One Step toward IoT Authentication in Mobile Systems

HungYu Chienhor
Department of Information Management, National Chi Nan University
hyhien@ncnu.edu.tw

Abstract

In the coming era of Internet-of-Things (IoT) scenarios, there will be billions of devices frequently accessing the networks. To leverage the merits of high coverage of very wide areas and the very competitive cost-to-performance ratio, many widely-deployed IoT applications would choose public mobile communication systems as their backbone. Among many challenges of deploying mobile-system-based IoT applications, two of the critical challenges regarding of authenticating the devices is tackled in this paper: the intensive communications between the visited networks and the home networks, and the aggregated authentication overhead issue. In this article, we introduce a range-bound key assignment technique to tackle the challenges. The proposed scheme drastically reduces the communication overhead and greatly strengthens the security robustness. The securities are analyzed and are verified using the AVISPA toolset.

Keywords: Authentication, Key Agreement, 5G, Internet-of-Things, formal proof.

1. Introduction

Various Internet-of-Things (IoT) applications have become more and more popular in our daily life. It has been estimated that there will be billions of devices accessing the networks in the coming future. The deployment of these devices can be roughly classified as two types: static and mobile. Examples of mobile devices include mobile phones and various Radio Frequency IDentification (RFID)/sensors embedded on mobile equipment and vehicles, and examples of statically-deployed devices include environment sensors, etc. Even though many IoT devices are statically deployed, many of them are geographical-widely deployed; therefore, the applications would also choose public mobile communication systems as their networking backbones to leverage the advantages of ubiquitous coverage and very competitive cost-to-performance ratio. The studies [4][22][23] shows that, although fixed and short range will be a significant part of IoT communications, cellular technology is forecasted to grow as the technology of choice for IoT applications owing to the increased pervasiveness of mobile broadband and stable transmissions. Inspired by the studies on

machine type communications [32][40], Chen et al. [16] further identify five advantages of adopting cellular networks for IoT applications: 1) quality of services, 2) support of devices with/without SIM cards, 3) ubiquitous coverage with relatively long distance, 4) mobility tolerance, and 5) service-level agreements (SLA) [16].

Some popular mobile communication systems include the third generation mobile system UMTS (Universal Mobile Telecommunications System) [1][2][3] and the fourth generation mobile communication system LTE (Long Term Evolution) [5][7], and the coming 5G [23][24][38][41].

The studies and improvements on 3G/4G pave the foundations and bridges toward the future 5G

Soldani and Manzalini [41] discuss several expected key performance indicators for 5G: very high throughput, low latency, wide coverage, battery lifetime, and qualities of services. Several prestigious experts from the academia and industry share their visions that future 5G and IoT will come from multiple infrastructures, not a single one; advances and improvements in conventional mobile systems pave the bridges toward the furture ideal IoT in 5G [23][24][38][41]. Salman et al. [38] point out that the high level of security ensured in the old mobile generations (GSM, UMTS and LTE) act as solid foundations for 5G and IoT, and the continuous improvements on these conventional systems are crucial for the future ideal 5G and IoT eras. Therefore, continual study and improvements on the securities of 3G, 4G and 5G are important.

Several critical challenges of authenticating devices in mobile-systems

The performance and security of authenticated key agreement scheme plays a very important factor for deploying IoT services in various mobile systems. Several studies like [15][16][20][25][27][28][30][31][32][33][40] have emphasized that, due to potential billions of devices accessing the networks, the long latency of authentication process and the large aggregated communication overhead of authentications would be big obstacles to successfully deploying mobile-system-based IoT applications. We call it the aggregated authentication overhead issue.

One another challenge comes from the heterogeneity of various IoT devices and various IoT applications. Some devices are resource-abundant and security-robust while others being quite simple and easy to be compromised. Various IoT applications own quite different characteristics. Salman et al. [38] also observe that, due to varying requirements in IoT and 5 G, applying uniform security measures is a waste of resources (processing, memory, and network bandwidth). This implies that, for those simple devices, a service provider might would like to delegate these devices less opportunities of accessing the networks directly, and

then it would examine the access records before it further delegates more access privileges. Here, we would like to point out one weakness that has been ignored by all the previous efforts. All the previous schemes deem all the devices are homogeneous and treat them equally trust-worthy. None of existing schemes address the challenge and provide any solutions. We call this the homogeneous trust and authorization issue.

Many IoT devices are resource-limited and cannot ensure tamper-proof. Therefore, the solutions should be lightweight and very efficient. We refer this as the light-weight computation challenge. The situation also calls for several desirable security properties. The Authenticated Key Agreement (AKA) schemes should provide session key forward/backward secrecy, light-weight computation, and immunity to other device compromise.

In this article, we introduce one novel authenticated key agreement scheme- the Range-Bound Authenticated Key Agreement (RB-AKA) scheme [19] that drastically reduces the communication overhead for device authentication in the convention mobile systems like #G and 4G, and it can be extended to fit the architectures of the coming 5G.

2. Related Work

Regarding the authentication and key agreement in conventional mobile systems like 3G, there are basically three kinds of entities: Mobile Stations (MS), Servicing Networks (SN), and Home Networks (HNs). MSs are registered in some HNs, and earn access grants from some SNs. The HNs delegate the authorities of authentication to SNs to perform the process of authenticating and key establishing with MSs. Basically, existent mobile-system-based AKA schemes follow the three-party-authenticated-key-agreement (3PAKA) scenarios, where both MSs and SNs trust the HNs, and they proceed the authenticated key agreement process via the help of the HNs.

UMTS. The Authentication and Key Agreement (AKA) protocols adopted in UMTS and LTE are called UMTS-AKA and LTE-AKA respectively. UMTS-AKA and LTE-AKA are symmetric-key-based protocols [1-3]. Initially, an MS and its HN share some secret keys. When an MS visits a SN, it launches a service request upon which the SN forwards the request to the MS's HN, and the HN returns some authentication vectors (AVs) for the SN to share authenticated session keys with the MS. The long latency between a SN and an HN could incur unbearable communication burden on the authentication process; therefore, the core of both UMTS-AKA [3] and LTE-AKA [1] is to reduce the number of interactions between a SN and an HN. This is achieved by delivering a list of AVs to a SN when the SN forwards a request to an HN. UMTS-AKA and LTE-AKA do reduce the number of lengthy

communications between SNs and HNs, and do work quite well for the conventional human to human communications. However, they are not suitable for handling the authentication and key agreement for the IoT scenarios; the overhead of transmitting tons of AVs will be very heavy and the number of interactions between SNs and HNs will be very intensive when there are many devices frequently accessing the networks in the IoT scenarios.

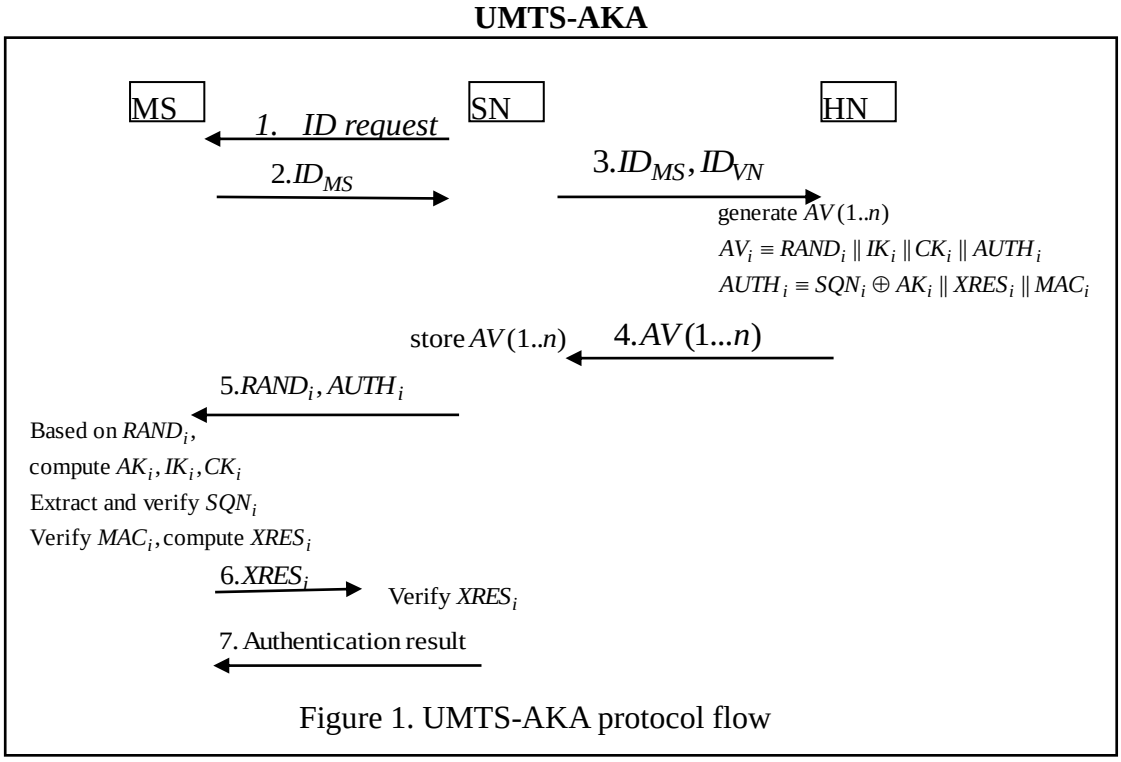

Figure 1. UMTS-AKA protocol flow

UMTS-AKA is an authentication and key Agreement protocol that helps an MS and an SN accomplish mutual authentication and establish authenticated keys via the help of the MS's HN. A simplified flow of UMTS-AKA is shown in Figure 1, and the notation used is listed in Table 1.

Table 1. The notation

MS, SN, HN	Mobile station (MS), Servicing network (SN), Home network (HN)
HSS, MME, UE	home subscriber server (HSS), mobility management entity (MME), user equipment (UE)
ID_{MS} ; AV	Identity of aMS ; Authentication vectors (AVs)
K_{MS}	A pre-shared secret key between an MS and its HN
AK, IK	Authentication key (*AK*), Integrity Key (*IK*)

R_{SN} , R_{MS} , *RAND*	Random number
AMF, *AUTH*	Authentication Management Field (AMF), Authenticator (*AUTH*)
GF, g	*GF is a Galois field, where the computational Diffie-Hellman problem is hard. g is an generator for GF.*
TZone, TSlot, AT	Time zone, Time slot, Authentication Token
$Seed_a$, $Seed_b$	Two random hash values which an RH uses to generate *AT*s and *AK*s for an AP
AT_a , AT_b	Two secret tokens which an RH generates and assigns to an AP
$AK_{MS,SN,t}$	Authentication key for the specified MS, SN and time slot *t*.
$K_{MS,SN}$	The session key between MS and SN

Initially, an MS registers at its home network (HN), and they share a secret key K_{MS}. When an MS visits an SN, the SN issues a request to the MS. The MS responds its identity ID_{MS}, and the SN forwards the ID_{MS} to the MS's HN in Step 3. Based on ID_{MS}, the HN looks up the key K_{MS} and the sequence numbers SQN_is. It then chooses random challenges $RAND_i$s, and prepares a list of AVs. Each entry in the AV list consists of $RAND_i$, $XRES_i$,CK_i, IK_i, and $AUTH_i$, where $CK_i=f3(K_{MS} \| RAND_i)$, $IK_i=f4(K_{MS} \| RAND_i)$, $AK_i=f5(K_{MS}\|RAND_i)$, $XRES_i=f2(K_{MS}\|RAND_i)$, $AUTN_i= SQN_i \oplus AK_i \|AMF\|MAC_i$, $MAC_i=f1(K_{MS}\| SQN_i \oplus AK_i \|RAND_i\|AMF)$, and $fj, 1 \le j \le 5$ are key derivation functions. AMF stands for Authentication Management Field. The AV list consists of n ($1 \le i \le n$) entries.

For each authentication instance between the SN and the MS, the SN chooses one entry from the list, and forwards ($RAND_i$, $AUTH_i$) to the MS, where $RAND_i$ acts as a challenge from the HN and $AUTH_i$ acts as the HN's authenticator. The MS verifies the validity of MAC_i, extracts SQN_i from $SQN_i \oplus AK_i$ and checks whether SQN_i is within the correct range. If all the verifications succeed, the MS accepts the SN and responds with its $RES_i = f2(K_{MS} \| RAND_i)$. The SN verifies whether $RES_i \stackrel{?}{=} XRES_i$ holds. If so, they share the keys (IK_i, CK_i).

Limitations/weaknesses of UMTS-AKA

From the above description, we note that UMTS-AKA does not fully obey the principle of the challenge-response technique since the calculation of MAC_i does not involve any challenges from the MS. The freshness of MAC_i and the message depends on the

synchronization of SQN_i; if out-of-synchronization happens due to any network problems or any malicious manipulation, it would trigger the costly re-synchronization process. We also notice that even though the number of interactions between SNs and HNs is reduced, the length of the AV list is still proportional to the length of the number of AV entries; it increases the overhead of storing secret AV lists and the transmission overhead; the overhead is amplified significantly when there are billions of devices frequently accessing the networks in the IoT scenarios [27, 33]. Additionally, the scheme only provides key distribution; once a station is compromised, then the attacker who gets the secret keys and eavesdrops on the random numbers can derive all the previous session keys. That is, it cannot provide session key forward secrecy.

LTE and LTE-AKA

Figure 2 depict the authentication network architecture in LTE networks [28]. LTE-AKA [1] basically follows the same principles and the same flows of UMTS-AKA; it, therefore, shares the same features and weaknesses.

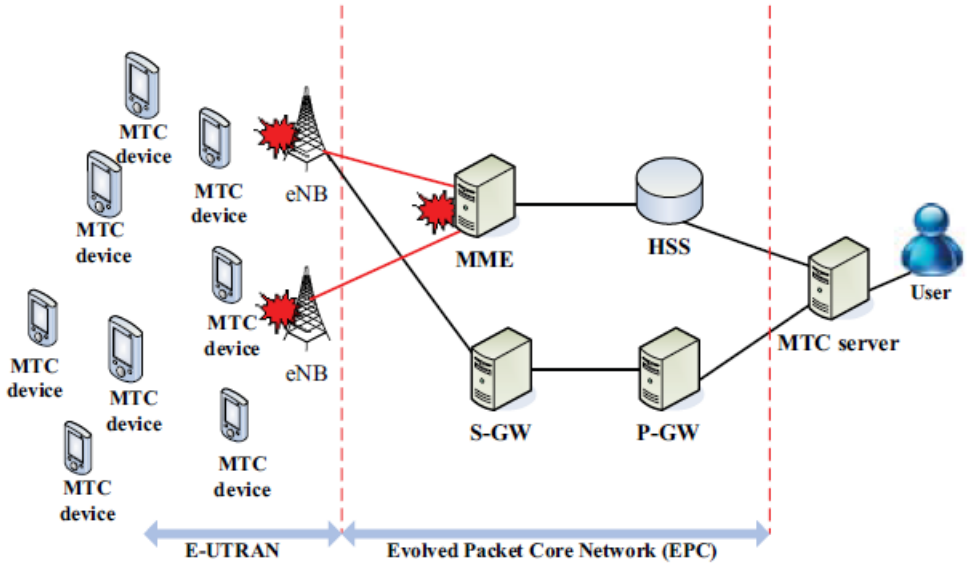

Figure 2. Network architecture in LTE networks [Figure 1 from 49]

Some improvements on 3G/4G securities and performances

There are many publications aiming at improving the performance of UMTS-AKA, LTE-AKA or other AKA-like protocols. The security robustness like sequence number synchronization issue and the costly resynchronization process are addressed in [5][20][39]. Reducing the communication overhead of the AV lists or the signaling cost is discussed in [20] [27][33]. Protecting clients' identities privacy is provided in [7][8][26][39]. Provision of key agreement and session key forward secrecy are introduced in [7][9][26][43].

Abdrabou et al. [5], based on the Simple Password Exponential Key Exchange (SPEKE) [7][35] protocol, proposed their improved scheme to tackle the costly resynchronization and the identity disclosure issue of LTE-AKA. However, the scheme still demands heavy communications and requires an impractical assumption: each MS should pre-share a secret password with each possible SN. Aminmoghadam and Mirghadri [8] proposed to deploy servers' certificates to build Tunneled TLS (TTLS) channels between clients and servers and to protect clients' identities. This approach incurs high computational load and it causes inconvenience to the clients. Sarmah et al.'s [39] proposed to synchronize a pseudonym between an MS and its HN such that the privacy of the identity of MS is ensured even it visits a new SN for the first time; however, this synchronization requirement would cause vulnerabilities to DOS (denial of service) attacks.

Some publications focus on the provision of key agreement and session key forward secrecy in the UMTS/LTE networks. Arkko et al. [9] applied public key cryptographies to achieve perfect forward secrecy. Sridevi et al. [43] apply certificates to facilitate authenticated key agreement in LTE networks. These two approaches inevitably incur high computational load on clients and the heavy communication overhead issue is not addressed. Alezabi et al. [7] applied Diffie-Hellman keys to improve the identity protection and session key forward secrecy; but each authentication needs the involvement of home network; it incurs unbearable communication overhead.

Degefa et al. [20] have done a thorough analysis on the security, communication and computational performance of LTE-AKA protocol; they applied secret-key cryptographies to improve the computation, the security and the communication; however, the scheme is based on one impractical assumption: there is one secret function $f()$ which would be kept secret even if the mobile station is compromised. Karuppiah and Saravanan [26] improved Rhee et al.'s scheme [36] to protect mobile station's identity and provide forward secrecy; the protection of the identity is achieved through an encryption using an ephemeral Diffie-Hellman key between a mobile station and its home network; however, each authentication needs the involvement of the station's home network; this worsens the communication burden.

One another group of improvements focuses on reducing the communication overhead. Kim et al. [27] analyzed the signaling cost of authentication in LTE networks, and proposed an algorithm for finding the optimal size of the authentication vectors to minimize the signaling cost on a per-user basis; this idea does not apply well in a large scalable networks like IoTs since it still incurs lots of communication overhead of transmitting AVs.

Some studies like [15][25][33] aim at providing more efficient authentication of machine type communications and IoT applications. One popular idea to achieve this goal is to group

the devices and then take proper actions to reduce the communications or the computations. We can further classify these grouping ideas into four categories. The first one is to group those devices that are deployed in dense areas and can communicate with each other using blue tooth or other short-range communications; in such scenarios, a group leader is selected and is responsible for the mobile broadband communication, and other group members can transfer and receive data through the group leader [25]; however, the authentication mechanism is not addressed in the scheme, and the application of their grouping idea is limited to those scenarios where devices are densely deployed and can communicate with each other using short-range communications.

The second one is to group these devices such that HNs will deliver all the authentication vectors (AVs) for all the members of the group to the requesting SNs in advance when any member from the group request for the services. This approach does reduce the number of interactions between SNs and HNs, but it does not reduce the transmission overhead for AV lists.

The third one like [15][31] focuses on aggregating group members' signatures and verifying the aggregated signatures to save verification cost; if the verification fails, then the divide-and-conquer technique is adopted to identify invalid any signatures. This approach requires computation-expensive operations and the issue of heavy communication overhead is not tackled.

The fourth one is to manage devices into groups and an HN only delegate simple tokens and keys to a SN such that the SN can authenticate all the members from the same group using the simple tokens [17][30]. Lai et al.'s scheme [30] uses Elliptic Curve Diffie-Hellman (ECDH) to realize key forward/backward secrecy, and it also adopts an asymmetric key cryptosystem to protect users' privacy; unfortunately, the authentication in [30] depends on the synchronization of sequence number and the shared temporary group key. It incurs the vulnerability of Denial of Services (DoS) attacks and one single compromised member endangers the security of the whole group- *we call this the single-node-compromise issue*. Chen et al.'s scheme [17] adopt similar design rationale as that of [30], and share the same weaknesses: vulnerability to DOS attacks and one single node compromise endangers the security of the whole group. *Here, we notice that how to solve the single-node-compromise issue is still a challenge for this category of schemes.*

Recently Chien [19] proposes a range-bound key agreement protocol (RB-AKA) that drastically reduces the communication overhead, and achieves the security properties like sequence number independence, key agreement, and session key forward secrecy. Here, we introduce this scheme because it drastically reduces the communication overhead, and it can be extended to fit the architectures of the coming 5G systems.

3. The RB-AKA

The Range-Bound Authentication and Key Agreement (RB-AKA) is reviewed here. The system model is described in Section 3.1, the idea is introduced in Section 3.2, and the protocol is described in Section 3.3.

3.1 The System Model

The system consists of three kinds of entities: a trusted Home Network (HN), several Serving Networks (SNs), and Mobile Stations (MSs). HN is trusted by both SNs and MSs. SN and MS respectively share secret keys with HN, and would like to establish authenticated session keys between them via the help of HN. SNs work as intermediaries between MSs and HN. The wireless communications between SNs and MSs are vulnerable to various attacks. The communications between SNs and HN would be assumed to be secure. MSs might be compromised and disclose the content. Here, we concern mutual authentication between MSs and SNs, privacy of the session keys, and forward secrecy of the session keys.

3.2 Authorization Delegation and Range-Bound Key Assignment

The system divides the time into successive time zones $TZone_1$, $TZone_2$, ..., etc. Each time zone consists of successive time slots $Tslot_i$s. The parameter z is the maximum number of time slots within a time zone. Each time zone could be one day, one week, one month, etc. The size of a time zone and that of a time slot depend on the system security and performance consideration. For example, if a time zone is one month, then a time slot could be one day and z equals 31. Another example is letting one time zone equals one day, and then one time slot could be one hour and z equals 24 in this setting.

The idea of authentication delegation is that an HN will assign a requesting SN a set of Authentication Tokens (*AT*) such that the SN can generate the corresponding authentication keys for each time slot within the authorized time zone. *Each authentication key exclusively corresponds to the specified MS, the SN, the time zone and the time slot.*

We take one example to illustrate the idea. Let one time zone equals one day, one time slot equals one hour, and z= 24. Assume an SN requests for authentication tokens (*AT*) on May 30 2016 (30/05/2016). Let $Seed_a = h(K_{MS} \| ID_{MS} \| ID_{SN} \| "05/30/2016" \| a)$ and $Seed_b = h(b \| K_{MS} \| ID_{MS} \| ID_{SN} \| "05/30/2016")$, where a and b could be pre-defined constants or a Diffie-Hellman value (we will illustrate the details later). Then an authentication key

$AK_{MS,SN,t}$ for the time slot t is defined as $AK_{MS,SN,t} = h(h^{t-1}(seed_a) \| h^{24-t}(seed_b))$, where $1 \le t \le 24$.

If the HN would like to authorize the SN the authentication authority from 8:00 am to 11:00 am (time slot 9 ~12). Then the HN computes $AT_a = h^8(Seed_a)$ and $AT_b = h^{12}(Seed_b)$, and securely sends (AT_a, AT_b) to the SN. Based on (AT_a, AT_b), the SN can compute any authentication keys $AK_{MS,SN,t} = h(h^{t-1}(seed_a) \| h^{24-t}(seed_b)) = h(h^{t-9}(AT_a) \| h^{12-t}(AT_b))$, where $9 \le t \le 12$. The example is depicted in Figure 3.

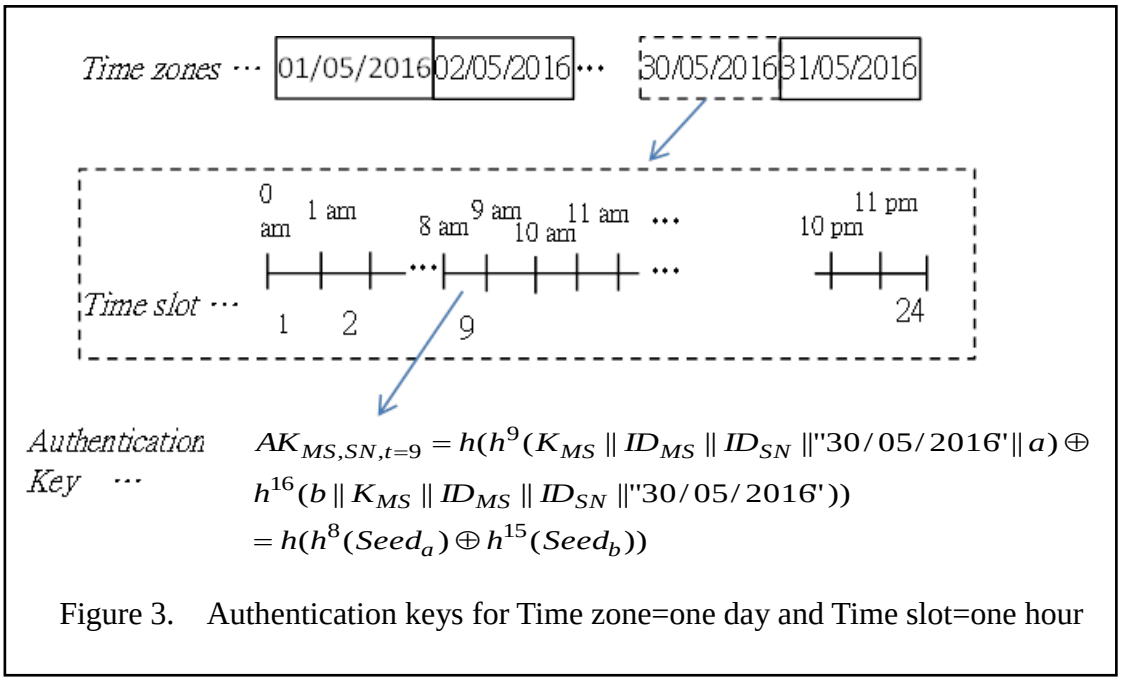

Figure 3. Authentication keys for Time zone=one day and Time slot=one hour

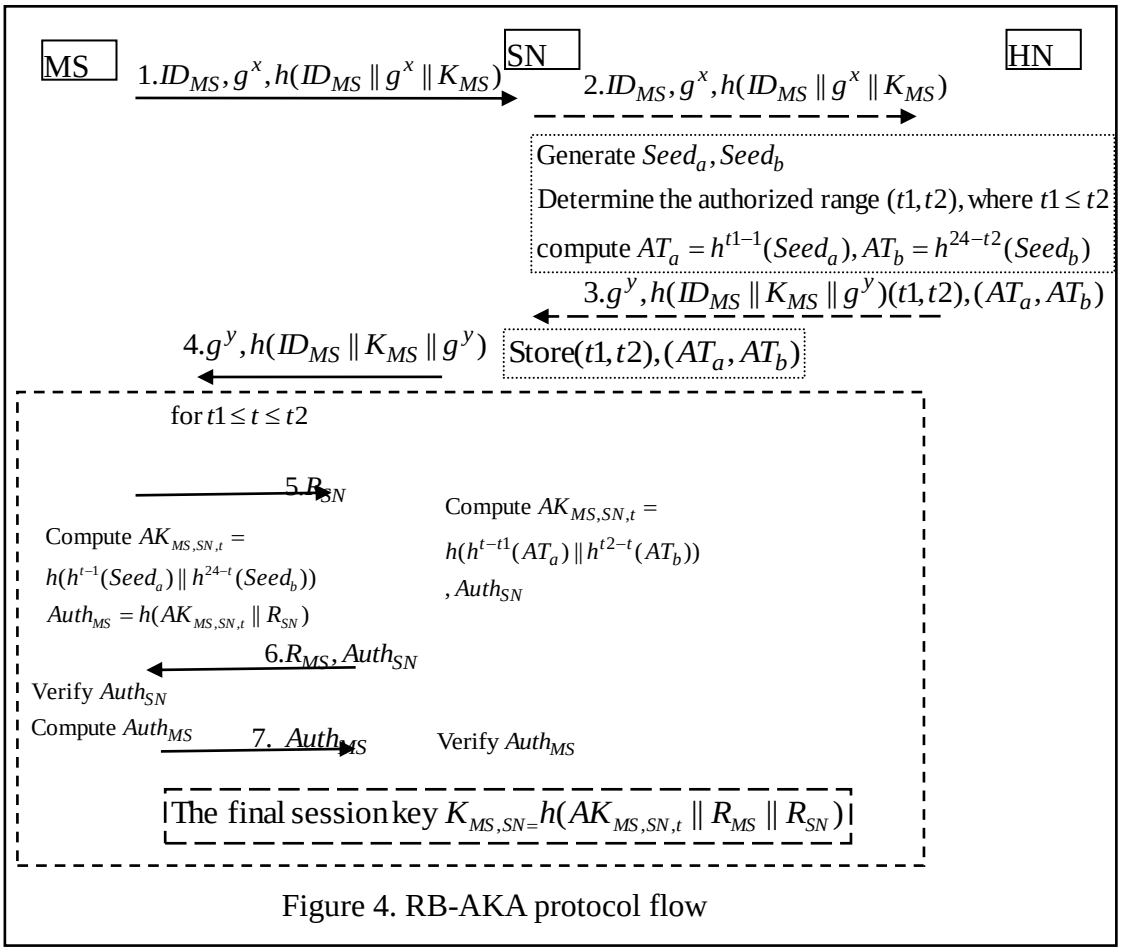

Figure 4. RB-AKA protocol flow

3.3 The proposed scheme

In this section, we use the above example of setting to illustrate the protocol; that is, one time zone equals one day, one time slot equals one hour, and *z*= 24. It is easy to extend it to other settings of different time zones and time slots.

The scheme consists of two phases: the initialization phase and the authentication phase.

The Initialization phase

Initially, the HN assigns a secret key K_{MS} and an identity ID_{MS} for each registered MS. In addition to the system parameters {*time zones, time slots, z*}, the HN also publishes a cryptographic hash function *h*. The HN writes both the secret key and the public parameters into the MS.

The Authentication phase

When an MS visits an SN and requests for services, it launches the following process. In the following, "→" denotes an insecure channel and "➔" denotes a secure channel.

Step 1. MS → SN: ID_{MS}, g^x, $h(ID_{MS} \| g^x \| K_{MS})$

The MS selects a random integer x and sends the above values to the SN.

Step 2. SN ➔ HN: ID_{MS}, g^x, $h(ID_{MS} \| g^x \| K_{MS})$

The SN forwards the MS's request with its own parameters to the HN.

Step 3. HN ➔ SN: g^y, $h(ID_{MS} \| K_{MS} \| g^y)$, $(t1, t2)$,(AT_a, AT_b)

Based on the request, the HN first looks up the key K_{MS} and checks the integrity of the parameters. If the verification succeeds, it chooses a random integer y, determines the authorized range $(t1, t2)$, and calculates $AT_a = h^{t1}(K_{MS} \| ID_{MS} \| ID_{SN} \| date \| g^{xy})$ and $AT_b = h^{z-t2+1}(g^{xy} \| K_{MS} \| ID_{MS} \| ID_{SN} \| date)$. The HN returns the values g^y, $h(ID_{MS} \| K_{MS} \| g^y)$, $(t1, t2)$ and (AT_a, AT_b). The SN forwards $\{ g^y, h(ID_{MS} \| K_{MS} \| g^y) \}$ in the next step and stores the values $(t1, t2)$ and (AT_a, AT_b).

Step 4. SN → MS: g^y, $h(ID_{MS} \| K_{MS} \| g^y)$

MS verifies the integrity of the received values. If the verification succeeds, then it calculates $Seed_a = h(K_{MS} \| ID_{MS} \| ID_{SN} \| date \| g^{xy})$, $Seed_b = h(g^{xy} \| K_{MS} \| ID_{MS} \| ID_{SN} \| date)$.

For any t that satisfies $t1 \le t \le t2$, the MS and the SN perform the following steps.

Step 5. MS → SN : R_{MS}

The MS chooses a random challenge R_{MS}, and sends it as a challenge to the SN. Based on the time slot t, the MS computes $AK_{MS,SN,t} = h(h^{t-1}(seed_a) \| h^{24-t}(seed_b))$.

Step 6. SN → MS: R_{SN}, $Auth_{SN}$

The SN first computes $AK_{MS,SN,t} = h(h^{t-t1}(AT_a) \oplus h^{t2-t}(AT_b))$, and computes $Auth_{SN} = h(AK_{MS,SN.t} \| R_{MS})$. It then chooses a random number R_{SN}, and sends the above values back to the MS.

Step 7. MS → SN: $Auth_{MS}$

The MS verifies the received $Auth_{SN}$ and computes $Auth_{MS} = h(AK_{MS,SN.t} \| R_{SN})$. It computes the session key $K_{MS,SN} = h(AK_{MS,SN,t} \| R_{MS} \| R_{SN})$, and sends $Auth_{MS}$ back to the SN.

Step 8. SN :

The SN verifies the received $Auth_{MS}$. If the verification succeeds, then it accepts the MS, and computes the session key $K_{MS,SN} = h(AK_{MS,SN,t} \| R_{MS} \| R_{SN})$.

4. Security Properties Verification, Analysis and Comparison

The protocols have been verified using AVISPA, and have been analyzed. Here, we review some key results in Section 4.1. Section 4.2 give a comparison of the security properties of

related publications. Section 4.3 reviews the communication performance. Interested readers are referred to the publication for the details of verification and the security analysis.

4.1 Protocol Verification Using AVISPA

In the HLPSL specification, we model two roles: “mobile” and “server”. “mobile” models the entity MS and “server” models an integrated entity of SN and HN. This integration is because the communications between HN and SN are assumed to be secure in our model but, in the current version of the AVISPA tool, only Dolev-Yao channels are supported. In Dolev-Yao channels, the intruder, in addition to having all the capabilities of an honest agent, may divert sent messages and send new ones impersonating other agents. The integration of the two entities simplifies the specification without losing the security semantics of our protocol.

We have two specifications. One specification is modelling the mutual authentication and the privacy of the session keys $K_{MS,SN}$ and the authentication keys $AK_{MS,SN,t}$. The authentication is modeled using the predicates “request” and “authentication_on” on the responses $Auth_{MS}$ and $Auth_{SN}$. The modeling of the privacy uses the predicate “secrecy_of”. The message sequence chart from the SPAN animator is depicted in Figure 5. The OFMC verifier confirms the security of the two goals (authentication and privacy) in Figure 6.

The second HLPSL specification of our protocol focuses on modeling the session key forward secrecy. Here, we let the role “mobile” and the role “server” run two instances of authentications to have one set of keys (denoted as ak1, k1, seeda1 in our specification) in the first instance and the other set of keys (denoted as ak2, k2, seeda2) in the second instance. Then, we facilitate the intruder/attacker own the knowledge of the second set of keys to model the compromise of the mobile station MS. The goals include the authentication and the privacy of the first set of keys. The message sequence chart is depicted in Figure 7 and the ATSE verifier reports “safe” in Figure 8.

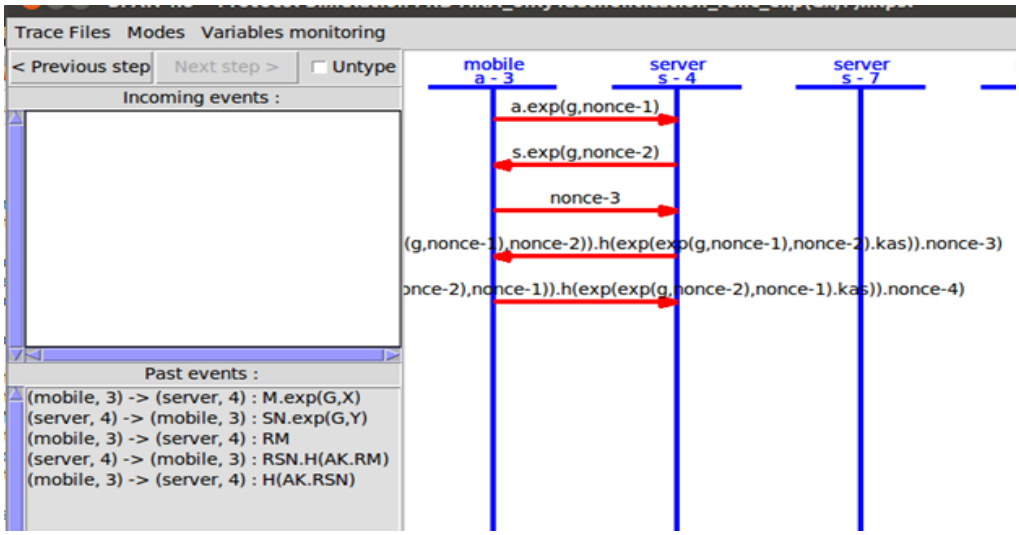

Figure 5. The message sequence chart of the HLPSL specification 1 of our protocol.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/RB-AKA_only1authentication_func_exp(Gx,Y).if
GOAL
  as_specified
BACKEND
  OFMC
```

Figure 6.The OFMC reports on the HLPSL specification 1 of our protocol

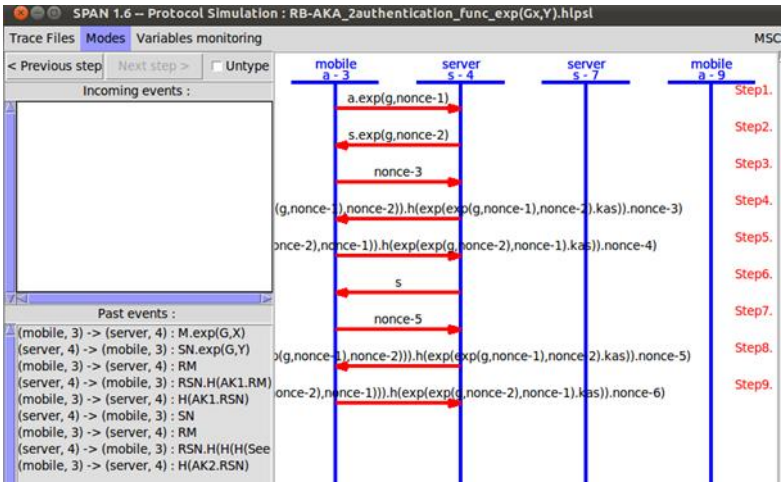

Figure 7. The message sequence chart of the HLPSL specification 2 of our

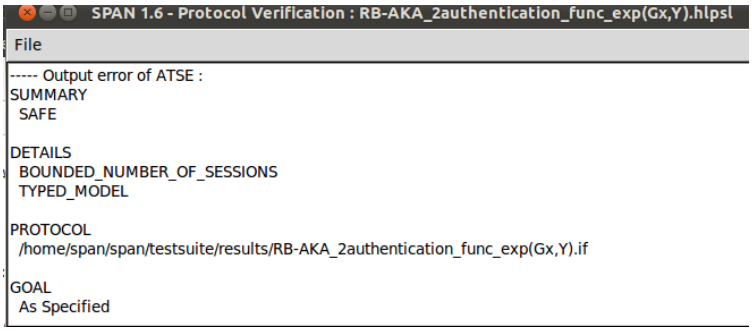

Figure 8. The OFMC reports on the HLPSL specification 2 of our protocol

4.2 Comparison of Security properties of Related Schemes

Table 2 summarizes the security performance of related schemes. Among them, the RB-AKA is the only one that provides both session key forward secrecy and time-bound authorization. UMTS-AKA and LTE-AKA only provide key distribution and they cannot provide session key forward secrecy since an attacker who gets a compromised station's secret key can derive all the session keys, using the secret key and the eavesdropped random numbers. The authentication in UMTS-AKA/LTE-AKA depends on the synchronization of sequence numbers; it, therefore, require the costly re-synchronization process to cope possible out-of-sequence issue. Based on the comparison, the proposed scheme is much robust than UMTS-AKA/LTE-AKA.

Even though Degefa et al. [20] claimed that their scheme provides forward secrecy, we find the claim does not hold. Their claims are based on the arguments that the session key computations involve the secret value *S*, and the secret *S* is derived using the secret key and

the secret function *f*. The assumption that the one-way hash function *f*() could not be compromised even if the station is compromised is too strong to be practical; this assumption violates the general notation of station compromise.

The forward secrecy of Karuppiah- Saravanan's scheme [26] depends on the assumption that the secret key *K* is still secure, even if the station is compromised; Their argument for this is that the secret key is encrypted using the key derived from user's password and some secret parameters inside the station. We find this argument should be further carefully examined because, once a station is compromised, the parameters inside could be used to verify the user's password and then the verified password could be used to decrypt the encrypted key.

Table 2. The security performance comparison

	RB-AKA	LTE-AKA	[7]	[20]	[26]	[9]	[8]	[5][1]
Identity privacy	No	No	Yes	Yes	Yes	No	Yes	Yes
Key distribution (dist.) vs key agreement (agr.)	Agr.	Dist.	Agr.	Dist.	Dist.	Agr.	Dist.	Agr.
Challenge-response security principle	Yes	No[2]	Yes	Yes	No[2]	No[2]	No[2]	Yes
Session key forward secrecy	Yes	No	Yes	No	Partial	Yes	Yes	Yes
Time-bound authorization	Yes	No	No	No	No	No	No	No
Requirement of SQN re-synchronization	No	Yes	No	Yes	No	Yes	No	No

1. Each MS should pre-share a secret key with each possible visited network. This assumption is not practical.
2. The authenticators from SN/HN do not depend on any challenges from MS; therefore, UMTS-AKA and LTE-AKA only partially follow the challenge-response principle. The schemes [8][9][26] share the same property.

4.3 Accumulated Communication Overhead between SNs and HNs

Here, we focus on the accumulated overhead between SNs and HNs for one time-zone because of the following reasons. First, the communication delay between MSs and SNs is much shorter than that between SNs and HNs. Second, the interactions between MSs and SNs of all the schemes (RBK-AKA, UMTS-AKA/LTE-AKA, and others) are quite simple and

similar, and the differences are insignificant. Third, according to previous studies like [20, 27], it is the overhead between SNs and HNs that would deteriorate the overall performance when there are intensive connection requests. In the following comparison, we let *m* denotes the number of services per MS within one time-zone. Assume an MS averagely accesses the services 40 times per day. Then *m* equals 40 when one time-zone is one day, and it equals 1200 when one time-zone is one month. Let *L* denotes the length of each value (we can simplify the comparison by taking all values like hashing, encryptions, etc, having the same length, without losing the correctness of the comparison).

We first examine that of our RB-AKA. For an authorized time zone, an HN only sends six values $\{g^y,\ h(ID_{MS} \| K_{MS} \| g^y),\ (t1, t2),(AT_a,\ AT_b)\}$ in RB-AKA. On the contrary, the HN in UMTS-AKA/LTE-AKA needs to send the *AV* list $AV(1 \ldots n)$ per SN's request. Each AV entry consists of 6 values and n=5 is usually recommended in the standards. So the accumulated overhead is 6*mL*. Aminmoghadam-Mirghadri's scheme [8] basically transmits similar AV lists like UMTS-AKA/LTE-AKA does; therefore, it demands 6*mL*.

In Alezabi et al.'s scheme, each authentication demands *6L* between an SN and an HN, and each authentication needs the interaction between the SN and the HN; therefore, the total communication overhead between an SN and an HN is 6*mL*. In Degefa et al.'s scheme [20], each authentication requires the home network to deliver two values to the visited network; so the total communication overhead between an SN and an HN is 2*mL*. In Karuppiah-Saravanan's scheme [26], each authentication requires the home network to deliver twelve values to the visited network; so the total communication overhead between an SN and an HN is 12*mL*. In Abdrabou et al. scheme [5], each authentication requires the home network to deliver ten values to the visited network; therefore, it needs 10*mL*. The AV list in Arkko et al.' scheme [9] is basically similar to UMTS-AKA, except that an extra parameter is added for SNs to compute the corresponding Diffie-Hellman key; therefore, the overhead is 7*mL*. Table 3 summarizes the overhead.

Table 3. The accumulated communication overhead between SNs and HNs for one time-zone.

	RB-AKA	LTE-AKA	[7]	[20]	[26]	[5]	[8]	[9]
Length of AT/AV for time zone	*6L*	*6mL*	*6mL*	*2mL*	*12mL*	*10mL*	*6mL*	*7mL*

- *L* denotes the length of each value; *m* denotes the number of services per MS within one time-zone.

We now take two practical settings to get simple insights of the costs in Table 4. Assume an MS averagely access the services 40 times per day. The first setting is let one time-zone in our

scheme be one day. In this setting, and the length of authentication tokens sent by HN for one time-zone is 6*L* in our scheme, and that for UMTS-AKA/LTE-AKA is 40*6=240*L*.

The second possible setting is to let one time-zone in our scheme be one month. In this setting, the overhead of RB-AKA is still 6*L*. The overhead of UMTS-AKA/LTE-AKA is 240*30 *L* (we take one month has 30 days). The overhead of other schemes for this setting are summarized in the third column in Table 5. From Table 5, the communication overhead of our scheme is only 1/40 that of UMTS-AKA/LTE-AKA for time zone=1 day, and it is only 1/1200 that of UMTS-AKA/LTE-AKA for time zone=1 month.

To get an insight of the improvement, we show in Figure 9 the accumulated overhead between one SN and one HN within one time-zone for varying *m* values. The values in the Y-axis represent the number of overhead in unit of "*L*". The values on the X-axis represents the *m* (the number of requests per MS within one time-zone). The value *m* ranges from 40 to 43200; The case of *m*=40 might represents the scenario where one time-zone is one day and an MS averagely access the network 40 times per day. The case of *m*=43200 might represents the scenarios of IoT applications where one time-zone is one month and one device accesses the network per minute. From Figure 9, we can see that the reduction of the overhead between an SN and an HN is very huge.

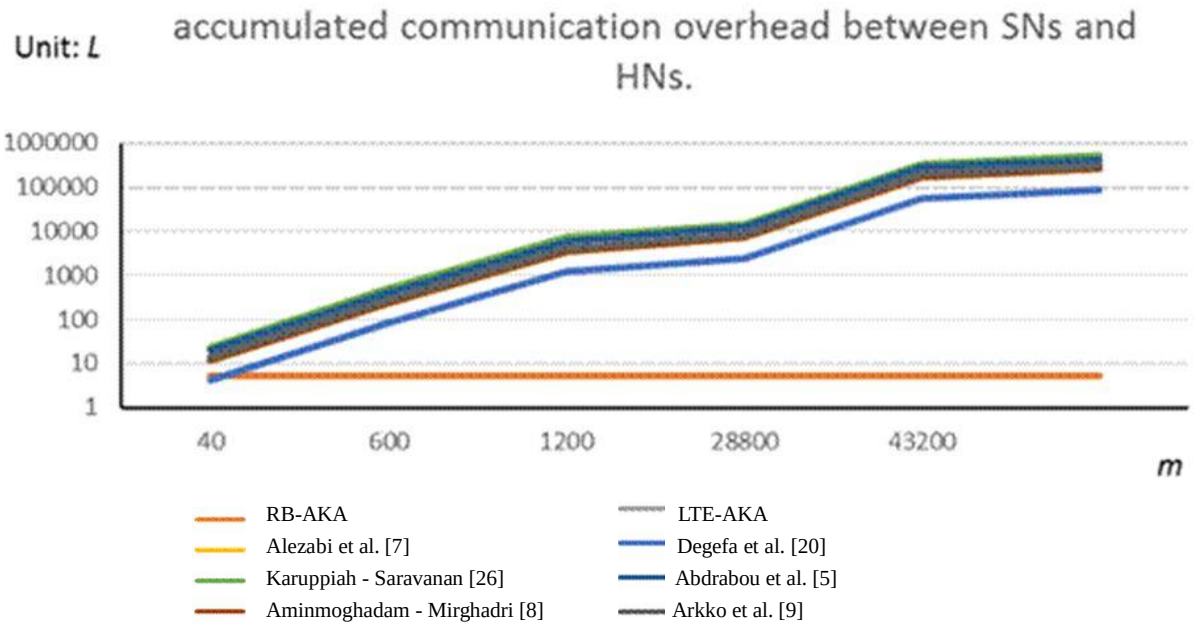

Figure 9. The accumulated overhead between one SN and one HN for one time-zone.

5. Conclusions

In this article, we have reviewed the range-bound key assignment and RBK-AKA key agreement scheme. The securities have been verified using AVISPA. For detailed security analysis and performance, readers are referred to the publication. Compared to UMTS-AKA/LTE-AKA, the RB-AKA scheme greatly strengthens the security robustness by ensuring key agreement, session key forward secrecy between time slots, and independence of sequence number synchronization. Most importantly, it drastically reduces the communication overhead. All these excellent features show that the RB-AKA is much attractive, especially when there are billions of devices accessing the networks in the coming IoT eras. The author is extended the scheme to fit the requirements and architectures of possible 5G systems.

References

[1] 3GPP, "3GPP TS 33.401 3GPP System Architecture Evolution (SAE); Security architecture," ed, 2015.

[2] 3GPP, "3GPP TS 23.002 Network architecture," ed, 2015.

[3] 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; "Formal Analysis of the 3G Authentication Protocol," 3GPP TR 33.902 version 4.0.0, Sep. 2001.

[4] 4G Americas, "Cellular Technologies Enabling the Internet of Things," White paper, Nov. 2015.

[5] M. A. Abdrabou, A. D. E. Elbayoumy and E. A. EI-Wanis, "LTE Authentication Protocol (EPS-AKA) Weaknesses Solution," *2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS)*, IEEE, 2015.

[6] A. M. Alberti, G. D. Scarpioni, V. J. Magalhaes, S. A. Cerqueira, J. J. P. C. Rodrigues and R. d. R. Righi, "Advancing NovaGenesis Architecture Towards Future Internet of Things," *IEEE Internet of Things Journal*, vol. PP, Issue 99, pp. 1-1, Jul. 2017.

[7] K. A. Alezabi, F. Hashim, S. 1. Hashim and B. M. Ali, "An efficient authentication and key agreement protocol for 4G (LTE) networks," *2014 IEEE REGION 10 SYMPOSIUM*, IEEE, 2014.

[8] E. Aminmoghadam and A. Mirghadri, "A Forward Secure PKI-based UMTS-AKA with Tunneling Authentication," *2015 Third International Conference on Digital Information, Networking, and Wireless Communications (DINWC)*, IEEE, 2015.

[9] J. Arkko, K. Norrman, M. Näslund and B. Sahlin, "A USIM compatible 5G AKA protocol with perfect forward secrecy," *2015 IEEE Trustcom/BigDataSE/ISPA*, IEEE, p. 1205-1209, 2015.

[10] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuellar, P. Hankes Drielsma, P.-C. H´eam, O. Kouchnarenko, J. Mantovani, S. M¨odersheim, D. von Oheimb, M. Rusinowitch, J. Santos Santiago, M. Turuani, L. Vigan`o and L. Vigneron, "The AVISPA Tool for the automated validation of internet security protocols and applications," *International Conference on Computer Aided Verification*, Springer, Berlin, Heidelberg, pp. 281-285, 2005.

[11] G. Ateniese, A. D. Santis, A. L. Ferrara and B. Masucci, "Provably-secure time-bound hierarchical key assignment schemes," *Journal of cryptology*, 25.2: 243-270, 2012.

[12] Avispa – a tool for Automated Validation of Internet Security Protocols. http://www.avispa-project.org.

[13] D. Basin, S. M¨odersheim and L. Vigan`o , "OFMC: A Symbolic Model-Checker for Security Protocols," *International Journal of Information Security*, 2004.

[14] J.-L. Beuchat, J. E. González-Díaz, S. G. Mitsunari, E. J. Okamoto, F. Rodríguez-Henríquez and T. Teruya, "High-Speed Software Implementation of the Optimal Ate Pairing over Barreto-Naehrig Curves," *International Conference on Pairing-Based Cryptography- Pairing 2000*, pp.21-39.

[15] J. Cao, M. Ma and H. Li, "A group-based authentication and key agreement for MTC in LTE networks," *2012 IEEE Global Communications Conference (GLOBECOM)*, IEEE, pp. 1017-1022, 2012.

[16] S. Chen, R. Ma, H.-H. Chen, H. Zhang, W. Meng and J. Liu, "Machine-to-Machine Communications in Ultra-Dense Networks – A Survey," IEEE COMMUNICATIONS SURVEYS & TUTORIALS", *IEEE Communications Surveys & Tutorials*, 2017.

[17] Y.-W. Chen, J.-T. Wang, K.-H. Chi and C.-C. Tseng, "Group-based authentication and key agreement," *Wireless Personal Communications*, 62.4: 965-979, 2012.

[18] H.Y. Chien, "Efficient Time-Bound Hierarchical Key Assignment Scheme," *IEEE Transactions on Knowledge and Data Engineering*16.10: 1301-1304, 2004.

[19] H.-Y. Chien, "An Effective Approach to Solving Large Communication Overhead Issue and Strengthening the Securities of AKA Protocols," *International Journal of Communication Systems*, 2017.

[20] F. B. Degefa, D. H. Lee, J. Kim, Y. S. Choi and D. H. Won, "Performance and security enhanced authentication and key agreement protocol for SAE/LTE network," Computer Networks, Volume 94, pp. 145-163, Jan. 2016.

[21] K. Fan, Y. Gong, Z. Du, H. Li and Y. Yang, "RFID Secure Application Revocation for IoT in 5G," *2015 IEEE Trustcom/BigDataSE/ISPA*, Vol. 1, IEEE, Aug. 2015.

[22] Global mobile Suppliers Association (GSMA): "LTE-Advanced, LTEAdvanced Pro global status – commitments, launches, devices ecosystem," GSA Evolution to LTE Report, online: http://gsacom.com/paper/lte-advanced-lte-advanced-pro-global-statuscommitments-launches-devices-ecosystem/, Jul. 2016.

[23] A. Haidine and S. E. Hassani, "LTE-a pro (4.5G) as pre-phase for 5G deployment: Closing the gap between technical requirements and network performance," *2016 International Conference on Advanced Communication Systems and Information Security (ACOSIS)*, IEEE, Oct. 2016.

[24] C.-L. I, M. A. Uusitalo and K. Moessner, "The 5G Huddle: from the guest editors," *IEEE Vehicular Technology Magazine*10.1: 28-31, 2015.

[25] K.-R. Jung, A. Park and S. Lee, "Machine-type-communication (MTC) device grouping algorithm for congestion avoidance of MTC oriented LTE network," *Security-Enriched Urban Computing and Smart Grid*, pp. 167-178, 2010.

[26] M. Karuppiah and R Saravanan, "A secure authentication scheme with user anonymity for roaming service in global mobility networks," *Wireless Personal Communications* 84.3: 2055-2078, 2015.

[27] S. Kim, J.-Y. Choi and J. Jeong, "On Authentication Signaling Costs in Hierarchical LTE Networks," *2014 7th International Conference on Ubi-Media Computing and Workshops*, IEEE, pp. 11-16, 2014.

[28] C. Lai, H. Li, R. Lu, R. Jiang and X. Shen, "LGTH: A Lightweight Group Authentication Protocol for Machine-Type Communication in LTE Networks," *2013 IEEE Global Communications Conference (GLOBECOM)*, IEEE, pp. 832-837, Dec. 2013.

[29] 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," *Computer Networks* 57.17: 3492-3510, 2013.

[30] C. Lai, H. Li, R. Lu, R. Jiang and X. Shen, "SEGR: A Secure and Efficient Group Roaming Scheme for Machine to Machine Communications between 3GPP and WiMAX Networks," *2014 IEEE International Conference on Communications (ICC)*, IEEE, pp. 1011-1016, Jun. 2014.

[31] C. Lai, R. Lu, D. Zheng, H. Li and X. Shen, "Toward Secure Large-scale Machine-to-machine Communications in 3GPP Networks: Challenges and Solutions," *IEEE Communications Magazine* 53.12: 12-19, Dec. 2015.

[32] M. Laner, P. Svoboda, N. Nikaein and M. Rupp, "Traffic Models for Machine Type Communications," *ISWCS 2013; The Tenth International Symposium on Wireless Communication Systems*, VDE, pp. 1-5, Aug. 2013.

[33] J. Li, M. Wen and T. Zhang, "Group-Based Authentication and Key Agreement With Dynamic Policy Updating for MTC in LTE-A Networks," *IEEE Internet of Things Journal* 3.3: 408-417, 2016.

[34] D. von Oheimb, "Specification language hlpsl developed in the eu project Avispa," In APPSEM, 2005.

[35] A. K. Rai, V. Kumar and S. Mishra, "An efficient password authenticated key exchange protocol for WLAN and WiMAX," *Proceedings of the International Conference & Workshop on Emerging Trends in Technology*, ACM, pp. 881-885, Feb. 2011.

[36] H. S. Rhee, "Improved user authentication scheme with user anonymity for wireless communications," *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, *94*(2), 860-864, Feb. 2011.

[37] F. B. Saghezchi, G. Mantas, J. Ribeiro, M. Al-Rawi, S. Mumtaz, J. Rodriguez, "Towards a Secure Network Architecture for Smart Grids in 5G Era," *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, IEEE, Jun. 2017.

[38] O. Salman, A. Kayssi, A. Chehab and I. Elhajj, "Multi-Level Security for the 5G/IoT Ubiquitous Network," *2017 Second International Conference on Fog and Mobile Edge Computing (FMEC)*, IEEE, pp. 188-193, 2017.

[39] S. Sarmah, P. Kalita and J. Devi, "A New Approach to Authentication and Key Agreement in LTE 3GPP," *Int J Comput Sci Res Tech (IJCSRT)* 1.4: 116-120, 2013.

[40] M. Shafiq, L. Ji, A. Liu, J. Pang and J. Wang, "Large-Scale Measurement and Characterization of Cellular Machine-to-Machine Traffic," *IEEE/ACM Transactions on Networking*, vol. 21, no. 6, pp. 1960-1973, Dec. 2013.

[41] D. Soldani and A. Manzalini, "Horizon 2020 and Beyond: On the 5G Operating System for a True Digital Society," *IEEE Vehicular Technology Magazine*, Vol. 10, Issue 1, pp. 32-42, Mar. 2015.

[42] SPAN: a Security Protocol ANimator for AVISPA, http://people.irisa.fr/ Thomas.Genet/span/.

[43] B. Sridevi1, D. Mohan and R. Neelaveni, "Secured Handover Key Management Among LTE Entities Using Device Certification," *2014 3rd International Conference on Eco-friendly Computing and Communication Systems*, IEEE, pp. 155-160, Dec. 2014.

[44] T. H. Szymanski, "Securing the Industrial-Tactile Internet of Things With Deterministic Silicon Photonics Switches," *IEEE Access*, vol. 4, pp. 8236-8249, 2016.

[45] M. Turuani, "The CL-Atse Protocol Analyser," *International Conference on Rewriting Techniques and Applications*, Springer, Berlin, Heidelberg, pp. 277-286, Aug. 2006.

[46] W. G. Tzeng, "A Time-Bound Cryptographic Key Assignment Scheme for Access Control in a Hierarchy," *IEEE Transactions on Knowledge and Data Engineering*, 14.1: 182-188, 2002.

[47] Q. Wang, D. Chen, N. Zhang, Z. Qin and Z. Qin, "LACS: A Lightweight Label-Based Access Control Scheme in IoT-Based 5G Caching Context," *IEEE Access*, Vol. 5, pp. 4018-4027, 2017.

[48] D. Zhang, Z. Zhou, S. Mumtaz, J. Rodriguez and T. Sato, "One Integrated Energy Efficiency Proposal for 5G IoT Communications," *IEEE Internet of Things Journal*, Volume: 3, Issue: 6 ,pp.1346-1354, Dec. 2016.