Services and Applications Security in IoT Enabled Networks

Philani Khumalo[1], Bakhe Nleya[2], A. Gomba[2] A Mutsvangwa[3]
[1,2]*Electronic Engineering Department, Steve Biko Campus, DUT*
South Africa
[1]massygomba@gmail.com, [2]bmnleya@ieee.org, [3]andrew.mutsvangwa@nwu.ac.za

***Abstract*—5G wireless together with optical backbone networks are expected to be the main pillars of the envisaged next /future generation networking (N/FGN) infrastructures. This is an impetus to practical realization of an IoT network that will support and ensure relatively higher bandwidth as well as enhanced quality of service (QoS) in both access and core network sections. The high-speed wireless links at the network peripherals will serve as a conducive platform for device-to-device (D2D) communication. D2D driven applications and services can only be effective as well as secure assuming the associated machine type communication devices (MTCDs) have been successfully verified and authenticated. Typically, D2D type services and applications involve the interaction of several MTCDs in a group. As such, secure and effective D2D group-based authentication and key agreement (AKA) protocols are necessary. They need to inherently achieve efficacy in maintaining the group key unlink-ability as well as generate minimal signalling overheads that otherwise may lead to network congestion. In this paper we detail a secure and efficient Group AKA (Gr-AKA) protocol for D2D communication. Its performance is compared to that of existing similar protocols and is found to comparably lower both computational as well as signalling overhead requirements. Overall the analysis shows that the Gr-AKA protocol improves performance in terms of fulfilling D2D communication's security requirements.**

Keywords—*5G network, group authentication, device-to-device (D2D), communication, security, privacy*

I. INTRODUCTION

The gradual shift from 3G/ 4G to 5G IoT enabled wireless as well as optical backbone networks will result in the provisioning of relatively higher bandwidth, lowered end-to-end latencies, massive device connectivity, reduced cost as well as consistent Quality of Experience (QoX). The shift also serves as a conducive platform for device-to-device (D2D) communication. Globally, D2D communication associated services and applications are steadily growing as billions of objects and devices are interconnected to form an Internet of Things (IoT) network [1].The combination of 5G wireless and optical backbones networks will enable capabilities of handling relatively larger mobile cellular data densities, support higher bandwidths to end users and serve vast numbers of objects and devices in comparison to current 3G/4G and optical networks.. This is illustratd in Figure 1. D2D communication will facilitate devices to interact directly without the aid of intermediary network elements [2]. In other words, it enables elements to communicate directly with each other without traversing fixed network infrastructural elements such as base stations or access points.

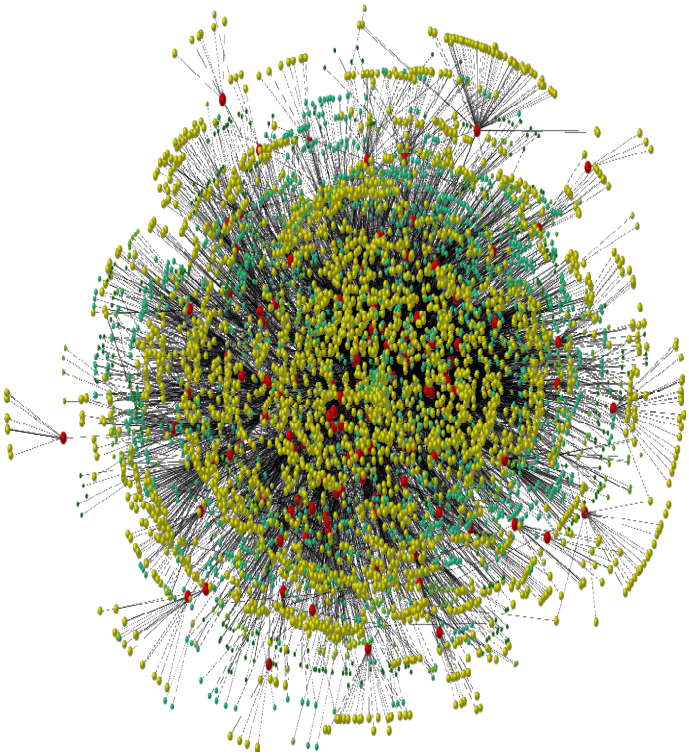

Figure 1. Devices in an IoT enabled Network

The main aim of D2D communication is to leverage the physical proximity of communicating objects as well as devices hence extending connectivity to sparse as well as remote environments [1]. D2D communication is primarily based in network infrastructural elements being involved in setting up a direct link between devices as well as the resources on which the communication will be facilitated. In classification terms, with *self-organized D2D communication*, the users themselves coordinate the setting up of the communication, whereas on the other hand with *network-assisted D2D communication*, the network infrastructural elements such as base stations (BS) assist direct data-transmission by means of control signaling and resource management [2]. D2D communication can further be distinguished as either,

978-1-5386-6477-3/18/$31.00 ©2018 IEEE

out-*band D2D* in which users **communicate** over the unlicensed spectrum or In-*band D2D***,** in which the users utilize the licensed spectrum of the network operator.
In-band D2D communication can further be subdivided into (i) *Underlay in-band D2D* in which the D2D and users share the same frequency bands in order to increase the spectrum efficiency of the network and ii) *Overlay in-band D2D* in which D2D and users transmit over non-overlapping frequency bands [3].

D2D communication can be differentiated with Machine-to-Machine Communication (M2M) as the latter is essentially a paradigm similar to it. [4], M2M communication can be defined as "Data communication among machines or devices that do not require human mediation nor impose specific restrictions on communication ranges and is based on traditional networks such as 3G and LTE. It links communicating devices via routable core networks and M2M servers, even if the two devices are in proximity. It is essentially an application-oriented technology. D2D communication on the other hand assumes close proximity between devices and relies only on local device capabilities without centralized infrastructure support. It can be used for M2M communication to improve network performance and QoS [4].

Consequently, the next step is to facilitate interactions between humans and the devices/objects literally independent of human involvement [4]. Hence a need to facilitate traffic to traverse any network infrastructure. The success of D2D communication will facilitate key services and applications such as, health-care monitoring systems, cloud computing, smart transportation, intelligent tracking and tracing systems, smart cities as well as smart power grids. Fig.1 illustrates a few examples: -
Social entertaining services-sharing of multimedia files between trusted users in close proximity using social networking applications will be easily facilitated.
Public safety-D2D communication technology will enable areas that suffer natural disasters (e.g., hurricanes and earthquakes) still maintain connectivity with rescue personnel since D2D communication focuses on relaying information over the still available reliable short-range links.
Vehicular to vehicle (V2V) networks services-These will support the exchange of information between vehicles in close proximity in order to avoid catastrophic accidents and to improve traffic management.

Security and privacy now remain a key issue to be addressed when establishing D2D communication for the multitudes of MTC type devices (MTCDs) in an IoT enabled network. Prevailing security related protocols and technologies suffice for addressing most of the security issues in legacy networks but are not directly applicable to the IoT network as they are often memory and computing resources intensive. IoT objects and devices are often deployed and work in harsh, erratic and even intimidating environments, where they can easily be prone to various security breaches.

The various D2D based services and applications can only be effective as well as secure assuming the MTCDs have been successfully verified and as well as authenticated. Most D2D services and applications involve interaction of a group of MTCDs.

In this regard, several group-based AKA protocols continue to be explored for achieving effective authentication. Primarily they all must satisfy security requirements such as confidentiality, mutual authentication, integrity, privacy preservation and most importantly utilizing a common and single security (encryption) key during the communication sessions in the IoT network. Such protocols need to inherently achieve efficacy in maintaining the group key unlink-ability as well as generate minimal overheads that otherwise may lead to network congestion [6].

A: 3GGP MTC Arcitecture

An MTC architecture in a IoT network was proposed by the 3GPP committee [6] to facilitate the authentication between MTCDs and MTC users as illustrated in Figure 2.

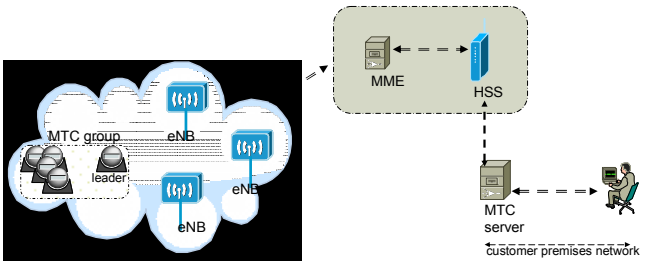

Figure 2. Communication Scenario [5]

The Mobile Management Entity (MME) as well as Home Subscriber Server (HSS) typically are located in network's pheriphery. The MTC server acts an as intermediary between MTC users and MTCDs. Users access it via an API. Secure communication between MTC serverand MTCDs requires prior authentication of the MTCDs by the network. This cannot be achieved using conventional protocols as they have short comings. Some of these short comings include, their tendency to generate excessive computational and signaling overheads as well as incapability to address key backward/foward secrecy (KBS/KFS) each time a device vacates or joins a group [7].

Generally, D2D devices do provide support for various features, normally geared towards optimizing the network resources available for use by applications. These are accessed as well as subscribed n the HSS. As such a single subscription can be utilized or shared by several devices. This will also include the associated security and access and control credentials associated with the subscription. Flexibility is granted in the form of subscribers being able to activate or deactivate some features whilst the subscription is valid. [8],[15].The 3GPP specification also addresses mechanisms for devices signaling congestion avoidance as well as traffic overload control. These are categorised as either soft or rigid.. With the soft apporoach, the service provider takes "soft" measures to try address any signalling congestion as well as traffic overload by minimising the number of attempts allowed per device.. With the rigid aproach associated devices are throttled altogether

Privacy and reliable mutual authentication and key negotiation are essential between the two communicating parties in D2D communication. In the absence of stringent privacy as well as security guarantees, data exchanged via the D2D communication links can easily be vulnerable to various attacks, e.g., eavesdrops, Man-in-the-Middle (MitM) and impersonation attacks. A user's identity can also be easily compromised [7].

The focus therefore is to secure D2D communication services such that user privacy and identity is never compromised. There is a tendency that a significant number of D2D applications and services will involve grouped devices and that they will be linked over several domains rather than within a single locality. In this kind of scenario, security issues are addressed taking into consideration that users are located under different domains, and that the D2D communication spans over several domains, operated by different operators, and with varying security policies.

Given the challenges and issues cited earlier, the Gr-AKA protocol for D2D communications (GAKA-D2D) that operates among multiple domains as well as operators. Its contributions are as follows: -

It ensures a group authentication mechanism that authenticates the group of participating MTCDs concurently.

- It preserves the security as well as privacy of the devices and at the same time maintains group key secrecy whenever a MTCD enters or vacates from it.
- For scalability as well as resource constraints, our proposal utilizes symmetric keys for authentication and authorization rather than asymmetric keys. As such it overcomes the security issuesof the network and generates relatively less signaling overheads.
- It satisfies key security requirements for D2D communication with moderate levels of both signaling as well as computational overhead.

II. PROPOSED PROTOCOL

The section commences by defining the system model as well as defining security assumptions. We then present the GAKA protocol. The system model is based on the generalized 3GPP MTC architecture as described in the previous section. We consider a conventional application scenario such as remote weather monitoring or crime surveillance. Initially, an MTC user registers with the local service provider for such a D2D communication service. This is followed by the network identifying a group of MTCDs ($MTCD_{grp_i}$) in the targeted area and initializing them. The group then designates a group leader ($grp_{i-leader}$) who will in turn negotiate both authentication and key establishment with the HSS/MME on behalf of the group. During this phase, the MTCDs and the HSS authenticate via the MME. Session keys are established between MTCDs and HSS for secure transmission of messages.

Session Key Compliance Stage: To ensure secured message exchanges between the MTCDs and the designated group leader, a session key is established among the group members. Because individual MTCDs may leave or exit the group; for each exit/or joining, key updating is necesary. A key generation center (KGC) communicates the updated information to all group member MTCDs.

MTCD join event: When a device joins a group, a new key is generated, so is the case when an existing member vacates the group.

MTCD exit event: A member can exit upon completing their task. In this case it must be prevented from accesing the group's resources, otherwise security is breached.. Hence the necessity to update current keys.

A. System Assumptions

We make the following assumptions:

- Considered is an applications or service in which several MTCDs together cooperate to form a group on one end and a single MTC user at the other end.
- The users are not necessary in their HOME location, and thus therefore prior registration (in case of roaming users) is necessary.
- The IoT service providers initially generate and agree on common system parameters as well as intra and inter operator agreements for D2D applications and services.

Asynchronous $(t;m;n)$ Group Authentication Scheme as proposed in [8] and further explored in [9] is utilized as the basis for carrying out group authentication.

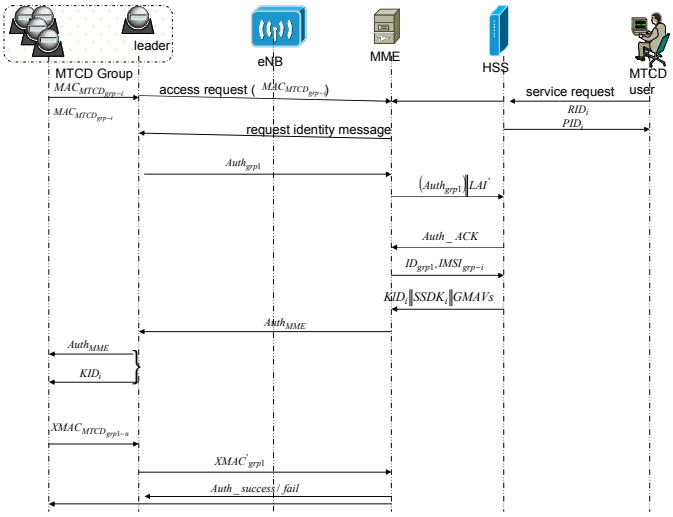

Figure 3. Sequence events for the proposed protocol

Asynchronous (t,m,n) group and authentication guarantees group authentication for m devices of a group with n members as well as being tolerant to t compromised tokens. In our protocol, it is considered that m has the same size of n, that is, all the members in a group are authenticated. Thus, it authenticates all the devices in a group simultaneously. The various sequence of events is summarised in Fig.ure 3. The detailed descriptions are narrated in the next section.

B. Session Request and Group Registration

A roaming MTC user registers for D2D services.

The MTC user (U_i) with a valid identifier (RID_i) performs user registration with the local HSS by furnishing his/her RID_i. If request is granted, the latter generates and issues a pseudonym ID (PID_i) to the user.

$$PID_i \overset{\text{def}}{=} (psedd, ExpiryTime) \tag{1}$$

The same RID_i will be used in the group initialization as well as key establishment process. The HSS also establishes and configures key parameters necessary for authenticating any formed MTC groups. Specifically, it generates a set of random numbers $\mathbf{R_z} \in \mathbf{Z_p^*}(z = 1,2,...i)$ and uses the set to compute a set of temporary identities $TID_{MTCD_{i-j}}$ to each $MTCD_{i-j}$ in a group:

$$TID_z = h_1(ID_{MTCD} \| \mathbf{R_z} * x) \tag{2}$$

where,
$h_1(.)$ is a secure hash function.
x is HSS's own secret key.

The HSS ultimately organizes the MTCD group into a binary tree [8]. Each node of the tree has a secret key that is known to each member MTCD. However, the secret keys of the nodes forming a path between a given MTCD and the root of the tree is not disclosed.
The HSS calculates a group key as follows:

$$GK_i = h_3(\sec_{i-1} \oplus \sec_{i-2} \oplus ..., \oplus \sec_{i-j} \oplus g * x) \tag{3}$$

where $g -$ is a random number, and $h_3(.)$ is a key generation function.

HSS further selects three hash functions; $h_1(.)$, $h_2(.)$, $h_3(.)$ which the key generation center (KGC) uses to generate an authentication message S to be used for group authentication. It also generates **k** tokens, all being a function of TID_{MTCD_i} to each device. These tokens must remain secret to any device outside the group. Finally the KGC computes and publishes the hash, function of S, $H(S)$ as well as hash function $H(.)$ that will be used to verify the validity of all MTCDs in the group.

C. MTCD Group Authentication and Key Agreement

This commences when a set of identified MTCDs within network coverage range request access so as part of a service/ application rendering. These are identified as a group ($MTCD_{grpi-j}$). We assume that within the group, the device with higher communication capability as well as battery reserve, will be designated as group leader($grp_{i-leader}$). The service provider then assigns a key (K_{grpi-j}) to each group member, as well as generating a group key which will be shared by both the MTCD group and HSS. The group key is used by individual MTCDs in the group for mutual authentication as well as privacy protection between MTCDs and service provider.

This is carried out mainly by the MTCDs' group leader and the HSS where the user is located. This is accomplished in the following sequence:

1. Each MTCD group member broadcasts a fresh temporary identifier $TID_{MTCD_{i-j}}$ and associated token $f(TID_{MTCD_{i-j}})$ to the group leader.

$$MTCD_{i-j} \rightarrow \lfloor TID_{MTCD_{i-j}}, f(TID_{MTCD_{i-j}} \rfloor \Rightarrow MTCD_{i-leader} \tag{4}$$

2. The group leader computes the Lagrange component vector for the group (LC_{MME}) using $TID_{MTCD_{i-j}}$ $f(TID_{MTCD_{i-j}})$ values received from the KGC.

$$MTCD_{i-j} \rightarrow LC_{grpi}$$

The general formula used is:

$$LC_{grp} = f(TID_{MTCD_{1-i}}) \prod_{q=1, q\neq j}^{\frac{n}{m}} \frac{-TID_{MTCD_{1-q}}}{TID_{MTCD_{i-j}} - TID_{MTCD_{i-q}}} \bmod p \tag{5}$$

This component is broadcast back to all group members. Each member uses it to verify whether all members are legitimate, by calculating the secret key S and comparing the result with $H(S)$ published by the KGC during registration phase.

3. The group leader further authenticates the group with the MME. In so doing, it first computes the group's $MACgrp_i$ and $Auth_{grp_i}$.

$$MAC_{grp_i} = h_2(GK \| ID_{grp_i} \| LAI \| S') \tag{6}$$

$$Auth_{grp_i} = (TID_{grp_i} \| MAC_{grp_i}) \tag{7}$$

$$MTCD_{grp_{i-leader}} \xrightarrow{Auth_{grp_i}, TID_{MTCD_{1-1}}, ...TID_{MTCD_{1-j}}} \rightarrow MME \tag{8}$$

4. The MME confirms with the corresponding HSS on whether the MTCD group is legitimate or not.

$$MME \rightarrow \underrightarrow{Auth_{grp_i}, LAI} \rightarrow HSS \tag{9}$$

5. Upon receipt of authentication verification request message from MME, the HSS authenticates the group by computing the group's MAC_{grp_i} using values received from the MME versus those it has in store.

$$MAC'_{grp_i} = h_2(GK \| ID_{grp_i} \| LAI \| S) \tag{10}$$

$MAC'_{grp_i} = MAC_{grp_i}$ implies successful authentication by the HSS, and the MTCD group leader will be informed accordingly.

HSS also further generates a temporary group key GTK for the MTCG Group.

$$GTK_{grp_i} = h_3(GK \| r_{HSS}) \tag{11}$$

Where, r_{HSS} is a random number. It also generates a token to MME that will enable the devices to authenticate the MME in future sessions

$$HSS \rightarrow \underrightarrow{f(ID_{MME})\|GTK_{grp_i}\|r_{HSS}} r \rightarrow MME \quad (12)$$

6. Upon receipt of messages from HSS, MME calculates its own Lagrange component LC_{MME} as well as $Auth_{MME}$ and broadcasts them to the MTCD group leader.

$$LC_{MME} = f(ID_{MME}) \prod_{q=1}^{\frac{n}{m}} \frac{-TID_{MTCD_{i-q}}}{ID_{MME} - TID_{MTCD_{i-q}}} \times \bmod p \quad (13)$$

$$Auth_{MME} = (LC_{MME}\|r_{MME} \oplus GTK\|r_{HSS}\|ID_{MME}) \quad (14)$$

Upon receiving $Auth_{MME}$, and encrypted KID_i the group leader broadcasts them to the rest of the group members.

7. Upon receiving the messages relayed from the MME each device updates its Lagrange component as follows:

$$LC\_new_{MTCD_{i-j}} = LC_{MTCD_{i-j}} * \frac{-ID_{MME}}{TID_{MTCD_{i-j}} - ID_{MME}} \quad (15)$$

Each device also uses the received r_{HSS} value to calculate GTK:

$$GTK_{grp_i} = h_3(GK\|r_{HSS}) \quad (16)$$

It also computes its integrity and cipher keys as well as $K_{asme}^{'MTCD_{grp_{i-j}}}$:

$$IK'_{grp_{i-j}} = h_4(ID_{grp_i}\|r_{HSS})K_{grp_{i-j}} \quad (17)$$

$$CK'_{grp_{i-j}} = h_5(ID_{grp_i}\|r_{HSS})K_{grp_{i-j}} \quad (18)$$

$$K_{asme}^{'MTCD_{grpi}} = KDF(GTK_{grpi}\|IK'_{grp_{i-j}}\|CK'_{grp_{i-ij}} \|ID_{grp_i}\|IMSI_{grp_{i-j}} \quad (19)$$

It further computes its own response value and sends it to the group leader.

$$XMAC_{MTCD}{}'_{grp_{i-j}} = h_1(ID_{grp_i}\|r_{HSS}\|IMSI_{grp_{i-j}})_{GTK_{grp_i}} \quad (20)$$

The group leader uses the response values from each of the group members to finally compute the group response.

$$XMAC_{grp_i} = h_1(XMAC_{MTCD_{grp1}} \oplus XMAC_{MTCD_{grp1-2}} \oplus \oplus,....XMAC_{MTCD_{grpn}})_{GRPK_1} \quad (21)$$

The Group leader finally passes the group response ($XMAC_{grp}$) to the MME for final authentication of each MTCD .

D. MTCD Joining or Exiting

In the event that a MTCD joins or vacates an already authenticated group, the secret S must be updated to avoid the old member to continue knowing the secret and to avoid new members discovering and exploiting previous secret values S. As illustrated in Fig. 4, when a MTCD joins, a new group key is generated:

$$GK_i^{'} = h_3(GK \oplus \sec_{i-j}) \quad (22)$$

Where $\sec_{i-j}$ is the secret value of the node to which the new MTCD is located. Likewise, HSS generates a new value for S as follows:

$$S_{new} = S + \delta S\ , \quad (23)$$

Where δS is a random value of S generated each time a member joins or exits.

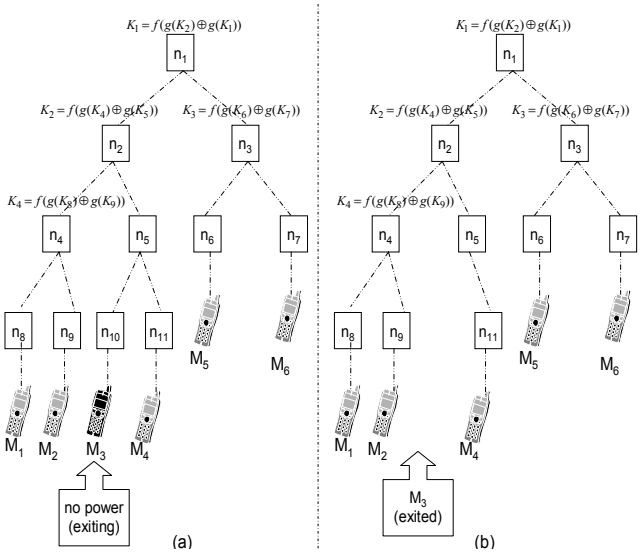

Figure 4. Example MTCD join/exit event tree

When a member exits, a new group key is computed according to:

$$GK_i^{''} = GK \oplus \sec_{i-j} \quad (24)$$

III. ANALYSIS AND PEFORMANCE EVALUATION

In this section we provide a general security analysis of the Gr-AKA protocol . Firstly we discuss its general security capabilities. We then go on to analyse metrics such as the amount of signalling overheads generated and exchanged bandwidth requirements, computational complexity as well as overall efficiency. We test some of the aspects of the Gr-AKA protocol using the AVISPA tool [10], [11].

A. Security Analysis

User's Privacy: At registration phase, **t**he MTC User's identity is mapped to a pseudonym ID (PID_i) and thereafter the latter is used for authentication purposes rather than the real User's name.. In this way the user's real identity is concealed hence privacy is guaranteed .

Mutual authentication: The Gr-AKA protocol provides robust mutual authentications between $User\_HSS$, as well as among the individual $MTCD_{grp_i}$ members. HSS authenticates the user by way of verifying MAC values computed using the User's credentials such as RID and PID. To authenticate HSS, the User checks the received MAC'' from the MME and if they both match with the $XMAC'$, then both MME and HSS are authenticated.

Similarly, HSS verifies and authenticates the $MTCD$ group by verifying ttheir their Lagrange components. Each member then uses these Lagrange components to compute the secret S and compares it with the same value that was sent from the KGC.

Backward /Forward Key Secrecy . With the Gr-AKA protocol the group key (GK) is updated and changed each time a device leaves or joins the group .When a device joins the group, HSS is compelled to broadcast its secret node, thus a new GK is computed using equation (22).
Similarly when a device exits, the remaining devices are compelled to update their GK using equation (24).

Attack resistivity: The channel between the MTC user and MTCD group is open to various attacks. To safeguard against replay attacks, time-stamped key hint messages are periodically exchanged between the two parties. A hacker who successfully intercepts the key hints exchange will not be able to replay a message for the next key hint exchange message because of the time stamping.

MiTM attack: The channel between the MME and HSS is assumed to be secure (in terms of integrity, confidentiality, and entity authentication), and only the channel between $MTCD_{grp_{i-j}}$ and MME may be vulnerable to MiTM attacks. However in the proposed protocol, the use of Shamir's secret sharing [9], together with the Lagrange component, makes it extremely difficult to recover the secret token. Furthermore, the group's ID is secret thus further making it difficult for attackers to generate or verify the MAC_{grp_i}.

B. Performance Analysis

The protocol is analysed in terms of its general security capabilities, computational demands/complexity as well as signalling overheads. The main security aspects of the protocol is tested using the Automated Validation of the Internet Security Protocols and Applications (AVISPA) tool[11]. We first created the model as in Figure 2, and also specifying the basic roles. We were able to verify that it can guarantee the privacy of a generated session key, as well as general authentication between MME and HSS.

To evaluate the total computational overheads, we compare the protocol to similar proposed protocols such as PPAKA-HMAC [13], G-AKA[13], and GBS-AKA [14]. In our analysis, we assume that overall there are n $MTCDs$ and each can have up to m members each. The following cryptographic computational times are utilised; maptopoint hash operation ($T_{mtp} = 0.07ms$), $MTCD$ Lagrange component computational time ($T_{L-MTCD} = 0.06ms$), HSS Lagrange computational time ($T_{L-HSS} = 0.04ms$), , multiplication over an elliptical curving ($T_{mul} = 0.6ms$), pairing ($T_{pair} = 4.5ms$), hash operation ($T_{hash} = 0.07ms$), symmetric ciphering/deciphering ($T_{aes} = 0.16ms$),. Table 1, summaries the computational overheads of the 4 protocols.

TABLE I. COMPUTATION COMPLEXITY OF GROUP PROTOCOLS [10]

AKA Protocol	Computational overhead		
	MTC Devices	Network	Total (ms)
PPAKA-HMAC [12]	$3T_{hash}*n+(T_{hash})*m$	$2T_{hash}*n+(T_{hash})*m$	$5T_{hash}*n+(2T_{hash})*m$ $2*Rand+(n+10)$ $*\exp(5+n)$ $*H_{mac}+(2n-1)$ $*Mul+1*Hash$
G-AKA[13]	$4T_{hash}+(4T_{hash})(n-1)$	$3(T_{hash})*n*(2T_{has}h)*m$	$7T_{hash}*n+(2T_{hash})*m$
GBS-AKA [14]	$(T_{\mathrm{mod}}+2T_{hash})*n+2T_{hash}*m$	$(T_{\mathrm{mod}}+2T_{hash})*n$ $+(6T_{hash}+T_{aes})*m$	$(2T_{\mathrm{mod}}+4T_{hash})*n$ $+(8T_{hash}+T_{aes})*m$
GR-AKA (proposed)	$(4T_{hash}+2T_{aes})*n$ $+(2T_{hash})*m$	$(3T_{hash}+2T_{aes})*n$ $+(2T_{hash})*m$	$(7T_{hash}+4T_{aes})*n$ $+(4T_{hash})*m$

We explored execution key generation time as a function of key size in bits. The key size is varied from 2^1 to about 2^{13} bits. In Figure.5, it is observed that as the key size is increased, so does the key generation time. However, increasing the key length makes it more secure. We thus chose to fix the key size to 12 bits (2^{10}), which is a 7 seconds delay. This is not so much a hindrance as key generation is a once off operation during initialization.

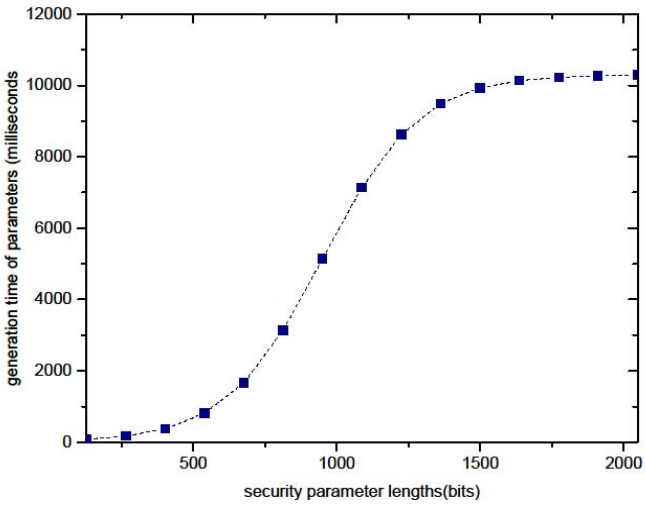

Figure 5. Overall protocol key generation time versus size

We further explore the proposed protocol's execution time and compare it with that of the PPAKA-HMAC [13], G-AKA[13],.GBS-AKA [14].

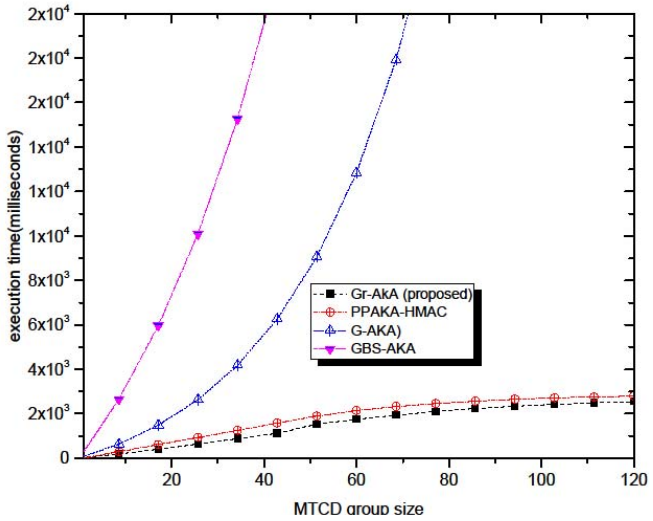

Figure 6. Execution time comparisons

The execution time more less increases linearly with increase in MTCD group size for the proposed protocol as well as PPAKA-HMAC[13]. However, we see an exponential increase in execution times with the other two protocols as clearly indicated by the graph of Figure. 6. We also analyse the overall magnitude signalling (communication) overheads of the proposed protocol. Overall the total signalling bits are computed from all the messages exchanged during the authentication process. Figure. 7 shows plots of total signalling (communication) overhead as a function of the number of MTCD groups, each comprising 5 members. Overall, the proposed protocol together with the PPAKA-HMAC generate more or less the same levels of signalling data, and not so excessive to cause congestion in the signalling channels.

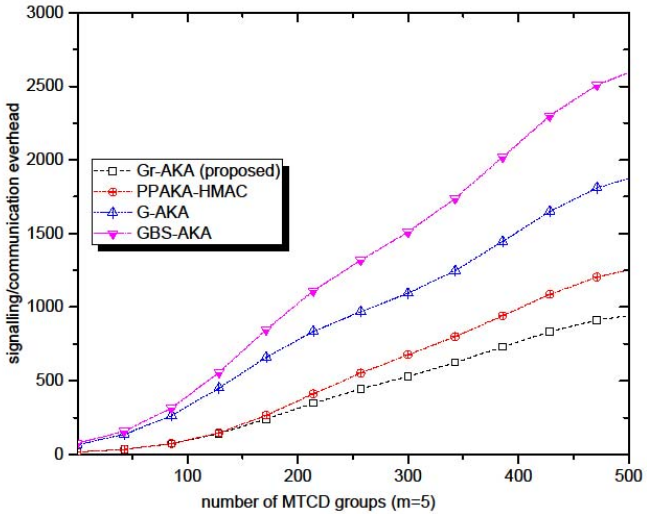

Figure 7. Signalling overhead

IV. CONCLUSION

We proposed the Gr-AKA protocol for enhancing security in D2D communication. It incorporates a secure group authentication mechanism that ensures MTCDs are authenticated as a group. As such, it preserves the privacy of the MTCDs and at the same time maintains the group's secrecy whenever a group member joins or vacates from it. The protocol is also designed to overcome the security problems of the network as well as the generation of relatively lesser overhead compared to other existing group-based AKA protocols.

REFERENCES

[1] E. Hossain and M. Hasan, "5G cellular: key enabling technologies and research challenges," IEEE Instrumentation & Measurement Magazine, vol. 18, no. 3, pp. 11–21, 2015.

[2] M. Gomba and B. Nleya, "Architecture and security considerations for Internet of Things," 2017 Global Wireless Summit (GWS), Cape Town, 2017, pp. 252-256, doi: 10.1109/GWS.2017.8300477.

[3] G. Fodor, E. Dahlman, G. Mildh, S. Parkvall, N. Reider, G. Miklos, and Z. Turanyi, "Design aspects of network assisted device-to-device communications," IEEE Communications Magazine, vol. 50, no. 3, 2012.

[4] D. Feng, L. Lu, Y. Yuan-Wu, G. Y. Li, G. Feng, and S. Li, "Device to-Device Communications Underlaying Cellular Networks," IEEE Transactions on Communications, vol. 61, no. 8, pp. 3541–3551, 2013.

[5] Technical Specification Group Services and System Aspects; 3GPP System Architecture Evolution (SAE); Security Aspects of Non-3GPP Accesses (Release 11), document 3GPP TS 33.402 V11.4.0, 3GPP, Jun. 2012.

[6] M. Haus, M.Waqas, A. Y. Ding, Y. Li, S. Tarkoma, and J. Ott, "Security and Privacy in Device-to-Device (D2D) Communication: A Review," IEEE Communications Surveys & Tutorials, vol. 19, no. 2, pp. 1054–1079, 2017.

[7] C. Lai, H. Li, R. Lu, and X. S. Shen, ''SE-AKA: A secure and efficient group authentication and key agreement protocol for LTE networks,'' *Comput. Netw.*, vol. 57, no. 17, pp. 3492–3510, 2013.

[8] Harn, L. Group Authentication. *IEEE Transactions on Computers*, v. 62, n. 9, p. 1893-1898, 2012.

[9] Shamir, A. How to share a secret. *Communications,* pp. 612-613, 1979.

[10] A. Armando, "The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications", In Computer Aided Verification. CAV 2005. Lecture Notes in Computer Science, vol 3576. Springer, Berlin, Heidelberg.

[11] http://www.irisa.fr/lande/genet/span.

[12] M. Wang and Z.Yan,"Privacy-Preserving Authentication and Key Agremment Protocvols for D2D Group Communications", IEEE Transactions on Industrial Informatics, vol 17, 2017.

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

[14] J. Yao, T. Wang, M. Chen, L. Wang, and G. Chen, "GBS-AKA: Groupbased secure authentication and key agreement for M2M in 4G network," in *Proc. IEEE Int. Conf. Cloud Computing. Reearchs. In-nov. (ICCCRI)*, May 2016, pp. 42-48.

[15] G. Singh, D. D. Shrimankar, "Dynamic Group Based Efficient Access Authentication, and Key Agreement Protocol for MTC in LTE-A", Wireless Networks, Personal Communications, Published online, 13 April, 2018.

[16] 3GPP, "Service Requirements for Machine-Type Communications," TS 22.368 V10.1.0, June 2010.

[17] Gomba and B. Nleya, "Overview Access and Control Considerations for Internet of Things," 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), Durban, 2018, pp. 1-7, doi: 10.1109/ICABCD.2018.8465435.

[18] M. Gomba , Bakhe Nleya, A. Mutsvangwa, " Applications Security for D2D Communication in IoT Enabled Networks", Proceeding of the Southern Africa Telecommunication Networks and Applications Conference (SATNAC) Arabella, Hermanus, Western Cape, South Africa, 2 - 5 September 2018.