posed threat model, and discusses the findings.

Outline. The paper is organized as follows. Section 2 introduces the reader to the basics of the AKA protocol. Section 3 motivates the need of group-based AKA protocols and defines the threat model. Section 4 describes the analysis of four group-based AKA proposals and discusses the results. Section 5 concludes the paper.

2 BACKGROUND

The goals of the AKA protocol are identification of subscriber, mutual authentication between the terminal and the serving network as well as generation of a session master key agreed between the terminal and the serving network. With a security take, the AKA protocol is strategical as it bootstraps the security parameters needed to form a security context that is agreed among the parties. In this section, we provide an overview of the current AKA protocol by shortly discussing the different roles, security requirements, and the protocol messages.

2.1 Roles

The terminology used in mobile telephony has changed each time a new standard was released. It happened with GSM, UMTS, and LTE. However, the set of tasks performed by each role has not changed despite the different names. We refer to the terminology adopted in LTE. The three roles concerning AKA are as follows.

- The *User Equipment* UE role is the combination of the tasks of the terminal device and USIM (or UICC). Each UE can be uniquely identified by a permanent subscriber identity (*IMSI*). At time of subscription, the UE is given a long-term secret key that is shared with the authentication server. In this paper, we use the term *machine-type communication* MTC to refer to the UE. This term is more appropriate in the context of 5G and IoT. In fact, the 3GPP consortium released a specification for MTC devices to enhance the LTE suitability for the IoT market (3GPP, 2011).
- The *Mobile Management Equipment* MME role concerns the tasks of covering the mobility of the MTC. A specific MME serves an MTC depending on the geographical area in which the MTC is located. The MME is part of the *serving network*, and we use both terms interchangeably. In the context of AKA, the MME authenticates the MTC when the latter wants to access the network. MME and MTC agree on a session master key $\mathrm{K}_{\mathrm{asme}}$ from which they can derive further keys to protect the signaling data.
- The *Home Subscriber Server* HSS is the authentication server that assists the MME to authenticate the MTC. The HSS knows the MTC identity and its long-term secret key. Moreover, HSS and MTC keep track of a *sequence number* (*SQN*) to support authentication. The communication between HSS and MME is normally secured with RADIUS or more recently with Diameter protocols (3GPP, 2008). As we shall see later, when an MTC requests network access to the MME, the latter forwards the request to the HSS, which provides an authentication vector that enables mutual authentication between MME and MTC.

2.2 Security Requirements

The security requirements of AKA have historically concerned the authentication of the user and the confidentiality of the session master key. In the last release, also authentication of the serving network has been considered, and more emphasis on protection of MTC identity has been posed. We briefly present the desired security requirements that an AKA protocol aims to achieve.

- *MTC authentication*: This requirement ensures that the MTC with the claimed identity was involved in the AKA protocol run with the MME.
- *Serving network authentication*: This requirement states that the MME with the claimed identity was involved in the AKA protocol run with the MTC, and that the HSS authorized that MME to provide network access to the MTC.
- *Session master key confidentiality*: This requirement prescribes that the session master key agreed between MTC, MME, and HSS is known only to them.
- *MTC identity privacy*: This requirement ensures that only legitimate parties can learn information regarding the MTC identity from messages occurring in an AKA protocol run.

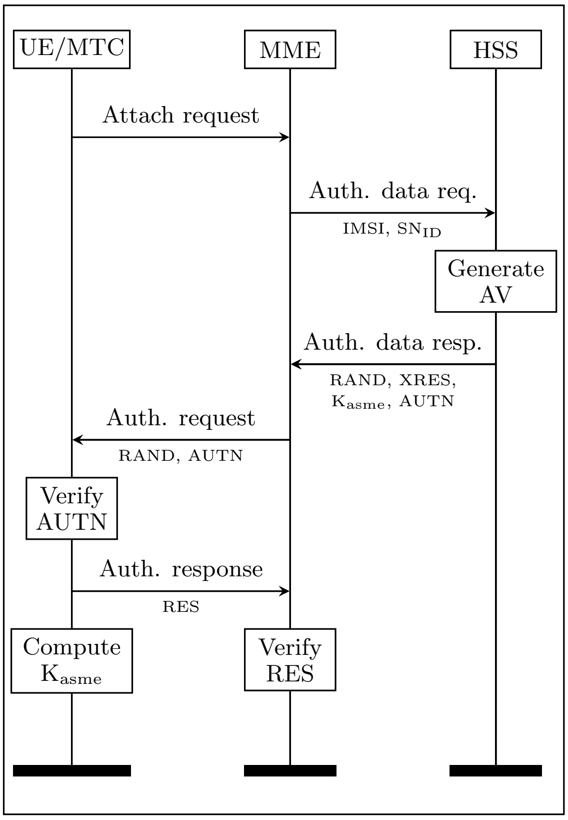

Figure 1: EPS-AKA message sequence chart

2.3 Overview of EPS-AKA

EPS-AKA is the last of the AKA protocol family as it is the standard for LTE. The goals of EPS-AKA match the security requirements outlined above. We give a high level description of messages and actions of EPS-AKA (see Figure 1).

The protocol begins with the *Attach request* message sent by the MTC to the MME. The message includes the IMSI of the MTC, when the device visits the MME for the first time. If the MTC has already visited the MME in the past, the message contains the Global Unique Temporary Identity (GUTI), which was assigned to the MTC by the MME in the previous visit. In doing so, the MME can translate the GUTI in the corresponding IMSI, and the privacy of the MTC can be assured. Then, the MME forwards its identifier ($\mathrm{SN}_{\mathrm{ID}}$) and the IMSI to the HSS in the *Authentication data request* message. The HSS generates an authentication vector containing:

- A random value *RAND* that provides freshness of the session;
- an expected response *XRES* that is based on RAND and long-term key.
- a session master key K_{asme} to achieve data confidentiality in the signaling between MTC and serving network;
- an authentication token $AUTN$ that is based on RAND, long-term key, and SQN. It allows the MTC to authenticate the serving network.

The *Authentication data response* message contains the generated authentication vector and is transmitted to the MME. The MME forwards RAND and AUTN to the MTC in the *Authentication request* message. The MTC retrieves the sequence number and checks if it matches a valid one. If so, the serving network is authenticated. The MTC computes the session master key and the response RES, which based on its long-term key and on the received RAND. It then sends the response to the MME in the *Authentication response.* If the received response RES corresponds to the expected response XRES, the MME successfully authenticates the MTC. From now on, signaling between serving network and MTC can be protected with keys derived from $\mathrm{K}_{\mathrm{asme}}$.

3 TOWARDS GROUP-BASED AKA

We now investigate on how the AKA procedure can benefit from a group-based approach.

A group is formed by one or more members that share similar features. Examples of common features include members that do the same task, members located in the same geographical area, or members that belong to the same owner. A group may also share a macro feature that is derived by a combination of single features. This scenario is the natural consequence of combining mobile communication and IoT. Thus, we shall unfold two use cases in support of such scenario.

Marathon. Our first use case is a marathon that gathers many participants who are equipped with MTC devices. Such devices gather some information, such as the position of the participant, and continuously need to access the network to upload the data. Entities in support of the marathon may be equipped with such devices as well. The devices should be able to continuously communicate to a remote service, and attach procedures can be executed simultaneously. Hence, the signaling between MTC and MME, and between MME and HSS may increase considerably. Grouping such devices according location and owner can reduce the signaling between MME and HSS, at least regarding the authentication of the devices.

Monitoring of goods. Another use case that may benefit from the presence of groups is the monitoring of goods during shipment (Seitz et al., 2016). Companies employ MTC sensors to monitor altitude, temperature, humidity, climate, or other environmental conditions of sensitive goods such as perishable food. Thus, the MTC sensors need network access to communicate with their owner, who may collect sensors values at specific times of the day. Additionally, a cargo container may contain goods and MTC sensors of different owners. Grouping the MTC sensor according tasks and owners avoid burdening the HSS due to the simultaneous authentication requests.

Therefore, the functional goal of a group-based AKA protocol is to authenticate a group of devices efficiently, minimizing the cost of repeated message exchanges and communication delays. More specifically, a group-based AKA protocol aims to reduce the signaling between MME and HSS when a large group of MTC with similar features requires network access simultaneously. While the literature has mainly focused on the functional goals of group-based AKA, we now concentrate on its security aspect.

Group-based authentication has seen several definitions and different threat models. Martucci et al. (Martucci et al., 2004) define group-based authentication as the process of verifying whether a device belongs or does not belong to a trusted group, considering honest group members only. Nguyen and Roscoe (Nguyen and Roscoe, 2006) specify group-based authentication as the mutual authentication of each member, with the main application of establishing a commonly shared secret key among the group members even in the presence of corrupt participants. In the context of AKA, we consider the following definition of group-based MTC authentication.

- *Group-based MTC authentication*: This requirement states that the MTC with the claimed individual and group identities was involved in the AKA message exchange with the MME.

Group-based MTC authentication naturally extends MTC authentication seen in section 2.2 with an explicit reference to the group identity. Thus, a group-based AKA protocol should ensure group-based MTC authentication and all the security requirements seen in section 2.2.

3.1 Threat Model

In the context of the AKA protocol, the threat model has historically concerned an intruder who wish to break subscriber's authentication or derive the session master key agreed between an MTC and the MME. Threats concerning privacy have been overlooked, and privacy issues emerged in AKA implementations. In fact, EPS-AKA is vulnerable to active attacks against the subscriber identity. One well-known problem is that the initial attach of the UE to the network requires the IMSI to be transmitted in clear text. Since the IMSI is unique for each subscriber, its leakage leads to subscriber tracking attacks. However, both UMTS and EPS AKA protocols proved to protect the subscriber identity against passive attacks if the MTC sends in the attach request the temporal identity GUTI. Both protocols also meet authentication of the subscriber and secrecy of the session key against active attacks (3GPP, 2001; Zhang and Fang, 2005).

In the scenario of group-based AKA, the historical threat model concerning the traditional AKA should be extended with additional threats stemmed from the group approach. A comprehensive list of such threats is outlined below.

- *The intruder is authenticated as MTC by the serving network.*

  This threat concerns the identification of the MTC by the serving network and originates from the traditional AKA threat model. The intruder may try to impersonate another MTC to get access to the network. The serving network shall ensure that network access is granted only to correctly identified MTC.

- *The intruder is authenticated as serving network by the MTC.*

  This threat also comes from the traditional AKA threat model. It concerns the identification of the serving network by the MTC. The MTC shall access the network only through a correctly identified serving network. A protocol that meets mutual authentication protects against this threat and the previous one.

- *The intruder derives the session master key agreed between an MTC and the serving network.*

  This threat concerns the secrecy of the session master key, which should be known by MTC and serving network only. It also originates from the traditional AKA threat model.

- *The intruder identifies and tracks an MTC.*

This is a privacy threat that allows the intruder to learn the IMSI, and use it to track the MTC via handover signaling messages. EPS-AKA resists to this threat only if perpetrated by a passive intruder.

- *The intruder is authenticated as member of the group by the serving network.*

  This is a novel threat introduced by the group approach. It is similar to the first threat outlined in this list but with a subtle difference: in this case the intruder does not need to impersonate another MTC. In fact, it is suffice to convince the serving network to be a member of the group to get access to the network. It follows that this threat must be addressed with appropriate mechanisms.

- *A corrupted member of the group is authenticated as another member of the group by the serving network.*

  So far the intruder has been considered as an external entity. This novel threat involves an intruder that is also member of the group, namely it corrupts and has the total control of an MTC. It signifies that this threat considers a more powerful intruder with additional knowledge derived from being part of the group. As the intruder may try to impersonate another member of the group, the serving network shall correctly identify the MTC before grant network access.

- *A corrupted member of the group is authenticated as serving network by the another member of the group.*

  This threat also involves an intruder that is part of the group. The goal of the intruder is to impersonate the serving network when an MTC, which is a member of the group as well, seeks for network access.

- *Colluding corrupted members of the group derive the session master key agreed between a third group member and the serving network.*

  This threat further extends the intruder capabilities with the ability to corrupt multiple MTCs that are members of the group. The intruder's goal is to learn the session master key agreed between the serving network and a third MTC not controlled by the intruder.

- *Colluding corrupted members of the group identify and track a third group member.*

  This last threat concerns the privacy of the group members. Again, the intruder might control a number of corrupted MTC. The goal of the intruder is to track an MTC that is not under its control.

The list outlined above contains nine threats, five of which are novel because the group approach. The new threats involve an intruder with the ability to corrupt and control MTC that are members of the group. It follows that no member of the group should be trusted.

It appears to be challenging how to ensure privacy and authentication in presence of one or several corrupted MTCs. The next section analyzes the state-of-the-art group-based AKA proposals to check how they address this issue.

4 SECURITY ANALYSIS

In this section, we analyze recently proposed schemes for group-based AKA, and discuss whether they are suitable candidate for 5G, given the threat model.

4.1 Broustis et al. Schemes

Broustis et al. (Broustis et al., 2012) propose three group-based AKA schemes that aim to i) ensure the same security as in individual device authentication, ii) protect against potential attacks originated by corrupted members, and iii) reduce the signaling as compared to individual device authentication. The underlying idea is to introduce a gateway that mediates between each device and the MME, and to base group authentication on global values that are valid for all devices in the group, hence minimizing the traditional AKA signaling in favor of broadcast messages.

4.1.1 First Scheme

In the first scheme, a group key is shared between the HSS and the gateway. The gateway knows the group members and sends a group authentication request to the MME, which forwards the request to the HSS. The HSS generates an *aggregated authentication vector* that consists of the sequence of each device authentication vector with the global values G_RAND and G_AUTN, which are generated using the group key. However, the HSS generates individual responses XRES, one per device. The aggregated authentication vector and the sequence of individual responses are sent back to the MME. The MME forwards the aggregated authentication vector to the gateway. The gateway authenticates the serving network using G_AUTN and forwards the challenge G_RAND to the devices, each replying with their response RES. The gateway forwards each response to the MME, which can authenticate each device.

4.1.2 Second Scheme

In the second scheme, the group key is shared between each device and the HSS. The aggregated authentication vector consists of the global values G_RAND, G_AUTN, and also G_XRES, all of them generated using the group key. Since the gateway does not know the group key, it cannot authenticate the serving network and has to forward both G_RAND and G_AUTN to the devices. Thus, each device can authenticate the serving network and generate the same response G_RES. The gateway checks that each device has sent the same G_RES, and forwards the response to the MME, which authenticates the group.

4.1.3 Third Scheme

The last scheme is similar to the second one with the sole difference that the MME forwards to the gateway G_RAND, G_AUTN, and also G_XRES. In doing so, the gateway can authenticate the group of devices on behalf of the serving network, hence it can reduce the signaling with the MME.

Discussion. All the proposed schemes reduce the signaling because they use global values based on a shared group key. However, the size of the aggregated authentication vector increases as the size of the group increases, affecting the bandwidth requirements between HSS and MME. The introduction of a new role as the gateway in the AKA procedure is also critical because it requires several changes at the architecture level.

We argue that each of the three different schemes fails to provide an adequate level of security according to our threat model. In the first scheme, mutual authentication cannot be achieved because the authentication of the serving network for each device is done by the gateway, hence a corrupted gateway can successfully impersonate as serving network to all the group members. The second and third schemes fail to meet the individual authentication of the devices: the global G_RES cannot be uniquely associated to a member of the group. Thus, colluding corrupted group members can successfully authenticate a third member without its participation.

4.2 SE-AKA

Lai et al. (Lai et al., 2013) design a group-based AKA protocol, called SE-AKA, for LTE networks. The key idea of SE-AKA is to provide each member of the group with the same group key but with different *synchronization values.* The synchronization values behave as sequence numbers for the synchronization between each MTC and the serving network. The protocol adopts an asymmetric key cryptosystem supported by a PKI to allow the MTC to send their IMSI encrypted to the serving network, and uses Elliptic Curve Diffie-Hellman to achieve key forward and backward secrecy.

SE-AKA distinguishes two protocol procedures. One procedure is the authentication of the first group member that visits the serving network. The other procedure regards the authentication of the remaining members. The message flow occurring during the authentication of the first member is similar to EPS-AKA. A major difference is that the HSS sends to the MME the authentication vector plus a list that contains all the synchronization values of the group members. In doing so, the MME will be able to run the AKA procedure with the remaining members without involving the HSS.

Discussion. SE-AKA observes the same roles described in LTE and reduces the communication overhead between MME and HSS to only one message exchange, independently on the size of the group. However, it increases the size of the authentication data response that the HSS sends to the MME, because the message includes also the list of synchronization values. The size of the list depends on the size of the group, hence the protocol may not be suitable for very large groups. Also, low-end MTC may not be able to support ECDH and asymmetric encryption.

As a general note, we observe that a potential security issue of SE-AKA is that the MME is provided with more information than needed in group-based AKA. Since the synchronization values behave as sequence numbers, and the HSS sends to the MME the list of synchronization values of *all* group members, the MME also obtains data regarding MTC that eventually will not visit that serving network.

The authors prove mutual authentication, session master key confidentiality, and privacy of the identifier in ProVerif. The proofs do not consider an intruder able to corrupt members of the group, as we advocate in the proposed threat model. Since all the members of the group share a single group key, and the AKA procedure to authenticate the remaining group members does not require the use of devices' pre-shared keys, an intruder that corrupts two MTCs can break authentication by swapping the two synchronization values assigned to the corrupted MTC.

4.3 Choi-Choi-Lee Scheme

Choi et al. (Choi et al., 2014) propose a new group-based AKA protocol that uses symmetric cryptography only. Their solution adopts an inverted hash tree (Page, 2009), in which each node is associated to a secret value. The node value is derived from the hashed value of the node's parent. Each MTC is assigned to a leaf node value and is given a set of secret values. The set contains all the secret values of the tree, except the secret value assigned to the MTC and all the secret values of its ancestor nodes. The MME is also assigned to a leaf node value. The idea of using an inverted hash tree is to allow each pair of MTC and MME to agree on a session master key, which is based on the common node values share by the pair. The Choi-Choi-Lee protocol distinguishes the role of *leader* among one of the members of the group. The leader bootstraps the AKA procedure and mediates as gateway between the MME and the rest of the group. The message flow of the Choi-Choi-Lee protocol is similar to the second scheme proposed by Broustis et al. in section 4.1 with two main differences: i) in Choi-Choi-Lee, the HSS generates a global authentication vector based on a group key that is shared with the members of the groups, and ii) the responses RES differ from MTC to MTC such that the leader can generate a global response G_RES by applying the Chinese Remainder Theorem (CRT) on the aggregated individual responses RES.

Discussion. The global authentication vector reduces the bandwidth requirements between HSS and MME, and the presence of the leader minimizes the signaling between MTC and MME. As in Broustis et al. schemes, the introduction of a gateway may require additional changes to the mobile telephony architecture.

Although a ProVerif analysis of the protocol seems to confirm that it meets mutual authentication and confidentiality of the master session key, we find that no property can be ensured against an intruder that controls some members of the group. The authentication of MME to MTC can be broken because the MAC inside the authentication vector is generated from the group key shared between HSS and the members of the group. Any corrupted MTC that is part of the group can generate the MAC, hence they can impersonate the MME to a third group member. A second problem lies in the use of the inverted hash tree for the generation of the master session keys. Since each MTC knows all the node values but ones of its ancestors, any two corrupted MTCs with no common ancestor nodes, except the root, can calculate the master session keys agreed between MME and any other group member. It follows that the protocol does not meet confidentiality of the master session key in presence of corrupted group members. A last issue concerns a denial of service attack related to the global response G_RES. The leader can correctly generate this value only if *all* the members of the group provide the leader with their individual RES, otherwise the CRT returns an incorrect G_RES to the MME, and no member can be authenticated. Thus, it is suffice that one group member omits its response RES to inhibit the authentication of all the other members.

4.4 GBAAM

Cao et al. (Cao et al., 2015) advance a Group-Based Access Authentication protocol for MTC (GBAAM) based on pairing cryptography. The idea is to use identity-based aggregate signatures to reduce the signaling between MME and MTC without affecting bandwidth requirements. The protocol has two phases: *registration* and *group-based access authentication*. At registration, each MTC executes a classic AKA procedure at end of which MTC, MME, and HSS agree on a long-term private key. In the second phase, each MTC generates the material to create a distinct master session key and signs it with its long-term private key. A leader collects the signatures received from the members, aggregates them, and forwards the aggregate signature to the MME. The MME generates the session keys and sends back a response message signed with its public key.

Discussion. This protocol introduces the role of the leader to reduce the signaling between the group and the MME. The protocol benefits from identity-based cryptography as it removes the need of a PKI and enables the construction of short yet secure aggregated signature. The protocol is formally analyzed by model checking in TLA+/TLC against a Dolev-Yao intruder model, but only two MTCs are considered in the security analysis. Hence, the intruder cannot corrupt multiple MTCs as prescribed in our threat model. However, the major issue of the protocol is that registration and group-based access authentication must be executed with the same MME. This choice cancels the benefits provided by the group-based approach because the required signaling between MME and HSS is the same as required in traditional AKA. The devices normally require to access the network in a different geographic location than the location where they registered. Moreover, the MTCs may be in different locations when they registers to the group. This limits the suitability of the protocol as group-based AKA.

5 CONCLUSION

This paper has described the threat model that a secure group-based AKA protocol is expected to withstand. In particular, it has identified nine threats, five of which introduced by the group approach. The analysis of four recent group-based AKA protocols has reveled that either they are not immune to the prescribed threats or they fail to achieve the functional goal of group-based AKA.

5G and Internet of Things represent the last challenge about the convergence of mobile communications and computing. 5G must be designed to support the massive growth of IoT devices with fast connections and without compromising the overall security. The authentication of group of devices is one the IoT-related use case in 5G, and a secure group-based AKA protocol is expected to be the next enhancement in mobile telephony. Thus, this paper advocates the proposed threat model and analysis as basis to design the future 5G group-based AKA protocol.

ACKNOWLEDGEMENTS

This research has been performed within the 5G-ENSURE project (www.5GEnsure.eu) and received funding from the EU Framework Programme for Research and Innovation Horizon 2020 under grant agreement no. 671562. The authors are grateful to Markus Ahlström and Simon Holmberg for helping with the analysis of the Choi-Choi-Lee scheme.

REFERENCES

3GPP (2001). Formal Analysis of the 3G Authentication Protocol. TR 33.902, 3GPP.

3GPP (2008). MME Related Interfaces Based on Diameter Protocol. TS 29.272, 3GPP.

3GPP (2011). Service requirements for Machine-Type Communications (MTC); Stage 1. TR 22.368, 3GPP.

Broustis, I., Sundaram, G. S., and Viswanathan, H. (2012). Group authentication: A new paradigm for emerging applications. *Bell Labs Technical Journal*, 17(3):157–173.

Cao, J., Ma, M., and Li, H. (2015). Gbaam: group-based access authentication for mtc in lte networks. *Security and Communication Networks*, 8(17):3282–3299.

Choi, D., Choi, H.-K., and Lee, S.-Y. (2014). A group-based security protocol for machine-type communications in lte-advanced. *Wireless Networks*, 21(2):405–419.

Lai, C., Li, H., Lu, R., and Shen, X. S. (2013). Se-aka: A secure and efficient group authentication and key agreement protocol for lte networks. *Computer Networks*, 57(17).

Martucci, L. A., Carvalho, T. C. M. B., and Ruggiero, W. V. (2004). A Lightweight Distributed Group Authentication Mechanism. In Furnell, S. M. and Downland, P. S., editors, *4^{th} International Network Conference (INC 2004)*, pages 393–400.

Nguyen, L. H. and Roscoe, A. W. (2006). Efficient group authentication protocol based on human interaction. In *In Proceedings of the Workshop on Foundation of Computer Security and Automated Reasoning Protocol Security Analysis (FCS-ARSPA*, pages 9–33.

Page, T. (2009). The application of hash chains and hash structures to cryptography. Tr, Royal Holloway, University of London.

Seitz, L., Gerdes, S., Selander, G., Mani, M., and Kumar, S. (2016). Use Cases for Authentication and Authorization in Constrained Environments. RFC 7744 (Informational).

Svensson, M., Paladi, N., and Giustolisi, R. (2015). 5g: Towards secure ubiquitous connectivity beyond 2020. Tr, Swedish Institute of Computer Science.

Zhang, M. and Fang, Y. (2005). Security analysis and enhancements of 3gpp authentication and key agreement protocol. *IEEE Transactions on Wireless Communications*, 4(2):734–742.