International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Dynamic Authentication Protocol for Mobile Networks Using Public-Key Cryptography

Mustafa AL-Fayoumi[1], Mohammed Nababteh[2], Mohammad Sh. Daoud[3], Mohammad Alhawarat[1]

[1]College of Computer Engineering and Sciences, Salman bin Abdulaziz University,
Computer Science Department, Al-Kharj, Saudi Arabia

[2]FESA University, UNRWA, Amman, Jordan

[3]Al Ain University of Science and Technology, Abu Dhabi, UAE

Abstract: ***The authentication and key agreement (AKA) protocol of Universal Mobile Telecommunication System (UMTS) is still vulnerable to redirection attack which allows an adversary to redirect user traffic form a network to another and eavesdrop or mischarge the subscribers in the system. Moreover, the International Mobile Subscriber Identity (IMSI) which uniquely identifies a user, is still reveal to the visited network and can still be demanded by an attacker who impersonates a base station, as there is no network authentication in this case, and the non-repudiation services requirement which provide the protection for the subscribers from incorrect bill charging, and the service providers with legal evidence when collecting the bills, are two important points in the non-repudiation requirement. In this paper, a dynamic authentication protocol by integrating the public-key cryptography with the hash-chaining technique is presented to significantly improve the security level as well as to improve the performance.***

Keywords: Mobile Security, 3G Mobile Network Security and Authentication, public-key, cryptography**.**

1. Introduction

In order 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]. 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.

There are three entities participating in the UMTS security architecture, home environment (HE), serving network (SN) and mobile station (MS). Figure 1 illustrates the UMTS architecture. 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 the SGSN handles the packet switched traffic [2].

Authentication procedure is executed when the MS moves from one registration area (RA) to another one (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 module (USIM). When the LAI is different than the MS executes authentication procedure.

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 codesf_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***[3-5]. 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 zero [4].

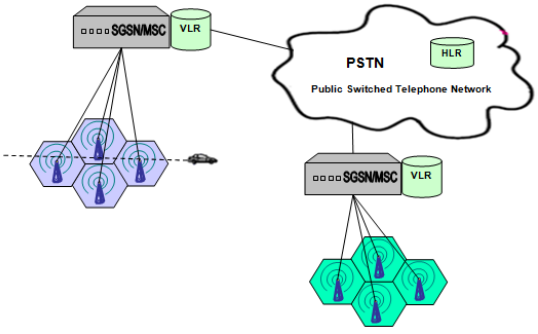

Figure 1: UMTS Architecture

There are three goals for the UMTS AKA: a mutual authentication between the user and the network; an establishment of a cipher key and an integrity key upon successful authentication; and a freshness assurance to the user of the established cipher and integrity keys. There are two phases in AKA protocol [1]:

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

Figure 2 describes authentication mechanism as follows:

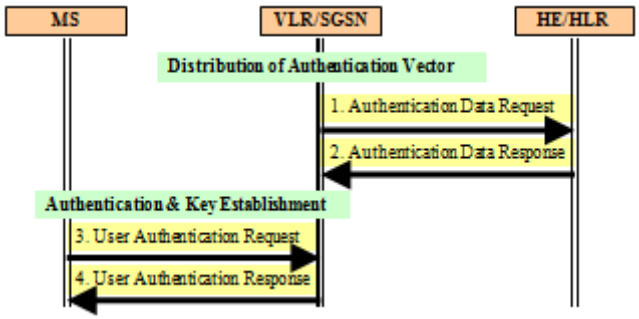

Figure 2: Authentications and Key Agreement Protocol

- When the MS moves to new ***VLR/SGSN*** area then MS sends ***(IMSI)*** authentication request to ***VLR/SGSN***.
- ***VLR*** passes this authentication request to ***HLR***.
- ***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:
  - ***HLR/AuC*** generates SQN_{HLR} and ***RAND***.
  - ***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*** isMessage Authentication Field and ***AUTN*** = $(SQN \oplus AK||AMF||MAC)$ where $\oplus$ is exclusive OR operation.
  - $HLR/AuCSQN_{HLR}$ is increased by 1.
- ***VLR*** stores authentication vectors. In thei^{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.
- ***MS*** computes and retrieves the following:
  - 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,
  - 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 ***(K, RAND)***, and ***CK*** = f_3 ***(K, Rand)***,
  - After that, it sends ***RES*** to ***VLR/SGSN.***
- ***VLR*** compares the received ***RES*** with ***XRES***. If they match, then authentication is successfully completed.

2. Related Works

Several authentication schemes have been proposed for mobile networks to enhance the security of mobile communication systems based on several authentication techniques. These techniques only provide some security features and have some weaknesses. Firstly, some of these schemes are based on the use of symmetric key cryptosystems and a challenge-response exchange. In this context, Many symmetric key basedAKAprotocols [6-9] were proposed forUMTS network to improve the security of UMTSAKAandeffective utilization of bandwidth during the authentication.

However, the UMTS authentication protocol still has a security problem. It allows an adversary to redirect user traffic from one network to another. The redirection attack would cause billing problem. For instance, the user is in the territory of his home network but gets charged by a foreign network based on rate higher than that offered by the home network. Zhang and Fang [10] proposed a new protocolnamelyAP-AKA, to defeat the redirection attack and intensely inferior the effect of corruptednetwork.

However, both UMTS-AKA and AP-AKA protocols have 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. Yet with this improvement, Lin and Chen [11] and AL-Saraireh and Yousef [12] are still there are bandwidth consumption. Unfortunately, the performance drawbacks still strike as follows. First, the space overhead strikes when n AVs in the SN are being stored. Second, there is bandwidth consumption between SN and HN since HN needs to pass n AVs to SN. The two problems can be solved by several techniques.

Harn and Hsin[13] proposed an enhanced registration and AKA scheme for UMTS. By introducing a combination of hash-chaining and keyed HMAC techniques, in their proposed protocol they claim it can provide strong periodically mutual authentication, strong key agreement, and a non-repudiation service in a simple and elegant way. However, due to the underlying hash chaining technique [14], the security was enhanced while more computation overhead of hash chaining was incurred at MS and SN in each session. This could have a negative impact on the performance of this protocol. However,this protocol also does not clear the security issues against various attacks.

X-AKAprotocol [15] was proposed an extension of the UMTS-AKA protocol to prune off the transmission of authentication vectors (AV) and improves its bandwidth utilization. In X-AKA, SN must continually generate random numbers to challenge MS to reply corresponding responses for every authentication. It is noticeable, random challenge generation overhead occurs in SN.
Both Harn-Hsin's and Huang-Li's protocols, the security for MS to identify the active SN is not mentioned so that an adversary can redirect the user traffic from the active SN to

another SN. The redirection and man-in-the-middle attacks are not prevent in the two protocols.

Al-Saraireh and Yousef's protocol [8] primary emphasis on reducing the bandwidthfor transmitted authentication vectors during authentication and therefore, the AVs areonly generated by the MS instead of by the VLR. Al-Saraireh and Yousef's protocol eliminates the cost of delivering AVs . the protocol doesnot clear the security issues with redirection as well as man-in-the-middle attacks.

Ou, Hwang, andJan [16] proposed a new protocol COCKTAIL-AKA, to overcome the congenital defects of UMTSAKAprotocol. In this protocol, each service network produces its own AVs (MAVs) in advance. These MAVs are produced only once but can be reused later. While authenticating the MS, the HLR/AuC calculates a private authentication vector (PAV) for MS. The PAV is transferred to the SGSN. Then, the SGSN uses the PAV and MAV to generate several effective AVs for subsequent authentications. Cocktail-AKA is penetrable to DoS attack and impersonation attack [17]. It alsodoes not solve the synchronization problem between MS and HLR.

Huang and et al [18] proposed a new protocolnamelyS-AKA, to defeat the redirection, man-in-the-middle and denial of service attacks. However, the S-AKA reduces bandwidthconsumption up to 38%and also decreases the number of messages required in authenticating mobile subscribers. In S-AKA, SN must continually generate random numbers to challenge MS to reply corresponding responses for every authentication. It is noticeable, random challenge generation overhead occurs in SN.TheNS-AKA protocolin [6] reduces the overheads, and is free from redirection andMITMattacks, but does not provideresistance against denial of service attack.

Secondly, other of these schemes are based on the use of asymmetric key cryptosystems. The public key certificates and timestamps are combined to provide user identity confidentiality and unilateral entity authentication in a single mechanism. In this context, Many asymmetric key basedAKAprotocols were proposed forUMTS network. Asymmetric cryptography in UMTS networks is proposed by Grecas, et al. [19]. This method consists of the introduction of public-private key pairs for the transactions between the VLR and HLR, as well as the MS and VLR. However, according to specifications that define GSM, GPRS and UMTS, there is no mutual authentication between the VLR and HLR, and no data encryption takes place when these two nodes communicate.A novel asymmetric end to end authentication protocol that is based on the concept of using the wireless access home network of a mobile station to assist its authentication with a service provider is presented by He and Zhang [20]. Jun and Chen [21] presented a novel mutual authentication and key agreement protocol based on the Number Theory Research Unit (NTRU) public key cryptography. The symmetric encryption, hash and "challenge-response" techniques were adopted to build their protocol.

Gódor and Imre [22] suggested a novel authentication algorithm (GSZV) based on public key infrastructure by using digital signatures, certifications, and two different sequence numbers. The main goal of that algorithm is to guarantee a secure and confidential communication between the users and the network. In that algorithm all information, including the IMSI of MS is encrypted on the air interface which is needed since if the IMSI gets known an attacker can misuse it.

Yeh and Lee [23] suggested a dual-purpose signature for authentication on UMTS which provides valuable improvements to UMTS by using the digital signature technique to reduce the storage needed at the HLR and guarantees the access rights of the mobile station (MS). The Dual-Purpose signature concept provides an alternative application for signature technique in an efficient way. With the suggested method, the UMTS will benefit from the elimination of bulky storage and face fewer security threats.

3. Framework for Proposed Protocol

To enhance the 3G AKA protocol, the proposed authentication protocol has adopted three major techniques: digital signature, Message Authentication Code (MAC) and hash chaining. Public key cryptography has not previously been used in mobile communication environments due to performance constraints. It was not consider suitable for second generation systems because of the resulting length of messages and the necessary computational loads. New protocols for authentication between user and network have been developed to overcome these problems. The proposed protocol is based on a digital signature cryptography scheme. A true non-repudiation service among HLR, VLR and MS can only be achieved via a public-key system using digital signatures [13]. A digital signature can be used in a public-key system to replace HMAC.

One-way function is a variation of the message authentication code as with the message authentication code, a hash function accepts a variable size message M as input and produces a fixed size output, referred to as a hash code *H(M)*. The hash code is a function of all the bits of the message and provides an error detection capability. When it changes any bits in the message result in a change to the hash code, a hash function *H* has some properties [24].

The proposed protocol uses a one-time password/hash-chaining technique which was proposed by Lamport [14]. It used a hash function with one-way property to construct a sequence of hashing value. They designed it in a remotely accessed computer system. One of the aims of the one-way hash function is to prevent eavesdroppers discovering the password and to reduce the computing time, which this technique has used in many applications [1, 13].

In this method, let the user (claimant) and the server (verifier) deal with the secret(M) as a seed of hash value and $f(M)$ be a one-way function, when a user (i.e., the one wishes to be authenticated) wants to register or log in the system, then the user should construct$f^n(M) = f\left(f\left(\ldots(f(M)\ldots)\right)\right)$, where$n$ represents the maximum number of services that the user can request after the registration phase (i.e., the composition ofnfs), and sends

$f^n(M)$to the server (i.e., the one decides whether the user is who it is). Then the server uses it to compute a sequence of passwords $f^{n-1}(M), f^{n-2}(M), \dots, f\left(f(f(M))\right), f(f(M)), f(M)$and the server stores those. The user holds$f^n(M), \dots, f\left(f(f(M))\right), f(f(M))$.

After the registration is completed, each hash chain can be used by the claimant to prove itself to the server Ntimes. In the j^{th}session, the user provides$f^{n-1}(M)$to ask for a connection to prove itself. The server can verify the correctness of$f^{n-1}(M)$ by means of the one way function by computing$f\left(f^{n-1}(M)\right)$and the server needs to store $f^{n-1}(M)$as the last value of user to authenticate the next visit. So, the user reveals$f^{n-1}(M), f^{n-2}(M), \dots, f(M)$, and $M = f^0(M)$ in sequence to prove itself $n$times. In this way $f^{n-j}(M)$ can be used as a proof of the j^{th}connection.

The proposed protocol can satisfy this requirement by considering the requirement of non-repudiation. This is achieved by using the digital signature during registration to provide a nonce random number $(NONCE_{MS}, NONCE_{HN})$ which can be used to construct the secret seed$M$ of hash chaining function$f^n(M)$ dynamically in both of MS and VLR/SGSN in the visited network. So, by combination of $f^{n-J}(M)$and signature$Sig(NONCE_{MS})$which lead to construct$f^n(M)$then can be achieve non-repudiation proof by the VLR/SGSN as an evidence for all $n$visits made by the MS. Specifically, for all $n$visits, the VLR/SGSN only needs to store the most recently released $f$value (i.e., $f^{n-j}(M)$), and does not need to keep all other values that it has received (i.e., $f^n(M), f^{n-1}(M), \dots, f^{n-j+1}(M)$) before the i^{th}visit. The VLR/SGSN can produce a proof of the claimant'sj^{th} visit, where$1 \le j \le n-1$, by simply computing$f^{n-j}\left(f^{N-n}(M)\right)$. This feature is especially good for applications with limited storage space such as mobile handsets. To extend the life time of a hash chain, an additional dimension can be added to the above technique as follows:

The claimant (MS) and verifier (VLR) cooperate to construct index(idx) seeds$M_1, M_2, \dots, M_{idx}$ to compute$f^n(M_1), f^2(M_1), \dots, f^n(M_{idx})$. Since each hash value allows for up to n non-repudiation connections the signature can be used for$n \times idx$ non-repudiation connections. The merit this method is that MS and VLR construct the secret message M_{idx} dynamically during the MS stay in same Routing Area (RA), this means the VLR does not need to bring back to the HLR in the home network. This method is performed as follows:

- When a claimant (MS) wants to register itself to the HN, by using the digital signature send a nonce random$(NONCE_{MS})$ to HN, which $NONCE_{MS}$plays here to guarantee the freshness of the message. A signature on this message allows the MS to prove its authenticity to its HLR. This message provides the legal evidence of the user's intention to make use of the service.
- After HLR/HN verify the MS, they compute the authentication token AUTHN and send it to the verifier (VLR/SN).
- A new nonce random number ($NONCE_{SN}$) is generated by the verifier (VLR/SN). A $NONCE_{SN}$ along with of $AUTHN$ which is received from HLR, the VLR generates a seed messageM_{idx}which is indexed by integer(idx)dynamically as follows:$M_{idx} = NONCE_{SN} + idx.AUTHN$.
- The verifier VLR/SN computes a new set of chained hash values $f^n(M_{idx})$ which is indexed by an integer (idx) for n hash values, which n represents the maximum number of services that the MS can request after initial authentication for the idx set of chained hash values. It is then stored for subsequent authentication.
- When it receives the challenge response from VLR/SN, the claimant MS just produces the same that seed message $M_{idx}$with the same parameters and computes a set of chained hash values $f^n(M_{idx})$. This is used for n times. When the n is used up, both MS and VLR/SN will reinitiate the new set of hashed values which is indexed by a new integer as (new $idx = idx + 1$).

By using one single message, one signature between an MS and an HLR is all that is needed for establishing the initial registration. Each MS only needs to sign once and is able to prove itself for$n \times idx$times. Therefore, the proposed scheme is efficient, and meets the security requirements: mutual authentication, non-repudiation services, and minimization of resource utilization.

4. Proposed Protocol Description

Like the UMTS AKA authentication protocol, in the proposed protocol a device is authenticated, the communication link between a VLR and HLR is adequately secure, and an MS and its HN share a common secret key K and certain cryptographic algorithms with its HN. Under the proposed protocol, the shared cryptographic algorithms between MS and its HN include; one signed message by using the digital signature; one message authentication codef^1, and three key generation functionsf^3, f^4, f^x.

Unlike for the UMTS AKA, the functions$f^2, f^{1*}$and$f^{5*}$are not necessarily needed in the proposed protocol. Since the proposed protocol uses the digital signature to let HN verify the user rather than SN, during inter-network roaming authentication, a MS and its HN will challenge each other using the nonce random number. Another enhancement of the proposed protocol is a temporary key mechanism with the management of a hash chain. This is a simple and elegant method compared to the SEQ mechanism. However, since the security level depends on the key length and the function f^5only produces a 48-bit hash result, then the security level off^5 is not sufficient to generate a robust key. Therefore, the proposed protocol uses another key generation functionf^x, which generates a 128-bit or higher hash result, to get a better security level.

The proposed authentication protocol is divided into two procedures; the first one is called the initial authentication procedure, which flow from MS⇔VLR⇔HLR. The second one is limited between MS⇔VLR and is called the subsequent authentication procedure.

4.1 Initial Authentication Procedure

The proposed protocol assumes the following operations are performed whenMS makes a service contract with his/her home network HLR:

1. MS generates the Public and Private Keys.
2. MS subscribes (Public Keys) to HLR/HN.
3. HLR produces a NONCE, TMSI and keeps it in its database. HLR writes K_S,TMSI in the SIM/USIM of MS. The format of TMSI is illustrated in Figure 3.
4. Uponreceiptthe SIM/USIM, MS verifies the related values stored in the SIM/USIM.

HLR.id	$E_{Ks}(IMSI \| NONCE_{HN})$

Ks: the secret key of HLR $NONCE_{HN}$: nonce random number
IMSI: International Mobile Subscriber identity

Figure 3: TMSI Format

At first, the scheme consists of four messages exchanged between the MS, VLR and HLR. The message flows are indicated in Figure 4. The notations are defined as follows:

- $\boldsymbol{IMSI}$: International Mobile Subscriber Identity.
- $\boldsymbol{TMSI}$: Temporary Mobile Subscriber Identity generated by HLR/HN.
- $\boldsymbol{ID_{SN}}$: The identity of the SN.
- $\boldsymbol{NONCE_{MS}}$: A nonce random number selected by MS.
- $\boldsymbol{NONCE_{SN}}$: A nonce random number selected by VLR/SN.
- $\boldsymbol{NONCE_{HN}}$: A nonce random number selected by HLR/HN.
- $\boldsymbol{CK_{HN}}$: The Cipher Key generated by an HLR, using HLR/HN-selected $NONCE_{HN}$. An MS can also generate this when given a $NONCE_{HN}$.
- $\boldsymbol{IK_{HN}}$: The Integrity Key generated by an HLR, using HLR/HN-selected $NONCE_{HN}$. An MS can also generate this when given a $NONCE_{HN}$.
- $\boldsymbol{CK_{j,idx}}$: The Cipher Key with id (j, idx) generated by an MS and a VLR and for use between the MS and the VLR/SN.
- $\boldsymbol{IK_{j,idx}}$: The Integrity Key with id (j, idx) generated by an MS and a VLR and for use between the MS and the VLR/SN>
- $\boldsymbol{f^n(M_{idx})}$: One-way hash function with idx^{th} random seed M_{idx} and n^{th}composition, where $idx \leq IDX$and $n \leq N$, for use in the subsequent authentication between MS⇔VLR/SN.
- $\boldsymbol{N}$: The maximum number of f hash chaining composition.
- $\boldsymbol{idx}$: The maximum number of random seeds for $f$hash chaining.
- ? = : An equality comparison operator.

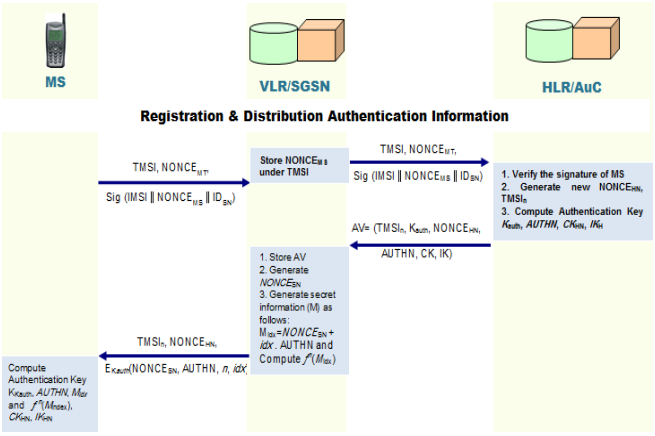

Figure 4: Message Flows in the Proposed Protocol

Step 1: M1 Authentication Request Message

When an MS needs to authenticate itself to all entities of network to access or utilize network services, the MS invokes the distribution of authentication procedure by sending the authentication request messages to the HLR/AuC ($AUTH_{MH}$) through VLR in the serving network. Authentication between the MS and his HLR/AuC relies on the use of its public-key digital signature. This message provides the legal proof of the MS's intent to register itself. Furthermore, the nonce random number here guarantees the freshness of the message and a signature on this message allows the MS to prove its authenticity to his HLR. This process is achieved as follows:

- The MS generates the following :
  - The Nonce Numbers$NONCE_{MS}$
  - The signature Sig (IMSI, Nonce, ID_{SN})
- The MS sends $AUTH_{HM}$ to VLR/SN:$AUTH_{HM} = TMSI, NONCE_{MS}, Sig(IMSI \| NONCE_{MS} \| ID_{SN})$Where: *TMSI* is the Temporary identification of the MS so that HLR can verify its signature.

Step 2: M2 Authentication Request Message

When the VLR/SN receives the message from the MS, the VLR/SN is able to recognize the HLR/HN to which MS belongs by reading the TMSI; and then it passes the message ($AUTH_{HM}$) to the intended HLR/HN. The VLR/SN maintains a profile for that MS under the identity of user (TMSI) which contains the privileges of a registered user for subsequent authentication. So, The VLR/SN waits to receive the authentication result from HLR/HN.

Step 3:M3 Authentication data Response Message

Upon receipt of the $AUTH_{HM}$, the HLR/AuC in the home network verify the MS according to the information that have been received, and then builds the Authentication Data Response message for MS and VLR/SN. In order to accomplish the authentication process, HLR/AuC will do the following:

- Decrypts the *TMSI*by the HLR's secret keyK_S to get the IMSI and$NONCE_{HN}$, after that it verifies the unique identity *IMSI*of the UE, to make sure that the user is legal one.

- The HLR/AuC in home network generates the following:
  - A new $NONCE_{HN}$.
  - A new temporary identity $TMSI_n$.
  - Computes an authentication key: $K_{auth} = \int_K^x (NONCE_{MS}, NONCE_{HN}, ID_{SN})$
  - Computes an cipher key $CK_{HN} = \int_K^3 (NONCE_{HN})$
  - Computes an integrity key $IK_{HN} = \int_K^4 (NONCE_{HN})$
  - Computes $AUTHN = \int_K^1 (NONCE_{MS}, NONCE_{HN})$
- HLR/HN sends the authentication vector to SN via secure channel as follows:

$AV = TMSI_n, K_{auth}, AUTHN, NONCE, CK_{HN}, IK_{HN}, idx$

Step 4: M4 Authentication Response Message

When the VLR/SN receives the response message from HLR/HN, it means that the MS has proved itself to its HN successfully. Therefore, the VLR/SN stores the authentication vector for performing the subsequent authentication

In order for the MS to verify the authenticity of the SN in the subsequent authentication AKA procedure, the SN generates response information for the MS and sends them to MS. Therefore, every idx^{th} performing the second procedure to produce n hash values, the SN generates a nonce random number $NONCE_{SN}$ and computes the dynamic seed M_{idx} for hash chaining function $f^n(M_{idx})$. So, performing the AKA procedure between MS and SN for $(n, idx)^{th}$ times without intervention of the HLR in the home network. The mathematical expression for the generation of the seed of hash chaining function is as $M_{idx} = NONCE_{SN} + idx.AUTHN$

After that, SN prepares the authentication response message and sends it to the MS.

$AUTH_{SN} = TMSI_n, NONCE_{HN}, E_{K_{auth}}(AUTH, NONCE_{SN}, n.idx)$

Upon receipt of $AUTH_{SN}$, the MS authenticates HN and SN by decrypting the $AUTH_{SN}$ and then verifying the AUTHN that is sent by the HN. In order to accomplish the authentication process, MS will do the following:

- Compute an authentication key $K_{auth} = \int_K^x (NONCE_{MS}, NONCE_{HN}, ID_{SN})$
- Decrypt the $AUTH_{SN}$ by the authentication key K_{auth} to get the *AUTHN* and $NONCE_{SN}$, *n*, and *idx*.
- Compute $AUTHN' = \int_K^1 (NONCE_{MS}, NONCE_{HN})$ and verify the authenticity of $AUTHN'$ by applying the confirming equation $AUTHN' ? = AUTHN$. If the equation holds, it implies that it is from the HN and the HN trusts the SN. Moreover, it means that the SN is authenticated by the MS. Otherwise; it means that both the HN and the SN are invalid. Then the MS will reject the procedure. If the HN and SN are validated the MS will start to construct the seed M_{idx} to produce n set of hash chaining function $f^n(M_{idx})$ for subsequent authentication between the MS and the VLR/SN. Meanwhile, the MS computes a cipher key $CK_{HN} = \int_K^3 (NONCE_{HN})$ and an integrity key $IK_{HN} = \int_K^4 (NONCE_{HN})$. K_{auth} becomes the shared authentication key used by the MS and SGSN/VLR, thus the authentication process is finished. After the first cipher key CK_H and integrity key IK_H are established, the MS and the VLR can make their communication instantly.

4.2 Subsequent Authentication Procedure

After the initial authentication, the VLR/SN gets a secret authentication key K_{auth} that it shares with the MS and subsequently can accomplish the mutual authentication by itself. That is, subsequent authentication only happens between the MS and the SGSN/VLR using two message exchanges. Since each authentication uses one from the set of hash chain values, then the MS can prove itself to the SN at most (n, idx) times. Within each set of hash chain values, it can be agreed that the chain with lower id (i.e., j) is used. If a set of hash chain values are used up (i.e., $j = n$) , the MS and the VLR/SN need to start another new set of hash chain functions dynamically. This is done by increasing the value of idx by one and establishing a set of hash chain values which is indexed by the new value of idx, and then set the value of j to 1.

For example, after the registration phase, the MS and the VLR/SN share the same authentication information to start the second phase AKA. Suppose the value of $n = 5$ and the value of $idx = 10$. First of all, the number of performing the second phase AKA will be 50 times without reference back to the HLR in the home network. According to the above parameters both MS and SN will set the initial value of $idx = 1$ and then initiate the seed of hash function M_1 and compute the set of hash chain function $f^5(M_1)$.

The parameter *j* will present the number of services that have been requested. In the 1st session (i.e., *j=1*), the claimant (MS) provides $f^{n-j}(M_1) \rightarrow f^4(M_1)$ to ask for a connection. The verifier (SN) computes $f\left(f^{n-j}(M_1)\right) \rightarrow f\left(f^4(M_1)\right)$ and verifies the correctness of $f^4(M_1)$ by applying the confirming equation: $f\left(f^{n-j}(M_1)\right) ? = f^{n-j+1}(M_1) \rightarrow f^5(M_1) ? = f^5(M_1)$. If the equation holds, it implies the MS has been authenticated successfully. Otherwise, it means that the MS is invalid, and then the SN will reject the procedure.

If the MS is validated, the SN will increase the counter j by one, while $j \leq n$ repeats that procedure. If the set of hash chain values are used up (i.e., $j = n$) , MS and VLR/SN need to start another new set of hash chain functions dynamically, $idx = idx + 1$ and establish a new seed of hash function M_2 and computes hash chain function $f^5(M_1)$. and then set the value of $j = 1$.

Similarly, if one side encounters problems in authenticating the other side, the verifier should send an error message with the problematic chain id to the claimant. The claimant then tries to authenticate itself to the verifier starting from the next fresh chain. For example, if the problematic chain idx in idx series is 7, then the MS should reveal $f^{n-1}(M_8)$ to the SN to try to correct the authentication problem. Figure 4 exhibits the subsequent authentication procedure, and the authentication steps are described as follows:

Step 1: The MS produces$f^{n-i}(M)$, where $i$is the number of services that have been requested, and M is the secret seed generated in the initial authentication. The MS sends $\boldsymbol{SAUTH_{MV}}$ as follows:$SAUTH_{MV}: TMSI, f^{n-1}(M_{idx})$

Step 2 VLR/SN first checks the subscribed service period of the mobile user for the requested service. If the service request is not made within the valid subscribed service period, the service request is rejected. The procedure then calls the initial authentication procedure (registration phase). Otherwise SGSN/VLR computes$f\left(f^{n-i}(M)\right)$ to verify whether it is the same as the number, $f^{n-i+1}(M)$, which VLR/SN saved in the last authentication. If they are identical, the MS has been authenticated successfully.
VLR/SN check$j = n$, if the equation holds, implies the set of hash chain values are used up. Otherwise, it means that the set of hash chain values are not finished and the MS and the SN use the same series of hash chain values in mutual authentication between them. If the equation is valid the MS and the SN simultaneously update the authentication information dynamically. The following steps describe the updating procedure.

1. Generate a new $NONCE_{\text{SN}}$.
2. Let$idx = idx + 1$and reset the value of$j$ to be equal 1.
3. Generate a new seed of hash chaining function $M_{idx+1} = NONCE_{SN} + idx.AUTHN$.
4. Compute a new set of hash chaining function $f^n(M_{idx+1})$.
5. Generate the response authentication message $SRAUTH_{\text{MV}}$ and send it to MS: $SRAUTH_{MV}: E_{K_{aut\,h}}(NONCE_{SN}, idx.j)$

When the MS receives of the response message, the MS decrypts the authenticator using $K_{aut\,h}$, and repeats the steps mentioned above. Now both MS and VLR establish a new cipher key$CK_{j.idx} = f^3\left(CK_H, f^{n-j}(M_{idx})\right)$and integrity key$IK_{j.idx} = f^3\left(IK_H, f^{n-j}(M_{idx})\right)$for the session$(j.idx)^{th}$, the *MS* and *VLR* can make their communication instantly. Therefore, stronger key agreement is achieved.

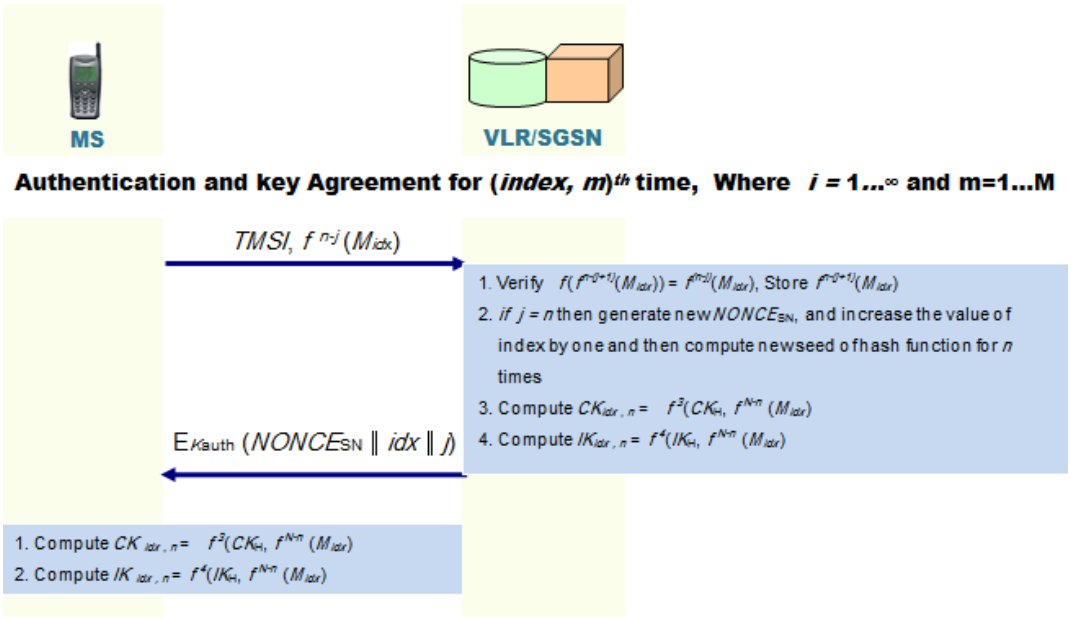

Figure 5: Subsequent Authentication Protocol

5. Security Analysis

In order to ensure that the proposed protocol is secure, the attack methods will be analyzed and discussed. The security requirements of third generation mobile systems are mutual authentication, MS anonymity, non-repudiation, and data integrity and data confidentiality. The proposed scheme can fulfil all of these requirements.

5.1 Mutual Authentication

It is clear that the proposed authentication protocol can authenticate MS, HLR/AuC and SGSN/VLR. The MS signs the message using a private key and then sends to the HLR/AuC. The HN confirms the identity of the MS by verifying the signed message using the MS's public key. Therefore, authentication between the MS and the HLR/AuC can be achieved by using the public key. Consequently, mutual authentication is achieved. In step 4 of the proposed protocol, the MS can decrypt the message which it has received and compute $AUTHN$. Therefore, the MS confirms the authenticity of the SN and HN together. After the initial authentication during the origination and termination call, the VLR/SN gets a secret authentication key$K_{aut\,h}$ that it shares with the MS and subsequently can accomplish the mutual authentication by itself. Since each authentication uses one of the set of hash chain values, then the MS can prove itself to the SN at most (n, idx) times.

5.2 Confiditionality

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 function $f^8$and session key *CK*. The proposed scheme assumes the link between the VLR/SGSN and the HLR/AuC is adequately secure like UMTS. Moreover, the communication content

through the wireless link is protected. Therefore, the attacker is not able to get any sensitive data.

5.3 Identity/Location confidentiality (User anonymity)

To provide MS anonymity in the authentication process, the permanent identity IMSI of the MS is never exposed in the plain-text mode whatever the situation is. A cracker cannot get the real identity of the MS by eavesdropping on the authentication messages on either wireless or wired networks. The UMTS fails to achieve this requirement. This goal is achieved by means of a TMSI throughout the entire authentication process. Therefore, the eavesdropper cannot get the real identity (IMSI) or be aware of the user's current location when system of VLR fails. However, the VLR has no knowledge of the user's IMSI. Furthermore, most of the other schemes did not consider this requirement.

5.4 Data Integrity

The integrity service in the proposed protocol was achieved by using the digital signature technique during the registration phase, as well as by using hash chaining function technique during origination and termination call phase. Therefore, throughout the entire authentication process the information exchanged between entities of the network cannot be altered without detection.

5.5 Non-Repudiation

The proposed protocol satisfies this requirement. The proposed protocol can provide service providers with legal evidence in order to collect bills that are denied by the user. In accordance with the goal of the proposed protocol, integrating a digital signature with a one-way hash chaining is achieved. In the $i-th$session, the user provides$f^{n-i}(M)$to ask for a connection. The SGSN/VLR can verify the correctness of$f^{n-i}(M)$by means of the one way function, but it cannot derive $f^{n-i}(M)$ from$f^{n-i+1}(M)$. In this way, $f^{n-i}(M)$can be used as a proof of the $i-th$ connection. Whenever a random challenge occurs, the SGSN/VLR can be required to show$f^{n-i}(M)$.

5.6 Minimize resource utilization

The proposed protocol satisfies this requirement by reducing the total of signaling between entities and decreasing the size of messages. Consequently, the delay time and bandwidth is minimized.

The proposed authentication protocol achieves all the requirements shown above. The proposed scheme is superior to other published schemes. The proposed protocol can prevent common attacks as follows:

Replay attacks: It can repulse replay attack, a form of network attack in which a valid data transmission is maliciously or fraudulently repeated or delayed. Hackers capture old messages and replay them at later times. By replying to the message it appears to be legal. Suppose *MS* wants to prove its identity to the *HLR*. The *HLR* requests its *IMSI* as proof of identity, which *MS* dutifully provides (possibly after some transformation like a hash function). Meanwhile, the hacker is eavesdropping on the conversation and keeps the *IMSI*. After the interchange is over, the hacker connects to the *HLR* posing as the first *MS*. When asked for proof of identity, the hacker sends the first MS's *IMSI* read from the last session, which the HLR must accept.

The proposed protocol can prevent the replay attack by the freshness of its process. The *MS* generates$NONCE_{MS}$, which is an unpredictable random number. The HLR in the home network also generates$NONCE_{HN}$. Both nonce's appear in the *AUTHN* and ensure the freshness of the authentication vector (*AV*). As well as the (*AV*) including authentication key as temporary key$K_{aut\,h}$. It refreshes the session key by using the nonce to ensure the freshness of authentication sessions. Thus the replay attack fails.

Redirection Attacks

In 3GPP-AKA, an authentication vector (AV) can be used by any SN. This situation can be abused to redirect data to an SN. This is called a redirection attack. In this case a false base station impersonates a SN. The false base station will then redirect all traffic to a SN of their choosing. The AKA procedure will normally succeed and the user will not notice being connected to another network. This can cause the user to get unusually high bills. Additionally it might be used to redirect traffic to networks with lower security, causing a wrong impression of the security level applied.

In the proposed protocol, an authentication vector (AV) generated by the user's HN can only be used by a particular serving network (SN). This is achieved by involving the identity of the SN in the generation and verification of the signature. Whenever an MS enters a new SN will get the identity of that SN, and then start a registration process to register itself to the network by replying with a new nonce random number$NONCE_{MS}$ and a signatureSigproviding integrity of$IMSI$, $NONCE_{MS}$, ID_{SN} and store a profile of the new SN in its database, which includes ($NONCE_{MS}, ID_{SN}$). When the *HN* receives the user's authentication request from the *SN*, the *HN* verifies the signature to ensure that the user is indeed in the territory of the SN. When the HN begins to generate the authentication vector, it should insert $K_{aut\,h}$and *AUTHN* into the authentication vector (*AV*). The first one is derived from($NONCE_{MS}, NONCE_{HN}, ID_{SN}$)and the second is derived from($NONCE_{MS}, NONCE_{HN}$). When the MS receives the authentication response sent from its *HN* through the *SN*, the user can determine if the message is sent by the indeed *SN* or by other *SN* by decrypting that message and verifying the (*AUTHN*) which is derived from($NONCE_{MS}, NONCE_{HN}$)..

6. Conclusion

In this paper, by integrating the public key digital signature with the hash-chaining technique, the security of the 3G protocols in network access is improved to provide key refreshment periodically, strong key management and a new non-repudiation service in a simple and elegant way. In addition, this mechanism has provided a new feature to let the encryption switches turn on before the authentication process commences and protect the subscriber's true identity. The bi-unilateral and mutual authentication among MS,

VLR/SGSN in the serving network and HLR/AuC in the home network has been adopted in the proposed scheme and result in a more secure protocol than the other available authentication protocols.

A new authentication protocol has been suggested to fulfil the security requirements of the third generation mobile systems and improve performance by reducing the communication times, and by creating fewer authentication messages and data sizes during the process of authentication. The proposed protocol significantly reduces the communication overhead between the home network and the visited network especially for roaming authentication. To avoid the complicated synchronization found in UMTS, the proposed protocol does not use SEQ, the management of a hash chain in the proposed protocol is simple and elegant compared to that of SEQ. This proposed protocol is also secure against network attacks, such as the replay attack and redirection attack.

Acknowledgment

This research was supported by the deanship of scientific research at Salman bin Abdulaziz University under the research project # 39/هـ/1432 .

References

[1] M. Al-Fayoumi, S. Nashwan, S. Yousef, A. Alzoubaidi, “A New Hybrid Approach of Symmetric/Asymmetric Authentication Protocol for Future Mobile Networks,” In Proceedings of theThird IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob, pp. 29, 2007.

[2] M. Al-Fayoumi, N. Shilbayeh, “Cloning SIM Cards Usability Reduction in Mobile Networks,” Springer’s Journal of the Network and Systems Management, 22(2), pp. 259-279, April 2014, doi: 10.1007/s10922-013-9299-8.

[3] 3GPP, “3G Security, Specification of the MILENAGE Algorithm Set: An Example Algorithm Set for the 3GPP Authentication and Key Generation Functions f1, f1*, f2, f3, f4, f5 and f5*, document 1: General,”3rd Generation Partnership Project, 2001.

[4] 3GPP, “Network Architecture,” 3rd Generation Partnership Project (3GPP), Technical Specification Group Services and System Aspects, 3GPP TS 23.002 V.4.4.0 (2002-1), Release 4.

[5] 3GPP, “3G Security, Specification of the MILENAGE Algorithm Set: An Example Algorithm Set for the 3GPP Authentication and Key Generation Functions f1, f1*, f2, f3, f4, f5 and f5*, document 3: Implementors, Test Dat,” 3rd Generation Partnership Project (3GPP), Technical Specification Group Services and System Aspects, 3GPP TS 35.207 V.5.0.0 (2006-04), Release 5.

[6] N. Saxena, N. S. Chaudhari, “NS-AKA: An improved and efficient AKA protocol for 3G (UMTS) networks,” In Proceedings of theInternational conference on advances in computer science and electronics engineering (CSEE’14), Kuala Lampur, Malaysia, pp. 220–224,2014.

[7] C. C. Lee, C. L. Chen, H. H.Ou, L. A. Chen, “Extension of an efficient 3GPP authentication and key agreement protocol,”Wireless Personal Communication, 68(3), pp.861–872, 2013.

[8] J. Al-Saraireh, S. Yousef, “A New Authentication Protocol for UMTS Mobile Networks,” EURASIP Journal on Wireless Communications and Networking, 2006 (2), pp.19-30, 2006.

[9] I. E. Chun, P. H. Ho, H. Y. Chen,“ Nested one-time secret mechanisms for fast mutual authentication in mobile communication,” In Proceedings of theIEEE Wireless Communication and Networking Conference (WCNC), pp.2714–2719, 2007.

[10] M. Zhang, Y. Fang,“Security analysis and enhancements of 3GPP authentication and key agreement protocol,” IEEE Transactions on Wireless Communication, 4(2), pp.734–742, 2005.

[11] Y. Lin, Y. Chen, “Reducing Authentication Signaling Traffic in Third-Generation Mobile Network,” IEEE Transactions on Wireless Communications, 2(3), pp. 493-501, 2003.

[12] J. Al-Saraireh, S. Yousef, “Analytical Model: Authentication Transmission Overhead Between Entities in Mobile Networks,” Elsevier, Computer Communications Journal, 30(9), pp.1713-1720, 2007.

[13] L. Harn, W. Hsin, “On the Security of Wireless Network Access with Enhancements,” In Proceedings of the 2003 ACM workshop on Wireless Security, San Diego, USA, pp.88-95, 2003.

[14] L. Lamport, “Password authentication with insecure communication,” Communication of ACM, 24(11), pp.770-772, 1981.

[15] C. Huang C., J. Li,“Authentication and Key Agreement Protocol for UMTS with Low Bandwidth Consumption,” In Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA'05), pp.392-397, 2005.

[16] H. H. Ou,M. S. Hwang, J. K. Jan, “A cocktail protocol with the authentication and key agreement on the UMTS,” Journal of Systems and Software, 83(2), pp.316–325, 2010.

[17] S. Wu, Y. Zhu, Q. Pu, “Security analysis of a cocktail protocol with the authentication and key agreement on the UMTS,” Communication Letters, 14(4), pp.366–368, 2010.

[18] Y. L. Huang, C. Y. Shen, S. W. Shieh, “S-AKA: A provable and secure authentication key agreement protocol for UMTS networks,” IEEE Transactions on Vehicular Technology, 60(9), pp.4509–4519, 2011.

[19] C. Grecas, S. Maniatis, I. Venieris, “Towards the introduction of the asymmetric cryptography in GSM,GPRS, and UMTS networks,” In Proceedings of the Sixth IEEE Symposium on Computers and Communications, pp. 15-21, 2001.

[20] L. He, N. Zhang, “An Asymmetric Authentication Protocol for M-Commerce Applications,” In Proceedings of the 8th IEEE International Symposium on Computers and Communication (ISCC’03), 1, pp.244-250, 2003.

[21] J. Jun, H.Chen, “A novel mutual authentication and key agreement protocol based on NTRU cryptography for wireless communications,” Journal of Zhejiang University Science, 6(5), pp. 399-404, 2005.

[22] G. Gódor, S. Imr,e, "Novel Authentication Algorithm – Public Key Based Cryptography in Mobile Phone Systems," IJCSNS International Journal of Computer Science and Network Security, 6(2), pp. 126-134, 2006.
[23] C. K. Yeh, W. B. Lee, "A Dual-Purpose Signature For Authentication On UMTS," Journal of the Chinese Institute of Engineers,30(2), pp. 343-347, 2007.
[24] W. Stalling, Cryptography and Network Security, Principles and Practice. 6th edition. USA: Prentice Hall, 2014.

Author Profile

Mustafa A. Al-Fayoumi received the B.S. degree in computer science from Yarmouk University, Irbid, Jordan, in 1988. He received the M.S. degree in computer science from the University of Jordan, Amman, Jordan, in 2003. In 2009, he received a Ph.D. degree in computer science from the Faculty of Science and Technology at Anglia University, UK. In 2009, he joined the Al-Zaytoonah University, in Jordan, as an assistant professor. Currently, he is assistant professor and chairman of computer science department at Salman bin Abdulaziz University, Saudi Arabia. His research interests include areas like computer security, cryptography, identification and authentication, wireless and mobile networks security, e-application security, simulation and modeling, algorithm analyzes and design, information retrieval, data mining and any other topics related to them.

Mohammed M. H. Alnababtehreceived the B.S. Degree inComputer Sciences from Philadelphia University, Jordanin 2001 and M.S. Degree in Computer InformationSystem from AABFS University, Jordan in 2005 andDoctor of Philosophy in computer information systemfrom AABFS University, Jordan in 2011. From 2002 to2005, he worked as a programmer in Web Developmentand Engineering company, Jordan. He was working as alecturer in Amman Training College during 2006-2011. He was an assistantprofessor in software Engineering Department at Jadara University, Jordanin 2011. Currently he is an assistant professor in Computer InformationSystem at Salman bin Abdulaziz University, KSA. His current researchfocuses on data mining, information retrieval and cloud computing.

Mohammad Sh. Daoud has been awarded BSc (Hons) degree in computer information system from Al-Zaytoonah University of Jordan in 2004. Four years later (2008), he has been awarded MSc degree in computer science from The University of Jordan. Furthermore, he received a Ph.D. degree in communication and media in location based-services from De Montfort University in the United Kingdom. He joined Al Ain University of Science and Technology in 2013. His specialist areas include wireless and mobile networks, mobility prediction, ants' colony optimization, networks security, Multi-Agent and real time multimedia over UMT All-IP network.

Mohammad O. Alhawarat has got his Ph.D. in chaotic neural networks from School ofTechnology of Oxford Brookes University, United Kingdom, in 2007. He has worked as anassistant professor in Petra University in Jordan for 1 year, he then joined the College ofComputer Engineering and Sciences of Salman bin Abdulaziz University since 2008 as anassistant professor of computer science. His research interests are chaotic neural networks,machine learning including Arabic natural language processing.