 18(9):2796, 2018.

[8] Jin Cao, Hui Li, and Maode Ma. GAHAP: A group-based anonymity handover authentication protocol for MTC in LTE-A networks. *2015 IEEE International Conference on Communications (ICC)*, 2015.

[9] Jin Cao, Hui Li, Maode Ma, and Fenghua Li. UGHA: Uniform group-based handover authentication for MTC within E-UTRAN in LTE-A networks. *2015 IEEE International Conference on Communications (ICC)*, 2015.

[10] Qinglei Kong, Rongxing Lu, Shuo Chen, and Hui Zhu. Achieve Secure Handover Session Key Management via Mobile Relay in LTE-Advanced

Networks. *IEEE Internet of Things Journal*, page 1âĂŞ1, 2016.

[11] Jin Cao, Hui Li, Maode Ma, and Fenghua Li. UPPGHA: Uniform Privacy Preservation Group Handover Authentication Mechanism for mMTC in LTE-A Networks. *Security and Communication Networks*, 2018:1âĂŞ16, 2018.

[12] Mohammad Mahdi Modiri, Javad Mohajeri, and Mahmoud Salmasizadeh. GSL-AKA: Group-based Secure Lightweight Authentication and Key Agreement Protocol for M2M Communication. *2018 9th International Symposium on Telecommunications (IST)*, 2018.

[13] Mourad Abdeljebbar and Rachid Elkouch. Security analysis of LTE/SAE networks over E-UTRAN. *2016 International Conference on Information Technology for Organizations Development (IT4OD)*, 2016.

[14] 3rd Generation Partnership Project (3GPP) TS 33.501, âĂIJTechnical Specification Group Services and System Aspects; Security architecture and procedures for 5G systemâĂİ,V.15.0.0, March 2018.

[15] Jin Cao, Hui Li, Maode Ma, Yueyu Zhang, and Chengzhe Lai. A simple and robust handover authentication between HeNB and eNB in LTE networks. *Computer Networks*, 56(8):2119âĂŞ2131, 2012.

Mohammad Mahdi Modiri received his B.S. degree in telecommunications engineering from University of Tehran, Tehran, Iran, in 2016. In addition, he received his M.S. degree in secure communication and cryptography engineering from Sharif University of Technology, Tehran, Iran, in 2018. He is currently working toward his Ph.D. degree at the electrical engineering department of Sharif University of Technology in the field of secure communication and cryptography engineering. His research interests include network security, Internet of Things, and cryptography.

Javad Mohajeri received the B.S. degree from Isfahan University in 1986 and the M.S. degree from Sharif University of Technology in 1989, both in mathematics. He has been a faculty member at the Electronics Research Institute of Sharif University of Technology since 1990. His research interests include cryptography and data security. He is the author/co-author of over 60 research articles in refereed Journals/ Conferences.

Mahmoud Salmasizadeh received the B.S. and M.S. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, in 1972 and 1989, respectively. He also received the Ph.D. degree in information technology from Queensland University of Technology, Australia, in 1997. Currently, he is an associate professor in the Electronics Research Institute and adjunct associate professor in the Electrical Engineering Department, Sharif University of Technology. His research interests include design and cryptanalysis of cryptographic algorithms and protocols, e-commerce security, and information theoretic secrecy. He is a founding member of Iranian Society of Cryptology.