Noname manuscript No.
(will be inserted by the editor)

On the Design of a Secure Proxy Signature-based Handover Authentication Scheme for LTE Wireless Networks

Behnam Zahednejad[1] · **Majid Bayat**[2] · **Ashok Kumar Das**[3]

Received: date / Accepted: date

Abstract Designing a secure and efficient handover authentication scheme has always been a concern of cellular networks especially in 4G Long Term Evolution (LTE) wireless networks. What makes their handover so complex, is the presence of different types of base stations namely eNodeB (eNB) and Home eNodeB (HeNB). In addition, they cannot directly communicate with each other. Recently, an efficient proxy signature-based handover authentication scheme has been suggested by Qui et al. Despite its better performance and security advantages than previous schemes, it suffers serious vulnerabilities, namely being prone to DoS attack , eNB impersonation attack and lack of perfect forward secrecy. In this paper, we propose an improved handover authentication scheme in LTE wireless networks that resists against such attacks. Further, we validate the security of the proposed scheme using Real-Or-Random (ROR) model and ProVerif analysis tool. The results confirm our security claims of the proposed scheme. In addition, the performance analysis shows that compared to other schemes, our proposed scheme is more efficient.

Keywords LTE · Handover authentication scheme · Proxy Signature · perfect forward secrecy · DoS attack · ProVerif · ROR model

[1]B. Zahednejad
Department of Cryptography and telecommunication, I.H University, Tehran, Iran
Tel.: +98 9101829672
Fax: +98 9101829672
E-mail: zahednejadb@gmail.com

[2] M. Bayat
Department of Computer Engineering, Shahed University, Tehran, Iran
E-mail: mbayat@shahed.ir

[3] A.K. Das
Center for Security, Theory and Algorithmic Research,
International Institute of Information Technology, Hyderabad 500 032, India
E-mail: iitkgp.akdas@gmail.com

1 Introduction

LTE (Long Term Evolution) is the standard of high-speed wireless communication mobile devices based on GSM/UDGE and UMTS/HSPA technologies. Being developed by 3GPP, it improves the core network and increases the capacity and speed by using different radio interfaces. In addition, it provides downlink peak rates of over 100 Mbit/s with speeds of over 200 Mbit/s. LTE networks use flexible carrier bandwidth from 5MHz to 20 MHz. FDD (Frequency Division Duplex) and TDD (Time Division Duplex) are supported in LTE as well. In addition to mobile phones, other portable devices such as notebooks, cameras, gaming devices, etc. implement LTE embedded modules [1]. The network consists of three main components: User Equipment (UE) , Evolved Universal Terrestrial Radio Access Networks (E-UTRAN), Evolved Packet Core (EPC).
User Equipment (UE): The user equipment is quite similar to the one used by UMTS,GSM. It is composed of the following modules:

- Mobile Termination (MT):This module is responsible for all communication functions.
- Terminal Equipment (TE): This module is the destination of data streams.
- Universal Integrated Circuit Card (UICC): This module is quite similar to SIM card of LTE equipment running Universal Subscriber Identity Module (USIM) application. This module stores information related to user such as his phone number, home network identity and security keys etc.

Evolved Universal Terrestrial Radio Access (E-UTRA): This network is the serving network which communicates with User Equipents (UEs). It includes the eNodeBs (eNB) which are normal base stations. Home eNodeBs (HeNB) are new kind of base stations for better network performance.
Evolved Packet Core (EPC): This network is the backbone network consisted of many components such as the Serving Gateway (S-GW), the Mobility Management Entity (MME), Evolved Serving Mobile Location Centre (E-SMLC) etc.
With the development of 4G LTEs, the performance and security of these networks have become so important [2]. Due to the lack of direct communication between base stations, handover and roaming to other base stations are so crucial that many roaming authentication schemes have been proposed to address this requirement [3,4,5,6,7,8,9,10,11,12]. In the following, we describe some of the main handover authentication schemes in LTE networks.

1.1 Related works

Mishra et al. [13] suggested a Security Context Transfer-based (SCT) scheme for the handover authentication of LTE Networks. In this scheme, before the beginning of the UE handover, the current eNB transfers the authentication information of the UE to the new eNB. In this scheme, a trust relationship is assumed between eNB which is not possible when the eNBs are in different networks.
Kim et al. [14] suggested an identity-based handover authentication scheme. The

scheme requires a Private Key Generator (PKG). However, the scheme has unresolved problems such as key escrow problem. In addition, pairing operations incur lots of computation costs. It cannot achieve perfect forward/backward secrecy as an important security requirement, as well.
Cao et al. [15] described replay attacks, de-synchronization attacks and lack of backward security as three main vulnerabilities of LTE handover scheme. To address these vulnerabilities, some schemes [16,17,18] have suggested the participation of Authentication, Authorizing, and Accounting (AAA) servers. However, these schemes incur so much authentication traffic. In addition, these server are far away from the base stations that incorporating them demands much time that makes these network not fast enough.
Choi et al. [19] suggested a credentials-based handover authentication using chameleon hash functions. The scheme utilizes short-term credentials generated by hash functions to provide authentication and key agreement in LTE networks. The scheme enjoys much more simplicity compared to AAA-based schemes. However, Yoon et al.[20] identified lack of perfect forward secrecy and perfect backward secrecy as the main vulnerability of this scheme.
Jing et al. [7] suggested a proxy signature-based roaming authentication protocol in which the HSS,AAA server or the base stations delegate their signing power to the UE for authentication. This method reduces the complexity of the entire network. Its main drawback is the one-way authentication, which makes the access point (AP) to authenticate the mobile user but not vice versa. This problem is resolved in [5] in which proxy signature is utilized to provide mutual authentication between the eNB and the UE with perfect forward/backward security. However, its main drawback is the heavy computation cost of the low-powered UEs who need to perform RSA verification for five times. For the sake of higher performance, an authentication scheme based on Elliptic Curve Cryptography (ECC) is introduced in [6] with smaller key sizes [20]. However, similar to [7], the Mesh AP (MAP) cannot be authenticated by the Mesh Host (MH).
Recently,Qui et al. [8] suggested an efficient proxy signature based handover authentication scheme based on the ECC. In addition to the better performance of the scheme, mutual authentication is achieved as well. The scheme is applicable in different handover scenarios including normal handover scenario, handover to HeNBs and handover to base stations connected to different MME. This issue has not been considered in any other schemes before.

1.2 Our contribution

In this paper, we discuss the vulnerabilities of the most recent LTE handover authentication scheme (Qui et al.′s scheme [8]). In particular, we have identified three main drawbacks: being prone to DoS attack , eNB impersonation attack and lack of perfect forward secrecy. We then propose an improved scheme resistant against these vulnerabilities. Further, we validate the security of the proposed scheme using Real-Or-Random (ROR) model and ProVerif automatic analysis tool.

1.3 Organization of the paper

The recently suggested Qui et al.′s roaming authentication protocol [8] and its vulnerabilities are discussed in section 2. In section 3, we propose our improved handover authentication scheme resistant against well-knows attacks. In section 4, we describe its security properties and performance analysis . Further, we discuss the formal security verification of the scheme using Real-Or-Random (ROR) model and ProVerif automatic analysis tool . Finally, a conclusion is given in section 5.

2 Review of Qui et al.′s scheme

The Qui et al.′s scheme utilizes the proxy signature concept first introduced by Mambo et al. [21]. According to Table 1, a cyclic group C is selected for the Elliptic Curve Cryptography (ECC) [22] which is public and known to everyone. It has a prime order p where $|p| = 160$ bits. In this network, each party has a private/public key selected from C denoted by X_i and Y_i respectively, where $Y_i = X_iG$. The hash functions used in the scheme are denoted by $h_1() : \{0,1\}^* \rightarrow \{0,1\}^n, h_2() : C \rightarrow \{0,1\}^n$ and $h_3() : \{0,1\}^* \rightarrow Z_p^*$. The scheme is consisted of two phases: attach phase and the handover phase. The attach phase occurs when a UE or base station first registers in the network. The handover phase describes the roaming process of the UE as it roams from one cell to another. In the following, we describe the attach phase and the handover phase of the Qui et al.′s ECC-based Proxy signature for Handover (EPH) scheme. The handover phase includes roaming to a base station associated to the same MME (normal handover phase) and to a base station associated to a different MME. For more information about different kinds of handover refer to [8].

Table 1 The notations of Qui et al.′s scheme

Notations	Description
C	Cyclic group of ECC
P	Prime order of group C, $\|p\| = 160$ bits
X_i	Private key of party i
$Y_i = X_iG$	Public key of party i
$h_1()$	Hash function $h_1() : \{0,1\}^* \rightarrow \{0,1\}^n$
$h_2()$	Hash function $h_2() : C \rightarrow \{0,1\}^n$
$h_3()$	Hash function $h_3() : \{0,1\}^* \rightarrow Z_p^*$

2.1 Attach phase

In this phase, eNBs/HeNBs, MMEs and HSS are connected to each other by wired links. They authenticate each other and establish shared secret keys using accepted protocols such as Internet Protocol Security (IPsec) and Internet Key Exchange version 2 (IKEv2) protocols. In addition, MME and HSS have a pre-shared secret key K_{MH} to communicate with each other. This phase is consisted of the following steps:

1. Similar to EPS-AKA scheme [23], the attach phase is initiated by the UE who first sends the MME its attach request containing its International Mobile Subscriber Identity (IMSI) number encrypted by the public key of MME.
2. After receiving the attach request, the IMSI of the UE is forwarded to the HSS by MME as an attachment data request. This request is encrypted with K_{MH}.
3. After receiving the attachment data request by the HSS, it chooses a proxy warrant w_{UE} for the UE as the expiration time of the proxy keys. Upon the expiration of the warrant, a message is sent to the HSS to request a new warrant. Using the random number r ($r \in Z_p^*$) ,the proxy keys of the UE (m_{UE}, η_{UE}) are generated as follows:
$$m_{UE} = rG \tag{1}$$
$$\eta_{UE} = X_{HSS} h_3(w_{UE} \| h_2(m_{UE})) + r \tag{2}$$
Then the authentication vectors (AVs) in addition to the set of $(m_{UE}, \eta_{UE}, w_{UE})$ are sent to the MME. The AVs defined in [23] includes the authentication information such as authentication token (AUTN),K_{ASME} which are all encrypted with K_{MH}.
4. As MME receives the above messages, it decrypts them with K_{MH}. The UE is then authenticated as step 4 and step 5 of the $EPS - AKA$. The MME sends the AV included in the user authentication request to the UE who responds with the user authentication response. Upon equality of the random numbers of the request and the response, both MME and the UE make sure over the agreement on the K_{ASME}. Upon successful verification of the UE by the MME, the proxy keys $(m_{UE}, \eta_{UE}, w_{UE})$ which are encrypted with the K_{ASME} will be forwarded to the UE.
5. As the UE decrypts the received message with the K_{ASME}, it checks the validity of the proxy keys using the following Equation:
$$\begin{aligned} \eta_{UE} G = X_{HSS} h_3(w_{UE} \| h_2(m_{UE})) G + rG = \\ h_3(w_{UE} \| h_2(m_{UE})) Y_{HSS} + m_{UE} \end{aligned} \tag{3}$$
If the above Equation does not hold, the UE sends an authentication reject message to the HSS to ask for new and valid proxy keys. Otherwise, its public and private keys are computed as follows:
$$X_{UE} = \eta_{UE} \tag{4}$$
$$Y_{UE} = \eta_{UE} G \tag{5}$$
In a similar fashion, the eNB receives the proxy tuple $(m_{UE}, \eta_{UE}, w_{UE})$ from the HSS to compute its privateX_{eNB} and public key Y_{eNB}.

2.2 Normal handover phase

The normal handover phase occurs when an UE roams to a foreign eNB associated to the same MME as the current eNB. In this phase, the UE should perform the following steps (Figure 1):

1. In the first step, the UE tries to find nearby legitimate base stations. In this regard, it receives the list of public keys of the close eNBs from the current eNB. Or, upon roaming to a new MME, the list of legitimate base station are given by MME.
2. After finding the new legitimate base station, the UE generates its public key by multiplying a random number ($r_{UE} \in Z_p^*$) by the public key of the eNB. (Equation 6). In addition, it multiplies the random number r_{UE} by the generator G to get the public parameter (R_{UE}). Then it signs the above parameters using private key X_{UE} and the random number r_{UE} as Equation 8.

$$PK_{UE} = r_{UE} Y_{eNB} \tag{6}$$

$$R_{UE} = r_{UE} G \tag{7}$$

$$s_{UE} = X_{UE} - r_{UE} h_3(h_2(PK_{UE}) \| h_2(R_{UE}) \| h_1(I_{UE})) \tag{8}$$

   Here, I_{UE} contains the necessary information such as temporary ID, Global Unique Temporary Identity (GUTI) and the security capabilities of the UE including the ciphering and integrity algorithms applied by the UE. Then the set of parameters $(R_{UE}, s_{UE}, m_{UE}, w_{UE}, Y_{UE}, I_{UE})$ are sent to the visited eNB.
3. After receiving the UE's request $(R_{UE}, s_{UE}, m_{UE}, w_{UE}, Y_{UE}, I_{UE})$, the eNB checks the validity of the proxy keys by checking their expiration time w_{UE}. The mechanism of verifying valid proxy keys are explained in section 2-1. After receiving valid proxy keys, the eNB computes the public key PK_{UE} according to Equation 9 . In addition, it checks the integrity of the received messages by verifying the signature s_{UE} according to Equation 10 .

$$PK_{UE} = r_{UE} Y_{eNB} = R_{UE} X_{eNB} \tag{9}$$

$$\begin{aligned} s_{UE} G + R_{UE} h_3(h_2(PK_{UE}) \| h_2(R_{UE}) \| h_1(I_{UE})) = \\ X_{UE} G = h_3(w_{UE} \| h_2(m_{UE})) Y_{HSS} + m_{UE} \end{aligned} \tag{10}$$

4. Upon successful authentication of the UE, the eNB generates a new random number r' and computes R' and session key K as follows:

$$R' = r' G \tag{11}$$

$$K = r' R_{UE} \tag{12}$$

   In addition, another random number $r_{eNB} (\in Z_p^*)$ is chosen to compute the signature s_{eNB} as follows:

$$PK_{eNB} = r_{eNB} Y_{UE} \tag{13}$$

$$R_{eNB} = r_{eNB} G \tag{14}$$

$$s_{eNB} = X_{eNB} - r_{eNB} h_3(h_2(PK_{eNB}) \| h_2(R_{eNB}) \| h_2(R') \| h_1(I_{eNB})) \tag{15}$$

   Here, I_{eNB} denotes necessary information such as selected encryption algorithms. In addition, w_{eNB} defines the expiration times of the proxy keys chosen by the HSS. eNB then sends the set of messages $(R_{eNB}, s_{eNB}, m_{eNB}, w_{eNB}, R', I_{eNB})$ to the UE.

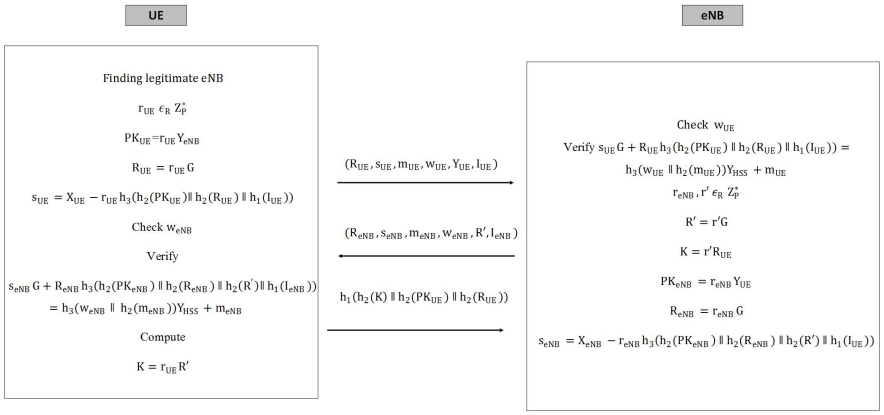

Fig. 1 Normal handover phase of the Qui et al.′s scheme

5. Upon receiving the messages $(R_{eNB}, s_{eNB}, m_{eNB}, w_{eNB}, R', I_{eNB})$ the UE first checks the validity of the proxy keys using w_{eNB}. Upon their validity, the UE computes PK_{eNB} using Equation 14. For the authentication of the eNB, the UE checks if Equation 15 holds. Otherwise, the UE looks for a new legitimate eNB.Upon successful authentication of the UE, the session key K is computed as follows:

$$K = r_{UE}R' \tag{16}$$

The UE sends the eNB the message $(h_1(h_2(K)\|(h_2(PK_{UE})\|h_2(R_{UE})))$ as a session key confirmation message.
6. Upon receiving the above message by the eNB, it computes the value of $(h_1(h_2(K)\|(h_2(PK_{UE})\|h_2(R_{UE}))))$ by itself to see if it is equal to the received message. If they are equal, the authentication process is over and a message is sent to the MME showing the establishment of the connection.

2.3 Handover to a base station associated with other MME

After connecting to a base station associated with other MME than the one associated to the current base station, the two MMEs need a handover authentication protocol as well. This protocol is performed between step 3 and 4 of the normal handover phase. Upon connecting to a HeNB, the legitimacy of the UE is checked after the completion of handover authentication protocol between the MMEs. The main steps of this phase include:

1. The target base station sends its own MME the essential information such as UEs GUTI and target MME ID (GUMMEI).
2. The MME checks GUTI and GUMMEI of the current MME to send an identification request to it.

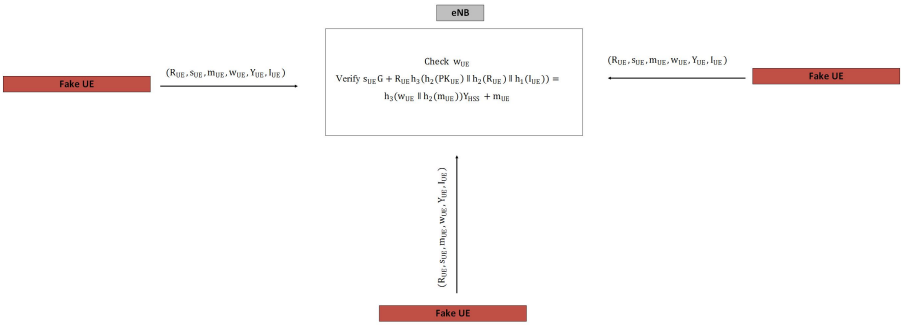

Fig. 2 DoS attack against Qui et al.′s handover scheme

3. As a response to this request, the current MME sends an identification response message containing $UE's$ information such as its IMSI number, network capability, etc.
4. After receiving the identification response by the target MME, it sends a handover request ACK to the target base station.

After completing the above steps, the target base station goes to the step 4 of the normal handover phase.

2.4 Vulnerabilities of Qui et al.′s scheme

Although the Qui et al.′s proxy-based scheme enjoys certain security and efficiency advantages such as mutual authentication, unforgeability and less computation and communication costs compared to the previous scheme, but the normal handover phase of the scheme suffers serious vulnerabilities such as being prone to DoS attack, impersonation of the eNB to the UE and lack of perfect forward secrecy. In the following, we discuss each of them in detail:

2.4.1 Being prone to the DoS attack

As shown in Figure 2, during each expiration time of the proxy keys denoted by w_{UE}, the adversary is able to replay the same authentication requests of the UE to the eNB. As a result of receiving so much authentication requests sent by the adversary, the eNB is unable to handle them all simultaneously. Thus, the availability of the eNB comes into danger.

2.4.2 Impersonation of the eNB to the UE

As shown in Figure 3, the adversary is able to impersonate a legal eNB to the UE. In this regard, it suffices for the attacker to replay the messages $(R_{eNB}, s_{eNB}, m_{eNB}, w_{eNB}, R', I_{eNB})$ as a response to the authentication requests of the UE during each expiration time of the proxy keys denoted by w_{eNB}. This is due to the fact that these parameters are not session-specific and may be easily replayed by the adversary.

2.4.3 Lack of perfect forward secrecy

Perfect Forward Secrecy (PFS) is an important security requirement of cryptography protocols. Any security protocol that provides perfect forward secrecy should prevent the adversary to learn previous session keys, when the long term secret key is compromised. However, the Qui et al.′s scheme fails to provide PFS. According to Equation (8), once the secret key X_{UE} is compromised, the attacker is able to obtain the secret parameter r_{UE} by some simple algebra ($r_{UE} = \frac{s_{UE}+X_{UE}}{h_3(h_2(PK_{UE})\|h_2(R_{UE})\|h_1(I_{UE}))}$). Then, the session key (K) can be easily computed according to Equation (16).

3 Our proposed scheme

In this section, we propose our improved authentication scheme that is resistant against above-mentioned attacks. In the following, we describe the different phases of our improved handover scheme:

3.1 Attach phase

In the attach phase, wired links are used to establish connection between eNBs/HeNBs, MMEs and HSS . Widely used protocols such as Internet Protocol Security (IPsec) and Internet Key Exchange version 2 (IKEv2) protocols are adopted to authenticate these parties together and establish shared keys between them. The MME and HSS have a common pre-shared secret key K_{MH}. The following steps should be taken in this phase:

1. The UE first sends the attach request to the MME that includes its encrypted International Mobile Subscriber Identity (IMSI) number by the public key of MME.
2. The MME encrypts this message with the pre-shared key K_{MH} and sends this encrypted message to the HSS as an attachment data request.
3. The HSS receives the attachment data request and generates the proxy keys of the UE with the expiration time defined by the proxy warrant w_{UE}. After choosing a random number r ($r \in Z_p^*$)), it computes the proxy keys ((m_{UE}, η_{UE})) of the UE:

$$m_{UE} = rG \tag{17}$$

$$\eta_{UE} = X_{HSS}h_3(w_{UE}\|h_2(m_{UE})) + r \tag{18}$$

   The HSS sends the authentication vector (AVs) [23] and the messages (m_{UE}, η_{UE}, w_{UE}) to the MME. The AVs consists of the authentication token (AUTN) and K_{ASME} encrypted under K_{MH}.
4. The MME decrypts the received message of the HSS with K_{MH}. Following the same procedure to the step 4 and 5 of the EPS-AKA, the UE is authenticated. The MME forwards the AV to the UE who replies the user authentication response. If the random numbers of the request and response are equal, the MME and UE agree on K_{ASME}. If the MME verifies the UE successfully, the encrypted proxy keys ($m_{UE}, \eta_{UE}, w_{UE}$) are encrypted by K_{ASME} and sent to the UE.

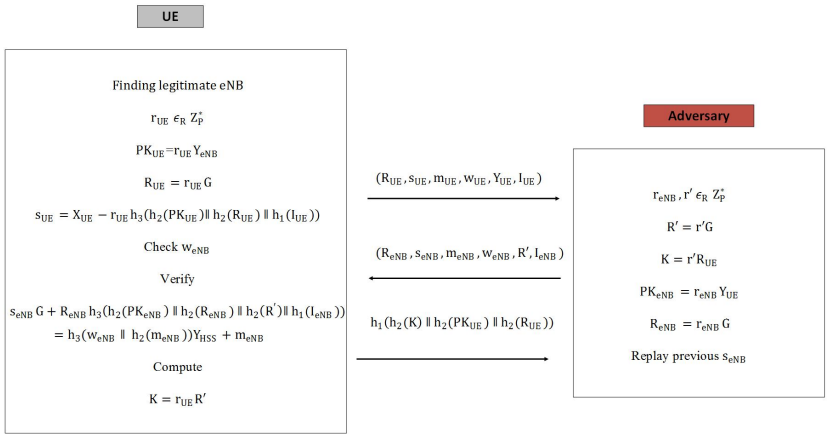

Fig. 3 eNB impersonation attack against Qui et al.′s scheme

5. The UE receives the message of MME and decrypts it with K_{ASME}. The proxy keys are validated as follows:

$$\eta_{UE}G = X_{HSS}h_3(w_{UE}\|h_2(m_{UE}))G + rG = h_3(w_{UE}\|h_2(m_{UE}))Y_{HSS} + m_{UE} \tag{19}$$

If Equation (19) holds, the private X_{UE} and public key Y_{UE} of the UE are calculated according to Equation (17-18). Otherwise, an authentication reject message is sent to the HSS to ask a new proxy keys.
Similarly, the eNB computes its private X_{eNB} and public key Y_{eNB} after receiving the proxy tuple from HSS.

3.2 Normal handover phase

The normal handover phase happens if the UE roams to a base station that is connected to the same MME as the current base station. The UE and the base station authenticate each other by performing the following steps:

1. As the first step, the UE looks for close base stations. It receives their public keys from the current base station. The public keys may be received from MME, if it roams to a new MME.
2. After the recognition of the new base station, the UE′s public key is computed as the multiplication of a random number $(r_{UE} \in Z_p^*)$ with the eNB ′s public key.(Equation 20)
The UE creates another random number $(r' \in Z_p^*)$. The random numbers r_{UE}, r'_{UE} are multiplied with the generator G to establish the public parameters R_{UE}, R'_{UE} respectively (Equation 21).Then it signs the public parameters with the current

time-stamp t_1 multiplied by the random number r_{UE} using the private key X_{UE} (Equation 22).

$$PK_{UE} = r_{UE}Y_{eNB} \tag{20}$$

$$R_{UE} = r_{UE}G, R'_{UE} = r'_{UE}G \tag{21}$$

$$s_{UE} = X_{UE} - r_{UE}h_3(h_2(PK_{UE})\|h_2(R_{UE})\|h_2(R'_{UE})\|h_1(I_{UE})\|t_1) \tag{22}$$

I_{UE} includes the information such as temporary ID, Global Unique Temporary Identity (GUTI) and the security capabilities of the UE like the ciphering and integrity algorithms of the UE. Then, the message $(R_{UE}, R'_{UE}, s_{UE}, m_{UE}, w_{UE}, Y_{UE}, I_{UE}, t_1)$ is sent to the visited eNB.

3. After receiving the request of the UE, $(R_{UE}, R'_{UE}, s_{UE}, m_{UE}, w_{UE}, Y_{UE}, I_{UE}, t_1)$, the expiration time w_{UE} of the proxy keys are checked by the eNB as explained in section 2-1. The eNB calculates the public key PK_{UE} according to Equation 23. Then, it checks if time-stamp t_1 does not exceed an allowable interval $\triangle T$. In addition, the signature s_{UE} is verified according to Equation 24 .

$$PK_{UE} = r_{UE}Y_{eNB} = R_{UE}X_{eNB} \tag{23}$$

$$\begin{aligned} s_{UE}G + R_{UE}h_3(h_2(PK_{UE})\|h_2(R_{UE})\|h_2(R'_{UE})\|h_1(I_{UE}\|t_1) = \\ X_{UE}G = h_3(w_{UE}\|h_2(m_{UE}))Y_{HSS} + m_{UE} \end{aligned} \tag{24}$$

4. After the authentication of the UE, the eNB creates a new random number r' and the public parameter R'. Further, the session key K is calculated as follows:

$$R' = r'G \tag{25}$$

$$K = r'R'_{UE} \tag{26}$$

Then, it chooses another random number $(r_{eNB} \in Z_p^*)$ to generate the signature s_{eNB}as follows:

$$PK_{eNB} = r_{eNB}Y_{UE} \tag{27}$$

$$R_{eNB} = r_{eNB}G \tag{28}$$

$$s_{eNB} = X_{eNB} - r_{eNB}h_3(h_2(PK_{eNB})\|h_2(R_{eNB})\|h_2(R')\|h_1(I_{eNB})) \tag{29}$$

Where, I_{eNB} contains information such as encryption algorithms and w_{eNB} denotes the expiration times of the proxy keys. Then eNB sends the messages $(R_{eNB}, s_{eNB}, m_{eNB}, w_{eNB}, R', I_{eNB})$ to the UE.

5. After receiving the $(R_{eNB}, s_{eNB}, m_{eNB}, w_{eNB}, R', I_{eNB})$ the UE checks the expiration time of the proxy keys w_{eNB}. If valid, the UE calculates the public key PK_{eNB} using Equation 30. If Equation 31 holds, the eNB is authenticated.

$$PK_{eNB} = r_{eNB}Y_{UE} = R_{eNB}X_{UE} \tag{30}$$

$$\begin{aligned} s_{eNB}G + R_{eNB}h_3(h_2(PK_{eNB})\|h_2(R_{eNB})\|h_2(R')\|h_1(I_{eNB}\|R_{UE}) \\ = X_{eNB}G = h_3(w_{eNB}\|h_2(m_{eNB}))Y_{HSS} + m_{eNB} \end{aligned} \tag{31}$$

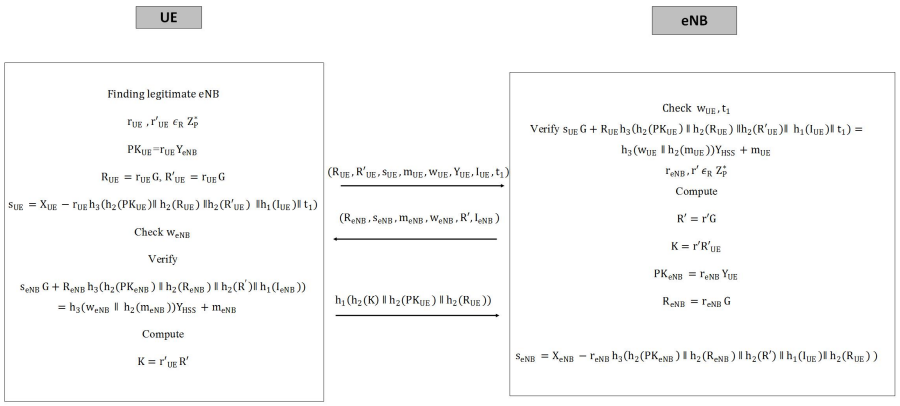

Fig. 4 Our proposed normal handover phase scheme

If the UE is authenticated, the session key K is calculated as follows:

$$K = r'R' \tag{32}$$

The UE sends the message $(h_1(h_2(K)\|h_2(PK_{UE})\|h_2(R_{UE})))$ as a session key acknowledgment message to the eNB.

6. After receiving the above message, the eNB calculates $(h_1(h_2(K)\|h_2(PK_{UE})\|h_2(R_{UE})))$ and checks if it is equal to the received message. After the successful authentication, a message is sent to the MME representing the establishment of the connection.

3.3 Handover to a base station associated with other MME

If the UE connects to a base station associated with a different MME than the one associated with the current base station, the MMEs perform a handover authentication protocol between the step 3 and 4 of the normal handover phase. If connected to a HeNB, the UE is authenticated after the handover authentication protocol between MMEs. The procedure of this protocol include:

1. The target base station sends information such as UE′s GUTI and the ID of the MME (GUMMEI) to the associated MME.
2. The MME sends an identification request to the current MME after checking the GUTI and GUMMEI.
3. The current MME responds to this request with a message that conveys the UE′s information like its IMSI number, network capacity, etc.
4. As the target MME receives the message of the current MME, it sends a handover request ACK to the target base station. If the above steps are finished, the target base station jumps to the step 4 of the normal handover phase.

4 Security and performance analysis

In this section, we describe the security properties of the proposed scheme based on an active attacker who can alter, construct, decompose or inject any messages to any entities.Then, we evaluate the efficiency of the proposed scheme compared to other previous schemes. In addition, we analyze the security of the scheme using Real-Or-Random (ROR) model and ProVerif analysis tool

4.1 Security properties

The main security properties of the proposed scheme include:

- **Strong unforgeability of the proxy signature algorithm.** Based on the security of ECC, the proxy signatures can never be forged. Only the party who is given the signing keys can generate proxy signatures on messages.
- **Mutual authentication of the UE and eNB.** Due to the unforgeability of proxy signatures, no body can sign messages on behalf of legitimate eNB and UEs. Thus, both eNB and UEs authenticate each other by verifying the signatures of each others.
- **Nonrepudiation of HSS over issuing the proxy keys to the UE or the eNB.** As the proxy keys are generated by the private keys of the HSS, nobody can forge the same proxy keys except HSS. Thus, it cannot deny issuing the proxy keys.
- **Resistance to man-in-the-middle attack.** The attacker has no chance to get the agreed session key, as it is not sent in the messages. The value of R' is sent instead. In addition, the public key of the UE is sent along the messages.

The above properties have also been achieved in Qui et al.′s scheme [8]. The main advantages of our proposed scheme compared to Qui et al.′s scheme include:

- **Resilience to DoS attack.** Despite the Qui et al.′s scheme [8], our proposed scheme is resilient to DoS attack. In this regard, we exploited time-stamp in the user authentication request and sign it according to Equation 22. Thus, the eNB no longer accepts fake user authentication requests.
- **Resistance to eNB impersonation attack.** In our proposed scheme, the session-specific parameter R_{UE} is included in the eNB′ s signature (Equation 29). As a result, the set of eNBs messages can no longer be replayed by the adversary to impersonate the eNB.
- **Perfect forward secrecy.** Despite the Qui et al′s scheme [8], according to Equation (32), the session key of our proposed scheme is constructed by a different random number (r'_{UE}) than the random number used in the signature s_{UE}(Equation 22). Thus, compromise of the long term key X_{UE} gives no chance to the attacker to obtain the session key K.

A brief security comparison of the proposed scheme with other similar schemes is shown in Table 2.

Table 2 Security comparison of our proposed scheme with Qui et al.′s scheme

Security properties	Qui et al.′s scheme	Our proposed scheme
Mutual authentication	Yes	Yes
Resilience to man-in-the middle attack	Yes	Yes
Resilience to DoS attack	No	Yes
Resistance to eNB impersonation attack	No	Yes
Achievement of perfect forward secrecy	No	Yes

4.2 Performance analysis

In this section, we evaluate the performance of our proposed scheme based on the computation costs of the involved cryptography algorithms. In addition, it is compared with Qui et al.′s scheme [8] and Jing et al.′s scheme [7], Cao et al.′s scheme [15] which have similar architecture and handover scenario with ours. The UE and eNB are modeled as a Celeron 1.1 GHz processor and a Dual-Core 2.6 GHz processor respectively. The computation cost of each computation type is estimated using C/C++ OPENSSL library. The results are shown in Table 3. As shown in Table 4, our proposed scheme is much more efficient than Choi et al.′s and Cao et al.′s scheme. Although it is less efficient for only 1.5 ms compared to Qui et al.′s scheme.

Table 3 The computation cost of each operation for UE and eNB.

Computation type	Symbol	Cost of UE (ms)	Cost of eNB (ms)
Modular exponentiation	T_e	1.7	0.5
RSA verification	T_{RV}	1.0	0.3
Elliptic curve point multiplication	T_M	1.5	0.5
Tate pairing	T_p	38	16

Table 4 Computation cost of different schemes.

	Jing et al.′s scheme (ms)	Cao et al.′s scheme (ms)	Qui et al.′s scheme (ms)	Our proposed scheme (ms)
UE	$2T_M = 3$	$5T_e + T_M = 10$	$3T_M = 4.5$	$4T_M = 6$
eNB	$6T_M = 3$	$5T_e + T_M = 3$	$3T_M = 1.5$	$3T_M = 1.5$
Total	6	13	6	7.5

4.3 Formal security analysis using ROR model

In this section, we prove the session key (SK) security of the proposed scheme formally using the broadly-accepted Real-Or-Random (ROR) model [25]. The ROR model-based formal security has gained popularity among the researchers recently and it is applied in analyzing the formal security in many authentication protocols [25,26,27,28,29]. To prove the session key (SK) security of the proposed scheme, we first discuss briefly the ROR model and then the formal security proof in Theorem

1. We also define the following computational problems which are needed for the security analysis.

The formal definition of a one-way hash function $h(\cdot)$ along with collision-resistant property [30], and also the elliptic curve computational problems are given below.

Definition 1 (Collision-resistant one-way hash function) A collision-resistant one-way hash function h: $\{0,1\}^* \rightarrow \{0,1\}^n$ is a deterministic mathematical function that takes a variable length input and produces a fixed length output, say n bits. If $Adv_{\mathcal{A}}^{HASH}(rt)$ denotes the advantage of an adversary $\mathcal{A}'$s finding a hash collision,

$$Adv_{\mathcal{A}}^{HASH}(rt) = Pr[(i_1, i_2) \in_R \mathcal{A} : i_1 \neq i_2, h(i_1) = h(i_2)],$$

where the probability of a random event X is $Pr[X]$, and the pair $(i_1, i_2) \in_R \mathcal{A}$ indicates that the inputs i_1 and i_2 are randomly chosen by $\mathcal{A}$. By an (ϵ, rt)-adversary $\mathcal{A}$ attacking the collision resistance of $h(\cdot)$, we mean that the run-time of $\mathcal{A}$ is at most rt and that $Adv_{(A)}^{HASH}(rt) \leq \epsilon$.

Definition 2 (Elliptic curve discrete logarithm problem (ECDLP)) Let $E_p(a, b)$ be an elliptic curve over a finite (Galois) field $GF(p)$ and $P \in E_p(a, b)$ be a point, where a and b are constants from $Z_p = \{0, 1, 2, \cdots, p-1\}$ such that $4a^3 + 27b^2 \neq 0 \pmod p$ and p is prime. The elliptic curve discrete logarithm problem (ECDLP) is that given the points P and $Q \in E_p(a, b)$, where $Q = d.P$, the scalar multiplication of the point P, to find the discrete logarithm d.

Definition 3 (Elliptic curve decisional Diffie-Hellman problem (ECDDHP)) Let $P \in E_p(a, b)$ be a point in $E_p(a, b)$. The ECDDHP states that given a quadruple (P, $k_1.P$, $k_2.P$, $k_3.P$), decide whether $k_3 = k_1k_2$ or a uniform random value, where $k_1, k_2, k_3 \in Z_p^*$, $Z_p^* = \{1, 2, \cdots, p-1\}$.

Both ECDLP and ECDDHP are computationally infeasible if the prime p is large. To make ECDLP and ECDDHP intractable, p should be chosen at least 160-bit prime based on the bit security recommended by NIST [31].

4.3.1 ROR model

Two entities, namely user equipment UE and $eNodeB$ are involved in the proposed scheme during the normal handover procedure. We have the following components associated with the ROR model [24].

- **Participants.** Let Π_{UE}^x and Π_{eNodeB}^y denote the x^{th} and y^{th} instances of UE and $eNodeB$, respectively, which are also called the oracles.
- **Accepted state.** If an instance, say Π^x goes to an accept state after receiving the last expected protocol message, it will be in accepted state. If all the communicated (sent and received) messages by Π^x are concatenated in sequence, it forms the session identification (sid) of Π^x for the current session.
- **Partnering.** Two instances Π^x and Π^y are called partners to each other if the following three criteria are simultaneously fulfilled: 1) both Π^x and Π^y are in accepted state; 2) both Π^x and Π^y mutually authenticate each other and share the same sid; and 3) both Π^x and Π^y are mutual partners of each other.

- **Freshness.** If the session key K between UE and $eNodeB$ is unknown to an adversary $\mathcal{A}$ using the following reveal oracle query $Reveal(\Pi^x)$ defined below, Π^x_{Ui} or Π^y_{SNj} is said to be fresh.
- **Adversary.** $\mathcal{A}$ is a polynomial-time adversary having full control over communication channels. The full control means that $\mathcal{A}$ can read/modify the exchanged messages as well as inject fabricated messages during the communication. In addition, $\mathcal{A}$ will have access to the following queries [32]:
  - $Execute(\Pi^x_{UE}, \Pi^y_{eNodeB})$: This query is modeled as a passive attack. Using this query, $\mathcal{A}$ can intercept the messages exchanged between any two instances of the corresponding participants UE and $eNodeB$.
  - $Reveal(\Pi^x)$: $\mathcal{A}$ can compromise the present session key K established by Π^x (and its partner).
  - $Send(\Pi^x, msg)$: This query is modeled an active attack, where a participant instance Π^x can transmit a message msg and also receive a response message.
  - $Test(\Pi^x)$: This query is modeled for the security of the session key K. The semantic security of session key K between UE and $eNodeB$ following the indistinguishability in the ROR model [29] is implemented with the help of this query. Before the game starts, an unbiased coin c is flipped whose output is only revealed to $\mathcal{A}$. This outcome determines if the output of the $Test$ query is consistent or not. If $\mathcal{A}$ executes this query and the session key K is fresh, Π^x returns K in case $c = 1$ or a random number when $c = 0$. Otherwise, a null ($\perp$) output is generated.
- **Random oracle.** Each entity and $\mathcal{A}$ are provided with the access of a collision-resistant one-way cryptographic hash function, which is modeled as a random oracle, say $\mathcal{OH}$.

4.3.2 Security proof

The session key security of the proposed protocol is given in Theorem 1.

Theorem 1 *Let $\mathcal{A}$ be a polynomial time adversary running against the proposed protocol, say $\mathcal{P}$ in the ROR model, $Adv^{ECDDHP}_{\mathcal{A}}(rt)$ the advantage of breaking the ECDDHP in time rt by $\mathcal{A}$, and q_h and $|Hash|$ represent the number of $\mathcal{OH}$ queries and range space of hash function, respectively. $\mathcal{A}$'s advantage in breaking semantic security of $\mathcal{P}$ for deriving the session key between UE and $eNodeB$ in time rt can be estimated by*

$$Adv^{ECDDHP}_{\mathcal{A},\mathcal{P}}(rt) \leq \frac{q_h^2}{|Hash|} + 2Adv^{ECDDHP}_{\mathcal{A}}(rt).$$

Proof The similar proof is followed in proving this theorem as done in other authentication protocols [31,32,33,34,35]. A sequence of four games, say Gm_i ($i = 0, 1, 2, 3$) are defined, where $Succ_i$ represents the winning probability of an adversary $\mathcal{A}$ in the game Gm_i in which $\mathcal{A}$ can guess the random bit c correctly. Let the advantage of $\mathcal{A}$ in game Gm_i be denoted by $Adv^{Gm_i}_{\mathcal{A}} = Pr[Succ_i]$.

The detailed description of each game is given as follows.

– Gm_0**:** This is the first game which is a real security experiment run between $\mathcal{A}$ and a challenger against the proposed protocol $\mathcal{P}$ in the ROR model. Since the bit c is chosen in the beginning of G_0, from the semantic security we have,

$$Adv_{\mathcal{A},\mathcal{P}}^{ECDDHP}(rt) = |2.Adv_{\mathcal{A}}^{Gm_0} - 1|. \tag{33}$$

– Gm_1**:** An eavesdropping attack is modeled in this game in which $\mathcal{A}$ intercepts the transmitted messages $Msg_1 = \{R_{UE}, R'_{UE}, s_{UE}, m_{UE}, w_{UE}, Y_{UE}, I_{UE}, t_1\}$, $Msg_2 = \{R_{eNB}, s_{eNB}, m_{eNB}, w_{eNB}, R', I_{eNB}\}$ in addition to the $Msg_3 = \{h_1(h_2(K)||h_2(PK_{UE})||h_2(R_{UE}))\}$ during the normal handover procedure of $\mathcal{P}$. This game simulates the eavesdropping attacks of $\mathcal{A}$ by making $Execute(\Pi_{UE}^{x}, \Pi_{eNodeB}^{y})$ query. $\mathcal{A}$ then finishes it by making the $Test$ query so as to decide whether the output of $Test$ query is a real session key or a random value. The session key established between UE and $eNodeB$ is given by $K = r'R'_{UE} = r'_{UE}R' = (r'_{UE}.r')G$.
To compute the session key K, $\mathcal{A}$ needs r'_{UE} and r'. However, the intercepted messages do not help in determining r' and r'_{UE} to construct K. Therefore, intercepting does not facilitate in winning this game, and we have,

$$Adv_{\mathcal{A}}^{Gm_1} = Adv_{\mathcal{A}}^{Gm_0}. \tag{34}$$

– Gm_2**:** This game differs from the previous game G_1 due to inclusion of the simulations of the $Send$ and the hash $(\mathcal{OH})$ queries by $\mathcal{A}$. Gm_2 simulates an active attack by $\mathcal{A}$ to convince an entity to accept $\mathcal{A}$'s fake (modified) messages. $\mathcal{A}$ can repetitively query $\mathcal{OH}$ queries to obtain collisions, but all the communicated messages Msg_1, Msg_2 and Msg_3 are associated with either a fresh random or a current time-stamp or both. Therefore, there is no possibility of collision while making $Send$ queries with the help of the one-way hash function (see Definition 1). Using the results from the birthday paradox, we obtain the following result:

$$|Adv_{\mathcal{A}}^{Gm_2} - Adv_{\mathcal{A}}^{Gm_1}| \leq \frac{q_h^2}{2|Hash|}. \tag{35}$$

– Gm_3**:** This is the final game, where $\mathcal{A}$ tries to compute the session key shared between UE and $eNodeB$. Note that the session key computed between UE and $eNodeB$ is $K = r'R'_{UE} = r'_{UE}R' = (r'_{UE}.r')G$. From the intercepted messages Msg_1, Msg_2 and Msg_3, $\mathcal{A}$ can try to derive $(r'_{UE}.r')G\ (= K)$ using $R'_{UE} = r'_{UE}G$ and $R' = r'G$. This is equivalent to solving the elliptic curve decisional Diffie-Hellman problem (ECDDHP) (see Definition 3). Hence, given the time rt to $\mathcal{A}$ and the intractability of ECCDHP produce the following relation:

$$|Adv_{\mathcal{A}}^{Gm_3} - Adv_{\mathcal{A}}^{Gm_2}| \leq Adv_{\mathcal{A}}^{ECDDHP}(rt). \tag{36}$$

Since all the queries are made by $\mathcal{A}$, only guessing the bit c is left in order to win the game after the $Test$ query is made by $\mathcal{A}$. Therefore, it follows that

$$Adv_{\mathcal{A}}^{Gm_3} = \frac{1}{2}. \tag{37}$$

Equations (33) and (34) give the following relation:

$$\frac{1}{2}Adv_{\mathcal{A},\mathcal{P}}^{ECDDHP}(rt) = |Adv_{\mathcal{A}}^{Gm_0} - \frac{1}{2}| = |Adv_{\mathcal{A}}^{Gm_1} - \frac{1}{2}|. \quad (38)$$

Equations (37) and (38) further give the following result:

$$\frac{1}{2}Adv_{\mathcal{A},\mathcal{P}}^{ECDDHP}(rt) = |Adv_{\mathcal{A}}^{Gm_1} - \frac{1}{2}| = |Adv_{\mathcal{A}}^{Gm_1} - Adv_{\mathcal{A}}^{Gm_3}|. \quad (39)$$

The triangular inequality gives the following result:

$$|Adv_{\mathcal{A}}^{Gm_1} - Adv_{\mathcal{A}}^{Gm_3}| \leq |Adv_{\mathcal{A}}^{Gm_1} - Adv_{\mathcal{A}}^{Gm_2}| + |Adv_{\mathcal{A}}^{Gm_2} - Adv_{\mathcal{A}}^{Gm_3}|. \quad (40)$$

From Equations (35), (36) and (40), we have the following relation:

$$|Adv_{\mathcal{A}}^{Gm_1} - Adv_{\mathcal{A}}^{Gm_3}| \leq \frac{q_h^2}{2|Hash|} + Adv_{\mathcal{A}}^{ECDDHP}(rt). \quad (41)$$

Equations (39) and (41) give the following relation:

$$\frac{1}{2}Adv_{\mathcal{A},\mathcal{P}}^{ECDDHP}(rt) \leq \frac{q_h^2}{2|Hash|} + Adv_{\mathcal{A}}^{ECDDHP}(rt). \quad (42)$$

Finally, multiplying both sides of Equation (42) by a factor of 2, the required result is obtained:

$$Adv_{\mathcal{A},\mathcal{P}}^{ECDDHP}(rt) \leq \frac{q_h^2}{|Hash|} + 2Adv_{\mathcal{A}}^{ECDDHP}(rt).$$

□

4.4 Automated verification of the proposed scheme using ProVerif analysis tool

To analyze and verify security properties of cryptographic protocols, many automatic verification tools were developed such as AVISPA [36], SATMC [37], and ProVerif [38] etc. ProVerif is the most professional and updated automatic verification tool capable of proving security properties such as secrecy, authentication and anonymity etc. Secrecy is defined as the confidentiality of a message. Secrecy generally means that the attacker should not be able to obtain a piece of data. This is generally referred to as syntactic secrecy. Sometimes a more general notion as strong secrecy is defined as the ability of the attacker to identify the changing of the value of a message. However he has no information on the value of the data. Authentication generally means that if an entity A is in contact with another party B, the party (B) should be contact with A as well. The two parties should share the same values of parameters as well. In this analysis tool, the protocol is converted into horn clauses [39]. The security properties are seen as queries into such clauses. For example, the " **query** attacker (Data)" asks the automatic tool to check the secrecy of the message *Data*. Authentication can be checked in this tool as well. This is achieved by a set of correspondence between events. For example the " **query** event end (data 2) → event begin (data 1)

" , checks if the event *"end"* has been executed with *data 2* and the event *"begin"* was executed with *data 1*. This tool has been used to validate many authentication schemes such as [40,41]. In the following we describe the results of the validation of our security protocol using ProVerif analysis tool.

– **Secrecy** To analyze the confidentiality of the session key (K), the following query was checked:

query attacker (K)

As expected, the ProVerif analysis tool confirms that adversary has no chance to obtain the secret session key (K).

– **Authentication of eNB to UE** For the sake of authentication of eNB to the UE, the following events were used :

query event endUE (K, ReNB) $\rightarrow$ event beginUE (K, ReNB)

As expected, the results confirm our claim with respect to the authentication of the eNB to the UE. Since, not only the event "beginUE" occurs before the event "endUE", but also the same parameters were exchanged in the same session.

– **Authentication of UE to eNB** For the sake of authentication of UE to the eNB, the following events were used :

query event endeNB (Rue,R$'$ue) $\rightarrow$ event begineNB (Rue,R$'$ue)

As shown by the ProVerif analysis tool, the event "begineNB" was executed before "endeNB" in the same session. This means that no execution of begineNB was executed after the execution of endeNB. Thus, the authentication of the UE to the eNB is validated. Full ProVerif script is available in [42].

5 Conclusion

In this paper, we analyzed a recently suggested handover authentication scheme based on proxy signature in LTE network. Although this scheme enjoys certain advantages such strong unforgrability, authenticity, user anonymity, however we indicated serious vulnerabilities such as being prone to DoS attack, eNB impersonation attack and lack of perfect forward secrecy. These attacks let the adversary make the availability of the network into danger and impersonate the eNB to the user. To resolve such threats, we proposed an improved scheme resistant against such attacks. Further we validated the security of the proposed scheme using ROR model and ProVerif

analysis tool. The results confirm our claims concerning the security of the proposed scheme. In addition,the performance analysis shows that compared to other schemes, the performance of our proposed scheme is more reasonable.

References

1. S. Sesia, I. Toufik, and M. Baker, LTE: The UMTS Long Term Evolution: from Theory to Practice, John Wiley and sons, 2011.
2. D. Forsberg, "LTE Key Management Analysis with Session Keys Context," Computer Communications, vol. 33, no. 16, pp. 1907-1915, 2010.
3. C. K. Han, H. K. Choi, and I. H Kim, "Building Femtocell More Secure with Improved Proxy Signature," Proceedings of IEEE GLOBECOM, pp. 1-6, December 2009.
4. J. Cao, M. Ma, H. Li, and Y. Zhang, "A Survey on Security Aspects for LTE and LTE-A Networks," IEEE Communications Surveys Tutorials, no. 99, pp. 1-20.
5. J. Cao, H. Li, M. Ma, Y. Zhang, and C. Lai, "A Simple and Robust Handover Authentication between HeNB and eNB in LTE Networks," Computer Networks, vol. 56, no. 8, pp. 2119-2113, 2012.
6. C. Ma, K. Xue, and P. Hong, "A Proxy Signature-Based Re-authentication Scheme for Secure Fast Handoff in Wireless Mesh Networks," International Journal of Network Security, vol. 15, no. 1, pp. 104-114, 2013.
7. Q. Jing, Y. Zhang, A. Fu, and X. Liu, "A Privacy Preserving Handover Authentication Scheme for EAP-Based Wireless Networks," Proceedings of IEEE GLOBECOM, pp. 1-6, December 2011.
8. Y.Qiu, M.Ma, X Wang, "A Proxy Signature-based Handover Authentication Scheme for LTE Wireless Networks," Journal of Network and Computer Applications, 2017.
9. H. Wang, A.R. Prasad, "Fast authentication for inter-domain handover Telecommunications and Networking," ICT , pp. 291-313, 2004.
10. C. Zhang, R. Lu, P. Ho, A. Chen, "A location privacy preserving authentication scheme in vehicular networks," in IEEE Wireless Communications and Networking Conference, 2008.
11. L Cai, S. Machiraju, H Chen, "CapAuth: A Capability-based Handover scheme," in Proceedings of IEEE INFOCOM , USA, 2010.
12. J. Qi, Y. Zhang, A. Fu, X. Liu, "A Privacy Preserving Handover Authentication Scheme for EAP-Based Wireless Networks," in Proceeding of GLOBECOM, 2011.
13. A. Mishra, M. Shin, W. Arbaugh, "An empirical analysis of the IEEE 802.11 MAC layer handoff process," ACM SIGCOMM Computer Communication, pp. 93-102, 2003.
14. Y. Kim, W. Ren, J. Jo, M. Yang, Y. Jiang, J. Zheng, "SFRIC: a secure fast roaming scheme in wireless LAN using ID-based cryptography," in Proceedings of IEEE International Conference on Communication, 2007.
15. J. Cao, M. Ma, H. Li, and Y. Zhang, "A Survey on Security Aspects for LTE and LTE-A Networks," IEEE Communications Surveys Tutorials, pp. 1-20.
16. A. Bohk, L. Buttyn, and L. Dra, "An Authentication Scheme for Fast Handover between WiFi Access Points," Proceedings of ACM Wireless Internet Conference, pp. 22-24, October 2007.
17. A. Mishra, H. S. Min, N. L. Petroni,, Jr, T.C. Clancy, and W. A. Arbaugh, "Proactive Key Distribution Using Neighbor Graphs," IEEE Wireless Communications, vol. 11, no. 1, pp. 26-36, 2004.
18. Y. E. H. E. Idrissi, N. Zahid, and M. Jedra, "Security Analysis of 3GPP (LTE) WLAN Interworking and A New Local Authentication Method Based on EAP-AKA," Proceedings of International Conference on Future Generation Communication Technology , pp. 137-142, 2012.
19. J. Choi, S. Jung, "A handover authentication using credentials based on chameleon hashing," IEEE Communications Letters, pp. 54-56, 2010.
20. R. L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-key Cryptosystems," Proceedings of ACM Communications, pp. 120-126, Feb 1978.
21. M. Mambo, K. Usuda, and E. Okamoto, "roxy signatures: Delegation of the Power to Sign Messages," IEICE Transaction of Fundamentals, Vols. E79-A, pp. 1338-1353, 1996.
22. N. Koblitz, "Elliptic Curve Cryptosystems," Mathematics of Computation, vol. 48, no. 177, pp. 203-209.
23. "3rd Generation Partnership Project; Technical Specification Group Service and System Aspects; 3GPP System Architecture Evolution (SAE); Security architecture (Rel 11) 3GPP TS 33.401," 2011.

24. M. Abdalla, P. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in 8th International Workshop on Theory and Practice in Public Key Cryptography (PKC'05), Lecture Notes in Computer Science (LNCS), vol. 3386, Les Diablerets, Switzerland, 2005, pp. 65-84.
25. M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo. "Secure Remote User Authenticated Key Establishment Protocol for Smart Home Environment," IEEE Transactions on Dependable and Secure Computing, 2017, DOI: 10.1109/TDSC.2017 .2764083.
26. A. K. Das, M. Wazid, N. Kumar, M. K. Khan, K. -K. R. Choo, and Y. Park. "Design of Secure and Lightweight Authentication Protocol for Wearable Devices Environment," IEEE Journal of Biomedical and Health Informatics 2017, DOI: 10.1109/ JBHI.2017.2753464.
27. S. Roy, S. Chatterjee, A. K. Das, S. Chattopadhyay, N. Kumar, and A. V. Vasilakos. "On the Design of Provably Secure Lightweight Remote User Authentication Scheme for Mobile Cloud Computing Services," IEEE Access, vol. 5, no. 1, pp. 25808-25825, 2017.
28. S. Roy, S. Chatterjee, A. K. Das, S. Chattopadhyay, S. Kumari, and M. Jo. "Chaotic Map-based Anonymous User Authentication Scheme with User Biometrics and Fuzzy Extractor for Crowdsourcing Internet of Things," IEEE Internet of Things Journal, 2017, DOI: 10.1109/JIOT.2017.2714179.
29. S. Chatterjee, S. Roy, A. K. Das, S. Chattopadhyay, N. Kumar, and A. V. Vasilakos. "Secure Biometric-Based Authentication Scheme using Chebyshev Chaotic Map for Multi-Server Environment," IEEE Transactions on Dependable and Secure Computing, 2016, DOI: 10.1109/TDSC.2016.2616876.
30. C.-C. Chang and H.-D. Le, "A Provably Secure, Efficient and Flexible Authentication Scheme for Ad hoc Wireless Sensor Networks," IEEE Transactions on Wireless Communications, vol. 15, no. 1, pp. 357-366, 2016.
31. V. Odelu, A. K. Das, and A. Goswami, "A Secure Biometrics-Based Multi-Server Authentication Protocol using Smart Cards," IEEE Transactions on Information Forensics and Security, vol. 10, no. 9, pp. 1953-1966, 2015.
32. P. Sarkar, "A Simple and Generic Construction of Authenticated Encryption with Associated Data," ACM Transactions on Information and System Security, vol. 13, no. 4, pp.33, 2010.
33. J. Srinivas, A. K. Das, M. Wazid, and N. Kumar, "Anonymous Lightweight Chaotic Map-Based Authenticated Key Agreement Protocol for Industrial Internet of Things," IEEE Transactions on Dependable and Secure Computing, 2018, DOI: 10.1109 TDSC.2018.2857811.
34. J. Srinivas, A. K. Das, N. Kumar, and J. J. P. C. Rodrigues, "Cloud Centric Authentication for Wearable Healthcare Monitoring System," IEEE Transactions on Dependable and Secure Computing, 2018, DOI: 10.1109/TDSC.2018.2828306.
35. E. Barker, "Recommendation for Key Management," Special Publication 800-57 Part 1 Rev. 4, NIST, 01/2016. Accessed on May 2018.
36. A.Armando, D.Basin, "The AVISPA tool for automated Validation of Internet Security protocols and Applications," in Computer Aided Verification, 2005.
37. A. Armando , R.Carbone, L.Compagna, "SATMC: a SAT-based Model Checker for Security-critical Systems," in Tools and Algorithms for the construction and Analysis of Systems, 2014.
38. B. Blanchet, "Automatic Proof of Strong Secrecy for Security Protocols," IEEE symposium on Security and privacy, 2004.
39. R.Kusters, T.Trudernung, "Using ProVerif to Analyze Protocols with Diffie-Hellman Exponentiation," in 22nd IEEE Computer Security Foundation Symposium, NewYork, 2009.
40. S. M.Pournaghi,B. Zahednejad, M.Bayat, Y.Farjami " NECPPA: A novel and efficient conditional privacy-preserving authentication scheme for VANET". Computer Networks, 134, 78-92,2018.
41. B. Blanchet, A.Chaudhuri. "Automated formal analysis of a protocol for secure file sharing on untrusted storage". In Security and Privacy, 2008. SP 2008. IEEE Symposium on (pp. 417-431), 2008.
42. B.Zahednejad, www.github.com, "https://github.com/lordbehnam/proverif-script /blob/master/ProVerif.txt" , 2018.