Efficient and Secure Authentication and Key Agreement Protocol

Ja'afer AL-Saraireh
Applied Science University, Amman-Jordan

1. Abstract

In the UMTS Authentication and Key Agreement (AKA) protocol only the home network can generate authentication vectors to its subscribers. Therefore; the home location register and authentication centre (HLR/AuC) actually suffers from the traffic bottleneck. AKA protocol has been enhanced by generating temporary key to enable visitor location register (VLR/SGSN) to authenticate mobile station (MS) without intervention of HLR/AuC.

The proposed protocol satisfies the security requirements of third generation (3G) mobile networks. In this research paper the current AKA has been enhanced by reducing the network traffic, signalling message between entities. This is achieved by reducing a size n array of authentication vector and the number of messages between MS and HLR/AuC. Hence, the traffic for the home network to generate authentication vectors is exponentially decreased, then reducing the authentication times, and setup time as well as improving authentication efficiency. Additionally, a mutual authentication between MS and its Home Network (HN) and between an MS and the Serving Network (SN) is achieved. A security analysis and comparison with related work shows that E-AKA is more efficient and a secure authentication is achieved.

Keywords: 3G, Authentication, Security, Mobile Station, and Authentication Vector.

2. Introduction

Wireless communication is a technology that is becoming a feature in many aspects of our daily life. Wireless networks face a large number of challenges that the conventional wired networks do not.

Wireless systems are more vulnerable to fraudulent access and eavesdropping. As a solution for this, mobile network systems are giving more importance to the privacy of users through an authentication process. The authentication process provides a reasonable level of security, but it overloads the network with significant signalling traffic and increases the call setup time [1].

Authentication is used as an initial process to authorize a mobile terminal for communication through secret credentials [2]. In authentication process, a mobile terminal is required to submit secret materials such as certificate or "challenge and response" values for verification. Without strong authentication, mobile networks access is unprotected through the release of message contents, modification of message or denial of service can be accomplished easily by an intruder [1, 2].

Figure 1 illustrates the Universal Mobile Telecommunication System (UMTS) architecture. There are three entities participating in the UMTS security architecture, home environment (***HE***), serving network (***SN***) and ***MS***. The ***HE*** contains the home location register (***HLR***) and authentication centre (***AuC***). The ***SN*** consists of the visited location register (***VLR***) and the Serving GPRS Support Node (***SGSN***). The ***VLR*** handles circuit switched traffic, but ***SGSN*** handles the packet switched traffic [2, 4].

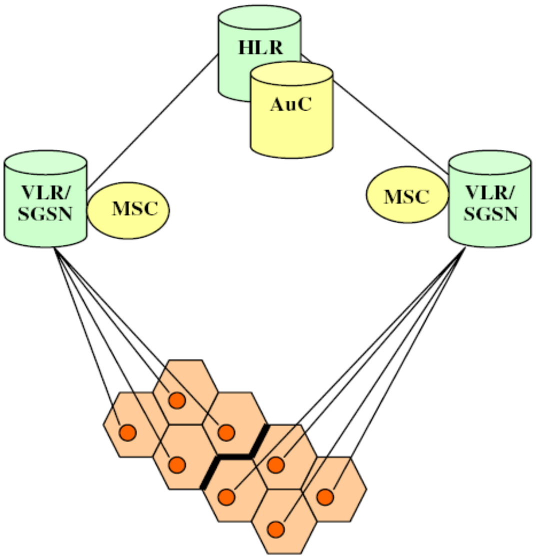

Figure 1 UMTS Architecture

To provide security services in wireless networks, authentication is used as an initial process to authorize a mobile terminal for communication through secret credentials [1, 2, 4]. Authentication procedure is executed when the ***MS*** moves from one registration area (***RA***) to another one (i.e., location update) during the process of calls origination and call termination. The ***MS*** is continuously listening to the broadcast message from ***VLR/SGSN*** to identify the location area by using location area identity (***LAI***) and the ***MS*** compares the ***LAI*** which is received with the ***LAI*** that stored in the Universal Subscriber Identity Mobile (***USIM***). When the ***LAI*** is different, then the MS executes authentication procedure [1, 2].

This paper is organized as follows. In section 2 the UMTS authentication and key agreement procedure is explained and its weaknesses are presented. Section 3, the literature review and related works is presented. In Section 4, the proposed protocol is described. The proposed protocol is securely analyzed and evaluated in Section 5. In Section 6, a comparison with the current UMTS-AKA and related works is presented. The paper is concluded in Section 7.

3. UMTS AKA Description and it Weaknesses

An authentication mechanism is a process designed to allow all participants show their legality and verify the other participant's identities that involved in the networks.

This mechanism using secret key (***K***), and cryptographic algorithms - include three message authentication codes f_1, f_1*and f_2 and four key generation functions f_3, f_4, f_5 ***and*** f_5*- that are shared between ***MS*** and the ***HLR/AuC*** [1, 5]. This is known as authentication and key agreement protocol (***AKA***). The ***AuC*** maintains a counter called sequence number (SQN_{HLR}), where user ***MS*** maintains a counter (SQN_{MS}), whose initial value for these counters are set to zeros [1, 5, 6].

There are three goals for the UMTS AKA protocol [1, 6]:

- **i.** A mutual authentication between the user and the network;
- **ii.** An establishment of a cipher key and an integrity key upon successful authentication; and
- **iii.** A freshness assurance to the user of the established cipher and integrity keys.

There are two phases in AKA protocol [2, 5]:

- **i.** ***MS*** registers with its ***HLR/AuC*** and then generates and distributes authentication vectors from the ***HLR/AuC*** to the ***VLR/SGSN***.
- **ii.** The authentication and key agreement procedure between the ***MS*** and the ***VLR/SGSN***.

Figure 2 describes authentication mechanism as follow:

1. When the ***MS*** moves to new ***VLR/SGSN*** area then ***MS*** sends International Mobile Subscriber Identity ***(IMSI)*** authentication request to ***VLR/SGSN***.
2. ***VLR*** passes this authentication request to ***HLR***.
3. ***HLR*** generates authentication vectors ***AV(1..n)*** and sends authentication data response ***AV(1..n)*** to ***VLR/SGSN***, where each authentication vector is called a quintet This ***AV*** consists of five components: random number (***RAND***), expected response (***XRES***), cipher key (***CK***), integrity key (***IK***) and authentication token (***AUTN***). The authentication vectors are ordered by the sequence number SQN_{HLR}. The authentication vector is generated according to the following sequence:
   - **i.** ***HLR/AuC*** generates SQN_{HLR} and ***RAND***.
   - **ii.** ***HLR/AuC*** computes $XRES = f_2(K,RAND)$, $CK = f_3(K,RAND)$, $IK = f_4(K,RAND)$, Anonymity Key $AK = f_5(K,RAND)$, Message Authentication Code $MAC = f_1(K,SQN \| RAND \| MAF)$, where ***MAF*** is Message Authentication Field and $AUTN = (SQN \oplus AK \| AMF \| MAC)$ where $\oplus$ is exclusive OR operation.
   - **iii.** ***HLR/AuC*** SQN_{HLR} is increased by 1.
4. ***VLR*** stores authentication vectors. In the i^{th} authentication and key agreement procedure, ***VLR/SGSN*** selects the i^{th} authentication vector ***AV(i)***, and sends (***RAND (i), AUTN(i)***) to ***MS***. In the ***VLR*** one authentication vector is needed for each authentication instance. This means that the signalling between ***VLR*** and ***HLR/AuC*** is not needed for every authentication events.
5. ***MS*** computes and retrieves the following:
   - **i.** Anonymity key $AK = f_5\ (Rand, K)$, $SQN =(\ (SQN \oplus AK) \oplus AK)$, computes expected message authentication code $XMAC = f_1\ (SQN, RAND, AMF)$ and then,
   - **ii.** Compares ***XMAC*** with ***MAC*** which is included in ***AUTN***. If ***XMAC*** is not equal to ***MAC*** then ***MS*** sends failure message to the ***VLR/SGSN***,

else if ***XMAC*** is equal ***MAC*** then ***MS*** checks that the received ***SQN*** is in the correct range i.e. $\boldsymbol{SQN} > \boldsymbol{SQN_{MS}}$. If ***SQN*** is not in the correct range then ***MS*** sends failure message to the ***VLR/SGSN***, else if it is in the correct range, then ***MS*** computes the Response $\boldsymbol{RES = f_2\ (K, RAND)}$, and $\boldsymbol{CK = f_3\ (K, Rand)}$,

iii. After that, it sends ***RES*** to ***VLR/SGSN.***

6. ***VLR*** compares the received ***RES*** with ***XRES***. If they match, then authentication is successfully completed.

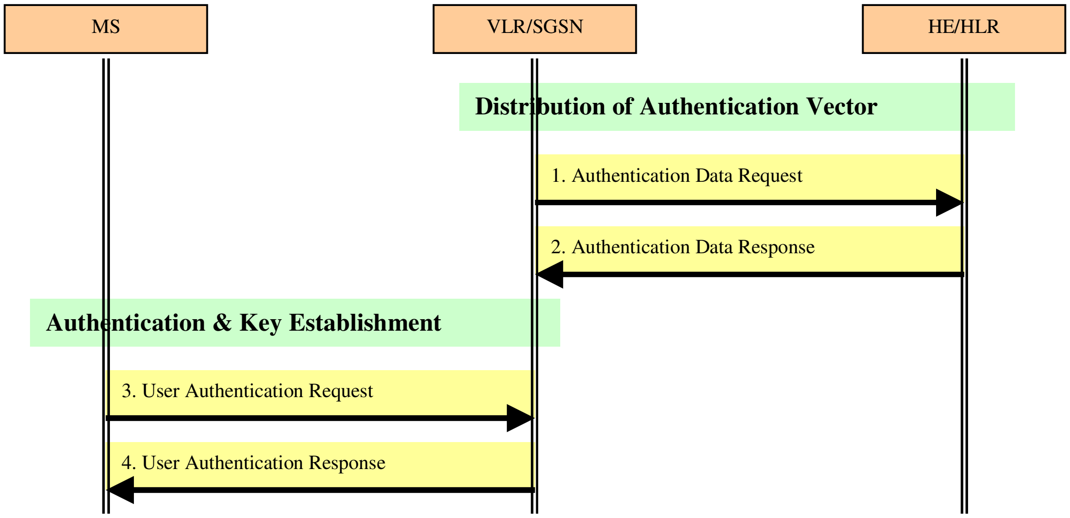

Figure 2 Authentications and Key Agreement Protocol

According to the previous description of UMTS AKA protocol, there are a set of UMTS AKA has set of weaknesses, such as:

i. The transmission between the ***HLR/AuC*** and ***VLR/SGSN*** is usually expensive. The ***AVs*** will consume network bandwidth for each transmission from authentication centre to ***VLR/SGSN*** [7].

ii. The storage space of ***VLR/SGSN***, An array of ***L*** authentication vectors for each ***MS*** must be stored in the ***VLR/SGSN***. If there are ***m MSs*** in a ***VLR/SGSN***, then the ***VLR/SGSN*** must store ***L*** * ***m*** authentication vectors. Therefore, a space overhead occurs [2].

iii. Bottleneck at ***HLR/AuC***. In UMTS AKA, the ***HLR/AuC*** is responsible for generating authentication vectors upon receipt of requests from all ***VLRs/SGSNs***. While the number of subscribers is usually large, the ***HLR/AuC*** experiences heavy authentication traffic. A size ***L*** (i.e., L is usually 5) array of authentication vectors can be used for ***L*** times authentication. Since only the ***HLR/AuC*** is responsible for generating and sending authentication vectors for all subscribers, it actually becomes the traffic bottleneck.

Therefore; an enhancement of the UMTS AKA protocol is proposed to provide solutions to the above mentioned weaknesses of the current UMTS AKA protocol. The proposed authentication and key agreement protocol is name E-AKA.

4. Related works

Several authentication schemes have been proposed for mobile networks to enhance the security of mobile communication systems. However, these schemes cannot fulfil the security requirements of 3G mobile systems [8]. Specifically, the schemes proposed by Horn et al. (2002), Lee et al. (2002), Lee et al. (2003), Lin and Shieh (2000), and Looi (2001) were not designed based on 3G mobile systems and incur much computational overheads.

Huang and Li (2005) propose an extension of UMTS AKA protocol, called UMTS X-AKA, to overcome some of problems of UMTS AKA protocol. The UMTS X-AKA protocol uses timestamp to manage re-freshness of the messages. To use timestamp a time synchronization infrastructure is required. So time-sync structure of the network has no security feature.

Daeyoung etal., proposed a privacy protecting UMTS AKA protocol providing perfect forward secrecy. The proposed protocol uses timestamp as X-AKA and use EC-based Diffie-Hellman key agreement protocol this cause authentication time and setup time.

Zhang and Fang, Zhang and Fujise, and Zhang showed that the 3GPP AKA protocol is vulnerable to a variant of the false base station attack. The vulnerability allows an adversary to redirect user traffic from one network to another. It also allows an adversary to use authentication vectors corrupted from one network to impersonate all other networks. Zhang and Fang presented a new authentication and key agreement protocol, which overcomes redirection attack and drastically lowers the impact of network corruption. The protocol, called adaptive protocol AKA (AP-AKA), also eliminates the need of synchronization between a mobile station and its home network.

Adi et al., proposed a technique for public key image authentication using fussy computations for El-Gamal authentication technique. A mutual authentication key and key exchange protocol suitable for application is proposed by Yijun et al.,. This protocol named F-MAKEP. The F-MAKEP scheme integrated into Wireless Transport Layer Security (WTSL) framework; the security was enhanced while more computation overhead was incurred.

The UMTS AKA protocol has the problem of the bandwidth consumption between ***SN*** and ***HN***. It is attractive to choose a suitable length (***L***) value for ***AV*** in the third generation mobile networks. So, many techniques are developed to minimize the authentication signalling cost and network bandwidth with consumption by selecting the dynamic length (***L***) for an authentication vector. But with this improvement by [3, 7] are still there are bandwidth consumption.

The technique of Lin and Chen basically estimates the number of authentication requests in current visited network based on the number in the previous visited network. Whereas the method of AL-Saraireh and Yousef, estimates the number of authentication requests in current visited network based on the history of mobile movements and the arrival rate for events.

Juang and Wu proposed an efficient 3GPP AKA with robust user privacy. A temporary key to authenticate ***MS*** and prevent the location privacy attack is used. In this proposed protocol, the ***VLR*** initiates the authentication process by sending a random number to the ***MS*** without using any ***MAC***. Therefore; denial of services (DoS) attack is possible. Additionally, the proposed protocol has seven steps.

A new UMTS AKA protocol called EAKAP is proposed by [21]. The EAKAP combines identification stage and AKA stage of UMTS AKA protocol. The problem in EAKAP is that the size of messages between ***MS, VLR/SGSN*** and ***HLR/AuC*** is increased. Therefore; the consumption of bandwidth is occurred.

Subscriber identity/location confidential and non-repudiation services are solved by [24], the propose scheme integrates symmetric and public key cryptosystem.

An Enhancement for UMTS AKA protocol is proposed by [19]. Harn and Hsin use hash chaining technique instead of using ***AVs***.

5. Description of Proposed UMTS E-AKA Protocol

This research work presents a secure and an efficient authentication protocol for mobile networks, where ***VLR/SGSN*** has the capability to authenticate the user without intervention of ***HLR/AuC*** in the home network during origination and termination of the call. Basically, the proposed authentication protocol consists of the same parts as in the UMTS systems, three nodes are involved in the authentication protocol; the ***MS***, ***VLR***, and ***HLR/AuC***.

Like the UMTS AKA authentication protocol the communication link between a ***VLR/SGSN*** and ***HLR/AuC*** is secure.

The E-AKA uses new key generation functions called it f_x to generate the temporary key (***TK***). The f_x function produce a 128 bits or higher bits to provide high level of security. Also the E-AKA uses the secrete key (***K***) and the cryptographic algorithms that are used in UMTS AKA protocol, include three message authentication codes f_1, $f_1^*$and f_2 and four key generation functions f_3, f_4, f_5, ***and*** f_5^*, that are shared between ***MS, VLR/SGSN*** and the ***HLR/AuC***.

In the proposed protocol after initial authentication has been performed, the ***VLR/SGSN*** is able to authenticate the ***MS*** when it is required. The proposed authentication protocol contains two operation modes for initial and subsequent authentication.

There are two phases in the E-AKA protocol:

- **i.** Registration and distribution of authentication information (Initial Authentication) and temporary key (***TK***) from the ***HLR/AuC*** to the ***VLR/SGSN***.
- **ii.** The authentication and key agreement procedure (Subsequent authentication) run between the ***MS*** and the ***VLR/SGSN***.

Figure 3 and 4 describe authentication mechanism for E-AKA protocol as follow:

1. When the ***MS*** moves to new ***VLR/SGSN*** area then registration and distribution of authentication information (i.e., initial authentication) as carried as follow:
    - **i.** ***MS*** generates random number $(Rand_{MS})$
    - **ii.** ***MS*** computes the Message Authentication Code (MAC_{MS}). $MAC_{MS}=f_1(K, Rand_{MS})$
    - **iii.** ***MS*** sends ***IMS***, $Rand_{MS}$ and MAC_{MS} as authentication request to ***VLR/SGSN***.
2. In this stage the ***VLR/SGN*** is unable to authenticate the ***MS*** by itself; therefore ***VLR/SGSN*** passes this authentication request to ***HLR/AuC***.
3. ***HLR/AuC*** receives the authentication request, and then verification procedure is performed by ***HLR/AuC***. A response message is generated. The following operations are carried by ***HLR/AuC***:
    - **i.** ***HLR/AuC*** computes expected message authentication code for mobile station $(XMAC_{MS})$ to verify the received message.

      $XMAC_{MS}=f_1(K, Rand_{MS})$
    - **ii.** Comparing the computed $XMAC_{MS}$ with received MAC_{MS}.

      $XMAC_{MS} ?= MAC_{Ms}$

      If mismatching occurred then registration failed else goes to next step.
    - **iii.** ***HLR/AuC*** generates SQN_{HLR} and $RAND_{HLR}$.
    - **iv.** ***HLR/AuC*** computes $XRES_{HLR} = f_2(K,RAND_{HLR})$, Anonymity Key $AK_{HLR} = f_5(K,RAND_{HLR})$, Message Authentication Code $MAC_{HLR} = f_1(K,SQN \| RAND_{HLR} \| MAF)$, where ***MAF*** is Message Authentication Field and $AUTN_{HLR} = (SQN \oplus AK_{HLR} \| AMF \| MAC_{HLR})$ where $\oplus$ is exclusive OR operation.
    - **v.** ***HLR/AuC*** computes temporary key $TK = f_x(K,RAND_{HLR})$.
    - **vi.** ***HLR/AuC*** generates one authentication vector $AV(1..n)$ (i.e., in the proposed protocol the ***AV*** contains one record), and sends authentication data response ***AV*** to ***VLR/SGSN***. This ***AV*** consists of four components: random number ($RAND_{HLR}$), expected response ($XRES_{HLR}$), temporary key (***TK***) and authentication token ($AUTN_{HLR}$).

      $AV = RAND_{HLR} \| XRES_{HLR} \| TK \| AUTN_{HLR}$
4. ***VLR/SGSN*** receives the response from ***HLR/AuC***. The ***VLR/SGSN*** executes the following operations:
    - **i.** ***VLR/SGSN*** stores the Temporary key ***TK***, $AUTN_{HLR}$ and generates random number $Rand_{VLR}$.
    - **ii.** ***VLR/SGSN*** computes $MAC_{VLR}=f_1(TK, MAC_{HLR} \| Rand_{VLR})$ where the MAC_{HLR} retrieved from $AUTN_{HLR}$ which stored in previous step.
    - **iii.** ***VLR/SGSN*** computes $AUTN_{VLR} = (SQN_{HLR} \oplus AK_{HLR} \| AMF \| MAC_{VLR})$
    - **iv.** ***VLR/SGSN*** sends $AUTH_{VLR}$, $Rand_{VLR}$ and $Rand_{HLR}$ to ***MS***
5. ***MS*** authenticates ***VLR/SGSN*** and ***HLR/AuC*** and generates response information. Upon receipt of $AUTN_{VLR}$, the ***MS*** authenticates ***HLR/AuC*** and ***VLR/SGSN***. ***MS*** computes and retrieves the following:
    - **i.** Computes the temporary key $TK=f_x(K, Rand_{HLR})$.
    - **ii.** The ***MS*** verifies that the received sequence number ***SQN*** is in the correct range. If the ***MS*** considers the sequence number to be not in the correct range, it sends synchronization failure back to the ***VLR/SGSN*** including an appropriate parameter, and abandon the procedure.

   - **iii.** Computes expected message authentication code for ***HLR*** and ***VLR***. $XMAC_{HLR} = f_1(K, AK_{MS} \oplus (SQN_{HLR} \oplus AK_{HLR}) \| Rand_{HLR} \| AMF)$ where $Rand_{HLR}$ and ***AMF*** are retrieved from $AUTN_{VLR}$
   - **iv.** Computes $XMAC_{VLR} = f_1(TK, XMAC_{HLR} \| Rand_{VLR})$. If $XMAC_{VLR}$ is equal $XMAC_{VLR}$ then ***HLR/AuC*** and ***VLR/SGSN*** are valid, the ***MS*** computes an expected response message $XRES = f_2(TK, Rand_{VLR})$
   - **v.** The ***MS*** sends ***XRES*** to ***VLR/SGSN***. While, the ***MS*** computes an integrity key as $IK = f_3(TK, Rand_{VLR})$ and a cipher key as $CK = f_4(TK, Rand_{VLR})$ to realize securely communication with ***VLR/SGSN*** subsequently.

6. ***VLR/SGSN*** compares the received ***RES*** with ***XRES***. ***VLR/SGSN*** authenticates the ***MS*** by verifying the following equation $XRES \mathrel{?=} RES = f_1(TK, Rand_{VLR})$. If they match, then authentication is successfully completed and ***VLR/SGSN*** computes integrity key as $IK = f_3(TK, Rand_{VLR})$ and a cipher key as $CK = f_4(TK, Rand_{VLR})$ to realize securely communication with MS subsequently

After the initial authentication, both the ***VLR/SGSN*** and ***MS*** obtain the authentication result from the ***HLR/AuC*** and share some secret information without intervention of ***HLR/AuC***. Here, the ***VLR/SGSN*** caches some authentication information, which can be used in subsequent authentication.

After initial authentication, the ***VLR/SGSN*** has the ability to authenticate the MS in subsequent authentication. If the ***MS*** remains in the same ***VLR/SGSN*** and requests services, then the user should ask for subsequent authentication. ***MS*** similarly generates an authentication request message, which should contain the information shared between the ***MS*** and ***VLR/SGSN***; the ***VLR/SGSN*** uses this information to authenticate the ***MS***. ***VLR/SGSN*** authenticates ***MS*** by using temporary key ***TK***.

As mentioned above, the ***VLR/SGSN*** has cached information needed to authenticate ***MS***. After authenticating the ***MS***, the ***VLR/SGSN*** sends a response message containing the authentication result to the ***MS***. The ***MS*** receives the response message and learns whether the authentication was successful or not.

The subsequent authentication occurs as the following:

1. ***MS*** sends ***TMSI*** to ***VLR/SGSN***
2. ***VLR/SGSN*** generates $Rand_{VLR}$
3. ***VLR/SGSN*** computes authenticate token $AUTN = SQN \oplus AK \| AMF \| MAC$ Where

   $AK = f_5(TK, RAND)$,

   $MAC = f_1(TK, SQN \| RAND \| MAF)$.
4. ***VLR/SGSN*** sends ***AUTN*** and ***RAND*** to ***MS***.
5. ***MS*** computes and retrieves the following:
   - **i.** Anonymity key $AK = f_5\ (TK, Rand)$, $SQN = ((SQN \oplus AK) \oplus AK)$, computes expected message authentication code $XMAC = f_1\ (SQN, RAND, AMF)$ and then,
   - **ii.** Compares ***XMAC*** with ***MAC*** which is included in ***AUTN***. If ***XMAC*** is not equal to ***MAC*** then ***MS*** sends failure message to the ***VLR/SGSN***, else if ***XMAC*** is equal ***MAC*** then ***MS*** checks that the received ***SQN*** is in the correct range i.e. $SQN > SQN_{MS}$. If ***SQN*** is not in the correct range then ***MS*** sends failure message to the ***VLR/SGSN***, else if it is in the correct range, then ***MS*** computes the Response $RES = f_2\ (TK, RAND)$, and $CK = f_3\ (TK, Rand)$,
   - **iii.** After that, it sends ***RES*** to ***VLR/SGSN.***
6. ***VLR*** compares the received ***RES*** with ***XRES***. If they match, then authentication is successfully completed.

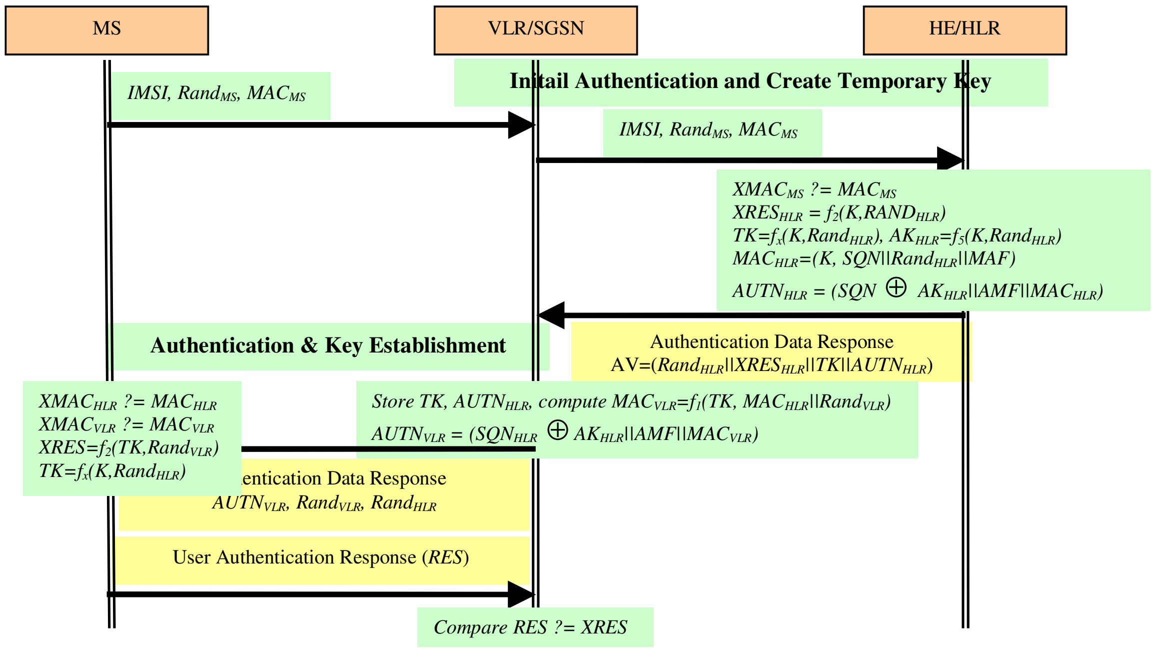

Figure 3 Registration and distribution of authentication information (Initial Authentication) in EAKA

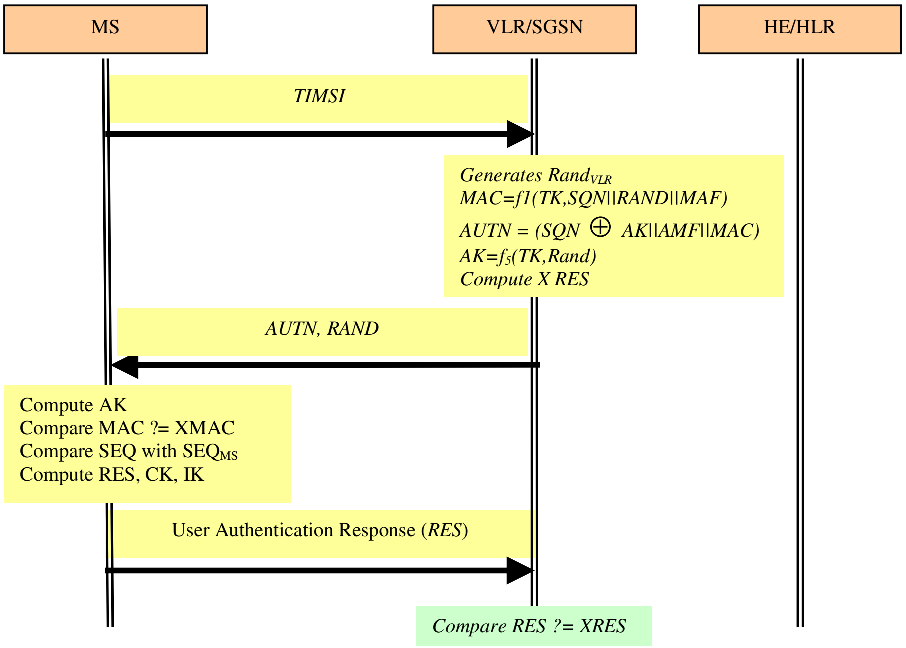

Figure 4 Subsequent authentications in EAKA

6. Security analysis

In order to ensure that the proposed protocol is secure, the attack methods will be analyzed and discussed. The proposed E-AKA fulfils all of the security requirements for UMTS AKA protocol include: mutual authentication, data integrity and data confidentiality and against various attacks.

The proposed E-AKA has enhanced security compared to UMTS AKA. In the first procedure, the distribution and establishment of authentication vector, all the principals (the ***MS***, the ***VLR/SGSN*** and the ***HLR/AuC***) contribute the generation of the temporary key ***TK*** and random numbers for all entities in UMTS. While in UMTS AKA, the ***AV***, and random number are generated only by the ***HLR/AuC***. In the second procedure, to achieve mutual authentication between the ***MS*** and the ***VLR/SGSN***, a ***Rand***$_{VLR}$ is sent to the ***MS*** to compute ***XRES***.Thus, a replay attack is prevented because the ***SEQ*** and ***Rand***$_{VLR}$ is not accepted by the ***MS*** and ***VLR/SGSN***, if has been used before.

By using the sequence numbers, the user is ensured that the authentication information ***RAND*** and ***AUTN*** cannot be reused by an adversary.

The user, however, can verify if an authentication vector was generated by the home network. Also the user determines if an authentication vector was requested by the ***VLR/SGSN***.

The response from the ***MS*** is generated based on the contributions of all principals. Hence, mutual authentication is stronger.

The ***TK*** helps the ***VLR/SGSN*** and the ***MS*** to generate the cipher key ***CK*** and the integrity key ***IK***.

5.1 Mutual Authentication: AKA provides procedures for mutual authentication of the ***MS*** and serving system [23]. It is clear that the proposed authentication protocol can authenticate ***MS, HLR/AuC*** and ***VLR/SGSN***. Meanwhile, the UMTS protocol has no mechanism to authenticate ***MS***.

The ***HLR/AuC*** verifies if the mobile user is a legitimate user and checks the correctness of the authentication message code ***MAC***$_{MS}$ through the shared secret key ***K***. If it fails, the ***HLR/AuC*** rejects the authentication request. Otherwise, the ***HLR/AuC*** succeeds the mobile user authentication.

The ***MS*** signs the message (i.e. $MAC_{MS} = f1(K,Rand_{MS})$) by using a secrete key and then sends it to the ***HLR/AuC***. The ***HLR/AuC*** confirms the identity of the ***MS*** by verifying the signed message by using the ***MS's*** secrete key. Therefore, authentication between the ***MS*** and the ***HLR/AuC*** can be achieved by using the secrete key. Consequently, mutual authentication is achieved in the proposed protocol, the ***MS*** can decrypt the message which it has received ***AUTH***$_{VLR}$ and verifies ***MAC***$_{VLR}$, ***MAC***$_{HLR}$. Therefore, the ***MS*** confirms the authenticity of the ***VLR/SGSN*** and ***HLR/AuC*** together. After the initial authentication during the origination and termination call, the ***VLR/SGSN*** gets a secret temporary key ***(TK)*** that it shares with the ***MS*** and subsequently can accomplish the mutual authentication by itself.

5.2 Temporary key

The temporary key is used in proposed E-AKA protocol to reduce the traffic between ***HLR/AuC*** and ***VLR/SGSN***.

The proposed protcol used f_x key generation function to generate temporary key. This functions produce a 128 bits or higher bits to provide high level of security. While the key generation function f_5 which used by UMTS AKA produce only 48-bit hash results, which is not sufficent security level.

The temporary key ***TK*** is generated as $TK = fx(K, Rand_{HLR})$. The ***Rand***$_{HLR}$ is transmitted in cleartext and the key generation function f_x is public, the temporary key cannot be generated without secret ***K*** owned by ***MS*** and ***HLR/AuC***.

In addition, ***HLR/AuC*** transmits the temporary key to ***VLR/SGSN*** via a secure channel. Therefore, ***VLR/SGSN*** owns the temporary key to authenticate ***MS*** on behalf of the ***HLR/AuC***.

In the second procedure of the E-AKA protocol, the temporary key is used for generating the authentication information using the cryptographic algorithms f_1 and f_2. If an intruder snoop this authentication information and reverses them to obtain the temporary key, it is very difficult because both f_1 and $f_2$are a one way function.

5.3 Integrity and Confidentiality: the signalling information that is sent between ***MS*** and the ***HLR/AuC*** are sensitive and must be integrity and confidentiality protected. The message authentication function integrity is used to be applied the signalling information elements transmitted between the ***MS, VLR/SGSN*** and ***HLR/AuC.***

The cipher key ***CK*** is used as input parameter for the ciphering algorithm ***f8*** to encrypt the plaintext transmitted between the ***MS, VLR/SGSN*** and ***HLR/AuC*** to provide the confidentiality.

The proposed protocol provides data integrity and origin authentication of signalling data. The receiving entity (i.e., ***MS, VLR/SGSN***) is able to verify that signalling data has not been modified in an unauthorised way since it was sent by the sending entity (.e., ***MS, VLR/SGSN***) and that the data origin of the signalling data received is in fact the one claimed;

In the proposed protocol the user is identified by a temporary identity by which is known by ***VLR/SGSN*** like UMTS. Therefore; the ***IMSI*** of user, location of user cannot be eavesdropped on the radio access. In the proposed protocol to avoid user traceability and to provide confidentiality, the ***TIMSI*** is not allowed to use it for long time like UMTS. Privacy extends to the radio network controller (RNC) for user traffic confidentiality like UMTS AKA, but after the RNC, data will be decrypted and transmitted in a plaintext form over the networks. This is done by using the ciphering algorithm f_8 to encrypt the plaintext by using cipher key ***CK***. Therefore, the attacker is not able to get any sensitive data.

The integrity service in the proposed protocol was achieved by using the integrity algorithm f_9 with ***IK*** to authenticate the data integrity of signalling message like UMTS AKA. Therefore, throughout the entire authentication process the information exchanged between entities of the network cannot be altered without detection.

5.4 Minimize resource utilization: the proposed protocol satisfies this requirement by reducing the total of signalling between entities and decreasing the size of messages. Consequently, the delay time and bandwidth is minimized.

5.5 Replay attacks: the attacker cannot forge an authentication data request message because this message should comprise a message authentication code from the ***MS***.

The proposed protocol can prevent the replay attack by the freshness of its process. The ***MS*** generates $Rand_{MS}$, which is an unpredictable random number. The ***HLR/AuC*** and ***VLR/SGSN*** generates $Rand_{HLR}$ and $Rand_{VLR}$. These random numbers appear in the $AUTH_{HLR}$, $AUTH_{VLR}$ and ensure the freshness of the authentication vector (***AV***). As well as the (***AV***) including authentication key as temporary key ***TK***. It refreshes the session key by using the Random number to ensure the freshness of authentication sessions. Thus the replay attack fails.

5.6 Redirection attacks: In UMTS AKA the redirection attack is possible because the authentication vector can be used by any serving network. In the proposed protocol this attack is not available. When the ***HLR/AuC*** receives the authentication data request message, it can check if that ***MS*** is really in the coverage of the supposed ***VLR/SGSN***. If it is not, the ***HLR***/AuC rejects the connection request. While the ***MS*** receives the authentication token $AUTN_{VLR}$, ***MS*** can check if they are really sent by the supposed ***HLR/AuC*** and ***VLR/SGSN*** through MAC_{VLR}, MAC_{MS}, ***and MACHLR*** since all three entities contribute to the generation of ***MAC***. Similarly, for each mutual authentication between the ***MS*** and the ***VLR/SGSN***, all entities also contribute to the generation of challenge/response message. Hence, E-AKA is preventing the redirection attack.

7. A Comparison with Related work

The Comparisons of the proposed Protocol with the current UMTS protocol and related works are presented in table 1.

The UMTS AKA and AP-AKA uses ***AVs*** to decrease the number of access to a ***HLR/AuC***. But, the use of ***AVs*** causes bandwidth consumption and storage overhead in ***VLR/SGSN***.

Harn and Hsin AKA and X-AKA protocol use hash chain. Using of several hash chains by causes bandwidth consumption and overhead.

In UMTS AKA ***HLR/AuC*** cannot authenticate ***MS***. While the proposed protocol, allows ***HLR/AuC*** to authenticate the ***MS*** by verifying the MAC_{MS}.

The VC-AKA protocol would bring more traffic than UMTS AKA tn the first procedure of authentication, because it needs more protocol rounds to establish the authentication vectors in ***VLR/SGSN*** and the ***MS***. While the E-AKA protocol decrease exponentially the traffic between the ***MS, VLR/SGSN*** and ***HLR/AuC***.

An analytic model is proposed to investigate the impact of the size of the authentication vector array in order to minimize the cost [7]. A dynamic length of authentication vector array based on prediction of the mobile user's residence time in the ***VLR/SGSN*** is proposed by Al-Saraireh and Sufian. Consequently, it is able to reduce the network traffic and to avoid the bottleneck at ***HLR/AuC***. There are differences between these works and the proposed E-AKA, because these works do not change the original UMTS AKA protocol and tries to find a suitable size for the authentication vector.

Table 1: A Comparison among the UMTS AKA Protocols

Comparison Items	UMTS	Harn& Hsia	AP-AKA	X-AKA	EAKAP	AL-Saraireh [2]	Lin	E-AKA
Mutual authentication Between MS and HE	No	No	No	No	Yes	No	No	Yes
Mutual authentication Between MS and SN	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
User traffic Confidentiality	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Signalling data integrity	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Reduction of bandwidth consumption between SN and HE	No	No	No	Yes	No	Yes	Yes	Yes
Reduction of Storage space for SN's database	No	Yes	No	Yes	No	Yes	Yes	Yes
Need Synchronization between MS and HE	Yes	No	No	No	Yes	Yes	Yes	Yes
Use Temporary Key	No	No	No	Yes	No	No	No	Yes
HN Involved in each Authentication Data Request	Yes	Yes	Yes	No	Yes	Yes	Yes	No
Use of AVs	Yes	No	Yes	No	Yes	Yes	Yes	No

8. Conclusion

The main advantage of E-AKA is that it releases the ***HLR/AuC*** from the bottleneck of authentication vectors generation and eliminates the traffic bottleneck.

The proposed authentication protocol has improved the performance of authentication by reducing the authentication times, setup time and data sizes. Also, the proposed authentication mechanism has less signalling traffic and consequently the bottleneck at authentication centre is avoided significantly, by reducing the number of messages between mobile and authentication centre.
The proposed authentication for UMTS has been generated while keeping in mind that the complexity of this function is as low as possible while keeping a high

The security analysis showed that E-AKA can defend against set of attacks. Also, E-AKA provides enhanced security by using temporary key. Through comparison with UMTS AKA and its improvements in literature, this work showed that E-AKA is more efficient and secure.

In addition, the bi-unilateral and mutual authentication among ***MS***, ***VLR*/SGSN** and ***HLR/AuC*** have been adopted that resulted in a more secure protocol than the other available authentication protocols. The proposed protocol fulfils the security requirements of the third generation mobile systems.

The proposed protocol achieved the following goals:

1. Provides mutual authentication between the user ***MS*** and the ***HLR/AuC***.
2. Provides mutual authentication between the user ***MS*** and the ***VLR/SGSN***.
3. The establishment of a cipher key and an integrity key upon successful authentication.
4. Reduces the signalling traffic between serving network and home network and reduces the size of authentication information to be stored in the serving network.
5. ***HLR/AuC*** allows ***VLR/SGSN*** to authenticate ***MS***, then ***VLR/SGSN*** authenticates ***MS*** without any intervention from the ***HLR/AuC***

References

[1] Al-Saraireh J., and Yousef S. (2006), "A New Authentication Protocol for UMTS Mobile Networks", *EURASIP Journal on wireless communications and networking,* vol. 2006. Pages(s): 1-10, Article ID 98107.

[2] Al-Saraireh J., and Yousef S. (2007), "Analytical Model: Authentication Transmission Overhead Between Entities in Mobile Networks", *Elsevier, Computer Communications Journal*, vol. 30, no. 9. Page(s):1713-1720.

[3] 3GPP TS 33.102 V8.0.0 (2008-06), "3GPP Technical Specification Group Services and System Aspects, 3G Security, Security Architecture (Release 8)", *3rd Generation Partnership Project.*

[4] Salgarelli L., Buddhikot M., Garay J., Patel S., and Miller S. (2003), "The Evaluation of wireless LANs and PANs – Efficient Authentication and Key Distribution in Wireless IP Networks". *IEEE Personal Communication on Wireless Communication*, vol. 10, no. 6. Page(s):52-61.

[5] Zhang M., and Fang Y. (2005), "Security Analysis and Enhancements of 3GPP Authentication and Key Agreement Protocol", *IEEE Transactions on wireless communications*, vol. 4, no. 2, Page(s): 734– 742.

[6] Huang Y., Shen Y., Shieh S., Wang H., and Lin C. (2009), "Provable Secure AKA Scheme with Reliable Key Delegation in UMTS ", *Secure Software Integration and Reliability Improvement, 2009. SSIRI 2009. Third IEEE International Conference,* vo. 2009, Page(s): 243 - 252.

[7] Lin Y, and Chen Y. (2003), "Reducing Authentication Signaling Traffic in Third-Generation Mobile Network", *IEEE Transactions on Wireless Communications*, vol. 2, no. 3, Page(s): 493-501.

[8] Cheng S., Shieh S., Yang W., Lee F., and Luo J. (2005), "Designing Authentication Protocols for Third Generation Mobile Communication Systems", *Journal of Information Science and Engineering*, vol. 21, Page(s): 361-378.

[9] Horn G., Martin K., and Mitchell C. (2002), "Authentication Protocols for Mobile Network Environment Value-Added Services", *IEEE Transactions on Vehicular Technology*, vol. 51, no. 2, Page(s): 383-392.

[10] Lee C., Li L., and Hwang M. (2002), "A remote User Authentication Scheme Using Hash Function", ACM Operating Systems Review, vol. 36 no. 4, Page(s): 23-29.

[11] Lee C., Hwang M., and Yang W. (2003), "Extension of Authentication Protocol for GSM", *IEE Proceeding Communication*, vol. 150, no. 2, Page(s): 91-95.

[12] Lin C., and Shieh S. (2000), "Chain authentication in mobile communication systems", *Journal of Telecommunication Systems*, vol. 13, Page(s): 213-240.

[13] Looi M. (2001),"Enhanced authentication services for internet systems using mobile networks", *IEEE Global Telecommunications Conference*, vol. 6, Page(s) 3468-3472.

[14] Huang C., and Li J. (2005), "Authentication and Key Agreement Protocol for UMTS with Low Bandwidth Consumption", *AINA2005*, *19th International Conference on Advanced Information Networking and Applications (AINA'05)*, vol. 1, Page(s): 392-397.

[15] Zhang M. (2003), "Provably-Secure Enhancement on 3GPP Authentication and Key Agreement Protocol", *Cryptology ePrint Archive, Report 2003/092, 2003*. [online]. Last accessed on 10 Feb 2011 as Available at http://eprint.iacr.org.

[16] Zhang Y., and Fujise M. (2006b), "Security Management in the Next Generation Wireless Networks", *International Journal of Network Security*, vol.3, no.1, Page(s):1-7

[17] Adi W., Dawood A., Mabrouk A., and Musa S. (2007), "Low complexity image authentication for mobile applications", *IEEE South East Conference*, Richmond, USA. Page(s): 20-20.

[18] Yijun H., Nan X., and Jie L. (2007), "A Secure Key Exchange and Mutual Authentication Protocol for Wireless Mobile Communication", *IEEE International Conference on Availability, Reliability and Security, ARES'07*, Vienna, Austria, Page(s): 558 – 563.

[19] Harn L., and Hsin W. (2003), “On the security of wireless network access with enhancements,” *in Proceedings of the 2003 ACM workshop on Wireless security*, San Diego, USA, Sep. 19 2003 Page(s): 88–95.

[20] Juang W.S., and Wu J.L. (2007), "Efficient 3GPP Authentication and Key Agreement with Robust User Privacy Protection", IEEE Communications Society, Proceedings of the WCNC, 2007.

[21] Farhat F., Salimi S., and Salahi A. (2009), "An Extended Authentication and Key Agreement Protocol of UMTS", *Information Security Practice and Experience, Lecture Notes in Computer Science, 2009*, vol. 5451/2009, Page(s): 230-244, DOI: 10.1007/978-3-642-00843-6_21

[22] Daeyoung K., Younggang C., Sangjin K. and Heekuck O. (2007), "A Privacy Protecting UMTS AKA Protocol Providing Perfect Forward Secrecy ",*Computational Science and Its Applications – ICCSA 2007, Lecture Notes in Computer Science*, 2007, vol. 4706/2007, Page(s): 987-995, DOI: 10.1007/978-3-540-74477-1_88

[23] Shankar R., Timothy Rajkumar K., and Dananjayan P. (2010), "Security Enhancement With Optimal QoS Using EAP-AKA In Hybrid Coupled 3G-WLAN Convergence Network", *International Journal Of UbiComp (IJU),* vol.1, No.3, July 2010, DOI : 10.5121/iju.2010.1303 31.

[24] Min-Shiang H., Song-Kong C., and Hsia-Hung O. (2010), "On the security of an enhanced UMTS authentication and key agreement protocol", *European Transactions on Telecommunications*. DOI: 10.1002/ett.1460

Acknowledgement

Dr. Ja'afer AL-Saraireh is grateful to Applied Science Private University Amman, Jordan, for the full financial support granted to this research project.