A Secure and light-weighted Group based Authentication and Key Agreement Protocol involving ECDH for Machine Type Communications in 3GPP Networks

Geeta Kakarla
Assistant Professor, Dept. of Computer Science and Engineering
Sreenidhi Institute of Science and Technology
Research Scholar, GITAM (Deemed to be University)
Hyderabad, India
geetavemula333@gmail.com

Phanikumar Singamsetty
Professor, Head, Dept. of Computer Science and Engineering
GITAM (Deemed to be University), Hyderabad, India
phanikumar.s@gmail.com

***Abstract*—Machine-Type Communications (MTC) is one of the primary aspects of Internet of Things (IoT) which has gained vast markets and application scenarios. In the recent times, an enormous surge in the number of low cost and low-powered Machine Type Communicating Devices (MTCDs) is observed that tend to connect and cause severe congestions, at times, in the network thus bringing down network's quality of service drastically. Conditionally, all the MTCDs must be mutually authenticated successfully for the corresponding application to be reliable. In the process of authentication and key agreement, the participating MTCDs, high or low configured, need to exchange several control messages with the other participating entities and the servers of the network, leading to overburden and ineffective service delivery. As most of the MTCDs are the constrained devices, lightweight computations are desirable for them without any compromise in its security aspect. In our paper, we have proposed the Secure and Lightweight Group Based Authentication and Key Agreement (SL-Grp-AKA) protocol based on Elliptic-Curve Diffie–Hellman cryptography, where the group of such MTCDs is authenticated simultaneously by the authorized Home Subscriber Server (HSS), also responsible for authentication in the wireless networks. Also, the independent session keys are established with the individual MTCD, to be used for further data transmission securely in our protocol. SL-Grp-AKA guarantees the efficient usage of bandwidth by reducing the congestions caused by the exchange of the control signals. The formal analysis is also performed on SL-Grp-AKA using the well-known AVISPA tool and achieved desirable results and enlisted. The cryptanalysis is performed for establishing its strength as well as the performance evaluation for its computational speed and bandwidth consumption. Towards the end of the paper, we have explored and presented the future scope in this direction.**

***Keywords*—**group authentication, key agreement, machine type communications, subscriber server, mobile management entity, Elliptic-curve Diffie–Hellman (ECDH), AVISPA

I. Introduction

In the recent years, Internet of Things (IoT) has garnered much momentum and playing key role in digital transformation of several businesses, public sectors, health-care industry, and entertainment services and in our day-to-day lives. IoT is visualized as a web of several low cost, resource constrained miscellaneous objects/machines which are connected to the internet and can communicate with each other thus reducing the human intervention and erroneous handling to give the best possible results. The communications that happen among the smart objects are called Machine Type Communications and the machines involved in these communications are termed as Machine Type Communication Devices (MTCDs). However, unlike the traditional Human-to-Human (H2H) communications, thousands of objects would generate a great deal of signaling, consumption and data process information under different environments, resulting in an explosive increase in the data traffic flow among the Radio Access Network (RAN) and the Core Network (CN) as in 3GPP specification [1]. This gives rise to a risk of eavesdropping or manipulation within the Wireless communication rampantly as the data is originally sent from/to a user may be received and unlawfully used by an unintended user. To overcome repudiation events and to protect the traffic between the MTCDs and the network from several attacks, bi-authentication procedures play the significant role. However, with thousands of devices performing the *authentication and key agreement (AKA)* process almost simultaneously, the network faces the heavy burden and congestion and can even lead to the network collapse. In order to tackle the congestion issues, several researchers came up with the group based authentication techniques and schemes, as briefed in the next section, projecting the scheme where group of MTCDs are authenticated simultaneously by the Home Subscriber Server(HSS),through the proxy authority, aka Mobile Management Entity (MME). The issues surfacing from using the existing Authentication and Key Agreement procedures are often seen as the increased number of subdivisions of the MTCDs, additional computational cost involved in electing the Group Leader (GL) among the set of MTCDs and making GL respondent to manage the authentication and key exchange process among its group members with overheads of storage and computing capacity borne by the GL itself, minimal congestion control between group of MTCDs and the MME or the usage of the demanding computations among the set of MTCDs. Addressing these issues, SL-Grp-AKA: a secure and light-weighted group authenticating and key agreement protocol among the machine type communications in 3rd Generation Partnership Project (3GPP) networks is proposed here involving Elliptic Curve Diffie Hellman (ECDH) key agreement protocol. We focussed on the congestion issue among the entities of 3GPP i.e., MTCDs, MME and HSS, that are used to exchange the control messages for the mutual authentication in the heterogeneous architecture where individual MTCD is a low end device having limited computational capability and GL is of higher capabilities as compared to any of the MTCDs.

The organization of this paper is as follows. In Section II, a study is made for the existing authentication and key agreement protocols. The architecture of the 3GPP network is presented in the Section III. In Section IV, the algorithmic details, along with phases, of the proposed protocol: SL-Grp-AKA are discussed. The formal analysis is performed using AVISPA and the programs with outcomes are enlisted in Section V. In the next section, the cryptanalysis is detailed for establishing the strength of the algorithm. The performance evaluation is discussed in Section VII, relative to the other earlier existing protocols. Further, in Section VIII, future scope is explored in addition to the concluding remarks.

II. Existing Authentication and Key Agreement Protocols

The authentication and key agreement protocols consider the tasks of clustering the MTCDs, contextual key generation, key exchange and authentication among the players of the 3GPP network. The groups of MTCDs formed play the significant role in the key distribution. The formation of groups take place based on discreetness criteria as a set of communicating devices installed in a local area premises or the devices owned by the single or common user. In other cases, a set of MTCDs is considered as single group which are transmitting and receiving the data to and from an application and become intrinsic part of the same system. In certain contexts, the group is governed by the nominated group leader and in other cases, the MTCDs of a group act autonomously.

In S-AKA by Yu-Lun Huang et. al.[2], G-AKA [3] and SE-AKA [4], MTC-AKA[5] also by Lai et al., DGBAKA [6] by Zhang et al.,the groups are without the explicit group leader and the first communicating MTCD in the group performs the full round of AKA with MME and HSS, followed by remaining MTCDs performing the AKA process only with MME without the intervention of HSS, as MME has sufficient information to authenticate each and every MTCD in the group. In this, there is enough congestion control in the control plane comprising of the MME and HSS in Evolved Packet Core (EPC), but the congestion problem remains same in the user plane, also termed as Evolved Universal Terrestrial Radio Access Network (E-UTRAN), encompassing MTCDs and Evolved Node Base Stations (eNBs) . GLARM by Chengzhe Lai et. al. [7], SEGR [8] , GAKALTE [9], GRAKA by Jinguo Li et. al. [10], LGTH by Chengzhe Lai et. al. [11] follow a commonality of suggesting the GL for the each group of MTCDs which is responsible for aggregating all the authentication requests from individual MTCDs and forward as the aggregated request to HSS channeling through the MME. HSS in turn produces the key generation information meant for each group member and securely conveys the same to the group members through the MME and the GL, to derive their own key. After each round of AKA using this technique, all the individual MTCDs of a group calculate their corresponding independent secret session key simultaneously from the information shared and thus authenticated by the HSS. But these schemes rely on the symmetric key shared between devices and controlling authority which makes these schemes vulnerable to brute force attacks.

In HGMAKA by Probidita Roychoudhury et. al. [12], the division of entities in Evolved Universal Terrestrial Radio Access Networks (E-UTRAN) into data generating MTCDs grouped as tier 1 entities, e.g., medical equipment, sensors etc., tier 2 elements, which aggregate the multiple streams of data coming from tier 1 elements and tier 3 elements that provide the connectivity to the network by forwarding the traffic to the connected MME. The approach used to elect the GL in HGMAKA, although remains lightweight, it introduces computational delay in the election process as well as cause considerable bottleneck in the transmission relayed to and from the GL. Besides, the complete cycle of requests for new shared session key from the individual MTCDs to HSS must be executed for AKA in the advent of maliciously functioning of even a single MTCD within a group.

The computations involved to ensure the Key Forward Secrecy (KFS) and Key Backward Secrecy (KBS) among the group members and the network thereof in the above discussed schemes, place additional computational overheads to perform the weighty calculations to re-generate the key and get re-authenticated in the 3GPP. To overcome this added computational cost, DBGES [13] and SEGB [14] suggested the dynamic group formation technique using binary trees, in which all the MTCDs including group leader are connected as leaves of the binary tree. Secret keys are calculated by using the secret keys of parent, grandparent and ancestor nodes which in turn increase computational load on the devices if the tree height increases in order to expand the group size. In [15] and [16], majority of the computations are done by intermediate key generation center (KGC). All entities taking part in communication has to depend on KGC which is vulnerable to DOS attacks and hence will bring down the network performance abruptly. In [17], privacy of all communicating devices are preserved with the help of random numbers and sequence numbers, but the scheme relies on pre- shared secret key between MTCDs and the authenticating agents scheme which increases its vulnerability to password guessing or brute force attacks.

III. 3GPP Network Architecture

The physical entities involved in the 3GPP network architecture, as in [18], [19], are sub-grouped into the following 3 domains as mentioned below in Figure 1:

- Evolved Universal Terrestrial Radio Access Network
- Evolved Packet Core
- Non-3GPP domain

A. Evolved Universal Terrestrial Radio Access Network (E-UTRAN) consist of a set of MTC devices with one among them declared as the group leader, equipped with high processing speed and possessing more memory comparatively and likewise, and mobile towers termed as evolved Node Base stations (eNBs).

B. Evolved Packet Core (EPC) comprises of Mobile Management Entity (MME) that accomplish the access authentication procedures and routing the key-related information among the stakeholders, Home Subscriber Server (HSS) for authenticating the MTCDs within group(s) through MME(s), Serving Gateway (S-GW) and Packet Data Network Gateway (P-GW) for transmitting the data traffic upon the completing the mutual authentications with the MTCDs of the group(s).

C. Non-3GPP domain is the external network acting as public channel for interacting with the other geographically located 3GPP networks.

Potential Congestion Control Enforcements

The numerous potential avenues and the channels in the 3GPP network giving rise to the bottlenecks out of congestion can be seen as

1. In the absence of GL, all independent MTCDs directly communicate with the eNB of the vicinity, giving rise to bottleneck in the Radio Access Network (RAN).
   a) The congestion arising within the RAN can be eased with the introduction of GL, representing a group of MTCDs. In this case, each of the eNBs receives a single aggregated request from the GL instead of numerous requests from the set of the autonomous MTCDs.

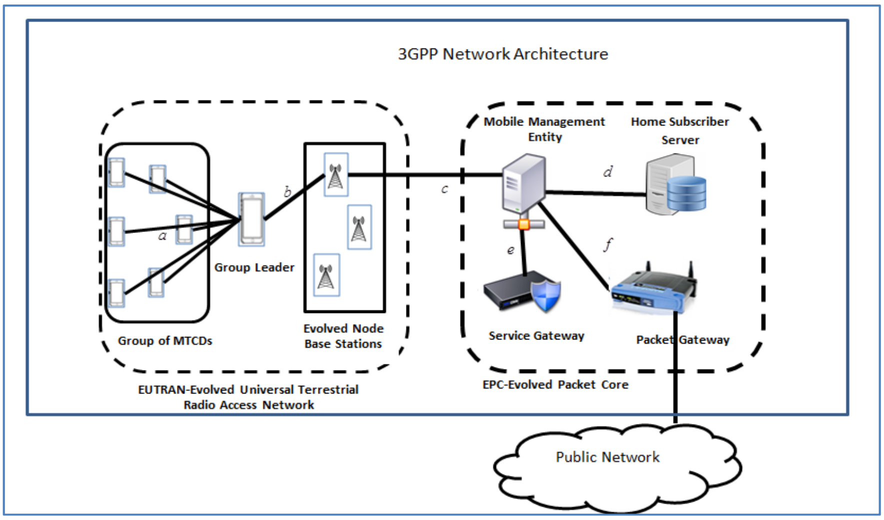

Figure 1: 3GPP Network Architecture with 3 domains

2. The second possibility of congestion arising in the network is the channel $'c'$ as shown between eNB and MME of EPC-Evolved Packet Core, due to multiple forwards by each of the eNB.
3. Of all, the channel transmission which is more often degraded by the congestion related issues is the $'d'$ in between MME and the HSS as MME is accountable for forwarding all the requests emanating from the eNBs and receive the corresponding key related information from HSS as response.
4. The focal points for exploring and employing the congestion control mechanisms remain the channels '***b***' $'c'$ and $'d'$.

IV. PROPOSED PROTOCOL: SL-Grp-AKA ALGORITHM

In the proposed secure and light-weighted group authentication and key agreement protocol, under the heterogeneous context, we assimilate the advantages of the of the security strengths provided by the Elliptic Curve Diffie Hellman Problem (ECDHP)[20], Elliptic Curve Discrete Logarithmic Problem (ECDLP) [21], [22], invertible hash functions and xor operations.

ECDHP refers to the intractability existing in computing the $d_i d_j \Psi$ given the $d_i \Psi$, $d_j \Psi$ $\in E_{\Psi}(a,b)$ over an Elliptic Curve E_{Ψ} . The protocol is presented in the following phases: 1. Registration and Activation Phase, 2. Authentication and key Agreement Phase.

Registration and Activation Phase

Primarily, a finite field F_p is chosen by the HSS over a large range such that $p > 2^{160}$ and an elliptic curve $E_p(a,b)$ is defined by the governing principle:

$$y^2 \bmod p = (x_3 + ax + b) \bmod p$$

where $(a,b) \in F_p$ and $(4a_3 + 27b^2) \bmod p \neq 0$ with order n over F_p .

An arbitrary base point Ψ of order n is chosen over E_p along with a large prime integer d_{HSS} as private key of HSS. All the entities participating in the communication agree on E_{Ψ} coefficients: Ψ_x and Ψ_y . The public key, U_{HSS}, is obtained from $(d_{HSS} \cdot \Psi_y) \bmod \Psi_x$ and published within its network thus facilitating the communicating devices to interact and avail its services in a secure way. The MME, acting as a transmission controller in the 3GPP network, randomly selects the arbitrary large prime integer as private key d_{MME}, obtain the public key: U_{MME} following the $(d_{MME} \cdot \Psi_y) \bmod \Psi_x$ and share in the corresponding network for serving the communicating devices. The GL and individual MTCDs, similarly, choose he private keys: $d_{GL_i} \mid 1 \leq i \leq m$ and $d_{MTCD_{i,J}} \mid 1 \leq i \leq m, 1 \leq j \leq n_i$ where m denotes the number of groups and n_i denotes the number of available MTCDs in the i^{th} group and obtain the corresponding public keys: U_{GL_i} and $U_{MTCD_{i,j}}$ from the $(d_{GL_i} \cdot \Psi_y) \bmod \Psi_x$ and $(d_{MTCD_{i,j}} \cdot \Psi_y) \bmod \Psi_x$, respectively.

Table 1: Initial Secret Keys' Hash Table (ISKHT)

MTCD ID	Secret Key
$ID_{MTCD_{1,1}}$	$H_{KMTCD_{1,1}}$
$ID_{MTCD_{1,2}}$	$H_{KMTCD_{1,2}}$
⋮	⋮
$ID_{MTCD_{1,n1}}$	$H_{KMTCD_{1,n1}}$
⋮	⋮
$ID_{MTCD_{2,1}}$	$H_{KMTCD_{2,1}}$
$ID_{MTCD_{2,2}}$	$H_{KMTCD_{2,2}}$
⋮	⋮
$ID_{MTCD_{2,n2}}$	$H_{KMTCD_{2,n2}}$
⋮	⋮
$ID_{MTCD_{m,1}}$	$H_{KMTCD_{m,1}}$
$ID_{MTCD_{m,2}}$	$H_{KMTCD_{m,2}}$
⋮	⋮
$ID_{MTCD_{m,nm}}$	$H_{KMTCD_{m,nm}}$

Each one of all the MTCDs is mandated to register with the HSS of the host network, for the first time, before availing the services. Thereupon, the individual MTCD is assigned with the identity, say ID, and the secret key by the HSS and allotted to a group using the pre-determined group formation scheme. The related information pertaining to that MTCD, that is the ID of registered MTCD and the hash value of the secret key shared, are maintained as a record securely at the HSS site, as shown in Table 1, along with already registered MTCDs across all groups. ISKHT is stored only in HSS and the corresponding secret key is hardwired into network card of individual MTCD.

Besides, the HSS also maintain a Groups Information Table (GIT) group-wise, as mentioned in Table 2, comprising of the IDs and the corresponding public keys of each group leader and all the MTCDs enrolled, and disseminated among all entities of the host network.

Table 2: Groups Information Table (GIT)

Group	Group Leader ID	Group LeaderPublic Key	MTCD ID	MTCD Public Key
G_1	ID_{GL_1}	U_{GL_1}	$ID_{MTCD_{1,1}}$	$U_{MTCD_{1,1}}$
			$ID_{MTCD_{1,2}}$	$U_{MTCD_{1,2}}$
			⋮	⋮
			$ID_{MTCD_{1,n1}}$	$U_{MTCD_{1,n1}}$
G_2	ID_{GL_2}	U_{GL_2}	$ID_{MTCD_{2,1}}$	$U_{MTCD_{2,1}}$
			$ID_{MTCD_{2,2}}$	$U_{MTCD_{2,2}}$
			⋮	⋮
			$ID_{MTCD_{2,n2}}$	$U_{MTCD_{2,n2}}$
⋮	⋮	⋮	⋮	⋮
G_m	ID_{GL_m}	U_{GL_m}	$ID_{MTCD_{m,1}}$	$U_{MTCD_{m,1}}$
			$ID_{MTCD_{m,2}}$	$U_{MTCD_{m,2}}$
			⋮	⋮
			$ID_{MTCD_{m,nm}}$	$U_{MTCD_{m,mn}}$

Authentication and Key Agreement Phase

In the second phase, each of the MTCDs is required to undergo the authentication and key agreement procedure with HSS to acquire a unique key for each session. In response, a series of messages are communicated between the MTCD and the HSS of the host network. The sequence of requests and the corresponding responses among the entities of the E-UTRAN and EPC-Core involved in AKA phase, with MTCD as initiator, are depicted in the Figure 2.

Notations Used

$aReq_{MTCD_{i,j}}$ → Request for AKA from j^{th} MTCD of i^{th} group

$bReq_{GL_i}$ → Group request generated by i^{th} group Group Leader

$cReq_{GL_i}$ → Group request forwarded by eNB_k

$dReq_{GL_i}$ → Group request forwarded by MME

$dRes_{GL_i}$ → Group response sent by HSS

$cRes_{GL_i}$ → Group response sent by MME

$bRes_{GL_i}$ → Group response sent by eNB_k

$MAC - X_Y$ → MAC generated by Y to be authenticated by X

$EXP - CH - RES_x$ → Expected challenge response generated for entity x

$CH - REQ_x$ → Challenge request generated by entity x

$CH - RES_x$ → Challenge response generated by entity x

TS_x → Time-stamp generated by entity x

R → Random number generated by x

R_x → Random number generated by entity x

KDF → Key Derivation Function

$f1$ → Common Function

SSK → Secret Session Key

The elements involved in the proposed protocol is detailed as hereunder:

Step 1: $\boldsymbol{ID_{MTCD_{i,j}} \rightarrow GL_i : aReq_{MTCD_{i,j}}}$

Every MTCD, desiring to undergo the AKA process, send the authentication request to the respective GL of the group in the form: $aReq_{MTCD_{i,j}}$, comprising of IDs, MACs and the timestamp, as shown below:

$$\boldsymbol{aReq_{MTCD_{i,j}}} = ID_{MTCD_{i,j}} \,||\, ID_{GL_i} \,||\, MAC\text{ - }GL_{MTCD_{i,j}} \,||\, MAC\text{ - }HSS_{MTCD_{i,j}} \,||\, TS_{MTCD_{i,j}}$$

$$\boldsymbol{MAC - GL_{MTCD_{i,j}}} = (ID_{MTCD_{i,j}} \,||\, ID_{GL_i}) \oplus [(d_{MTCD_{i,j}} \cdot U_{GL_i}) mod\, \psi_x]$$

$$\boldsymbol{MAC - HSS_{MTCD_{i,j}}} = (ID_{MTCD_{i,j}} \,||\, ID_{GL_i} \,||\, LAI) \oplus H_{KMTCD_{i,j}}$$

Step 2: $\boldsymbol{GL_i \rightarrow eNB_k : bReq_{GL_i}}$

The group leader receive authentication requests from the MTCDs in the group, calculate $\boldsymbol{MAC - GL_{MTCD_{i,j}}}$ with the key $(d_{GL_i} \cdot U_{MTCD_{i,j}}) mod \psi_x$ for authentication. Upon validation, it retrieves $MAC - HSS_{MTCD_{i,j}}$ from all the MTCDs, calculate the group MAC, $MAC - HSS_{GL_i}$, $MAC - MME_{GL_i}$ and forward $bReq_{GL_i}$ to one of eNBs as $bReq_{GL_i}$

$$\boldsymbol{bReq_{GL_i}} = ID_{MTCD_{i,1}} \,||\, D_{MTCD_{i,2}} \,||\cdots||\, D_{MTCD_{i,n}} \,||\, ID_{GL_i} \,||\, MAC\text{-}MME_{GL_i} \,||\, MAC\text{-}HSS_{GL_i} \,||\, TS_{GL_i}$$

$$\boldsymbol{MAC - MME_{GL_i}} = (ID_{MME} \,||\, ID_{GL_i}) \oplus ((d_{GL_i} \cdot U_{MME}) mod \psi_x)$$

$$\boldsymbol{MAC - HSS_{GL_i}} = MAC\text{-}HSS_{MTCD_{i,1}} \oplus MAC\text{-}HSS_{MTCD_{i,2}} \oplus \ldots \oplus MAC\text{-}HSS_{MTCD_{i,n}}$$

Step 3: $\boldsymbol{eNB_k \rightarrow MME : cReq_{GL_i}}$

The eNB receive and forward $bReq_{GL_i}$ to MME as $cReq_{GL_i}$.

Step 4: $\boldsymbol{MME \rightarrow HSS : dReq_{GL_i}}$

MME calculates its MAC-HSS$_{MME}$ and forwards it along with the group request and LAI as $dReq_{GL_i}$.

$$\boldsymbol{dReq_{GL_i}} = ID_{MTCD_{i,1}} \,||\, ID_{MTCD_{i,2}} \,||\ldots\ldots||\, ID_{MTCD_{i,ni}} \,||\, ID_{GL_i} \,||\, ID_{MME} \,||\, MAC\text{-}HSS_{MME} \,||\, MAC\text{-}HSS_{GL_i} \,||\, TS_{MME} \,||\, LAI$$

$$\boldsymbol{MAC\text{-}HSS_{MME}} = (ID_{MME} \,||\, ID_{HSS}) \oplus [(d_{MME} \cdot U_{HSS}) \bmod \psi_x]$$

$$(MAC\text{-}HSS_{GL_i} \textit{ is forwarded as it is})$$

Step 5: $\boldsymbol{HSS \rightarrow MME : dRes_{GL_i}}$

Authentication and $dRes_{GL_i}$ *Generation*: HSS, upon receiving $dReq_{GL_i}$, authenticate MME, group leader and the group members with the initial shared secret key as mentioned below:

$$\boldsymbol{MAC\text{-}HSS_{MME}} = (ID_{MME} \,||\, ID_{HSS}) \oplus [(d_{HSS} \cdot U_{MME}) \bmod \psi_x]$$

$$(Note :- d_{MME} \cdot U_{HSS} mod \psi_x = d_{HSS} \cdot U_{MME} \bmod \psi_x$$
$$= d_{HSS} \cdot d_{MME} \cdot \psi_y \bmod \psi_x)$$

$$\boldsymbol{MAC - HSS_{MTCD_{i,j}}} = (ID_{MTCD_{i,j}} \,||\, ID_{GL_i} \,||\, LAI) \oplus H_{KMTCD_{i,j}}$$

$$\boldsymbol{MAC\text{-}HSS_{GL_i}} = MAC\text{-}HSS_{MTCD_{i,1}} \oplus MAC\text{-}HSS_{MTCD_{i,2}} \ldots MAC\text{-}HSS_{MTCD_{i,ni}}$$

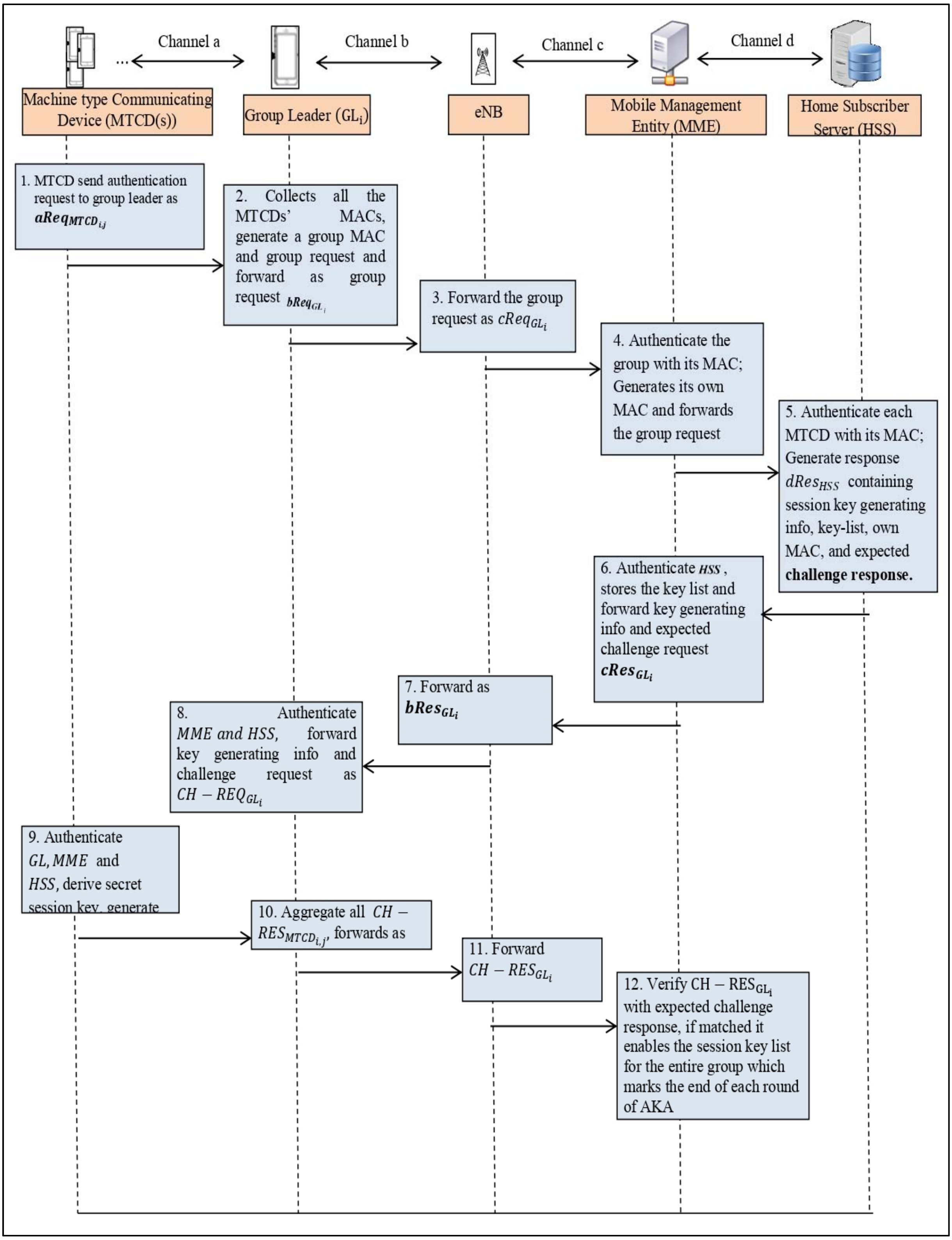
Channel a
Channel b
Channel c
Channel d
Machine type Communicating Device (MTCD(s))
Group Leader (GL_i)
eNB
Mobile Management Entity (MME)
Home Subscriber Server (HSS)
1. MTCD send authentication request to group leader as $aReq_{MTCD_{i,j}}$
2. Collects all the MTCDs' MACs, generate a group MAC and group request and forward as group request $bReq_{GL_i}$
3. Forward the group request as $cReq_{GL_i}$
4. Authenticate the group with its MAC; Generates its own MAC and forwards the group request
5. Authenticate each MTCD with its MAC; Generate response $dRes_{HSS}$ containing session key generating info, key-list, own MAC, and expected challenge response.
6. Authenticate HSS, stores the key list and forward key generating info and expected challenge request $cRes_{GL_i}$
7. Forward as $bRes_{GL_i}$
8. Authenticate MME and HSS, forward key generating info and challenge request as $CH-REQ_{GL_i}$
9. Authenticate GL, MME and HSS, derive secret session key, generate
10. Aggregate all $CH-RES_{MTCD_{i,j}}$, forwards as
11. Forward $CH-RES_{GL_i}$
12. Verify CH – RES_{GL_i} with expected challenge response, if matched it enables the session key list for the entire group which marks the end of each round of AKA

Figure. 2: Flow diagram depicting exchange of messages among the entities of the 3GPP network for AKA

Also, HSS compare the LAI embedded in $MAC-HSS_{MTCD_{i,j}}$ as well as conveyed by the MME. If invalidated, a cascade of failure messages is sent to MME and the group leader. With the authentication step remaining successful, the HSS proceeds the generation of the $dRes_{HSS}$, comprising of a random number, IDs, MACs, expected challenge response and key-list, as shown in table 3, containing individual MTCDs' IDs with corresponding hash values of secret session keys, as detailed below:

$$\boldsymbol{dRes_{GL_i}} = R_{HSS} \,||\, ID_{HSS} \,||\, ID_{GL_i} \,||\, MAC\text{-}MME_{HSS} \,||\, MAC\text{-}GL_{i_{HSS}} \,||\, EXP\text{-}CH\text{-}RES_{GL_i} \,||\, KeyList_{GL_i} \,||\, TS_{HSS}$$

$$\boldsymbol{R_{HSS}} = R \oplus [(d_{HSS}.U_{GL_i}) \bmod \psi_x]$$

$$\boldsymbol{MAC\text{-}MME_{HSS}} = (ID_{MME} \,||\, ID_{HSS}) \oplus [(d_{HSS}.U_{MME}) \bmod \psi_x]$$

$$\boldsymbol{MAC\text{-}GL_{i_{HSS}}} = Enc(((U_{GL_i}.d_{HSS}) \bmod \psi_x),(ID_{HSS} \,||\, ID_{GL_i}))$$

$$\boldsymbol{EXP\text{-}CH\text{-}RES_{GL_i}} = EXP\text{-}CH\text{-}RES_{MTCD_{i,1}} \oplus EXP\text{-}CH\text{-}RES_{MTCD_{i,2}} \oplus \ldots \oplus EXP\text{-}CH\text{-}RES_{MTCD_{i,ni}}$$

$$\boldsymbol{EXP\text{-}CH\text{-}RES_{MTCD_{i,j}}} = Enc(H_{KMTCD_{i,j}},(f_1(R)))$$

$$\boldsymbol{SSK_{MTCD_{i,j}}} = KDF(R, H_{KMTCD_{i,j}})$$

Table 3: $KeyList_{GL_i}$

MTCD ID	Secret Session Key
$ID_{MTCD_{i,1}}$	$H_{SSK-MTCD_{i,1}}$
$ID_{MTCD_{i,2}}$	$H_{SSK-MTCD_{i,2}}$
⋮	⋮
$ID_{MTCD_{i,n}}$	$H_{SSK-MTCD_{i,n}}$

Step 6:- $MME \rightarrow eNB_k : cRes_{GL_i}$

$\boldsymbol{MME}$ stores the $KeyList_{GL_i}$ and $\boldsymbol{EXP-CH-RES_{GL_i}}$, verifies the $\boldsymbol{MAC-MME_{HSS}}$, and sends $\boldsymbol{cRes_{GL_i}}$ consisting of R_{HSS}, IDs, MACs for the entities next in network and time stamp, as shown below:

$$\boldsymbol{cRes_{GL_i}} = R_{HSS} \,||\, ID_{MME} \,||\, ID_{GL_i} \,||\, ID_{HSS} \,||\, MAC\text{-}GL_{i_{MME}} \,||\, MAC\text{-}GL_{i_{HSS}} \,||\, TS_{MME}$$

$$\boldsymbol{MAC\text{-}GL_{i_{MME}}} = Enc(((d_{MME}.U_{GL_i}) \bmod \psi_x),(ID_{GL_i} \,||\, ID_{MME})$$

Step 7:- $eNB_k \rightarrow GL_i : bRes_{GL_i}$

eNB_k receives and forward the $\boldsymbol{cRes_{GL_i}}$ to GL_i as $bRes_{GL_i}$.

Step 8:- $GL_i \rightarrow MTCD_{i,j}$: $CH\text{-}REQ_{GL_i}$

Group leader authenticates HSS and MME as

$$\boldsymbol{MAC\text{-}GL_{i_{MME}}} = Enc(((d_{GL_i} \cdot U_{MME}) \bmod \psi_x),(ID_{GL_i} \,||\, ID_{MME}))$$

$$\boldsymbol{MAC\text{-}GL_{i_{HSS}}} = Enc(((U_{HSS} \cdot d_{GL_i}) \bmod \psi_x),(ID_{HSS} \,||\, ID_{GL_i}))$$

Following, $\boldsymbol{GL_i}$ retrieves R with its private key as mentioned:

$$\boldsymbol{R} = R_{HSS} \oplus \left(\left(U_{HSS}.d_{GL_i}\right) \bmod \psi_x\right)$$

Subsequently, challenge request $CH-REQ_{GL_i}$ is generated, with the below mentioned elements, for individual MTCD and forwarded in the group.

$$\boldsymbol{CH\_REQ_{GL_i}} = R_{GL_i} \,||\, ID_{GL_i} \,||\, ID_{MTCD_{i,j}}$$

$$R_{GL_i} = R \oplus \left(\left(d_{GL_i} \cdot U_{MTCD_{i,j}}\right) \bmod \psi_x\right) // \; re\text{-}encrypting\; R$$

Step 9: $MTCD_{i,j} \rightarrow GL_i : CH-RES_{MTCD_{i,j}}$

$MTCD_{i,j}$ computes its secret session key, $SSK_{MTCD_{i,j}}$ and generates $CH - RES_{MTCD_{i,j}}$. It forwards the challenge response to group leader and stores the session key for secure information transmission.

$$\boldsymbol{SSK_{MTCD_{i,j}}} = KDF\left(R, H_{KMTCD_{i,j}}\right)$$

$$\boldsymbol{CH\text{-}RES_{MTCD_{i,j}}} = Enc(H_{KMTCD_{i,j}},(f_1(R)))$$

$$\boldsymbol{R} = R_{HSS} \oplus \left(\left(d_{MTCD_{i,j}}.U_{GL_i}\right) \bmod \psi_x\right)$$

Step 10:- $\boldsymbol{GL_i \rightarrow eNB_k : CH - RES_{GL_i}}$

Group leader perform the xor operation for all responses, generates group challenge response as $CH - RES_{GL_i}$ and send to eNB_k using public key encryption.

$$\boldsymbol{CH\_RES_{MTCD_{i,j}}} = CH\text{-}RES_{MTCD_{i,j}} \oplus \left(\left(d_{GL_i} \cdot U_{MTCD_{i,j}}\right) mod\, \psi_x\right)$$

$$\boldsymbol{CH\_RES_{GL_i}} = Enc((d_{GL_i} \cdot U_{MME}),(CH\text{-}RES_{MTCD_{i,1}} \oplus CH\text{-}RES_{MTCD_{i,2}} \oplus \ldots \oplus CH\text{-}RES_{MTCD_{i,ni}}))$$

Step 11:- $\boldsymbol{eNB_k \rightarrow MME : CH - RES_{GL_i}}$

eNB_k forward the group response securely to MME.

Step 12:- **MME enables key-list for the group GL_i, sent by the HSS**

Upon receiving $CH - RES_{GL_i}$, MME compares with $EXP - CH - RES_{GL_i}$, and enables the keylist if found same as communicated by HSS.

After full round of AKA, each MTCD shares a secret session key with MME as well as HSS, which can be used to secure data for subsequent data transmissions.

V. FORMAL SECURITY ANALYSIS OF PROPOSED SL-Grp-AKA PROTOCOL USING AVISPA

Automated Validation of Internet Security Protocols and Applications, namely AVISPA [23][24], remain the widely used tool as a suite of applications for validating formally the security protocol models designed to be used over the internet, for its strength. Few of the earlier designed protocols, namely [25][26][27], have been simulated using AVISPA. For specifying the protocol to be validated, AVISPA supports High Level Protocol Specification Language (HLPSL). The HLPSL is a role-oriented language where every entity is simulated as a role during protocol execution and communicates over the insecure public channel with the other roles of the network. The HLPSL specification is interpreted into intermediate form (IF), a lower level language generated by hlpsl2if translator and directly read by the AVISPA back-ends [28] [29] [30]: OFMC, CLAtSe, SATMC, and TA4SP. The intruder is framed as one of the roles holding sufficient knowledge of the traffic transmitting across the channel of the communicating parties..

```
goal
secrecy_of sec1
secrecy_of sec2
secrecy_of sec3
secrecy_of sec4
secrecy_of sec5

authentication_on gl_mme
authentication_on gl_hss
authentication_on mme_hss
authentication_on hss_gl
authentication_on hss_mme
authentication_on mme_gl
end goal
```

Figure 3: Goals set for proposed SL-Grp-AKA

The proposed protocol of SL-Grp-AKA specification is validated using the OFMC and CLAtSe back-ends; goals mentioned in Figure 3, and the corresponding algorithms for the roles involved in the network, as mentioned in the Figure 4 for the group leader (GL), Figure 5 for MME, Figure 6 for HSS, whereas figure 6 indicate algorithm for the session and environment using the HLPSL. Finally, OFMC and CL-AtSe back end simulation results of the SL-Grp-AKA protocol are shown in the Figure 7 and Figure 8, respectively.

```
role gl(GL,M,HSS:agent,
         PSIX,PSIY,LAI :text,
         KDF,MUL,H : hash_func,
         K : symmetric_key,
         Snd,Rcv:channel(dy))
played_by GL
def=
  local State : nat,

EXP_CH_RES,ID_MME,ID_HSS,ID_GL,Dgl,Ugl,Umme,Uhss,MAC_GL_MME,MAC_GL_HSS,MAC_
MME_GL,MAC_MME_HSS,MAC_HSS_GL,MAC_HSS_MMER,Rhss,R,SSK : text,
       F1,Inc : hash_func
      const sec1,sec2,sec3,sec4,sec5,gl_mme,gl_hss,mme_gl,mme_hss,hss_gl,hss_mme : protocol_id

  init State := 0

  transition
1. State = 0 /\ Rcv(start) =|> State' := 1
/\ ID_GL' := new()
/\ Dgl' := new()
/\ Ugl' :=mod(MUL(Dgl',PSIX),PSIY)
/\ Snd(ID_GL'.Ugl')
/\ secret({Dgl},sec1,{GL})

2. State = 1 /\ Rcv(ID_MME'.Umme') /\ Rcv(ID_HSS'.Uhss') =|> State' := 2
/\ MAC_GL_MME' := {(ID_GL.ID_MME')}_MUL(Umme',Dgl)
/\ MAC_GL_HSS' := {(ID_GL.ID_HSS'.LAI)}_K
/\ Snd(ID_GL.ID_MME'.ID_HSS'.MAC_GL_MME'.MAC_GL_HSS')
/\ secret({K},sec4,{GL,HSS})
/\ request(GL,M,gl_mme,MAC_GL_MME')
/\ request(GL,HSS,gl_hss,MAC_GL_HSS')

3. State = 2 /\ Rcv(ID_GL'.ID_MME'.ID_HSS'.MAC_MME_GL'.MAC_HSS_GL'.Rhss')
/\ MAC_MME_GL = {(ID_GL.ID_MME)}_MUL(Umme,Dgl)
/\ MAC_HSS_GL = {(ID_GL.ID_HSS)}_MUL(Uhss,Dgl) =|> State' := 3
/\ R' := xor(Rhss',MUL(Uhss,Dgl))
/\ SSK' := KDF(R',H(K))
/\ EXP_CH_RES' :={F1(R)}_H(K)
/\ Snd(EXP_CH_RES')
/\ witness(GL,M,mme_gl,MAC_MME_GL')
/\ witness(HSS,GL,hss_gl,MAC_HSS_GL')
end role
```

Figure 4: HLPSL code for Group Leader of SL-Grp-AKA

```
role mme(GL,M,HSS:agent,
         PSIX,PSIY,LAI :text,
         KDF,MUL,H : hash_func,
         K : symmetric_key,
         Snd,Rcv:channel(dy))
played_by M
def=
  local State : nat,

EXP_CH_RES,ID_MME,ID_HSS,ID_GL,Dmme,Ugl,Umme,Uhss,MAC_GL_MME,MAC_GL_HSS,MAC_
MME_GL,MAC_MME_HSS,MAC_HSS_GL,MAC_HSS_MME,R,Rhss : text,
       F1,Inc : hash_func
  const sec1,sec2,sec3,sec4,sec5,gl_mme,gl_hss,mme_gl,mme_hss,hss_gl,hss_mme : protocol_id

      init State := 0

  transition
1. State = 0 /\ Rcv(start) =|> State' := 1
/\ ID_MME' := new()
/\ Dmme' := new()
/\ Umme' :=mod(MUL(Dmme,PSIX),PSIY)
/\ Snd(ID_MME'.Umme')
/\ secret({Dmme},sec2,{M})

2.State = 1 /\Rcv(ID_GL'.Ugl')/\Rcv(ID_HSS'.Uhss') /\
Rcv(ID_GL'.ID_MME'.ID_HSS'.MAC_GL_MME'.MAC_GL_HSS')
/\ MAC_GL_MME ={(ID_GL'.ID_MME)}_MUL(Ugl',Dmme) =|> State' := 2
/\ MAC_MME_HSS' := {(ID_MME'.ID_HSS')}_MUL(Uhss',Dmme)
/\ Snd(ID_GL'.ID_MME'.ID_HSS'.MAC_MME_HSS'.MAC_GL_HSS')
/\ witness(GL,M,gl_mme,MAC_GL_MME')
/\ request(M,HSS,mme_hss,MAC_MME_HSS')

3. State = 2 /\ Rcv(ID_GL'.ID_MME'.ID_HSS'.MAC_HSS_MME'.MAC_HSS_GL'.Rhss'.EXP_CH_RES')
/\ MAC_HSS_MME = {(ID_MME.ID_HSS)}_MUL(Uhss,Dmme) =|> State' := 3
/\ MAC_MME_GL' := {(ID_GL.ID_MME)}_MUL(Ugl,Dmme)
/\ Snd(ID_GL'.ID_MME'.ID_HSS'.MAC_MME_GL'.MAC_HSS_GL'.Rhss')
/\ request(GL,M,mme_gl,MAC_MME_GL')
/\ witness(HSS,M,hss_mme,MAC_HSS_MME')

4. State = 3 /\ Rcv(EXP_CH_RES' ) =|> State' :=4
end role
```

Figure 5: HLPSL code for MME of SL-Grp-AKA

```
role hss(GL,M,HSS:agent,
         PSIX,PSIY,LAI :text,
         KDF,MUL,H : hash_func,
         K : symmetric_key,
         Snd,Rcv:channel(dy))
played_by HSS
def=
  local State : nat,

EXP_CH_RES,SSK,ID_MME,ID_HSS,ID_GL,Dhss,Ugl,Umme,Uhss,MAC_GL_MME,MAC_GL_HSS
,MAC_MME_GL,MAC_MME_HSS,MAC_HSS_GL,MAC_HSS_MME,R,Rhss : text,
       F1,Inc : hash_func
 const sec1,sec2,sec3,sec4,sec5,gl_mme,gl_hss,mme_gl,mme_hss,hss_gl,hss_mme : protocol_id

  init State := 0

   transition
1. State = 0 /\ Rcv(start) =|> State' := 1
/\ ID_HSS' := new()
/\ Dhss' := new()
/\ Uhss' :=mod(MUL(Dhss,PSIX),PSIY)
/\ Snd(ID_HSS'.Uhss')
/\ secret({Dhss'},sec3,{HSS})

2.State = 1 /\ Rcv(ID_GL'.Ugl')/\Rcv(ID_MME'.Umme') /\
Rcv(ID_GL'.ID_MME'.ID_HSS.MAC_MME_HSS'.MAC_GL_HSS')
/\ MAC_MME_HSS = {(ID_MME.ID_HSS)}_MUL(Umme,Dhss)
/\ MAC_GL_HSS = {(ID_GL.ID_HSS.LAI)}_K =|> State' := 2
/\ R' := new()
/\ Rhss' := xor(R',MUL(Ugl',Dhss))
/\ SSK' := KDF(R',H(K))
/\ MAC_HSS_MME' := {(ID_MME'.ID_HSS)}_MUL(Umme',Dhss)
/\ MAC_HSS_GL' := {(ID_MME'.ID_HSS)}_MUL(Ugl',Dhss)
/\ EXP_CH_RES' :={F1(R)}_H(K)
/\ Snd(ID_GL'.ID_MME'.ID_HSS.MAC_HSS_MME'.MAC_HSS_GL'.Rhss'.EXP_CH_RES')
/\ secret(SSK',sec5,{GL,HSS})
/\ witness(GL,HSS,gl_hss,MAC_GL_HSS')
/\ witness(M,HSS,mme_hss,MAC_MME_HSS')
/\ request(HSS,M,hss_mme,MAC_HSS_MME')
/\ request(HSS,GL,hss_gl,MAC_HSS_GL')

end role
```

Figure 6: HLPSL code for HSS of SL-Grp-AKA

```
role session(GL,M,HSS:agent,
         PSIX,PSIY,LAI :text,
         KDF,MUL,H : hash_func,
         K : symmetric_key)
def=
    local
       Sndgl,Rcvgl,Sndm,Rcvm,Sndh,Rcvh:channel(dy)
composition

gl(GL,M,HSS,PSIX,PSIY,LAI,KDF,MUL,H,K,Sndgl,Rcvgl)
/\ mme(GL,M,HSS,PSIX,PSIY,LAI,KDF,MUL,H,K,Sndm,Rcvm)
/\ hss(GL,M,HSS,PSIX,PSIY,LAI,KDF,MUL,H,K,Sndh,Rcvh)
 end role
role environment()
def=
const gl,m,h: agent,
exp_ch_res,psix,psiy,lai,id_mme,id_hss,id_gl,dhss,dmme,dgl,ugl,umme,uhss,mac_gl_mme,mac_gl_hss,
mac_mme_gl,mac_mme_hss,mac_hss_gl,mac_hss_mme,r,rhss : text,
      f1,kdf,mul,h1 : hash_func,
      k : symmetric_key,
      sec1,sec2,sec3,sec4,sec5,gl_mme,gl_hss,mme_gl,mme_hss,hss_gl,hss_mme : protocol_id
intruder_knowledge =
{gl,m,h,mac_gl_mme,mac_gl_hss,mac_mme_gl,mac_mme_hss,mac_hss_gl,mac_hss_mme,exp_ch_res}

composition
session(gl,m,h,psix,psiy,lai,kdf,mul,h1,k)
/\session(m,h,gl,psix,psiy,lai,kdf,mul,h1,k)
/\session(h,m,gl,psix,psiy,lai,kdf,mul,h1,k)
/\session(m,gl,h,psix,psiy,lai,kdf,mul,h1,k)
end role
```

Figure 7: HLPSL code for session and environment of SL-Grp-AKA protocol using AVISPA

VI. INFORMAL SECURITY ANALYSIS OF THE PROTOCOL

In the literature available, the protocol under study is required to resist the following attacks and remain robust to be considered as secure.

a) Stolen Secret Key Attack
b) Redirection Attack
c) Man in the Middle Attack
d) Session Key Computation Attack
e) Replay Attack
f) Denial of Service (DoS) Attack
g) Impersonation Attacks
   i) MTCD Impersonation Attack
   ii) Group Leader Impersonation Attack
   iii) MME Impersonation Attack.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/GRP_AKA999.if
GOAL
as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
parseTime: 0.00s
searchTime: 15.62s
visitedNodes: 4096 nodes
depth: 12 plies
```

Figure 8: OFMC back end simulation result of the SL-Grp-AKA protocol

```
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/GRP_AKA999.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

Analysed   : 0 states
Reachable  : 0 states
  Translation: 0.06 seconds
  Computation: 0.00 seconds
```

Figure 9: CL-AtSe back end simulation result of the SL-Grp-AKA protocol

In this section, an informal security analyses is presented to demonstrate the robustness of the proposed SL-Grp-AKA protocol against the protocol attacks mentioned above.

Stolen Secret Key Attack

In the stolen secret key attack, in agreement to the literature[31], the adversary, say A, steals the secret key and misuses it to hijack the information being transmitted among the entities of the 3GPP network.

Introducing ECDH strengths in the proposed protocol: The resilience offered by the proposed protocol is presented in the contrapositive manner. Suppose, the adversary steals the secret key, shared among the independent MTCD and the HSS to be used by the respective MTCD while sending the request to get the information required in computing session key from HSS and also by HSS while providing this information. For doing so, the request from MTCD reaches HSS via the group leader and MME in between. In the proposed scheme, every entity chooses their own private key and computes the public key to be distributed to other entities of the network for mutual common key, as per the ECDH principle, required for information sharing.

The HSS would transmit the *random number* R_{HSS} in response to the MTCD for the session key generation. The R_{HSS} reaches every entity along the network path to the MTCD in an encrypted form, along with other response parameters, which is decrypted at the entity site, using the public key cryptosystem, and re-encrypted for forwarding the same to the next entity. The probability of retrieving the R_{HSS} by the adversary A is trivial as the private key of every entity including MTCD is not known and cannot compute eventually the session key. Using the brute force approach, the number of combinations in the key space for determining the private keys is $2^{128} \times 2^{128} \times 2^{128} \times 2^{128} = 2^{512} = 2^{10\times 51.2} \approx 10^{3\times 51.2}$. If the time required for executing one combination of the key space is 10^{-7} seconds, then the total time required for the entire key space is $10^{138.6}$ years, a formidably large time.

Instead, if the adversary proceeds by the intuitive knowledge or the brute force method to identify the R_{HSS}, comprising of 128 bits, and compute secret session key, generation of challenge response for enabling the key-list in MME, essential for further information transmission, still remain the elusive task in the absence of the private keys.

Thus, in the advent of the stolen secret key by the adversary, the proposed protocol offers resilience to this type of attack.

Redirection Attack

According to the [32], the information, shared between a pair of entities in the network, may undergo redirection to the adversary A instead of the intended receiver.

Role of LAI and the cryptographic procedures used in the proposed protocol: In case of 3GPP, the redirection attack is mainly initiated by the adversary A impersonating as eNB to obtain the legitimate user information and forward them to the destination. In the proposed protocol, the user is the group leader of E-UTRAN and the destination is the HSS of the EPC. The redirection attack cannot take place in case the eNB fails to obtain the information of either the GL or the MME. The usage of public key cryptosystem, in transmitting the requests and responses among the entities, makes it impossible for any base station to obtain this information. Alternatively, a fake eNB may intercept and try to send the forged intermediate messages from the GL to MME or vice-versa. To counter this possibility, LAI is embedded by the MTCD in $MAC - HSS_{MTCD_{i,j}}$, shown in Step1 of the Section IV, and also by MME in the requests, shown in Step 4 of Section IV, which in turn is verified by HSS at its end. With the above procedure in place in the proposed protocol, the verification is bound to fail resulting in authentication failure and offering stiff resistance to redirection attack.

Man In The Middle Attack

In the advent of the Man in The Middle (MITM) attack, according to [33], the adversary A is able to sniff the requests and the corresponding responses from the senders in the network and manipulate them before sending them to the intended receivers.

Counter measures used in the proposed protocol: As described in the Step 2 of the Section IV, the request: $bReq_{GL_i}$ emanating from the group leader and the response: $cRes_{GL_i}$ generated by the MME, as detailed in the Step 6 of Section IV, the R_{HSS} may be compromised. However, computation of the secret session key is possible with the knowledge of the pre-shared secret key of the MTCD.

Other possibility to know the secret session key, required for further communication, is using the brute-force approach to deduce the 128 bit session key. With the above mentioned time constraints, the duration to correctly determine the same is $3.12 \times 10^{23.4}$ years, a very long time. In either of the possibilities, the proposed protocol offers stiff resistance to the MITM attack.

Session Key Computation Attack

The adversary determines the session key of the MTCD in order to gain unauthorized access to the network services offered by the network.

Role of inherent properties of the strong cryptic procedures in the proposed protocol: The public channel used for the information transmission may give rise to the scope for the session key attack, if any, which is only in between the group leader and the MME entities. For communicating the AKA requests: $bReq_{GL_i}$ and $cReq_{GL_i}$, as mentioned in the Step 2 and 3 of Section IV and the corresponding responses: $cRes_{GL_i}, CH - REQ_{GL_i}, and\ CH - Res_{GL_i}$ as described in the Steps 6, 8 and 11 of Section IV, between the E-UTRAN and EPC securely over the internet, derive the common secret key to be used as a key for encrypting the messages using one of the robust private key cryptosystems available.

To unravel the bits of the common secret key, the adversary A can implement known cipher text/plaintext attack or chosen plaintext/cipher text attacks. Usage of the known cipher text attack would require the sufficiently large computing time thereby losing the credibility of the key determined and render useless for A. The inherent strength offered by the used encryption technique resists the session key computation attack as the former uses the multiple rounds of the iteration process. The other two types, namely chosen plaintext/ciphertext attacks uses one's own intuition to figure out the sequence of bits of the secret key and by no means it can level the knowledge of the key.

Replay Attack

According to the [34][35] and [36], the replay attack is said to be delaying or resending the information to a victim thereby causing the network congestion and misdirecting the authorized entity to perform the tasks what the adversary wants.

Significant role of time-stamps in the proposed protocol: Time stamps [37] play a major role in combating replay attacks. In the proposed protocol, time stamps are generated by every active entity involved in the communicating network and embedded in the messages transmitted thereby restricting the replay attack.

DoS Attack

The Denial of Service (DoS) attack [38] results in the server or the network failure in extending the services to the legitimate clients. The most common DoS attack against a computer network include the bandwidth and the connectivity attacks, where the former refers to depleting the bandwidth capacity and the latter refers to cause exhaustion of the resources thereby denying the service requests of the legitimate users.

Using LAI and time-stamps in the proposed protocol: The HSS and MME are the servers in the proposed protocol; and prone to DoS attacks by adversary A, if any, by forging the messages: $aReq_{MTCD_{i,j}}$, $bReq_{GL_i}$ and $cReq_{GL_i}$. However, with the introduction of LAI and timestamps, the HSS and MME entities can easily detect the abnormality by checking the MACs, thus withstanding the DoS attacks effectively.

Impersonation Attack

An impersonation attack [39] masquerade the identity of one of the legitimate entities, participating in the communication protocol of a system. The goal of a strong identification and/or entity authentication protocol is to bring down the probability of stealing someone else identity and posing as the genuine entity to the other communicating entity.

Role of private keys and MACs in the proposed protocol: In order to deal with the possibilities when an adversary A can impersonate either of the individual MTCD, GL or the MME by forging their identities in the proposed SL-Grp-AKA, the generation and verification of entities' MACs are carried out with the help of the private keys of the communicating entities. As the private keys are never transmitted over network in the proposed protocol, acquiring these will cost humongous time and effort to A which drastically bring down its chances of posing this attack.

VII. PERFORMANCE EVALUATION

In this section, the evaluating parameters are chosen as the computational cost and the bandwidth consumption, as these remain vital in judging the efficiency of the protocol designed to cater the transmission amongst the devices of the heterogeneous environment. The comparative analyses are done amid the protocol services offered by the [4],[7],[11],[17],[19] and the proposed SL-Grp-AKA protocol, with due considerations to the architectural design followed in each of them with the components: HSS, MME, eNBs, MTCDs. The assumptions considered in the network architecture are the presence of the private and safe connectivity between the entities of each pair: HSS and the MME, group of MTCDs and GL and the public transmission media between the entities of each pair: GL and eNB, eNB and MME.

Installation Set-up and Computational cost

The SL-Grp-AKA protocol is simulated on the system with Intel(R) Core™ i3-5005U, CPU @ 2.00GHz, 12.0GB RAM, 64-bit operating system. A comparative study is made for the computational cost borne when the schemes are simulated with n devices in each of the m number of groups. The computational cost is evaluated as the throughput, in terms of milliseconds (ms), for executing each operation of the protocol. The Table 4 shows the values emerged as the computing cost, with some natural and trivial differentiation of the devices, omitting the xor operation in view of its negligible computing cost involved, universally. In the Table 5, the parametric cumulative costs of the protocols considered for comparative study are projected.

Table 4: Time elapsed in each operation

Operation	Notation	Time in (ms)
Multiplication	MUL	0.011
Modular	MOD	0.04
One way Hash Function	$HASH$	0.09
Encryption Algorithm	ENC	0.12

Table 5: Parametric Comparative and Cumulative Computing Costs Analyses of Protocols

Protocol	**Operations**	**Computations Cost (in ms)**
SE-AKA [4]	$(6n + 3m) \times HASH + 4n \times MUL$	$(0.984n + 0.27) \times m$
GLARM [7]	$(8n) \times HASH + 3 \times HASH$	$(0.72n + 0.27) \times m$
LGTH [11]	$(3 + 2n) \times HASH \times m$	$(0.18n + 0.27) \times m$
PP-AKA [17]	(8n+6m)× $HASH$ +(3n+m)× MUL	$(0.99n + 0.551) \times m$
GSL-AKA [19]	$(4n) \times HASH + (8) \times HASH$	$(0.36n + 0.72) \times m$
SL-Grp-AKA (Proposed Protocol)	$n \times MUL$+$n \times MOD$+$3 \times ENC +$ $5 \times MUL + 5 \times MOD$	$(0.051n + 0.615) \times m$

Although, the computational cost of the proposed one as compared to [4], [7], [17] and [19] is low, it is found to be bit higher as compared to the [11] owing to the inclusion of the encryption techniques in the SL-Grp-AKA to resist the attacks. With the inherent modularity introduced in the proposed protocol, a greater flexibility is accounted for replacing the encryption technique with the optimal available one. Also, the corresponding plots are shown in Figure 10(a), 10(b) and 10(c), for $m = 1, m = 10$ and $m = 100,$ respectively.

Bandwidth Consumption

The communication cost is measurable in terms of number of the binary bits used in the messages exchanged between the communicating entities. In the Table 6, the number of bits used for each parameter, carefully chosen, is shown.

Using the same installation setup and the environment mentioned above with n devices and m groups, the total number of bits accumulated in the messages shared among the 3GPP network entities for one cycle of AKA are compared between the [4], [[7], [11], [17], [19] and the proposed SL-Grp-AKA.

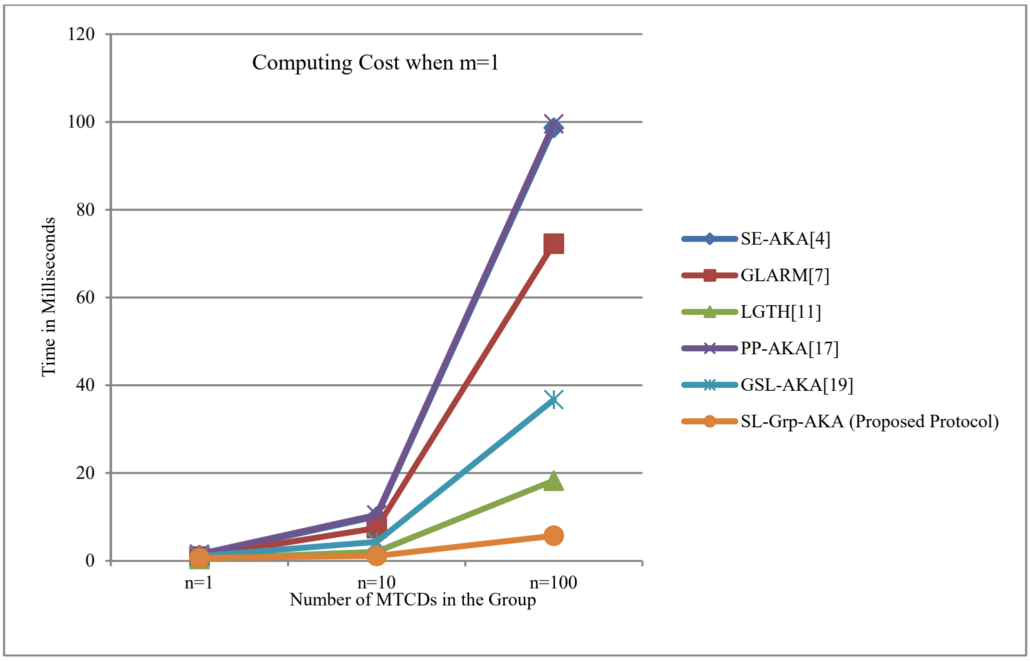

Figure 10(a): Computational cost comparison, for m=1

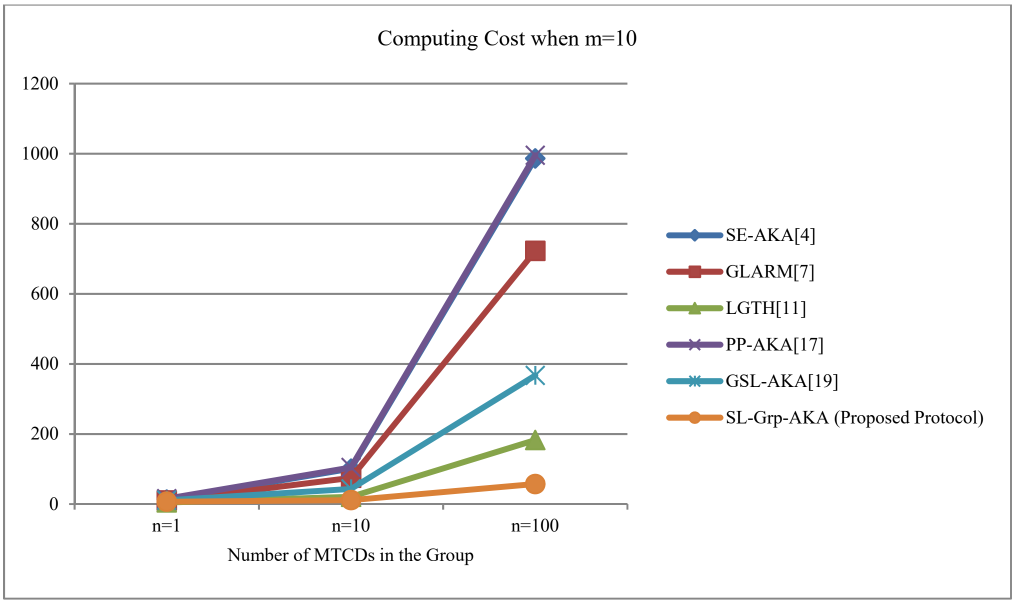

Figure 10(b): Computational cost comparison, for m=10

Table 6: Number of binary bits required for protocols' parameter

Parameter	Number of Bits
ID	128
TS	128
MAC	128
Hash Val	64
LAI	40
R	128
Secret Key Size	128
Session Key Size	128
Integrity and Cipher Keys	128

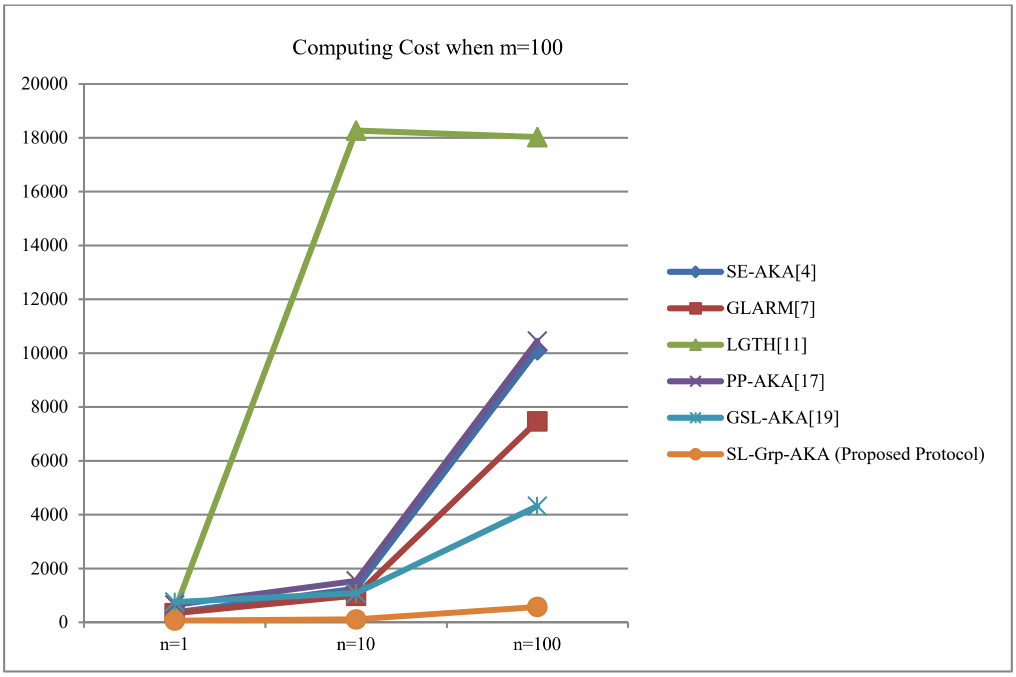

Figure 10(c): Computational cost comparison, for m=100

In the Table 7, the parametric bandwidth consumption in terms of n and m are projected, with proposed taking lesser as compared to rest except [11] for a larger value of n and more number of parameters introduced in the proposed to tackle the attacks.

Table 7: Parametric Comparative and Cumulative Bandwidth Consumption Analyses of Protocols

Protocol	**Number of messages in one of AKA**	**Parametric Bandwidth consumption (in bits)**
SE-AKA[4]	$\sum_{i=1}^{8} M_i$	$(1328n + 2184) \times m$
GLARM[7]	$\sum_{i=1}^{6} M_i$	$(768n + 1440) \times m$
LGTH[11]	$\sum_{i=1}^{6} M_i$	$(65n + 2252) \times m$
PP-AKA[17]	$\sum_{i=1}^{7} M_i$	$(384n + 1536)\times m$
GSL-AKA[19]	$\sum_{i=1}^{12} M_i$	$(896n + 1736) \times m$
SL-Grp-AKA (Proposed)	$\sum_{i=1}^{12} M_i$	$(384n + 1320) \times m$

In figures: 11(a), 11(b) and 11(c), the plots for the comparison of the bandwidth consumptions of protocols are also depicted for the values: $m = 1, m = 10\ and\ m = 100$, respectively. In the Table 8 as shown below, the status of the security objectives observed are enlisted for [4], [7], [11], [17], [19] and our proposed SL-Grp-AKA protocol.

VIII. CONCLUSION AND FUTURE SCOPE

In this paper, we have presented a novel, secure and light-weighted group authentication and key agreement protocol using ECDH mechanism where all the entities involved can mutually authenticate each other with minimal computations. The computational cost incurred and the bandwidth requirements are also found to be low as compared to the other earlier protocols in practice. The graphs are presented indicating the comparative performance of our protocol with respect to varying degree of n and m. The proposed SL-Grp-AKA is also offering greater resistance to the numerous popular attacks.

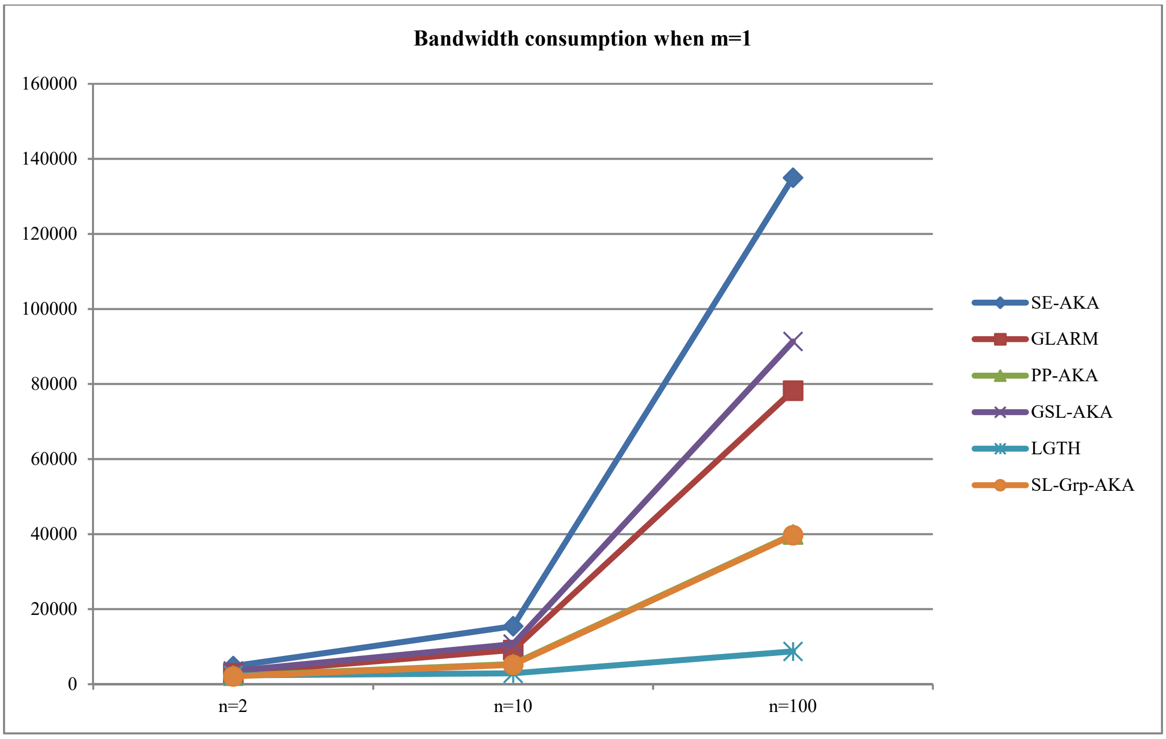

Figure 11(a): Bandwidth consumption comparison, for m=1

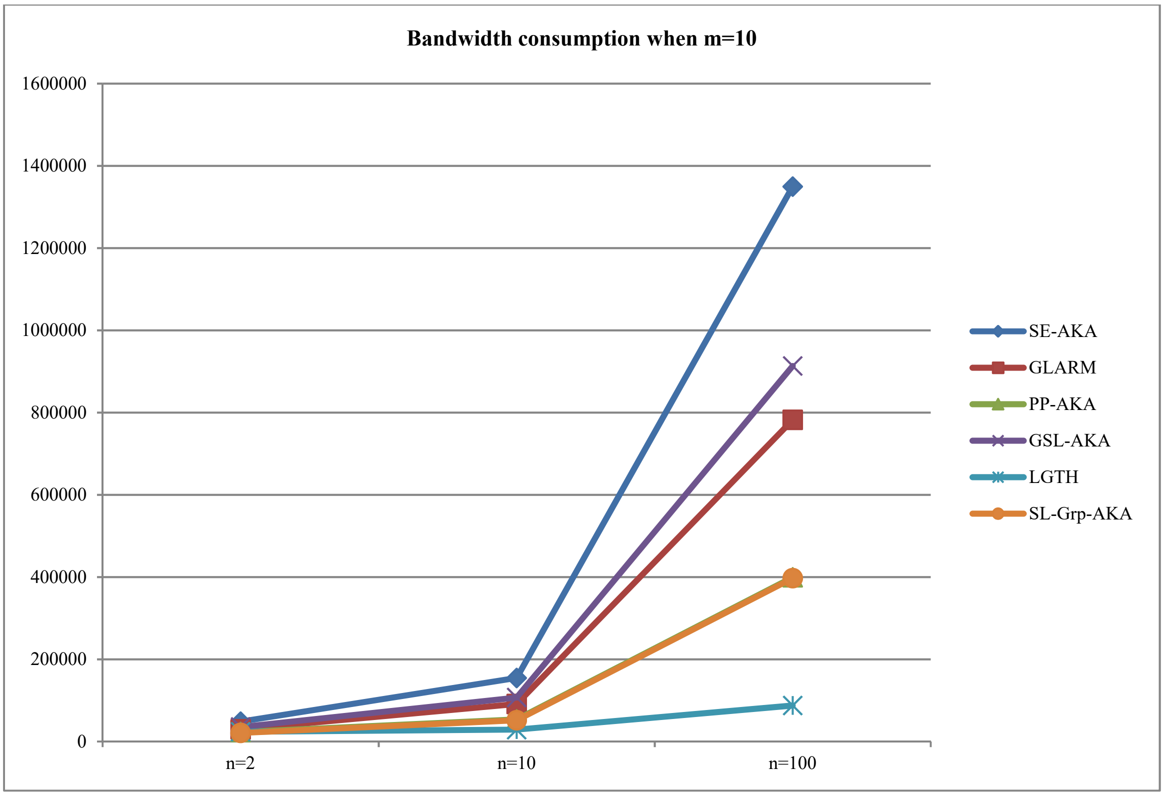

Figure 11(b): Bandwidth consumption comparison, for m=10

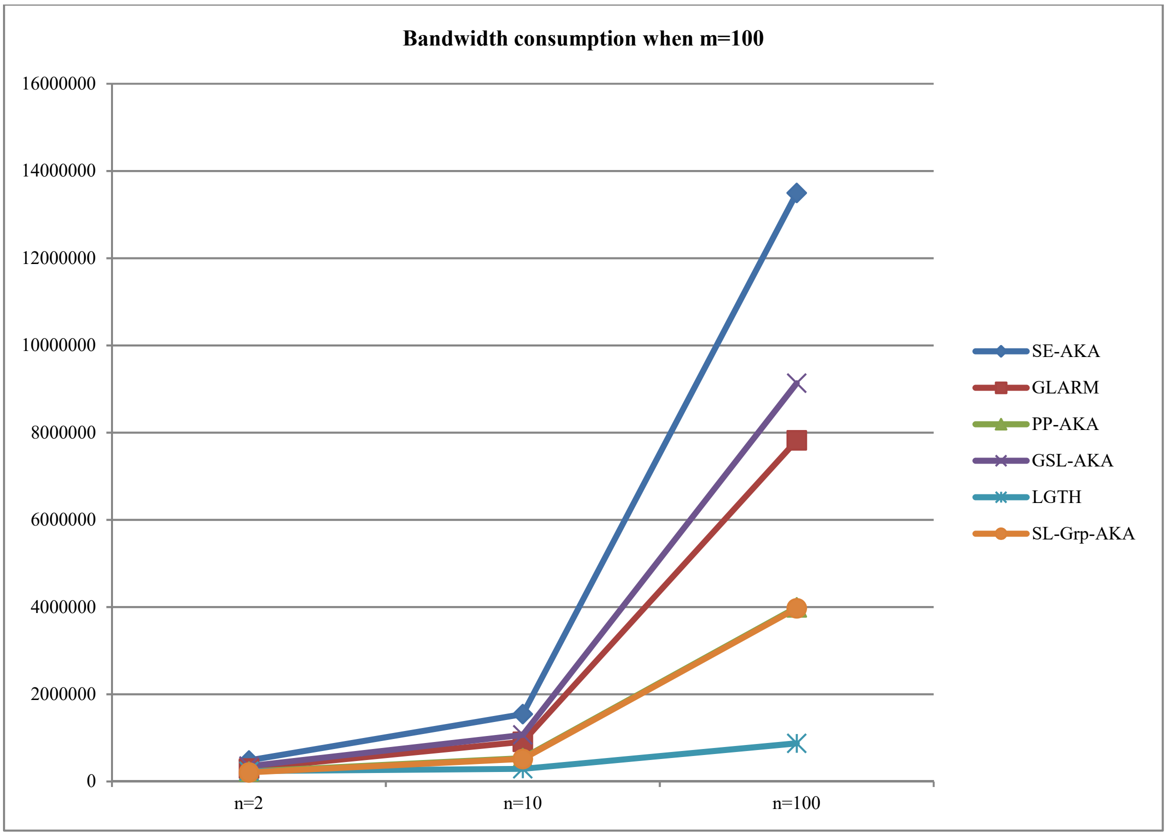

Figure 11(c): Bandwidth consumption comparison, for m=100

Table 8: Comparative observations of the Protocols' Security Objectives

Security Objectives	**SE-AKA [4]**	**GLARM [7]**	**LGTH [11]**	**PP-AKA [17]**	**GSL-AKA [19]**	**SL-Grp-AKA (Proposed)**
Mutual Authentication and key agreement	Yes	Yes	Yes	Yes	Yes	Yes
Type of cryptosystem	Symmetric	Symmetric	Symmetric	Symmetric	Symmetric	Asymmetric And Symmetric
Authentication of a group of MTCDs simultaneously	No	Yes	Yes	Yes	Yes	Yes
Congestion Prevention	No	Yes	Yes	Yes	Yes	Yes
Privacy Preservation	No	No	Yes	Yes	Yes	Yes
Key Backward and forward secrecy	Yes	Yes	Yes	Yes	Yes	Yes
Follow 3GPP standard	No	Yes	Yes	Yes	Yes	Yes
Resistance to DOS attack	Yes	No	No	Yes	Yes	Yes
Resistance to MITM attack	Yes	Yes	Yes	Yes	Yes	Yes
Resistance to Redirection attack	Yes	Yes	Yes	No	Yes	Yes
Resistance to Impersonation attack	No	No	No	Yes	Yes	Yes
Resistance to replay attack	No	No	No	Yes	Yes	Yes
Minimal Computational Complexity	Partially	Partially	Yes	Partially	Partially	Yes

The validation of this protocol is also performed formally with the well-known protocol validation tool: AVISPA and simulation results are presented in Section V. The cryptanalysis of the proposed protocol is also performed vigorously to determine its robustness and found to be tougher for the attackers. The expansion of this work can be carried out by designing light weighted and robust encryption techniques to suit the low configured devices in the constrained environment. Further, we would like to extend our work in pursuit of designing more robust and secure authentication mechanism for future 5G networks.

REFERENCES

[1] 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Security aspects of Machine-Type Communications (Rel 12), 3GPP TR 33.868 V0.10.0, sSeptemper 2012.

[2] Yu-Lun Huang, Chih-Ya Shen, Shiupyng Winston Shieh, "S-AKA : A Provable and Secure Authentication and Key Agreement Protocol for UMTS Networks"IEEE Transactions on Vehicular Technology Vol 60 No 9 Nov 2011.

[3] Yu-Wen Chen,Jui Tang Wang,Kuang Hui Chi,Chien Chao Tseng, "Group Based Authentication and Key Agreement," Springer Science + Business Media, LLC, 2010.

[4] Chengzhe Lai, Rongxing Lu, Xuemin "SE-AKA: A Secure and Efficient Group Authentication and Key Agreement Protocol for LTE Networks, ," ELSEVIER, Computer Networks, 2013.

[5] C. Lai, H. Li, X. Li, and J. Cao, "A novel group access authentication and key agreement protocol for machine-type communication," Transactions on Emerging Telecommunications Technologies,vol. 26, no. 3, pp. 414–431, 2015.

[6] Y. Zhang, J. Chen, H. Li, W. Zhang, J. Cao, and C. Lai, "Dynamic group based authentication protocol formachine type communications," in Proceedings of the 4th International Conference on Intelligent Networking and Collaborative Systems (INCoS '12), pp. 334–341, IEEE, Bucharest, Romania, September 2012.

[7] Chengzhe Lai, Rongxing Lu, Dong Zeng, Hui Li, Xuemin "GLARM:Group based Lightweight authentication scheme for resource constrained machine to machine communications," ELSEVIER, Computer Networks, 2016.

[8] Chengzhe Lai, Rongxing Lu, Xuemin, Hui Li, Rong Jiang, "SEGR: A Secure and Efficient Group Roaming Scheme for Machine to Machine Communications between 3GPP and WiMAX Networks" IEEE ICC 2014 - Communication and Information Systems Security Symposium.

[9] Jin Cao, Maode Ma, Hui Li, "A Group-based Authentication and Key Agreement for MTC in LTE Networks" Globecom Communication and Information System Security Symposium 2012.

[10] Jinguo Li, Mi Wen, Tao Zhang, "Group based authentication and key agreement with dynamic policy updating for MTC in LTE-A Networks" accepted for publication in IEEE Internet of Things Journal.

[11] Chengzhe Lai, Rongxing Lu, Xuemin, Hui Li, Rong Jiang, "LGTH: A Lightweight Group Authentication Protocol for Machine Type Communications in LTE Networks" Globecom Communication and Information System Security Symposium 2013.

[12] Probidita Roychoudhury, Basav. Roychoudhury, Dilip Kumar Saikia, "Hierarchical Group Based Mutual Authentication and Key Agreement for Machine Type Communication in LTE and Future 5G Networks" WILEY, Hindawi, Security and Communication Networks, Volume 2017.

[13] Shubham Gupta, Balu .L.Parne, Narendra S. Choudhary, "DGBES: Dynamic Group Based Efficient and Secure Authentication and Key Agreement Protocol for MTCin LTE/LTE-A Networks" Springer Science and Business Media, LLC, 2017.

[14] Shubham Gupta, Balu .L.Parne, Narendra S. Choudhary,SEGB: Security Enhanced Group Based AKA Protocol for M2M Communication in an IoT Enabled LTE/LTE-A Network" IEEE Access, accepted December 21, 2017, date of publication January 5, 2018, date of current version February 28, 2018.

[15] Hu Xiong, Zhi Guan, Zhong Chen, and Fagen Li, "An efficient certificateless aggregate signature with constant pairing computations" ELSEVIER, Information Sciences, 2013.

[16] S.K,Hafuzul Islam,Rahul Amin, G.P. Biswas, Mohammad Sabzinejad Farash, Xiong Li, Saru Kumari, "An improved three party authenticated key exchange protocol using hash funtion and elliptic curve cryptography for mobile-commerce environments", "Journal of King Suad University-Computer and Information Sciences", 2017.

[17] Anmin Fu, Jianye Song, Shuai Li, Gongxuan Zhang and Yuquing Zhang, "A Privacy Preserving group authentication protocol for machine type comminication in LTE/LTE-A networks" WILEY, Hindawi, Security and Communication Networks, Volume 2016.

[18] Ana Paula G Lopes, Lucas O Hilgert, Paulo R.L Gondim, Jaime Lloret, " Secret sharing-based authentication and key agreement protocol for machine-type communications", International Journal of Distributed Sensor Networks, Vol. 15(4), DOI: 10.1177/1550147719841003, 2019

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

[20] Menezes, A.J. Elliptic Curve Public Key Cryptosystems; Kluwer Academic Publishers: Boston, MA, USA, 1993.

[21] Hankerson D., Menezes A., Elliptic Curve Discrete Logarithm Problem. In: van Tilborg H.C.A., Jajodia S. (eds) Encyclopedia of Cryptography and Security. Springer, Boston, MA, 2011.

[22] Junichi Yarimizu1 , Yukihiro Uchida1 and Shigenori Uchiyama, "The elliptic curve Diffie-Hellman problem and an equivalent hard problem for elliptic divisibility sequences", JSIAM Letters Vol.6 pp.5–7 2014.

[23] A. Armando, D. Basin, Y. Boichut et al., "The AVISPA tool for the automated validation of internet security protocols and applications," in Proceedings of the 17th International conferenceon computer aided verification. 2005.

[24] AVISPA. "Automated validation of internet security protocols and applications," , http://www.avispa-project.org/, 2014.

[25] Rahul Amin, S.K,Hafuzul Islam, Muhammad Khurram Khan, Arijit Karati,Debasis Giri, Saru Kumari, " A two- factor-RSA based authentication systems for multiserver environments", Wiley, Hindawi, 2017.

[26] S. Chatterjee, S. Roy, A. K. Das, S. Chattopadhyay, N. Kumar and A. V. Vasilakos, "Secure Biometric-Based Authentication Scheme Using Chebyshev Chaotic Map for Multi-Server Environment," in IEEE Transactions on Dependable and Secure Computing, vol. 15, no. 5, pp. 824-839, doi: 10.1109/TDSC.2016.2616876 1 Sept.-Oct. 2018.

[27] M. Rekik, A. Meddeb-Makhlouf, F. Zarai, M. S. Obaidat and K. F. Hsiao, "A SCTP-based authentication protocol: SCTPAP," 2014 5th International Conference on Data Communication Networking (DCNET), , pp. 1-7, Vienna, 2014.

[28] Glouche, Yann & Genet, Thomas & Heen, Olivier & Courtay, Olivier, A security protocol animator tool for AVISPA, (2006).

[29] A. R. R. Shaikh and S. Devane, "Verification of security properties of payment protocol using AVISPA," 2009 International Conference for Internet Technology and Secured Transactions, (ICITST), London, pp. 1-6, doi: 10.1109/ICITST.2009.5402598, 2009.

[30] Vishesh, Kinchit & Verma, Amandeep, Formal verification of authenticated AODV protocol using AVISPA. International Journal of Computer Applications. 50. 38-43. 10.5120/7914-1179, 2012.

[31] Sun, Hung-Min & Wang, Feng., Defending Secret-Key Based Authentication Protocols against the Stolen-Secret Attack. Proceedings of the International Symposium on Electronic Commerce and Security, ISECS 2008. 385-389. 10.1109/ISECS.2008.36, 2008.

[32] Thing, Vrizlynn L. L, Lee, Henry C. J., Sloman, Morris", "Traffic Redirection Attack Protection System (TRAPS)", "Security and Privacy in the Age of Ubiquitous Computing",Springer US, pg 309-325, 2005.

[33] B. Bhushan, G. Sahoo and A. K. Rai, "Man-in-the-middle attack in wireless and computer networking — A review," 3rd International Conference on Advances in Computing,Communication & Automation (ICACCA) (Fall), Dehradun, pp. 1-6, doi: 10.1109/ICACCAF.2017.8344724, 2017.

[34] Malladi, Sreekanth & Alves-Foss, Jim & Heckendorn, Robert. "On Preventing Replay Attacks on Security Protocols. Proc. International Conference on Security and Management", 2002.

[35] Paul Syverson. A taxonomy of replay attacks. In Proceedings of the Computer Security Foundations Workshop (CSFW97), pages 187–191, June 1994.

[36] Li Gong and Paul Syverson. Fail-stop protocols: An approach to designing secure protocols. In 5th International Working Conference on Dependable Computing for Critical Applications, pages 44–55, September 1995.

[37] D. Denning and G. Sacco. Timestamps in key distribution protocols. Communications of the ACM, 24(8):553–536, August 1981.

[38] Z. Chao-yang, "DOS Attack Analysis and Study of New Measures to Prevent," 2011 International Conference on Intelligence Science and Information Engineering, Wuhan, pp. 426-429, doi: 10.1109/ISIE.2011.66, 2011.

[39] Yilmaz, Mustafa & Arslan, Huseyin. (2013). Impersonation attack identification for secure communication. IEEE Globecom Workshops, 1275-1279. 10.1109/GLOCOMW.2013.6825169, 2013.

BIOGRAPHIES

Geeta Kakarla, M.Tech in Software Engineering from JNTUH, Hyderabad. She possess 10 years of experience in Academic has guided many UG students. Currently she is working as Assistant Professor at Sreenidhi Institute of Science and Technology, Hyderabad. Her areas of interest include IoT, Web Technologies, Information Security, IoT and Network Security.

Dr. Phanikumar Singamsetty completed his B.E.(Computer Science & Engineering) from VTU, Belgaum M.Tech.(Software Engineering) and Ph.D. from Bharath University, Chennai. Currently he is working as Professor & Head, Department of Computer Science & Engineering, School of Technology, GITAM Deemed to be University, Hyderabad. He has 25 research papers in reputed peer reviewed journals in addition to 12 papers in International Conferences to his credit. He has co-authored 04 book chapters in Springer series. He is Life member of ISTE, member of CSI, member of Indian Science Congress Association. His research interests are software safety, safety critical systems, Machine Intelligence, Wireless Sensor Networks and IoT Security.