An Extended Authentication and Key Agreement Protocol of UMTS

Farshid Farhat, Somayeh Salimi, and Ahmad Salahi

Iran Telecommunication Research Center, North Karegar, Tehran, Iran
{farhat,ssalimi,salahi}@itrc.ac.ir

Abstract. Identification, authentication and key agreement protocol of UMTS networks have some weaknesses to provide DoS-attack resistance, mutual freshness, and efficient bandwidth consumption. In this article we consider UMTS AKA and some other proposed schemes. Then we explain the known weaknesses in the previous frameworks suggested for UMTS AKA protocol. After that we propose a new UMTS AKA protocol (called EAKAP) for UMTS mobile network that combines identification stage and AKA stage of UMTS AKA protocol as well as eliminating disadvantages of related works and bringing some new features to improve the UMTS AKA mechanism such as reducing the interactive rounds of the UMTS AKA protocol.

Keywords: Identification, Authentication and Key Agreement, UMTS, Mobile Network, Security Protocol.

1 Introduction

The wireless communications advances cause the ease of access to wireless services for individuals. Because of the air interface between the user and the network, the physical security of users' media is in serious danger with respect to the wired infrastructure. Physical layer security like spread spectrum methods for commercial usages is so expensive, hence wireless providers try to secure higher layers to obtain privacy and confidentiality features for their subscribers.

The most commonly used wireless communications are cellular communications. In the first-generation (1G) analog systems, security services were not addressed. Proper authentication of subscribers is an important feature for operators to charge them correctly. So in the second-generation (2G) digital cellular systems, the GSM communications, authentication and also confidentiality were taken into account and security measures were designed for these goals in 2G.

Some of the GSM security weaknesses are active attack by fake BTS, no secure communication between BTS and BSC and also between BSC and MSC, no data integrity check and weak stream cipher algorithm (A5/1,2) for confidentiality. These weaknesses were concentrated in the security design phase of UMTS, a major standard for the third-generation (3G). So, an enhanced authentication and key agreement protocol was considered for UMTS and integrity was added as well as using strong algorithms.

F. Bao, H. Li, and G. Wang (Eds.): ISPEC 2009, LNCS 5451, pp. 230–244, 2009.

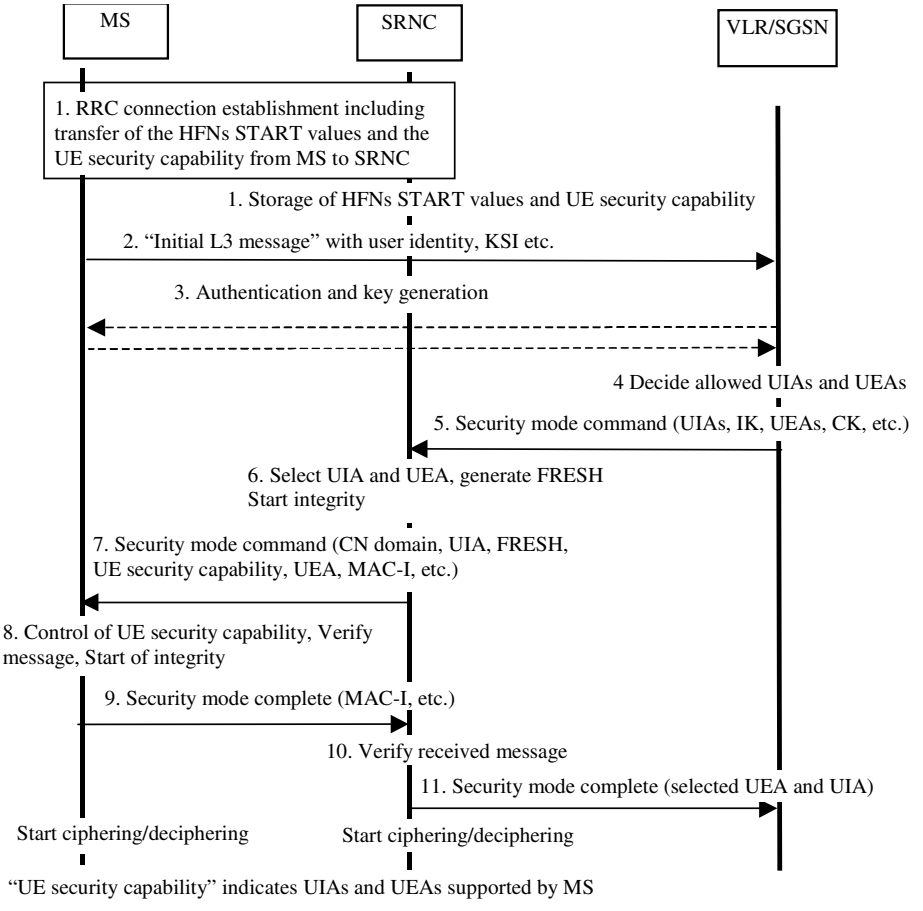

Fig. 1. The UMTS Identification and AKA procedure [1]

Between features mentioned above, the authentication and key agreement protocol has vital importance which the most prominent security features are based on. In this protocol, other than authentication, user and network agree on the cipher and integrity keys CK and IK respectively. If existence of any vulnerability in this protocol, other than these keys, the subscriber secret key K may be compromised and so, we focus on this protocol for the purpose of improving it.

This paper is organized as follows; in section II we explain the UMTS authentication and key agreement (AKA) procedure. Section III outlines related challenges to improve the security and performance of the UMTS AKA protocol. In section IV the new proposed protocol is described that covers the previous known weaknesses. In section V the EAKAP is analyzed and evaluated from the security point of view. We conclude the paper in section VI. The descriptions of abbreviations are given in appendix.

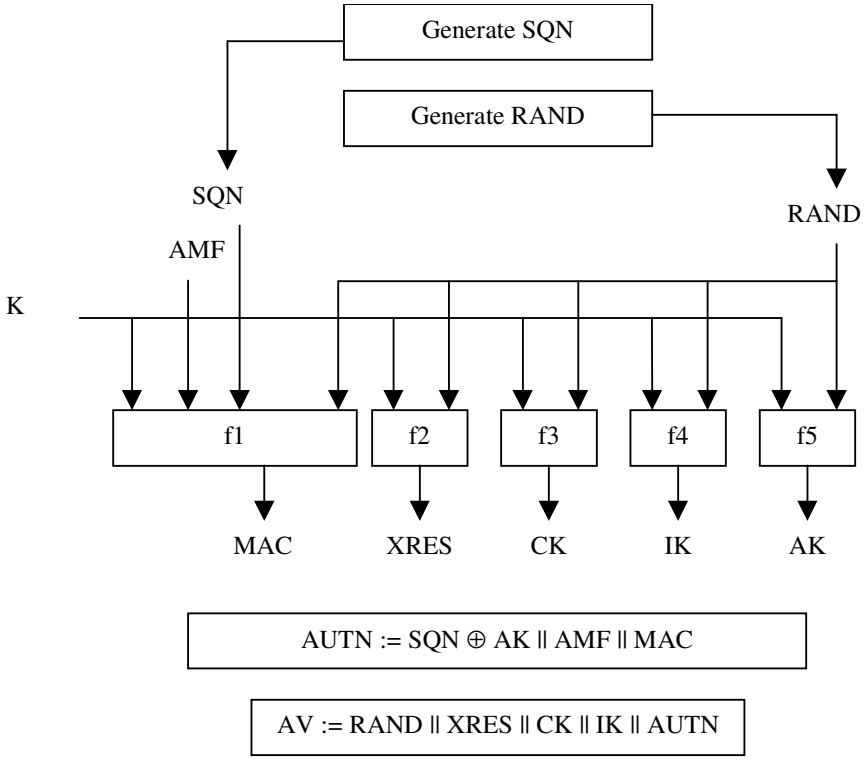

Fig. 2. Generation of Authentication Vectors [1]

2 UMTS AKA Description

The purpose of UMTS AKA is to authenticate the user and network to each other and also establish a new pair of cipher and integrity keys between the VLR/SGSN and the USIM [1]. MS, VLR/SGSN, and HLR/AuC (HE) are involved in UMTS AKA protocol. At identification stage, MS sends its identity to VLR/SGSN via SRNC.

The secret key (K) and cryptographic Algorithms including f_1, f_1^*, f_2, f_3, f_4, f_5, and f_5^*, shared between MS and HE, are used for UMTS AKA process. Furthermore HE and MS track the value of a counter (SQN_{HE}) and (SQN_{MS}) respectively. These sequence numbers are used for the purpose of freshness checking of the received messages. As shown in figure1, UMTS Identification, distribution of authentication data, and AKA procedure are performed as follows.

Identification

1- The MS sends the initial L3 message including its TIMSI and the KSI to VLR/SGSN via SRNC. By this message, the MS requests for services like Location Update, CM Service and Routing Area Update.

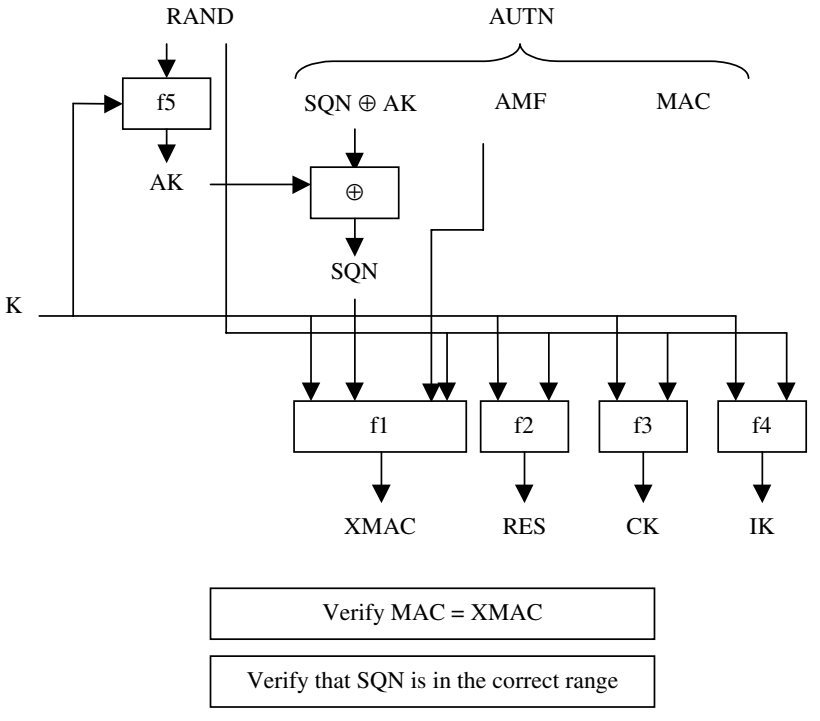

Fig. 3. User Authentication Function in the MS [1]

Distribution of Authentication Data

2- VLR/SGSN identifies MS by its TMSI and then sends the authentication data request including IMSI and requesting node type (PS or CS) to the HE.

3- Upon the receipt of the authentication data request, the HE sends an authentication data response back to the VLR/SGSN which contains an ordered array of n authentication vectors *AV (1...n)*. Each *AV* includes parameters *RAND*, *XRES*, *CK*, *IK*, and *AUTN*. Generation of *AV* is shown in figure2.

Authentication and Key Management

4- The VLR/SGSN chooses the next unused *AV* from the ordered array of *AV*s in the VLR/SGSN database on the basis of first-in/first-out. Then the VLR/SGSN sends to the MS the random challenge (*RAND*) and an authentication token (*AUTN*) from the chosen *AV*.

5- When the MS receives *RAND||AUTN*, it proceeds as illustrated in figure3. The MS first computes the anonymity key $AK=f5_K(RAND)$ and retrieves the sequence number. Then the MS computes $XMAC=f1_K(SQN||RAND||AMF)$ and compares it with MAC included in *AUTN*. If they are the same, the MS verifies if the *SQN* is in the correct range. Then, the MS calculates $RES=f2_K(RAND)$ and sends it to the VLR/SGSN. The VLR/SGSN compares the received *RES* with *XRES*. If they match, then the authentication process of the MS is successfully completed.

3 Related Works

In this section we consider the security analysis of UMTS AKA Protocols. Many protocols have been suggested for UMTS AKA improvement, but we choose some protocols that have novelty in their design and use symmetric algorithms. Nevertheless they have some security or performance weaknesses in their structure that we try to explain.

The UMTS X-AKA protocol [7] applies a temporary key mechanism with timestamp instead of the sequence number. The function *f5* is used for generating temporary keys. The UMTS X-AKA protocol consists of two procedures. First, the user registers on HN and then HN distributes temporary key (TK) and authentication information to SN. Second, the authentication and key agreement procedure is executed between SN and MS. The SN uses TK and authentication information to carry out the mutual authentication between SN and MS and then an agreed key and a cipher key are provided. The UMTS X-AKA protocol uses timestamp to manage freshness of the messages. The timestamp usage needs a robust time synchronization infrastructure. Time synchronization structure of the network has no security feature, so the usage of an independent structure with no security to refresh the exchanged messages is hazardous. Also the HN could not recognize the shared session keys between MS and SN, because SN generates the pseudo-random number needed to construct the session keys.

In [8] an AKA protocol with robust user privacy protection has been proposed. In this scheme, temporary key mechanism to authenticate MS and prevent the location privacy attack is used. In addition, it has lower overhead on VLR. Since MS can easily compute the temporary key through the shared secret key, VLR can be authenticated by MS successfully. In this protocol, the VLR initiates the authentication process by sending a nonce to the MS without any MAC, so DoS attack is probable to be imposed on the MS. Also the protocol has seven steps without identification and security mode set-up stages.

J. Al-Saraireh and S. Yousef proposed an AKA protocol [9] in which, the MS generates the AVs sending to the network. They provided an efficient bandwidth consuming framework with minimal ways for the AKA procedure, but the proposed protocol doesn't support mutual authentication i.e. only the network authenticates the MS. The protocol has 3 steps. The man in the middle attack scenario on interworking of UMTS and GSM [10] could be applied on this protocol [9], because the MS doesn't recognize the validity of the network. Furthermore, the DoS attack on the MS is possible, because the MS could only verify the network until a MAC is received from the network. With some modification in this protocol, it will be mutual. If the VLR/SGSN sends the RES received from the HE to the MS, the mutual authentication would be satisfied by checking the XRES and RES in the MS side.

The security of the wireless network access has been enhanced by Harn and Hsin [11] that uses timestamp and hash chain to provide non-repudiation and freshness. As mentioned earlier, using the timestamp needs independent secure infrastructure. Also the hash chain construction consumes much computation load at the end user side. Furthermore, the number of the protocol ways is six and the protocol doesn't contain the identification and security mode set-up stages.

An extension of UMTS AKA protocol has been proposed by J. Al-Saraireh and S. Yousef in [12] that provides mutual freshness of the MS and the HE but it doesn't use sequence number mechanism and instead, both the MS and HE generates random numbers. So verifying the freshness of the messages could be done by searching in a large database that contains all of the previous random numbers generated by the parties. Applying such a huge database is more expensive and complex than using sequence number method. Besides, the power of DoS attack diminishes when the parties can check the integrity and freshness of the messages faster.

M. Zhang and Y. Fang enhanced the security of the 3GPP AKA by a protocol called AP-AKA. They have projected a special scenario of the redirection attack that UMTS AKA is weak towards it and AP-AKA is robust against it. But the MS traffic redirection by a virtual relay to the neighbor VLR could charge the MS more than usual because the location of the MS has been virtually changed. The first step of the AP-AKA has not integrity protection, so it could be forged. Also the attack [10] can be executed on AP-AKA while interworking with GSM because the VLR initiates the AKA procedure without integrity check. Furthermore, the identification and security mode set-up stages are not considered to provide a better performance for the six-way AP-AKA protocol.

4 UMTS Extended AKA Protocol

In this section, we propose a new authentication protocol of UMTS mobile networks. The stages of the proposed protocol are illustrated in figure4 and figure5. UMTS AKA has some weaknesses to provide simple DoS-attack resistance. Also previous proposed schemes (described in section III) do not provide strong mutual freshness. Furthermore, the steps of the pre-proposed protocols could be reduced to save bandwidth consumption.

The new proposed UMTS AKA protocol, named EAKAP, combines identification stage and AKA stage with security mode set-up of described UMTS protocol. EAKAP is done by a 5-way handshake protocol between the MS, the VLR/SGSN, and the HE. Most of the previous schemes were done by a 5-way handshake in the phase of AKA without security mode set-up, so we could enhance bandwidth efficiency. The EAKAP applies the secret key (K) and the cryptographic algorithms that are used in UMTS AKA protocol shared between the MS and the VLR/SGSN. But the usage of the *f2* algorithm in EAKAP is not necessary because EAKAP does not generate the *RES* or the *XRES*.

Both the MS and the HE have sequence number and random number generator to provide complete freshness of their messages. On the basis of the *f5* structure [2], we could improve the confidentiality of the *SQN* by changing its encryption method. To protect VLR/SGSN against DoS attack, we assume some security capabilities at VLR/SGSN side. These security features contain the shared cryptographic algorithms (*fc* and *fi*) between MS and VLR/SGSN. The *fc* algorithm is used for ciphering and the *fi* algorithm is used to generate integrity check. The EAKAP procedure works as follows (shown in figure4).

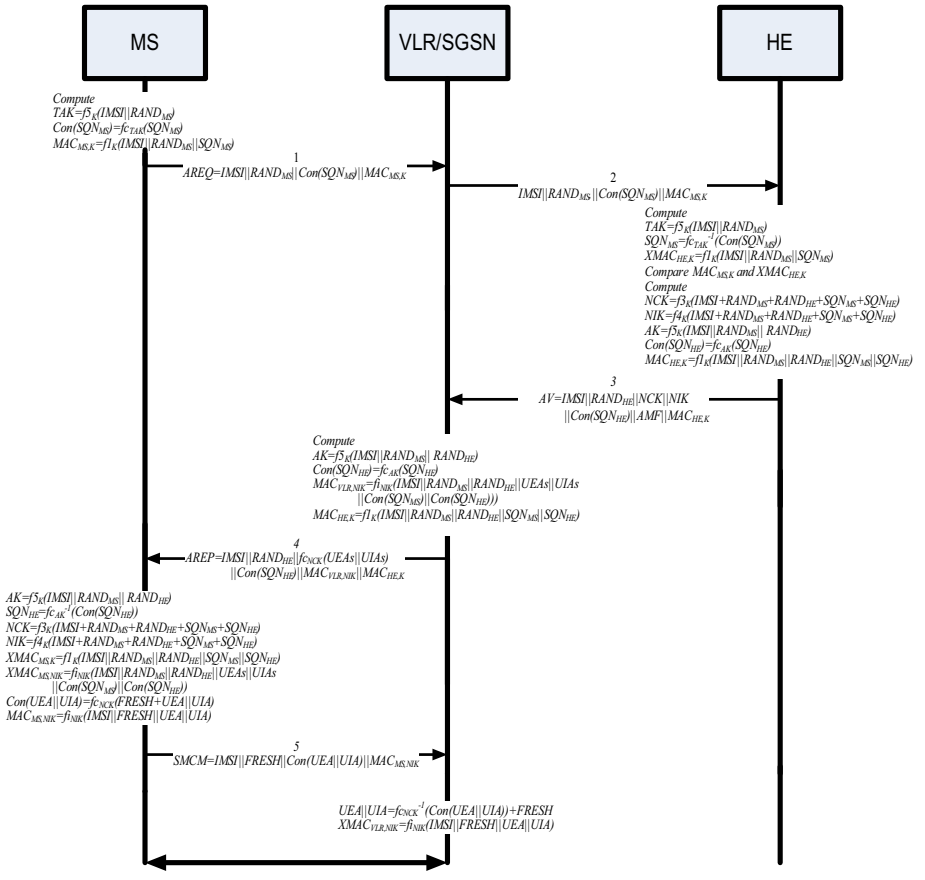

Fig. 4. EAKAP Structure when MS meets a new VLR/SGSN

Authentication Request

1- If no TMSI was set before, the MS generates the *AREQ* message as follows:

$$\text{AREQ}=\text{IMSI}||\text{RANDMS}||\text{Con}(\text{SQN}_{\text{MS}})||\text{MAC}_{\text{MS,K}} \quad (1)$$

Where

$$\text{TAK}=\text{f5}_{\text{K}}(\text{IMSI}||\text{RAND}_{\text{MS}}) \quad (2)$$

$$\text{Con}(\text{SQN}_{\text{MS}})=\text{fc}_{\text{TAK}}(\text{SQN}_{\text{MS}}) \quad (3)$$

$$\text{MAC}_{\text{MS,K}}=\text{f1}_{\text{K}}(\text{IMSI}||\text{RAND}_{\text{MS}}||\text{SQN}_{\text{MS}}) \quad (4)$$

If the VLR/SGSN has allocated a TMSI encrypted by previous *CK* (*OCK*) paired with previous *IK* (*OIK*) to the MS, the MS would generate the *AREQ* message as follows:

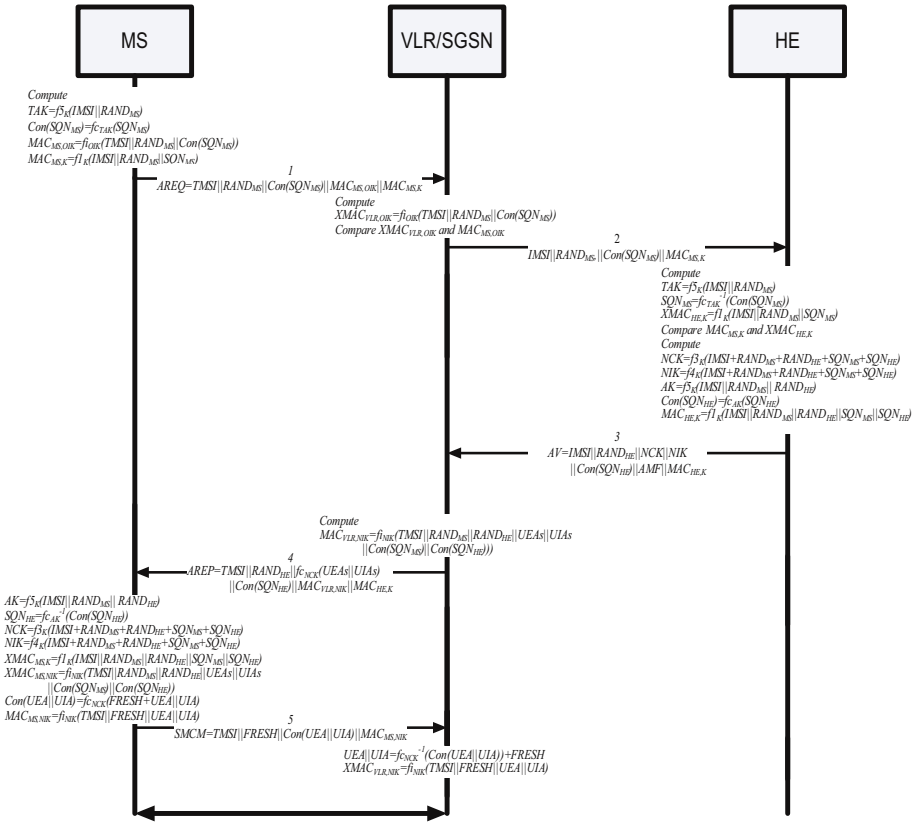

Fig. 5. EAKAP Structure when MS meets an old VLR/SGSN

$$AREQ=TMSI||RAND_{MS}||Con(SQN_{MS})||MAC_{MS,OIK}||MAC_{MS,K} \quad (5)$$

Where

$$TAK=f5_K(IMSI||RAND_{MS}) \quad (6)$$

$$Con(SQN_{MS})=fc_{TAK}(SQN_{MS}) \quad (7)$$

$$MAC_{MS,OIK}=fi_{OIK}(TMSI||RAND_{MS}||Con(SQN_{MS})) \quad (8)$$

$$MAC_{MS,K}=f1_K(IMSI||RAND_{MS}||SQN_{MS}) \quad (9)$$

Then the MS sends the calculated *AREQ* to the VLR/SGSN.

Distribution of Authentication Data

2- The VLR/SGSN identifies the MS by its sent IMSI via HE or TMSI via old VLR/SGSN. If TMSI related to current VLR/SGSN is sent, the VLR/SGSN computes the $XMAC_{VLR,OIK}$ (as given below) and compares this with $XMAC_{MS,OIK}$ which is included in *AREQ*.

$$XMAC_{VLR,OIK}=fi_{OIK}(TMSI||RAND_{MS}||Con(SQN_{MS})) \quad (10)$$

If they are different, VLR/SGSN does not distribute the authentication data request for HE and AKA process fails. Otherwise, VLR/SGSN sends the authentication data request included IMSI, $RAND_{MS}$, $Con(SQN_{MS})$ and $MAC_{MS,K}$ to HE.

3- When the HE receives the authentication data request from the VLR/SGSN, it first retrieves sequence number of the MS as follows.

$$TAK=f5_K(IMSI||RAND_{MS}) \quad (11)$$

$$SQN_{MS}=fc_{TAK}^{-1}(Con(SQNMS)) \quad (12)$$

Then the HE computes the $XMAC_{HE,K}$ (as given below) and compares it with $MAC_{MS,K}$ which is included in the VLR/SGSN's authentication data request.

$$XMAC_{HE,K}=f1_K(IMSI||RAND_{MS}||SQN_{MS}) \quad (13)$$

If the $MAC_{MS,K}$ and the $XMAC_{HE,K}$ are different, the HE detects the MS as a fraud user and does not generate any *AV*. If they are the same, the HE verifies the *SQN* is in the correct range. If the SQN_{MS} is considered to be in the correct range, the HE sends an authentication data response back to the VLR/SGSN that contains an authentication vector (*AV*). The generated *AV* included IMSI, $RAND_{HE}$, new *CK* (*NCK*), new *IK* (*NIK*), $Con(SQN_{HE})$, *AMF*, and $MAC_{HE,K}$ as shown in figure5.

$$AV=IMSI||RAND_{HE}||NCK||NIK||Con(SQN_{HE})||AMF||MAC_{HE,K} \quad (14)$$

Where

$$NCK=f3_K(IMSI+RAND_{MS}+RAND_{HE}+SQN_{MS}+SQN_{HE}) \quad (15)$$

$$NIK=f4_K(IMSI+RAND_{MS}+RAND_{HE}+SQN_{MS}+SQN_{HE}) \quad (16)$$

$$AK=f5_K(IMSI||RAND_{MS}||\ RAND_{HE}) \quad (17)$$

$$Con(SQN_{HE})=fc_{AK}(SQN_{HE}) \quad (18)$$

$$MAC_{HE,K}=f1_K(IMSI||RAND_{MS}||RAND_{HE}||SQN_{MS}||SQN_{HE}) \quad (19)$$

Authentication Reply

4- Upon the receipt of the AV from the HE, the VLR/SGSN computes the *AREP* for the MS. The VLR/SGSN determines which UIAs and UEAs that are allowed to be used in order of preference. If no TMSI was set before, the VLR/SGSN generates the *AREP* message as follows:

$$AREP=IMSI||RAND_{HE}||fc_{NCK}(UEAs||UIAs) \\ ||Con(SQN_{HE})||MAC_{VLR,NIK}||MAC_{HE,K} \quad (20)$$

Where

$$AK=f5_K(IMSI||RAND_{MS}||\ RAND_{HE}) \tag{21}$$

$$Con(SQN_{HE})=fc_{AK}(SQN_{HE}) \tag{22}$$

$$MAC_{VLR,NIK}=fi_{NIK}(IMSI||RAND_{MS}||RAND_{HE} \\ ||UEAs||UIAs||Con(SQN_{MS})||Con(SQN_{HE}))) \tag{23}$$

$$MAC_{HE,K}=f1_K(IMSI||RAND_{MS}||RAND_{HE}||SQN_{MS}||SQN_{HE}) \tag{24}$$

If the VLR/SGSN allocates a TMSI encrypted by previous *CK* (*OCK*) paired with old *IK* (*OIK*) to the MS, it would generate the *AREP* message as follows:

$$AREP=TIMSI||RAND_{HE}||fc_{NCK}(UEAs||UIAs) \\ ||Con(SQN_{HE})||MAC_{VLR,NIK}||MAC_{HE,K} \tag{25}$$

Where

$$MAC_{VLR,NIK}=fi_{NIK}(TIMSI||RAND_{MS}||RAND_{HE} \\ ||UEAs||UIAs||Con(SQN_{MS})||Con(SQN_{HE})) \tag{26}$$

5- Upon the receipt of the *AREP* message, the MS proceeds as illustrated in figure4 and figure5. If the MS has a TMSI, allocated by the VLR/SGSN, it first retrieves the HE's sequence number as follows.

$$AK=f5_K(IMSI||RAND_{MS}||\ RAND_{HE}) \tag{27}$$

$$SQN_{HE}=fc_{AK}^{-1}(Con(SQN_{HE})) \tag{28}$$

If the *SQN* is considered to be in the correct range, the MS calculates the new pair key, the new cipher key (*NCK*) and the new integrity key (*NIK*), as follows:

$$NCK=f3_K(IMSI+RAND_{MS}+RAND_{HE}+SQN_{MS}+SQN_{HE}) \tag{29}$$

$$NIK=f4_K(IMSI+RAND_{MS}+RAND_{HE}+SQN_{MS}+SQN_{HE}) \tag{30}$$

Then the MS calculates the $XMAC_{MS,K}$ (as given below) and checks the correspondence of the $MAC_{HE,K}$ and the $XMAC_{MS,K}$.

$$XMAC_{MS,K}=f1_K(IMSI||RAND_{MS}||RAND_{HE}||SQN_{MS}||SQN_{HE}) \tag{31}$$

If they are the same, the MS decrypts the $fc_{NCK}(UEAs||UIAs)$ and gets the cryptography capabilities of the VLR/SGSN in order of preference. Then the MS computes $XMAC_{MS,NIK}$ (as given below) with regard to the allocated TMSI.

$$XMAC_{MS,NIK}=fi_{NIK}(TMSI||RAND_{MS}||RAND_{HE} \\ ||UEAs||UIAs||Con(SQN_{MS})||Con(SQN_{HE})) \tag{32}$$

If no TMSI was allocated, the $XMAC_{MS,NIK}$ would be as follows:

$$XMAC_{MS,NIK}=fi_{NIK}(IMSI||RAND_{MS}||RAND_{HE} ||UEAs||UIAs||Con(SQN_{MS})||Con(SQN_{HE})) \quad (33)$$

The MS compares $XMAC_{MS,NIK}$ with $MAC_{VLR,NIK}$ included in the received *AREP* message. If they are different, the MS does not continue authentication process and terminate the connection. Otherwise, the MS generates the security mode complete message (SMCM) as follows:

$$SMCM=TIMSI||FRESH||Con(UEA||UIA)||MAC_{MS,NIK} \quad (34)$$

Where

$$Con(UEA||UIA)=fc_{NCK}(FRESH+UEA||UIA) \quad (35)$$

$$MAC_{MS,NIK}=fi_{NIK}(TMSI||FRESH||UEA||UIA) \quad (36)$$

The MS sends the SMCM to the VLR/SGSN and the VLR/SGSN decrypts the *Con(UEA||UIA)* by the new shared cipher key (*NCK*) to get preferred UEA and UIA.
$UEA||UIA=fc_{NCK}^{-1}(Con(UEA||UIA))+FRESH$

Then the VLR/SGSN verifies the integrity of the SMCM by generating the $XMAC_{VLR,NIK}$ with regard to the chosen UEA and UIA by the MS and comparing it with the $MAC_{MS,NIK}$.

$$XMAC_{VLR,NIK}=fi_{NIK}(TMSI||FRESH||UEA||UIA) \quad (37)$$

Consequently the IAKA (Identification and AKA) process with security mode setup is complete i.e. the parties authenticate each other mutually and the preferred algorithms for ciphering and integrity checking are chosen by them.

5 Security and Performance Features of the Proposed Protocol

In this section, we analyze the security features of the new proposed protocol explained in previous section and then we evaluate the performance of the suggested protocol relatively. The EAKAP provides the main security services issued in literature like authentication, confidentiality and integrity [3]. Furthermore, the EAKAP is more robust than the UMTS AKA and the previously suggested protocols against the DoS attack so with using the EAKAP, availability service would be provided properly.

Performance Evaluation towards Previous Works

In the EAKAP, three sections of the UMTS protocol including identification, AKA and security mode set-up are combined to set-up connection. In EAKAP, security mode setup is performed after key establishment so the attacker has no information about the shared algorithms for ciphering and integrity check. The EAKAP is completed after five-way interactions. Previous works do not consider the identification and security mode set-up steps. Furthermore, if we consider the whole protocol, the number of interactions has been reduced largely. So the load of the network signaling is reduced and performance of the IAKA procedure grows efficiently.

Confidentiality, Integrity, and Authentication

As described in previous section, the only shared secret between the MS and the HE is a private key (K). Sometimes the old session pair keys (*OCK*, *OIK*) are shared between MS and VLR/SGSN if the MS has visited the area of the VLR/SGSN already. After the IAKA process, the MS and the VLR/SGSN share the new session pair key (*NCK*, *NIK*). These keys are hidden from the sight of adversary and we show that they could not be guessed or eavesdropped easily.

The TMSI/IMSI fields are sent public without any encryption because anonymity service is out of the scope of the article. Also RAND fields are sent as plain text because they are needed for the other side to generate necessary fields. The sequence numbers, UEA, and UIA fields are encrypted by the shared keys (K or *NCK*). So the confidentiality service is provided properly. Also *MAC* fields are protected by the shared keys (K, *OIK* or *NIK*) and hence the integrity service is afforded appropriately.

The HE could authenticate the identity (IMSI) of the MS by verifying the integrity of the $MAC_{MS,K}$ generated by the MS. The MS could confirm the identity of its home network by checking the $MAC_{HE,K}$ calculated by the HE. So two parties authenticate each other suitably referred to mutual authentication. Besides, the MS and the VLR/SGSN could authenticate each other by verifying the $MAC_{MS,OIK}$, $MAC_{MS,NIK}$ and $MAC_{VLR,NIK}$, as the session keys (*OIK, NIK*) could be computed or available in both sides (MS side and VLR/HE side). So the mutual authentication with security mode set-up is done by the EAKAP.

Sequence Number Protection vs. Private Key Derivation

MS increments the SQN_{MS} for every IAKA procedure and HE runs the SQN_{HE} for every generated *AV*. MS and HE use sequence number and random number generator to achieve strong freshness of their messages. However pseudo-random numbers could provide the freshness of the ways of the EAKAP, but both MS and HE need a large directory to save used random numbers in it. Also to avoid the usage of the repeated random numbers they have to search in their directory or sort and update the directory for every procedure that it consumes a large amount of energy and memory.

In the EAKAP, anonymity protection of the sequence number is higher than the UMTS AKA because the sequence number of the UMTS AKA is XOR-ed by anonymity key derived from nonce and the private key [2] like stream ciphers. As sequence number value is guessable because it starts from zero and changes incrementally slowly, the known plaintext attack is imaginable on the *f5* algorithm to get private key (K). However the core of the *f5* algorithm is based on Advanced Encryption Standard (AES) [4] that the recent proposed attacks are not practical on it [5]. In the EAKAP, the sequence number is encrypted by a block cipher algorithm called *fc* with derived key from *f5* algorithm and so the privacy of the sequence number is strongly established.

Availability (Robustness against DoS Attack)

If the MS enters area of a new VLR/SGSN, it won't have a TMSI allocated by VLR/SGSN. So the MS must use its IMSI to initiate the procedure. In this situation, the MS actually aims its HE not VLR/SGSN. In fact SRNC and VLR/SGSN role as

relays to forward the MS messages to the HE. Here we don't struggle with the privacy establishment of the MS.

First, we consider the condition that no TMSI was set before. The MS and the VLR/SGSN have no shared key before and so, as mentioned in step1 of the previous section, the *AREQ* message has been sent to the HE via the intermediate SRNC and VLR/SGSN. As mentioned in step3, the HE could check the integrity of the $MAC_{MS,K}$, so the spam messages generated by spurious users are discarded by the HE. Furthermore, the DoS attack organized by unauthentic MS at the first step, could be detected at the HE side and no more traffic would be procreated in the network.

Second, we consider the situation where the VLR/SGSN has allocated the MS a TMSI so the MS and the VLR/SGSN have shared a pair key (*OCK* and *OIK*) with each other. Consequently, as explained in step2, the VLR/SGSN checks the integrity of the $MAC_{MS,OIK}$ and rejects the forged messages. Furthermore, with the usage of the pre-shared information of the MS, the network could prevent DoS attack of first step at the VLR/SGSN side.

In the other steps of the EAKAP, by using MAC mechanism, the vulnerability towards DoS attack is reduced as mentioned above. Although in some cases, the *SQN* should be computed before the MAC so the computation load would be increased. But no further signaling load would be injected to the network and the elements of the network could decide about the genuineness of the message by verifying the MAC integrity.

Mutual Freshness and Unguessable Keys

The SQN_{MS} is encrypted by temporary anonymity key (*TAK*) derived from the output of the *f5* algorithm by a high entropy seed which is $IMSI||RAND_{MS}$. The SQN_{HE} is protected by anonymity key (*AK*) derived from the *f5* algorithm by concatenation of the IMSI, $RAND_{MS}$ and $RAND_{HE}$ that is an entropic seed to generate a fresh key. The seed entropy of the *f5* algorithm depends on two random numbers $RAND_{MS}$ and $RAND_{HE}$ and hence is improbable to be guessed by adversary. So the sequence numbers could not be revealed. Also if an anonymity key is compromised at a session, no next-generated keys will be concealed i.e. our scheme supports forward security towards session key compromising. The MAC fields like the *SQN* fields are fresh. The entropic seed of the MAC fields is derived from the pseudo random numbers as well as IMSI and sequence number.

6 Conclusion

Most of the proposed protocols do not consider the previous and next stages of the UMTS AKA protocol and they try to improve the UMTS AKA protocol solely. Our proposed protocol, so called EAKAP, merges all of the stages in five ways so it improves performance by reducing the load of the network signaling. Also EAKAP support mutual freshness and is more robust against DoS attack by applying MAC mechanism between the MS and the network.

References

1. 3GPP TS 33.102 V8.0.0 (2008-06), 3GPP Technical Specification Group Services and System Aspects, 3G Security, Security Architecture (Release 8)
2. 3GPP TS 35.206 V7.0.0 (2007-06), Technical Specification Group Services and System Aspects, 3G Security, Algorithm Specification (Release 7)
3. Stallings, W.: Cryptography and Network Security: Principles and Practice. Prentice-Hall, Englewood Cliffs (1998)
4. National Institute of Standards and Technology (NIST), Advanced Encryption Standard (AES), Federal Information Processing Standards Publications (FIPS PUBS) 197 (2001)
5. Dobbertin, H., Knudsen, L., Robshaw, M.: The cryptanalysis of the AES - A brief survey. In: Dobbertin, H., Rijmen, V., Sowa, A. (eds.) AES 2005. LNCS, vol. 3373, pp. 1–10. Springer, Heidelberg (2005)
6. Shannon, C.E.: A Mathematical Theory of Communication. The Bell System Technical Journal 27, 379–423, 623–656 (1948)
7. Huang, C.M., Li, J.W.: Authentication and Key Agreement Protocol for UMTS with Low Bandwidth Consumption. In: Proceedings of the 19th International Conference on Advanced Information Networking and Applications (2005)
8. Juang, W.S., Wu, J.L.: Efficient 3GPP Authentication and Key Agreement with Robust User Privacy Protection. In: IEEE Communications Society. Proceedings of the WCNC (2007)
9. Al-Saraireh, J., Yousef, S.: A New Authentication Protocol for UMTS Mobile Networks. EURASIP Journal on Wireless Communications and Networking, Article ID 98107, 1–10 (2006)
10. Meyer, U., Wetzel, S.: A Man-in-the-Middle Attack on UMTS. In: Proceedings of the 3rd ACM workshop on Wireless security, pp. 90–97 (2004) ISBN:1-58113-925-X
11. Harn, L., Hsin, W.J.: On the Security of Wireless Network Access with Enhancements. In: Proceedings of the 2nd ACM workshop on Wireless security, pp. 88–95 (2003) ISBN:1-58113-769-9
12. AL-Saraireh, J., Yousef, S.: Extension of Authentication and Key Agreement Protocol (AKA) for Universal Mobile Telecommunication System (UMTS). International Journal of Theoretical and Applied Computer Sciences 1(1), 109–118 (2006)
13. Zhang, M., Fang, Y.: Security Analysis and Enhancements of 3GPP: Authentication and Key Agreement Protocol. IEEE Transactions on Wireless Communications 4(2) (March 2005)

APPENDIX (Abbreviations)

3GPP	Third-Generation Partnership Project
AK	Anonymity Key
AKA	Authentication and Key Agreement
AMF	Authentication Management Field
AREP	Authentication Reply
AREQ	Authentication Request
AuC	Authentication Center
AUTN	Authentication Token
AV	Authentication Vector

BSC	Base Station Controller
BTS	Base Transceiver Station
CK	Cipher Key
CM	Connection Management
CS	Circuit Switched
DoS	Denial of Service
EAKAP	Extended AKA Protocol
FRESH	Pseudo-Random Number Generated by MS
GSM	Global System for Mobile
HE	Home Environment included HLR and AuC
HLR	Home Location Register
HN	Home Network
IAKA	Identification and AKA
IK	Integrity Key
IMSI	International Mobile Subscriber Identity
KSI	Key Set Identifier
MAC	Message Authentication Code
MAC-I	Message Authentication Code for Integrity
MS	Mobile Station
MSC	Mobile Switching Center
PS	Packet Switched
$RAND_X$	Pseudo-Random Number Generated by X
RES	Response
SGSN	Serving GPRS Support Node
SMCM	Security Mode Complete Message
SQN	Sequence Number
SRNC	Serving Radio Network Controller
TAK	Temporary Anonymity Key
TIMSI	TMSI or IMSI
TMSI	Temporary Mobile Subscriber Identity
UEA	UMTS Encryption Algorithm
UIA	UMTS Integrity Algorithm
UMTS	Universal Mobile Telecommunications System
USIM	Universal Subscriber Identity Module
VLR	Visitor Location Register