An LTE Authentication and Key Agreement Protocol Based on the ECC Self-Certified Public Key

Xiaofeng Lu, *Member, IEEE*, Fan Yang, Luwen Zou, Pietro Lio, and Pan Hui, *Fellow, IEEE*

***Abstract*— After analyzing the long-term evolution (LTE) authentication and key agreement process (EPS-AKA), its existing security vulnerabilities are pointed out. Based on elliptic curve cryptography (ECC) self-certified public keys, this paper proposes an ECC self-certified authentication key agreement scheme (ESC-AKA). This scheme includes the addition of a trusted center (TC), which generates the public keys for the home subscriber server (HSS), the mobility management entity (MME), and the user equipment (UE). Three communication protocols are designed, including MME/HSS registration, UE registration, and UE access. A strand space model is used to carry out the formal analysis, and performance and security analyses are carried out. The results show that this scheme can compensate for the security vulnerabilities of the original EPS-AKA scheme. It implements the encrypted transmission of the international mobile subscriber identity (IMSI), and realizes the mutual authentication between the HSS and MME, the MME and UE, and the HSS and UE. Because the self-certified public key cryptosystem is adopted in this scheme, communication encryption is ensured, and the risk of the TC simultaneously mastering the public and private keys is avoided. This scheme is proven to be effective in protecting the communication security of the LTE network.**

***Index Terms*— LTE, EPS-AKA, self-certified public keys, authentication, key agreement protocol.**

I. Introduction

LONG-TERM evolution (LTE) [1] is a new generation of broadband wireless mobile communication technology, and includes the use of orthogonal frequency division multiplexing (OFDM), multiple-input multiple-output (MIMO), and other advanced technologies. These technologies greatly improve the data transmission speed and meet the growing demands of users for the quality of multimedia services. 4G LTE technology is undoubtedly the most widely used and mature technology. In the Ericsson Mobility report in June 2022, global 4G population coverage was reported to be around 85% at the end of 2021, and is forecast to reach around 95% in 2027. There are currently 809 commercial 4G networks deployed around the world [2].

The widespread use of LTE networks and the upcoming 5G network place more stringent requirements on the security of LTE networks [3]. The multi-access function provided by LTE networks greatly increases the risk of LTE systems, so the design of a secure and efficient authentication protocol is imperative. However, the authentication and key agreement protocol in LTE networks (EPS-AKA) [4] has been demonstrated to have numerous security vulnerabilities [5]. To ensure the security of the current LTE network and the future mobile Internet, the investigation of a new type of security protocol is not only related to the user's personal privacy and property, but also has an important impact on social and national long-term stability [6].

In the mobile communication network, if a user wants to obtain a network service, he/she must first perform user authentication and key negotiation through an access network. Compared with 3G access network, LTE network reduces the number of nodes [5]. There is only one node e-NodeB in the access network, which makes the original tree branch structure flat, thus reducing the system complexity and system delay.

EPS-AKA is the authentication and key agreement scheme developed by the 3rd Generation Partnership Project (3GPP) organization on the basis of the 3G-AKA scheme for LTE networks [7], and is based on the "challenge/response" process. Via the negotiation and authentication between mobile users and network elements, the process of communication key acquisition is completed, which provides security for encrypted communication.

By analyzing the EPS-AKA process, the following security problems can be found.

(1) Plaintext transmission of the international mobile subscriber identity (IMSI): when the user equipment (UE) accesses the network for the first time, or when the mobility management entity (MME) requires the user to transmit the IMSI, the user device transmits the IMSI in plaintext. In this way, IMSI can easily be intercepted, which can then lead to user identity disclosure [8], man-in-the-middle (MITM) attacks [9], ToRPEDO attacks [10], denial-of-service attacks (DoS) and other

Manuscript received 28 January 2022; revised 30 July 2022; accepted 13 September 2022; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor J. S. Sun. Date of publication 26 September 2022; date of current version 16 June 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62136006 and Grant 61472046 and in part by the National Key Research and Development Program of China under Grant 2020YFB2104700. *(Corresponding author: Xiaofeng Lu.)*

Xiaofeng Lu and Luwen Zou are with the National Engineering Research Center of Mobile Network Technologies, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: luxf@bupt.edu.cn).

Fan Yang is with the China Unicom Network Technology Research Institute, Beijing 100032, China (e-mail: yangf94@chinaunicom.cn).

Pietro Lio is with the Computer Laboratory, University of Cambridge, CB2 1TN Cambridge, U.K. (e-mail: pl219@cam.ac.uk).

Pan Hui is with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, China (e-mail: pan.hui@telekom.de).

Digital Object Identifier 10.1109/TNET.2022.3207360

dangerous events. In an MITM attack, the attacker obtains the IMSI and then tries to register with the base station [9]. ToRPEDO attack is able to verify whether a victim device is present in a geographical cell with less than 10 calls [10].

(2) Plaintext transmission of the service network identity (SNID) and authentication vector (AV) (1,…,n): the communication between the MME and the home subscriber server (HSS) uses plaintext transmission, so there is a high security risk. For example, SNID leakage will cause network spoofing, MITM attacks, pseudo base stations, and the disclosure of the AV (1,…n). This allows attackers to disguise themselves as an MME or HSS to communicate with users, thereby stealing private user data.

(3) The public key K between the UE and the HSS coexists for a long period of time. Because the encrypted transmission is based on the shared key K, once K is cracked, the data transferred by the user will no longer be confidential.

(4) The initial access message of the UE lacks authentication, and a DoS attack can be easily initiated. A DoS attack makes the service unavailable and prevents legitimate users of the service from using it [9].

Table I refers to the notations and their representations used throughout this paper.

In 1991, Girault first proposed the self-certified public key system [11]. The self-certified public key system is different from the certificate public key system because the authority only takes part in the generation of the user's self-certified public key, but does not have the user's private key, and the self-certified public key contains two secrets: the user's private key and the private key of the authority. So the authority cannot fake the certificates and fake the users [12].

The self-certified public key has the advantages of small overhead and high security, which make it very suitable for the authentication process of mobile devices. Aiming at the security flaws of the EPS-AKA protocol, an LTE network user authentication and key agreement scheme, namely ECC self-certified authentication key agreement (ESC-AKA), is put forward based on the elliptic curve self-certified public key. In the scheme, three communication protocols are designed, including MME/HSS registration, UE registration, and UE access. The authentication test theory based on the serial space model is used to conduct a formal analysis of each protocol.

The security and efficiency analysis results show that this scheme can effectively resist various attacks launched against the original EPS-AKA protocol vulnerabilities. ESC-AKA can resist internal and external eavesdropping, counterfeiting and tampering attacks, AK attacks caused by SQN guessing, attacks caused by the leakage of shared key K value, and DoS attacks, forged authentication request attack.

The ESC-AKA protocol achieves the third level of public key system security. The 5G-AKA protocol uses plaintext transmission to return error messages, which may leak the UE's location information and cause serious consequences [13], [14]. Compared with 5G-AKA, due to the handling of the returned error message, the confidentiality of the user's location is protected. Compared with the high deployment cost of 5G network elements [15], the ESC-AKA protocol could be used to improve the security of LTE networks before 5G is deployed in some countries and regions.

TABLE I
NOTATIONS AND THEIR REPRESENTATION

Notation	Representation
UE	User EquipMent
MME	Mobile ManagementEntity
HSS	Home Subscriber Server
TC	Trust Center
EPC	Evolved Packet Core
UMTS	Universal Mobile Telecommunications Service
IMSI	International Mobile Subscriber Identity
SNID	Serving Network Identity
AV	Authentication Vector
CK	Cipher Key
IK	Integrity Key
SQN	sequence number
IDC	identity certificate
USIM	Universal Subscriber Identity Module
ID_{HSS}	HSS identity
ID_{MME}	MME identity
ID_{UE}	UE identity
PK_{UE}	Public Key of UE
PK_{MME}	Public Key of MME
PK_{HSS}	Public Key of HSS
PK_{TC}	Public Key of TC
SK_{UE}	Secret Key of UE
SK_{MME}	Secret Key of MME
SK_{HSS}	Secret Key of HSS
SK_{TC}	Secret Key of TC
p	Large prime number
E	Elliptic curve order

II. RELATED WORK

In view of the security problems of the EPS-AKA protocol, many scholars have proposed their own schemes. To retain the EPS-AKA architecture to the greatest extent and make it a full mutual entity authentication protocol, Kien proposed the EAKA protocol, which is a mutual authentication protocol between the UE and HSS that operates by keeping a secret parameter of the AV in the HSS [16]; however, the problem of plaintext transmission still exists. Bai *et al.* studied the leak security risks of the key K and proposed the ES-AKA scheme [17], but the scheme does not solve the problem of plaintext transmission.

To solve the problem of plaintext transmission, Wang proposed an identity-based public key cryptosystem [18]. The scheme can truly protect the IMSI when transmitted, but there exists the security risk of the trusted center (TC) storing the public and private keys of users. Parne *et al.* also solved the problem of plaintext transmission by pre-sharing a dynamic symmetric key between the UE and HSS, and solved the problem of key exposure and key freshness [19]. In the scheme,

cocktail therapy is used to generate AVs, which reduces the computational overhead of the HSS and the communication overhead from the HSS to the MME. Based on symmetric and asymmetric encryption, Hamandi *et al.* proposed a scheme to enhance user identity privacy, which can defend against all kinds of active and passive attacks [20], [21]. However, the scheme does not consider non-traceability. Prasad and Manoharan proposed a two-way encryption and authentication algorithm (DS-AKA) [22] based on the combination of the EIGamal digital signature and the Diffie-Hellman key exchange protocol; the algorithm uses the certificate-based public key cryptosystem, which requires the TC to issue a digital certificate. Because there is no time stamp, the algorithm may be subject to replay attacks. The scheme proposed in this paper applies the self-certified cryptosystem to generate a public and private key, which can keep the secret of the key pair. The proposed scheme solves the security risk of the TC forging the certificate.

Abdrabou *et al.* proposed a scheme to solve the existing problems based on simple cryptographic exponential key exchange and symmetric encryption [23]. In the scheme, the UE and MME compute the secret dynamic key through pre-shared parameters, and then use the dynamic key to encrypt the transmitted data in the latter authentication. However, it is unreasonable to pre-share multiple parameters. Purkhiabani and Salahi proposed a scheme to improve the performance based on a shared key [24]. The scheme reduces the bandwidth overhead, but it uses plaintext in the data transmission.

To avoid using an asymmetric encryption system but keep the secret of the IMSI, Zhu *et al.* proposed an LTE authentication and key agreement scheme [25] to strengthen the protection of user identity confidentiality by changing the IMSI; however, the scheme does not solve the problem of the plaintext transmission of the SNID. Choudhury *et al.* used the Dynamic Mobile Subscriber Identity (DMSI) instead of the IMSI to prevent the leakage of the IMSI, and the value of the DMSI changes after every authentication [26]. The scheme can effectively enhance user privacy, but the new functions added and changing the DMSI increase the computational and communication overhead. Fan *et al.* considered that the HSS writes a singed pseudo IMSI in the Subscriber Identity Module (SIM) card of the UE in advance [27], and then the UE uses the pseudo IMSI for authentication. The scheme can solve the problem of privacy leakage to some extent, but is impractical for multiple UEs.

Singh and Saini provided a robust authentication scheme [28] to solve the problem of DDoS attacks in LTE systems. Ekene *et al.* proposed a certificate-based public key encryption scheme [29], but it cannot avoid the security problem of the TC forging certificates. Arapinis *et al.* proposed a set of patching methods for the original AKA protocol in view of the vulnerable replay attack in LTE systems [30]. Li and Wang proposed a public key-based SE-EPS-AKA scheme for IMSI plaintext transmission, AV risk transmission, and the long-term storage of a shared key K [31]. However, this scheme was also proved to be risky by Abdo *et al.* [32]. Panda and Chattopadhyay proposed an improved authentication scheme for LTE/LTA-A networks based on ECC, ECDH, and the Salasa2o algorithm [33]. The scheme can effectively solve the security problems in the authentication process and avoid the system suffering from security attacks. However, in the registration, the UE sends the IMSI as a parameter to the HSS in the form of plaintext transmission, which may lead to the leakage of the IMSI.

TABLE II
COMPARISON ON RSA AND ECC

Crack time (MIPS year)	RSA Key length	ECC Key length	RSA/ECC Key length ratio
10^4	512	106	5:1
10^8	768	132	6:1
10^{12}	1024	160	7:1
10^{20}	2048	210	10:1

Ma *et al.* considered authentication and key agreement, privacy, and perfect forward/backward secrecy, and proposed a scheme consisting of two phases: the initial attach phase and the handover authentication phase [34]. In the initial attach phase, the encryption algorithm elliptic curve E is used in the HSS to generate partial system parameters. One of its purposes is to prepare for the future handover authentication. In the handover authentication phase, three kinds of scenarios in LTE-A networks in which a UE moves from the source base station to the target base station are described. The scheme achieves secure authentication in different handover scenarios with enhanced complexity in the handover authentication phase.

III. SELF-CERTIFIED PUBLIC KEY CRYPTOSYSTEM BASED ON ELLIPTIC CURVE

A. Elliptic Curve Public Key Cryptosystem

Public key cryptography is based on mathematical difficulties, and the three most widely used mathematical problems are now:

1. Decomposition of large integers
2. The discrete logarithm problem on a finite field multiplicative group
3. The discrete logarithm problem on elliptic curves

At present, the first two mathematical problems have the sub exponential time algorithm.

In 1985, Indra and Taneja put forward the elliptic curve cryptosystem (ECC), and the security of ECC is based on the intractability of the elliptic curve discrete group [35]. The best solution to this problem is the exponential time algorithm. Generally speaking, RSA and Diffie-Hellman key exchange protocols need more than 1024 bits modulus to be secure, but for ECC, only 160 bits modulus can achieve the same security level [36]. Table II shows the comparison on RSA and ECC.

The ECC has the following advantages: it can achieve high security with a short key; digital signatures and certificates are small; calculation speed is fast. Due to these advantages, the ECC public key algorithm is widely used in the field of cryptography. Indra and Taneja proposed a two-way authentication key management scheme for wireless sensor networks based on ECC algorithm [35]. Haripriya and Kulothungan proposed an ECC-based IoT self-authentication key management scheme [36].

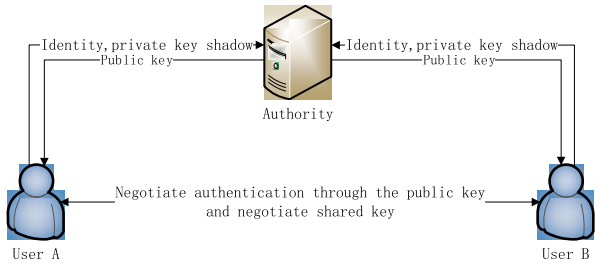

Fig. 1. Self-certified public key principle diagram.

B. The Basic Principle of Self-Certified Public Key

Fig. 1 is a schematic diagram of the authentication principle of self-certified public key. In this system, the user first generates its own private key, and the encrypted private key (called private key shadow) is sent to the authority. The authority makes a signature of the user's private key shadow and user's identity to produce the user's self-certified public key [37]. Any third party can verify the user's public key through the public key of the authoritative organization and the user's private key shadow; the authority does not participate in the authentication process of the user.

C. Security Level of Public Key Cryptography

Girault defines three security levels for public key cryptography [11]:

Level 1: Corresponding to identity based public key system, the authority grasps the public and private key of the users and can impersonate any user to communicate in the network, so the authority must ensure that it is fully trusted.

Level 2: Corresponding to certificate based public key cryptosystem, authority does not grasp or not easily calculate the user's private key, but can generate fake proof of public key and then fake the user to communicate, therefore the authority must also be fully trusted.

Level 3: Corresponding to self-certified public key system, authority does not grasp or not easily calculate the user's private key, and if the authority exists cheating, the cheating behavior can be proved.

The three security levels theory is widely used in the security level determination of certificateless public key system [38], [39], [40], [41]. ESC-AKA system uses the self-certified key generation algorithm in public key cryptosystem based on the elliptic curve, adds the TC to produce the public key of each element on the basis of the original LTE network. So, the security level of the ESC-AKA system can reach level 3.

IV. ESC-AKA Scheme Design

A. Introduction of Trusted Center TC

In this scheme, a TC is added to the original LTE network, and the TC can be set on the HSS side, which is responsible for the distribution and management of the public key.

TC's features include:

① TC accepts the registration request of the local HSS/MME/UE, and verifies the identity (ID) of the HSS/MME and the IMSI of the UE.

② In the registration phase of HSS/MME/UE, the self-certified public key is generated and distributed by TC according to the private key sent by each network element.

③ After the HSS/MME registration is completed, the HSS/MME's public key and the ID are stored in the local database if TC receives the authentication from the HSS/MME.

④ During the UE registration phase, TC is responsible for sending the HSS public key requested by UE to the UE to ensure that the IMSI uses the public key of the HSS to encrypt the transmission during the UE access phase.

⑤ The TC public key can be written into the SIM card when the SIM card is shipped out of the factory and can be updated periodically through the key exchange protocol, such as the Diffie-Hellman key exchange protocol.

⑥ TC has the anti DDoS attack characteristics, such as establishing a sound firewall and packet filtering mechanism, when a large number of registration requests are received from the same user in a short time. The user can be regarded as an attacker and these requests are shielded. If a large number of new users initiate registration requests, multiple TCs can be used to achieve load balancing. The TC will not have much communication pressure because the UE registration process is enabled only when the UE is first registered to the network or a HSS public key error occurred.

B. ESC-AKA Schema Overall Architecture

The ESC-AKA scheme is divided into four phases: TC system initialization, MME/HSS registration, UE registration, and UE access. After the TC system is initialized, other network elements can execute their registration process. The registration process is the process of application for the public key by each network element to the TC. The MME and HSS can apply the public key directly from the local TC, and the UE needs to apply it from the TC of the home through the MME. In this scheme, the self-certified public key generation algorithm is used only during the registration phase of each network element, and the original EPS-AKA authentication mechanism is still used in the UE access phase. Therefore, this protocol has three main parts: the first is the MME/HSS registration protocol, the second is the UE registration protocol, and the third is the UE access protocol.

The overall architecture of the ESC-AKA scheme is shown in Fig. 2, where the TC is responsible for the MME, the HSS, UE registration, ID plausibility checking, and the generation of a public key for each network element. The HSS is responsible for managing the UE contract data, UE authentication, the maintenance of the shared key K, and the generation of AVs. The MME is responsible for positioning the UE, paging process management, and forwarding the UE authentication data.

C. Trusted Center Initialization

The initialization process of the trusted center TC is according to the self-certified public key system based on elliptic curve, and the specific process is as follows:

(1) TC selects a large prime p of 200bits and the parameters a, b in elliptic curve equation y^2=x^3+ax^2 + b,meet a,

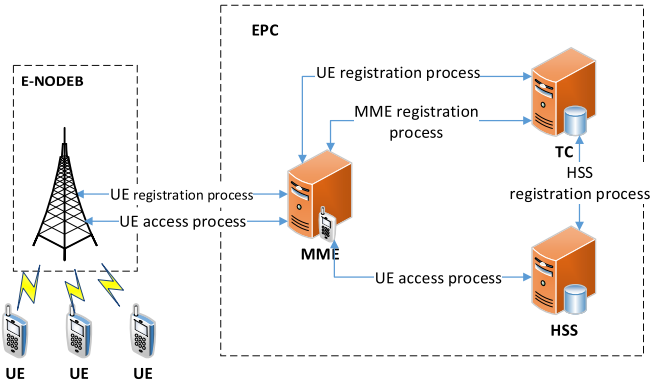

Fig. 2. The overall architecture of the ESC-AKA scheme.

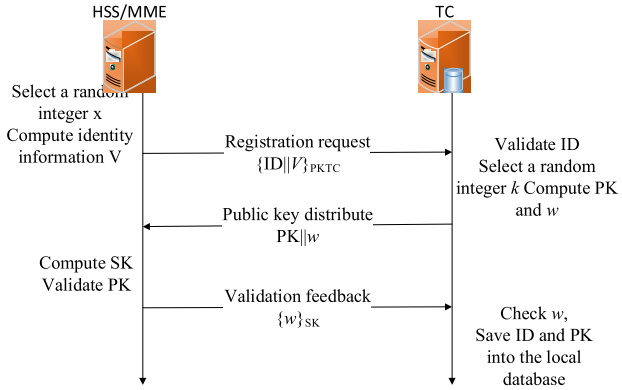

Fig. 3. The process of HSS/MME registration.

b less than p-1,and $4a^3 + 27b^2 \neq 0$, and determines the elliptic curve group $\mathrm{E_P}(a, b)$.

(2) Select a generator $M(x_\mathrm{M}, y_\mathrm{M})$ from the group $\mathrm{E_P}(a, b)$ as the base point of the elliptic curve. The order of M is n, x_M, y_M is required less than p-1, and n is a prime number.

(3) Select a random integer $\mathrm{SK_{TC}}$ as the private key of TC. $\mathrm{SK_{TC}}$ is a prime number between 1 to p-1. Calculate $\mathrm{PK_{TC}} = \mathrm{SK_{TC}}M$ as public key of TC.

(4) Select one-way hash function *SH*().

TC publics p, a, b, M, n, *SH*(), $\mathrm{PK_{TC}}$ and keeps $\mathrm{SK_{TC}}$ as secret.

D. HSS/MME Registration Process

The process of HSS/MME registration is shown in Fig. 3. The specific process:

(1) HSS/MME selects a random integer x, calculates identity information $V = SH(ID, x) \bullet M$, and then HSS/MME connects ID to V and uses TC's public key to encrypt it to form a registration request message $\{\mathrm{ID}||V\}_{\mathrm{PKTC}}$. The message is sent to TC then.

(2) TC receives and decrypts $\{\{\mathrm{ID}||V\}_{\mathrm{PKTC}}$, tests ID rationality and selects a random number k to calculates PK= $k \bullet M$+V as the public key of HSS/MME. It calculates w=(ID •k-X(PK) • $\mathrm{SK_{TC}}$) (mod n) as the proof of the public key, where X(PK) represents the abscissa of the PK. Then it sends the message PK$||w$ to HSS/MME.

(3) After the HSS/MME receives the message PK$||w$, the private key SK is first calculated by SK= (w+ *SH*(ID, x) • ID)(mod n). Then it checks whether SK •M=ID•PK-X(PK)•$\mathrm{PK_{TC}}$. After the test, HSS/MME sends message $\{w\}_{\mathrm{SK}}$to TC.

(4) After the TC receives the message $\{w\}_{\mathrm{SK}}$, the TC uses PK to decrypt the message and checks whether the w is equal to that w it sends. If they are equal, the ID of HSS/MME and its corresponding PK are added to the database.

The proof process for validating equations:

$$\begin{aligned}\mathrm{SK} \bullet M &= (w + SH(\mathrm{ID}, x) \bullet \mathrm{ID})(\mathrm{mod}n) \bullet M \\ &= \mathrm{ID} \bullet k \bullet M\text{-}X(\mathrm{PK}) \bullet \mathrm{SK_{TC}} \bullet M + SH(\mathrm{ID}, x) \\ &\quad \bullet \mathrm{ID} \bullet \mathrm{M} \\ &= \mathrm{ID} \bullet (k \bullet M + SH(\mathrm{ID}, x) \bullet M)\text{-}X(\mathrm{PK}) \bullet \mathrm{PK_{TC}} \\ &= \mathrm{ID} \bullet \mathrm{PK}\text{-}X(\mathrm{PK}) \bullet \mathrm{PK_{TC}}\end{aligned}$$

The MME/HSS public key distribution process is completed through the above registration protocol. In this process, only the user knows its private key, and TC only knows the user's ID and identity information V. Because of the intractability of the hash function and the elliptic curve discrete logarithm problem, TC cannot compute the random number x according to ID and V. So, the private key of the user can not be known by TC. In the registration process, the public key PK and the proof w are generated by TC using its private key $\mathrm{SK_{TC}}$ signature, that is to say TC is involved in the public key generation, so the MME/HSS validation of the public key can be passed. This eliminates the risk that an attacker forges the public key.

E. UE Registration Process

The process of UE registration is shown in Fig. 4. The specific process:

(1) UE selects a random integer x_{UE} and calculates $V_{\mathrm{UE}} = SH(IMSI, x) \bullet M$. Then it connects IMSI to $V_{\mathrm{UE}}$and uses TC's public key to encrypt it. UE connects the ciphertext to the target HSS's serial number $\mathrm{ID_{HSS}}$ and the target TC's serial number $\mathrm{ID_{TC}}$ to form a message $\{\mathrm{IMSI}||V_{\mathrm{UE}}\}_{\mathrm{PKTC}}||\mathrm{ID_{HSS}}||\mathrm{ID_{TC}}$ and sends it to MME to request a registration.

(2) After MME receives the registration request information from UE, MME connects SNID to its public key and uses TC's public key encryption, then connects received UE's identity information and UE's $\mathrm{ID_{HSS}}$to form a message $\{\mathrm{SNID}||\mathrm{PK_{MME}}\}_{\mathrm{PKTC}}||\{\mathrm{IMSI}||V_{\mathrm{UE}}\}_{\mathrm{PKTC}}||\mathrm{ID_{HSS}}$, sends to TC according to $\mathrm{ID_{TC}}$ received from UE, initiates public key request.

(3) TC decrypts $\{\mathrm{SNID}||\mathrm{PK_{MME}}\}_{\mathrm{PKTC}}$ and $\{\mathrm{IMSI}||V_{\mathrm{UE}}\}_{\mathrm{PKTC}}$ and validates SNID,IMSI and $\mathrm{ID_{HSS}}$ after it receives the UE's registration request

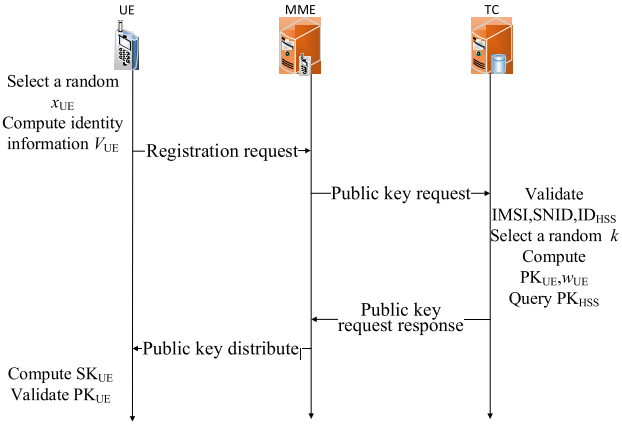

Fig. 4. The process of UE registration.

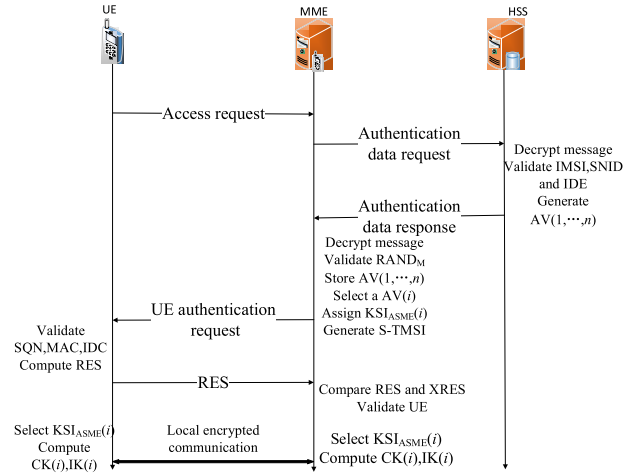

Fig. 5. The process of UE access.

information from MME. If the validation succeeds, TC selects a random number k, calculates $\mathrm{PK_{UE}} = k \bullet M + V_{\mathrm{UE}}$ as the public key of UE and calculates w_{UE}=(IMSI $\bullet k$-$X(\mathrm{PK_{UE}}) \bullet \mathrm{SK_{TC}})(\mathrm{mod}\ n)$ as the proof of the public key. Then it queries $\mathrm{PK_{HSS}}$ in the local database according to $\mathrm{ID_{HSS}}$. Then it connects generated $\mathrm{PK_{UE}}$, w_{UE} to SNID,IMSI and $\mathrm{ID_{HSS}}$ and uses MME's public key to encrypt it to form a message $\{\mathrm{PK_{UE}}||w_{\mathrm{UE}}||\mathrm{IMSI}||\mathrm{SNID}||\mathrm{PK_{HSS}}\}_{\mathrm{PKMME}}$. The message is sent to MME.

(4) MME receives the response message from TC, decrypts it and sends the message $\mathrm{PK_{UE}}||w_{\mathrm{UE}}||\mathrm{PK_{HSS}}$ to UE.

(5) After UE receives the message $\mathrm{PK_{UE}}||w_{\mathrm{UE}}$, the private key $\mathrm{SK_{UE}}$ is first calculated by $\mathrm{SK_{UE}} = (w_{\mathrm{UE}} + SH(\mathrm{IMSI}, x_{\mathrm{UE}}) \bullet \mathrm{IMSI})(\mathrm{mod}\ n)$ and UE checks whether there is $\mathrm{SK_{UE}} \bullet M = \mathrm{IMSI} \bullet \mathrm{PK_{UE}}$-$X(\mathrm{PK_{UE}}) \bullet \mathrm{PK_{TC}}$. If the check passes, the $\mathrm{PK_{UE}}$ is saved as its own public key, and $\mathrm{PK_{HSS}}$ is saved as the public key of HSS. Then the access process is started.

F. UE Access Process

The process of UE access is shown in Fig. 5.

The specific process:

(1) UE selects a random number $\mathrm{R_U}$ and uses the $f2$ function and shared key K to calculates identity certificate (IDC), IDC= $f2_K(\mathrm{R_U})$, and then it connects IMSI, $\mathrm{R_U}$, IDC to its public key $\mathrm{PK_{UE}}$and uses $\mathrm{PK_{HSS}}$to encrypt it. Then it connects the ciphertext to the public key $\mathrm{PK_{HSS}}$ and the serial number $\mathrm{ID_{HSS}}$ of the target HSS to form the message $\{\mathrm{IMSI}||\mathrm{R_U}||\mathrm{IDC}||\mathrm{PK_{UE}}\}_{\mathrm{PKHSS}}||\mathrm{ID_{HSS}}||\mathrm{PK_{HSS}}$. UE sends it to MME and requests access.

(2) After MME receives the UE's access request information, it selects a random number $\mathrm{RAND_M}$ and connects SNID, $\mathrm{PK_{MME}}$ to $\mathrm{RAND_M}$. Then MME uses $\mathrm{PK_{HSS}}$ which is included in the request information to encrypt $\{\mathrm{SNID}||\mathrm{PK_{MME}}||\mathrm{RAND_M}\}$, and then it connects the ciphertext to the identity information received from UE to form a message $\{\mathrm{SNID}||\mathrm{PK_{MME}}||\mathrm{RAND_M}\}_{\mathrm{PKHSS}}||\{\mathrm{IMSI}||\mathrm{R_U}||\mathrm{IDC}||\mathrm{PK_{UE}}\}_{\mathrm{PKHSS}}$. The message is sent to HSS according to $\mathrm{ID_{HSS}}$ provided by UE. The authentication data request is initiated.

(3) HSS decrypts $\{\mathrm{SNID}||\mathrm{PK_{MME}}||\mathrm{RAND_M}\}_{\mathrm{PKHSS}}$ and $\{\mathrm{IMSI}||\mathrm{R_U}||\mathrm{IDC}||\mathrm{PK_{UE}}\}_{\mathrm{PKHSS}}$ after it receives the authentication data request information from MME. It validates the SNID and IMSI and queries the database to find the shared key K through IMSI. Then it validates the IDC through the $f2$ function. After the validation, the authentication vector group AV (1,…,n) is generated. HSS connects IMSI, $\mathrm{PK_{UE}}$, $\mathrm{RAND_M}$, IDC to AV and uses $\mathrm{PK_{MME}}$to encrypt it to form message $\{\mathrm{AV}(1,\ldots,n)||\mathrm{IMSI}||\mathrm{PK_{UE}}||\mathrm{RAND_M}||\mathrm{IDC}\}_{\mathrm{PKMME}}$. HSS sends the message to MME to make the authenticate data response. The algorithm used to generate parameter of AV is consistent with the original EPS-AKA algorithm.

(4) MME decrypts the response message when it receives the message from HSS. It stores AV (1,…,n), chooses a AV(i) from AV, extracts RAND(i), AUTN(i), KASME(i) from AV(i) and assigns a key identifier $\mathrm{KSI_{ASME}}(i)$ to KASME(i). After that MME uses IMSI to generate temporary identity S-IMSI for UE, uses $\mathrm{PK_{UE}}$to form a message $\{\mathrm{RAND}(i)||\mathrm{AUTN}(i)||\mathrm{S\text{-}TMSI}||\mathrm{KSI_{ASME}}(i)||\mathrm{IDC}\}_{\mathrm{PKUE}}$ and sends it to UE. The user authentication request is initiated.

(5) After the authentication request is successfully received by UE, UE decrypts the message, extracts AUTN(i) and calculates XMAC(i). Then UE verifies that whether XMAC(i) and MAC(i) are equal, and also verifies that the received sequence number SQN(i) is within the valid range. If both are validated, K and RAND (i) are used to compute RES(i) through the $f2$ algorithm, and $\mathrm{K_{ASME}}(i)$ is computed. Then, the RES(i) is sent to the MME as an authentication response.

(6) MME receives the authentication response RES(i), and compares it with the XRES (i) in the received authentication vector AV (i). If they are consistent, the authentication is passed.

(7) After the mutual authentication, MME and UE use KASME (i) as the basic key. UE uses K and RAND(i) to calculate CK and IK by $f3$ and $f4$ algorithm. MME

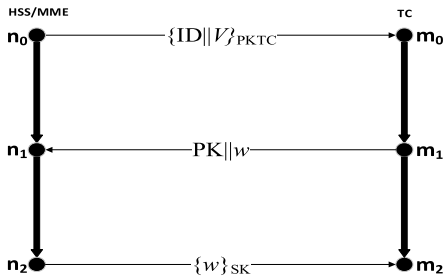

Fig. 6. HSS/MME registration protocol strand space model.

extracts CK and IK from AV(i) as the encryption key and integrity protection key, and then the secure communications are followed. The certification process is finished.

G. Error Message Handling

When UE calculates the authentication vectors and finds the authentication fails(e.g. SQN-Fail or MAC-Fail), UE generates error messages according to the original EPS-AKA protocol and then uses the public key of MME to encrypt the error message {Err-Message} to form a message {Err-Message}PK_{MME}. UE sends {Err-Message}PK_{MME} to the MME.

V. Formal Analysis of ESC-AKA Schemes

A. Formal Analysis Method of Secure Protocol in Strand Space Model

In 1998, Fabrega *et al.* established the strand space [42], [43] model, which opened up a new field for the formal analysis of security protocols. The strand space model combined with theorem proving and protocol trace, and it is efficient, rigorous and intuitive. It not only can prove the confidentiality and integrity of a protocol or an authentication scheme [44], but also can discover the inherent defect of a security protocol by constructing attack string, so it has gradually become the key direction of the security protocol formal research.

Authentication test method [45] is an analysis method to an authentication protocol based on strand space theory. This method can judge whether the security protocol can achieve the security target of identity authentication by constructing the authentication test component, establishing the authentication target of the security protocol and applying the authentication test rules. This paper will validate the protocol process based on this method.

B. HSS/MME Registration Protocol Formal Analysis

HSS/MME registration implements two-way authentications between HSS/MME and TC. The strand space model diagram and the proof process are as follows. Fig. 6 is a strand space model diagram of the HSS/MME registration protocol.

Subject set T,HSS/MME,TC∈T,Strand space($\sum$, *tr*)

The attack string is P, C is a bundle of $\sum$, $s_{\text{HSS/MME}}$, s_{TC}, $P \in C$, $\text{s}_{\text{HSS/MME}}$ ∈[ID, V, PK, w] with C-height 2

tr($\text{s}_{\text{HSS/MME}}$) =< +{ID||$V$}$_{\text{PKTC}}$,-(PK||$w$), +{$w$}$_{\text{SK}}$ >

s_{TC} ∈[ID, V, PK, w],

且 C-height=2

tr(s_{TC}) =<-{ID||V}$_{\text{PKTC}}$, +(PK||w),-{w}$_{\text{SK}}$ >

Attack string $p \in P$,K_{P} is the key set attackers master.

Basic assumptions:

(1) PK_{TC} and SK_{TC} are public key and private key of TC, and PK_{TC} is generated and published by TC. PK and SK are public key and private key of HSS/MME with PK, $\text{PK}_{\text{TC}} \in \text{K}_P$, SK, $\text{SK}_{\text{TC}} \notin \text{K}_P$.

(2) V is uniquely generated by HSS/MME. PK is generated by TC through V, and w is uniquely generated by TC through ID and SK_{TC}, and the attacker cannot launch SK or SK_{TC} from V, PK, w.

Proof: HSS/MME authentication process for TC:

(1) Construct test component. Since V is the only one generated by HSS/MME, the message {ID||V}$_{\text{PKTC}}$ containing V is not a proper sub-term for any node. So, the message can serve as a test component for V, and the edge $n_0 \Rightarrow^+ n_1$ constitutes an outgoing test for V in the {ID||V}$_{\text{PKTC}}$.

(2) By authentication test 1: there exist regular nodes m, $m' \in C$, such that {ID||V}$_{\text{PKTC}}$is a component of m and $m \Rightarrow^+ m'$ is a transforming edge for V.

(3) Define node m. The result from the second step is that m is the negative node. Suppose m is a node in a TC string s', s'∈[ID', V', PK', w'], $m =< s',1>$, and *term*($< s',1>$) ={ID'||V'}$_{\text{PKTC}'}$.

(4) The comparation between the components of *term*($< s',1>$) and TC string shows ID= ID',$V = V$'.

(5) Use the second part of the authentication test 1. Since PK is generated by V and the t'=(PK||w) is used as the test component of the V outgoing test, there exist a regular negative node m with *term*(m) = t'.

(6) Define node m''. If m'' is a node in a HSS/MME string s'', s'' ∈[ID", V", PK", w"], *m"*=<*s"*,2>,and *tr*<*s"*,2>=-(PK"||w'').

(7) Compare the components of string s and string *s"*. Since V is uniquely produced by HSS/MME and PK is generated by TC through V, and w is uniquely generated by TC through ID and SK_{TC}, it can be seen that PK=PK", $w = w''$. This proves that HSS/MME can complete the authentication of TC.

TC authentication process for HSS/MME:

(1) Construct the test component. Since w is only generated by node m_1 and the message {w}$_{\text{SK}}$ containing w is not a proper sub-term for any node, the message can serve as a test component for w. By assuming SK$\notin$ K_P, the edge $m_1 \Rightarrow^+ m_2$constitutes an incoming test for w in the {w}$_{\text{SK}}$.

(2) By authentication test 2: There exist regular nodes n, $n' \in C$, which makes {w}$_{\text{SK}}$ ∈*term*(n') and $n \Rightarrow^+ n'$ is a transforming edge for w.

(3) Define node n'. The result from the second step is that n is a positive node. Since w is only generated in node m_1, there must be a regular negative node n to receive the *w.* Suppose HSS/MME string $s' \in$[ID', V', PK', w'], and because of its structure, n is located in the node <s',2>. Because of <s',2>⇒$^+$<s',3>, n' is located in <s',3>.

(4) Compare the components of strings. Since w is only produced, by comparing the components in *term*(<s',2>)

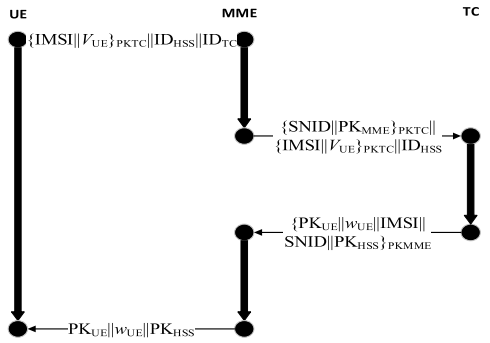

Fig. 7. UE registration protocol strand space model.

and HSS/MME string, what can obtain are $w = w'$, PK=PK', which proves that TC can complete the authentication to HSS/MME.

C. UE Registration Protocol Formal Analysis

The registration of the UE implements unidirectional authentication, which is the authentication of the UE to the MME and the authentication of the MME to the TC. It ensures that the UE can obtain a valid public key. The strand space model diagram and the proof process are as follows:

Fig. 7 is a strand space model diagram of the UE registration protocol.

Subject set T,UE,MME,TC∈T. Strand space($\sum$, *tr*)

The attack string is P, C is a bundle of $\sum$, s_{UE}, s_{MME}, s_{TC}, $P \in C$

UE String s_{UE} ∈[IMSI, V_{UE}, $\mathrm{ID_{HSS}}$, $\mathrm{ID_{TC}}$, $\mathrm{PK_{UE}}$, w_{UE}] with C-height=1

Trace $tr(s_{\mathrm{UE}})$ =< +{IMSI||V_{UE}}$_{\mathrm{PKTC}}$||$\mathrm{ID_{HSS}}$||$\mathrm{ID_{TC}}$,-$\mathrm{PK_{UE}}$||w_{UE}||$\mathrm{PK_{HSS}}$ >

MME string s_{MME} ∈[$\mathrm{ID_{HSS}}$, $\mathrm{ID_{TC}}$, SNID, $\mathrm{PK_{MME}}$, $\mathrm{PK_{UE}}$, w_{UE},$\mathrm{PK_{HSS}}$] with C-height =1

Trace $tr(s_{\mathrm{MME}})$ =<-{IMSI||V_{UE}}$_{\mathrm{PKTC}}$||$\mathrm{ID_{HSS}}$||$\mathrm{ID_{TC}}$, +{SNID||$\mathrm{PK_{MME}}$}$_{\mathrm{PKTC}}$||{IMSI||V_{UE}}$_{\mathrm{PKTC}}$||$\mathrm{ID_{HSS}}$, -{$\mathrm{PK_{UE}}$||w_{UE}||IMSI||SNID||$\mathrm{PK_{HSS}}$}$_{\mathrm{PKMME}}$, +$\mathrm{PK_{UE}}$||w_{UE}||$\mathrm{PK_{HSS}}$ >

s_{TC} ∈[SNID,$\mathrm{PK_{MME}}$,IMSI,V_{UE},$\mathrm{ID_{HSS}}$,$\mathrm{PK_{HSS}}$,$\mathrm{PK_{UE}}$, $\mathrm{PK_{MME}}$, w_{UE}] with C-height 1

$tr(s_{\mathrm{TC}})$ =<-{SNID||$\mathrm{PK_{MME}}$}$_{\mathrm{PKTC}}$||{IMSI||V_{UE}}$_{\mathrm{PKTC}}$|| $\mathrm{ID_{HSS}}$, +{$\mathrm{PK_{UE}}$||w_{UE}||IMSI||SNID||$\mathrm{PK_{HSS}}$}$_{\mathrm{PKMME}}$ >

Attack string $p \in P$,$\mathrm{K_P}$ is the key set that attackers master.

Basic assumptions:

(1) $\mathrm{PK_{TC}}$ and $\mathrm{SK_{TC}}$ are public key and private key of TC. $\mathrm{PK_{TC}}$ is generated and published by TC. $\mathrm{PK_{MME}}$ and $\mathrm{SK_{MME}}$ are public key and private key of MME. $\mathrm{PK_{UE}}$ and $\mathrm{SK_{UE}}$ are public key and private key of UE. $\mathrm{PK_{HSS}}$is the HSS public key applied by UE. And, $\mathrm{PK_{HSS}}$,$\mathrm{PK_{TC}}$,$\mathrm{PK_{UE}}$ ∈K_P, $\mathrm{PK_{MME}}$,$\mathrm{SK_{MME}}$,$\mathrm{SK_{TC}}$,$\mathrm{SK_{UE}}$ ∉ K_P.

(2) V_{UE} is generated by UE only. $\mathrm{PK_{UE}}$ is generated by TC through V_{UE}. w_{UE} is generated only by TC through IMSI and $\mathrm{SK_{TC}}$. The attacker cannot derive $\mathrm{SK_{UE}}$, $\mathrm{SK_{MME}}$ or $\mathrm{SK_{TC}}$ from $\mathrm{V_{UE}}$, $\mathrm{PK_{UE}}$, w_{UE}, $\mathrm{PK_{TC}}$.

Proof: UE authentication process for MME:

(1) Construct test component. Since V_{UE} is the only one generated by UE, the message {IMSI||V_{UE}}$_{\mathrm{PKTC}}$ containing V_{UE} is not a proper sub-term for any node. So, the message can serve as a test component for V_{UE}, and the edge <s_{UE},1>⇒$^+$<s_{UE},2> constitutes an outgoing test for V_{UE} in the {IMSI||V_{UE}}$_{\mathrm{PKTC}}$

(2) By authentication test 1: there exists regular nodes m, $m' \in C$, and this makes {IMSI||V_{UE}}$_{\mathrm{PKTC}}$be a component of m and $m \Rightarrow^+ m'$ is a transforming edge for V_{UE}.

(3) Define node m. The result from the second step is that m is the negative node. Suppose m to be a node in a MME string s', s'∈[$\mathrm{ID_{HSS}}$', $\mathrm{ID_{TC}}$', SNID', $\mathrm{PK_{MME}}$', $\mathrm{PK_{UE}}$', w_{UE}', $\mathrm{PK_{HSS}}$'], and *term*(< s',1>) ={IMSI'||V_{UE}'}$_{\mathrm{PKTC'}}$.

(4) The comparation between the components of *term*(< s',1>) and MME string shows IMSI= IMSI',$V_{\mathrm{UE}} = V_{\mathrm{UE}}$'.

(5) Use the second part of the authentication test 1, since $\mathrm{PK_{UE}}$ is generated by V_{UE}, the t'=($\mathrm{PK_{UE}}$||w_{UE}||$\mathrm{PK_{HSS}}$) is used as the test component of the V_{UE} outgoing test, and there exists a regular negative node u, and *term*(u) = t'.

(6) Define node u'. If u' is a node in a UE string s", u'=<s",2> and *tr*<s",2>=-($\mathrm{PK_{UE}}$"||w''_{UE}||$\mathrm{PK_{HSS}}$").

(7) Compare the components of string s and string s". V_{UE} is only produced by UE; $\mathrm{PK_{UE}}$ is generated by TC through V_{UE}, and w_{UE} is generated by TC through IMSI and $\mathrm{SK_{TC}}$ only. Thus it can be seen that $\mathrm{PK_{UE}}$ =$\mathrm{PK_{UE}}$", w_{UE}=w_{UE}", $\mathrm{PK_{HSS}}$ =$\mathrm{PK_{HSS}}$", which proves that UE can complete the authentication of MME.

MME authentication process for TC:

(1) Construct test component. Since SNID is only generated by node <s_{MME},2> and the message {$\mathrm{PK_{UE}}$||w_{UE}||IMSI||SNID||$\mathrm{PK_{HSS}}$}$_{\mathrm{PKMME}}$ containing SNID is not a proper sub-term for any node, the message can serve as a test component for SNID. Assuming $\mathrm{SK_{TC}} \notin \mathrm{K}_P$, the edge<$s_{\mathrm{MME}}$,2>⇒$^+$<$s_{\mathrm{MME}}$,3> constitutes an incoming test for SNID in the {$\mathrm{PK_{UE}}$||w_{UE}||IMSI||SNID||$\mathrm{PK_{HSS}}$}$_{\mathrm{PKMME}}$.

(2) By authentication test 2: there exist regular nodes n, $n' \in C$, which makes {$\mathrm{PK_{UE}}$||w_{UE}||IMSI||SNID||$\mathrm{PK_{HSS}}$}$_{\mathrm{PKMME}}$ ∈*term*(n') and $n \Rightarrow^+ n'$ is a transforming edge for SNID.

(3) Define node n'. It can be known that n is the positive node and {$\mathrm{PK_{UE}}$||w_{UE} ||IMSI||SNID||$\mathrm{PK_{HSS}}$}$_{\mathrm{PKMME}}$ ∈*term*(n') from the result of the second step. Since SNID is only generated in node <s_{MME},2>, there must be a regular negative node n to receive the SNID. Suppose TC string s'∈[SNID',$\mathrm{PK_{MME}}$',IMSI',V_{UE}', $\mathrm{ID_{HSS}}$',$\mathrm{PK_{HSS}}$',$\mathrm{PK_{UE}}$']. According to its structure, n is located in the node <s',1>.*B*ecause of the presence of <s',1>⇒$^+$<s',2>, n' is located in <s',2>.

(4) Compare the components of strings. Since SNID is only produced, the comparation between the components *term*(<s',2>) and TC string can get SNID= SNID′,$\mathrm{PK_{UE}}$ =$\mathrm{PK_{UE}}$', $w = w'$, IMSI=IMSI', $\mathrm{PK_{HSS}}$ = $\mathrm{PK_{HSS}}$', which proves that MME can complete the authentication of TC.

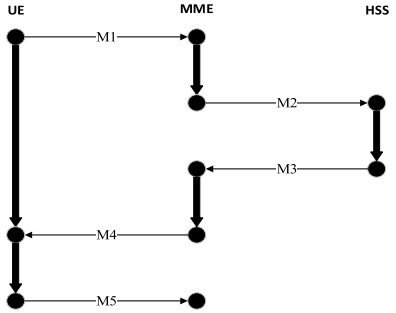

Fig. 8. UE access protocol strand space model.

D. UE Access Protocol Formal Analysis

UE access protocol achieves the mutual identification between HSS and MME, MME and UE, and HSS and UE. The strand space model diagram and the proof process are as follows:

Fig. 8 is a strand space model diagram of the UE access protocol.

M1=$\{\text{IMSI}||\text{R}_\text{U}||\text{IDC}||\text{PK}_\text{UE}\}_{\text{PKHSS}}||\text{ID}_\text{HSS}||\text{PK}_\text{HSS}$

M2=$\{\text{SNID}||\text{PK}_\text{MME}||\text{RAND}_\text{M}$ $\}_{\text{PKHSS}}||\{\text{IMSI}||\text{R}_\text{U}||\text{IDC}||\text{PK}_\text{UE}\}_{\text{PKHSS}}$

M3=$\{\text{AV}(1,\ldots,n)||\text{IMSI}||\text{PK}_\text{UE}||\text{RAND}_\text{M}||\text{IDC}\}_{\text{PKMME}}$

M4=$\{\text{RAND}(i)||\text{AUTN}(i)||\text{S-TMSI}||\text{KSI}_{\text{ASME}}(i)||$ IDC$\}$

M5=RES(i)

Subject set T, MME,UE,HSS$\in$T, strand space($\sum$, *tr*)

The attack string is P, C is a bundle of $\sum$, s_{UE}, s_{MME}, s_{HSS}, $P \in C$

s_{UE} $\in$[IMSI,R_U,IDC,ID_HSS,RAND(i),AUTN(i), S-TMSI, $\text{KSI}_{\text{ASME}}(i)$,$\text{PK}_\text{UE}$,$\text{PK}_\text{HSS}$] with C-height 2

$tr(s_{\text{UE}})$ =< +M1,-M4,+M5>

s_{MME} $\in$[PK_HSS, ID_HSS, SNID, RAND_M, PK_UE, IMSI, IDC, RAND, K_ASME, AMF, SQN$\oplus$AK, MAC, XRES, RES] with C-height 4

$tr(s_{\text{MME}})$ =<-M1,+M2,-M3,+M4,-M5>

s_{HSS} $\in$[IMSI,PK_UE,R_U,IDC,SNID,PK_MME,RAND_M, RAND,SQN$\oplus$AK, AMF,MAC,K_ASME] with C-height 1

$tr(s_{\text{HSS}})$ =<-M2,+M3>

Attack string $p \in P$, K_P is the key set that attackers master.

Subject set T,UE,MME,TC$\in$T,strand space($\sum$, *tr*).

The attack string is P, C is a bundle of $\sum$, s_{UE}, s_{MME}, s_{TC}, $P \in C$

s_{UE} $\in$[IMSI, V_{UE}, ID_HSS, ID_TC, PK_UE, w_{UE}] with C-height 1

$tr(s_{\text{UE}})$ =< +$\{\text{IMSI}||V_{\text{UE}}\}_{\text{PKTC}}||\text{ID}_\text{HSS}||\text{ID}_\text{TC}$,-$\text{PK}_\text{UE}||$ $w_{\text{UE}}||\text{PK}_\text{HSS}$ >

s_{MME} $\in$[ID_HSS, ID_TC, SNID, PK_MME, PK_UE, w_{UE}, PK_HSS] with C-height 1

$tr(s_{\text{MME}})$ =<-$\{\text{IMSI}||V_{\text{UE}}\}_{\text{PKTC}}||\text{ID}_\text{HSS}||\text{ID}_\text{TC}$, +$\{\text{SNID}||\text{PK}_\text{MME}\}_{\text{PKTC}}||\{\text{IMSI}||V_{\text{UE}}\}_{\text{PKTC}}||\text{ID}_\text{HSS}$, -$\{\text{PK}_\text{UE}||w_{\text{UE}}||\text{IMSI}||\text{SNID}||\text{PK}_\text{HSS}\}_{\text{PKMME}}$, +$\text{PK}_\text{UE}||w_{\text{UE}}||\text{PK}_\text{HSS}$ >

s_{TC} $\in$[SNID,PK_MME,IMSI,V_{UE},ID_HSS,PK_HSS,PK_UE, $\text{PK}_\text{MME}, w_{\text{UE}}$] with C-height 1

$tr(s_{\text{TC}})$ =<-$\{\text{SNID}||\text{PK}_\text{MME}\}_{\text{PKTC}}||\{\text{IMSI}||V_{\text{UE}}\}_{\text{PKTC}}||$ ID_HSS, +$\{\text{PK}_\text{UE}||w_{\text{UE}}||\text{IMSI}||\text{SNID}||\text{PK}_\text{HSS}\}_{\text{PKMME}}$ >

Attack string $p \in P$,K_P is the key set that attackers master.

Basic Assumptions:

(1) PK_HSS and SK_HSS are the public key and private key of HSS. PK_HSS is generated by TC and stored in the database during the HSS registration phase. UE gets the PK_HSS of the corresponding HSS from TC after the registration phase. PK_MME and SK_MME are the public key and private key of MME. PK_UE and SK_UE are the public key and private key of UE. The public keys are generated by TC during the registration phase, and the private keys are stored locally. PK_UE, PK_HSS $\in \text{K}_P$,SK_UE, SK_MME, SK_HSS, PK_MME $\notin \text{K}_P$.

(2) IMSI is a unique identifier for UE. R_U is generated by UE only, and IDC is calculated by UE through R_U; RAND_M is only generated by MME and S-TMSI is calculated by MME through IMSI. The authentication vector AV and each component of it is generated only by HSS.

Proof: **The UE authentication process for MME:**

(1) Construct test component. Since IMSI is a unique identifier for UE, the message r =$\{\text{IMSI}||\text{R}_\text{U}||\text{IDC}||\text{PK}_\text{UE}\}_{\text{PKHSS}}$ containing IMSI is not a proper sub-term for any node. S-TMSI is generated by IMSI and SK_MME, SK_HSS, $\notin \text{K}_P$, so the message r can serve as a test component for IMSI, and the edge <s_{UE},1>$\Rightarrow^+$<s_{UE},2> constitutes an outgoing test for V_{UE} in the r.

(2) By authentication test 1: there exist regular nodes m, $m' \in C$, which make $r$be a component of m and $m \Rightarrow^+ m'$ be a transforming edge for IMSI.

(3) Define node m. From the result of the second step, m is the negative node. Suppose m to be a node in a MME string s', m =< s',1> and *term*(< s',1>) =$\{\text{IMSI'}||\text{R}_\text{U}\text{'}||\text{IDC'}||\text{PK}_\text{UE}\text{'}\}_{\text{PKHSS'}}$.

(4) The comparation between the components of *term*(< s',1>) and MME string shows that IMSI=IMSI', R_U = R_U',IDC= IDC', PK_UE = PK_UE'.

(5) Use the second part of the authentication test 1. Since S-TMSI is generated by IMSI, and the t' =$\{\text{RAND}(i)||\text{AUTN}(i)||\text{S-TMSI}$ $||\text{KSI}_{\text{ASME}}(i)||\text{IDC}\}_{\text{PKUE}}$ is used as the test component of the IMSI outgoing test, there exists a regular negative node u, and *term*(u) = t'.

(6) Define node u'. If u' is a node in a UE string s'', *u'*=<*s"*,2>,and *tr*(<s'',2>)=-$\{\text{RAND}(i)\text{"}||\text{AUTN}(i)\text{"}$ $||\text{S-TMSI"}||\text{KSI}_{\text{ASME}}(i)\text{"}||\text{IDC"}\}_{\text{PKUE"}}$.

(7) Compare the components of string s and string *s"*. Since IMSI is only produced by UE, S-TMSI is generated by MME through IMSI. Thus, it can be known that S-TMSI=S-TMSI", RAND(i) =RAND(i)", AUTN(i) =AUTN(i)", $\text{KSI}_{\text{ASME}}(i)$ =$\text{KSI}_{\text{ASME}}(i)$" and IDC=IDC". This proves that UE can complete the authentication of MME.

MME authentication process for UE:

(1) Construct test component. Since RAND(i) is only generated in the node <s_{MME},3>, the message $\{\text{RAND}(i)||\text{AUTN}(i)||\text{S-TMSI}$ $||\text{KSI}_{\text{ASME}}(i)||\text{IDC}\}_{\text{PKUE}}$ containing RAND(i) is not a proper sub-term for any node and SK_MME, SK_UE $\notin$

K_P, so the message can serve as a test component for RAND(i), the edge <s_{MME},3>⇒$^+$<s_{MME},4> constitutes an outgoing test for RAND(i) in this message.

(2) By authentication test 1: there exist regular node *n*, $n' \in C$, such that {RAND(i)||AUTN(i)||S-TMSI(i)||$KSI_{ASME}(i)$||IDC}$_{PKUE}$ ∈*term*(n) and $n \Rightarrow^+ n'$ is a transforming edge for RAND(i).

(3) Define node n. The result from the second step shows that n is the negative node. Suppose n is a node in a UE string s'_{UE}, and s'_{UE} ∈[IMSI',R_U',IDC',ID_{HSS}',RAND(i)',AUTN(i)',S-TMSI',$KSI_{ASME}(i)$',PK_{UE}',PK_{HSS}'], by the structure of UE string, n is located in the node <s'_{UE},2>, and {RAND(i)'||AUTN(i)'||S-TMSI'||$KSI_{ASME}(i)$'||IDC'}$_{PKUE}$ ∈*term*(n).

(4) Because RAND(i) generated only by MME, < s_{UE} ',2>⇒ + <s_{UE} ',3> and {RAND(i)'||AUTN(i)'||S-TMSI'||$KSI_{ASME}(i)$'||IDC'}$_{PKUE}$ ∈*term*(< s_{UE}',2>), it can be known that RAND(i) = RAND(i)', S-TMSI =S-TMSI', AUTN(i) = AUTN(i)', $KSI_{ASME}(i)$ = $KSI_{ASME}(i)$ ', IDC= IDC', which proves that MME can complete the authentication for UE.

HSS authentication process for MME:

(1) Constructing test component. SNID is generated only in the node < s_{MME},2>, and the message {SNID||PK_{MME}||$RAND_M$}$_{PKHSS}$ is a component of SNID at <s_{HSS},1> and SNID does not appear in any component form except in {SNID||PK_{MME}||$RAND_M$}$_{PKHSS}$at < s_{MME},2>, so the message can serve as a test component for SNID. By the assumption $SK_{MME} \notin K_P$, the negative node < s_{HSS},1> constitutes an unsolicited test for SNID in this message.

(2) By the unsolicited test: there exist positive regular node *n*, $n' \in C$, such that {SNID||PK_{MME}||$RAND_M$ }$_{PKHSS}$ ∈ *term*(n).

(3) Define node n. From the results of the second step, n is a positive node. Suppose n is a node in a MME string s'_{MME}, and s'_{MME} ∈[PK_{HSS}',ID_{HSS}', SNID', $RAND_M$', PK_{UE}', IMSI', IDC', RAND', K_{ASME}', AMF',SQN⊕AK',MAC',XRES',RES'], n is located in the node < s'_{MME},2> and {SNID'||PK_{MME}'||$RAND_M$'}$_{PKHSS}$ ∈*term*(n) by the structure of MME string.

(4) Compare the contents of strings. Because SNID generated only by MME and {SNID'||PK_{MME}' ||$RAND_M$'}$_{PKHSS}$ ∈*term*(s'_{MME},2), SNID=SNID', PK_{MME} =PK_{MME}', $RAND_M$ =$RAND_M$', which proves that HSS can complete the authentication of MME.

MME authentication process for HSS:

(1) Constructing test component. $RAND_M$is generated only in node <s_{MME},2>, and the message {SNID||PK_{MME}||$RAND_M$}$_{PKHSS}$ containing SNID is not a proper sub-term for any node, so {SNID||PK_{MME}||$RAND_M$}$_{PKHSS}$ can serve as a test component for $RAND_M$ by the assumption $SK_{HSS} \notin K_P$. The edge <s_{MME},2>⇒$^+$<s_{MME},3> constitutes an outgoing test for $RAND_M$ in this message.

(2) The results by authentication test 1: there exists regular node *n*, $n' \in C$, such that {SNID||PK_{MME}||$RAND_M$ }$_{PKHSS}$ ∈ *term*(n) and $n \Rightarrow^+ n'$ is a transforming edge for $RAND_M$.

(3) Define node n. The result from the second step is that n is a negative node. Suppose n to be a node in a HHS string s'_{HSS} and s'_{HSS} ∈[IMSI', PK_{UE}',R_U',IDC',SNID',PK_{MME}',$RAND_M$', RAND',SQN'⊕AK',AMF',MAC',K_{ASME}'], by the structure of HHS string, n is located in the node <s'_{HSS}, 1> and {SNID'||PK_{MME}'||$RAND_M$'}$_{PKHSS'}$ ∈*term*(n).

(4) Because $RAND_M$generated only by MME, < $s\_'_{HSS}$,1>⇒$^+$< s'_{HSS},2> and {SNID'||PK_{MME}'||$RAND_M$'}$_{PKHSS'}$ ∈*term*(s'_{HSS},1), it can be known that $RAND_M$ =$RAND_M$', SNID=SNID', PK_{MME} =PK_{MME}', which proves that MME can complete the authentication of HSS.

The UE authentication for the HSS is done by checking the received MAC and SQN. Since K is the shared key between the UE and the HSS, a valid MAC can be generated from the RAND only when the key K is owned.

The HSS authentication for the UE is achieved by the validation of the IMSI and the IDC generated by R_U and the shared key K.

Therefore, this scheme achieves the mutual identifications between HSS and MME, MME and UE, and HSS and UE, which greatly improves the security of the communications.

VI. Security Analysis of the ESC-AKA Scheme

A. Security Attributes for the ESC-AKA Scheme

1) Confidentiality of Data Transmission: In the processes of registration and access, data clearly transmitted are non-critical data, such as PK_{UE} and ID_{HSS}. Even if the data are stolen, it will not affect the communication process. The fields of privacy, including the IMSI, SNID, and AV, are encrypted by public key encryption, so it is very difficult for attackers to obtain the key identity information from the messages without the private key.

Because the self-certified public key system is adopted, the private key of the network element is only known by itself, and the TC only participates in the generation of the public key. This avoids the risk of the TC mastering the private key, thereby improving the security of the communication.

2) Freshness of Session Keys: The UE generates the private key and gets its own public key when it is registered or accesses the network for the first time. The MME and HSS can also periodically change the public and private keys via the registration process to ensure the freshness of the public and private keys.

In each authentication, the HSS selects a new random number (RAND) to generate a cipher key (CK) and integrity key (IK), so the CK and IK can maintain their freshness after each certification. Additionally, the introduction of an independent sequence number (SQN) mechanism allows the system to resist replay attacks.

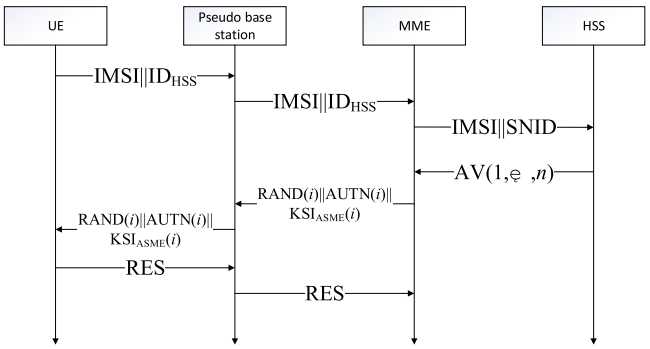

Fig. 9. The process of a pseudo base station attack.

B. Anti-Attack Ability of the ESC-AKA Scheme

1) Resisting Internal and External Eavesdropping, Counterfeiting, and Tampering Attacks: For attacks from within the system, such as eavesdropping or tampering with malicious insiders, because of the public key encryption system, valuable private information cannot be obtained without the private key, even if the eavesdropper has obtained the information on the channel. Furthermore, the risk of information being tampered with is avoided. For attacks from outside of the system, such as pseudo base station attacks, the achievement of mutual identification between the HSS and MME, the MME and UE, and the HSS and UE makes it difficult for the pseudo base station to hijack a UE that has been authenticated.

Regarding the vulnerability of the original EPS-AKA, namely that it is easily exploited by the pseudo base station, the scheme is also well compensated. For example, Fig. 9 presents the flow diagram of a pseudo base station attack on the original EPS-AKA.

(1) The pseudo base station selects a UE within the range of action and sets an abnormal TAC (tracking area code), which will cause the UE in the RRC idle state to be reallocated to the pseudo base station and to initiate TAU (Tracking, Area, Update) requests to the pseudo base station. After the pseudo base station rejects the request, the access request by UE will be initiated to the pseudo base station according to the protocol.
(2) After receiving the access request, the pseudo base station pretends that the user initiates an access request to the MME.
(3) After a series of communications between MME and HSS, the authentication vector is transmitted to the pseudo base station.
(4) The pseudo base station transmits the authentication vector to the blocked UE.
(5) UE sends the response information RES to pseudo base station.
(6) pseudo base station sends the response information RES to MME.

Through the above process, the attacker can masquerade as a legitimate UE, establish communication with MME, or pretend to be MME to communicate with the UE. Since AUTN(i) uses plaintext transmission in the user authentication request sent by MME to UE, the attacker can easily obtain the based key $\mathrm{K_{ASME}}(i)$. When the attacker masters the shared key K, it can calculate the encryption key CK and integrity protection key IK according to the algorithm, so as to it completely pretends to be the user or MME to establish communication, which has serious consequences.

In this scheme, the transmitted information on the channel is encrypted using the public key, and the attacker cannot get the correct key in the absence of the private key, even via eavesdropping. The self-certified public key system greatly improves the security of the communication.

2) Resisting AK Attacks Caused by SQN Guessing: If AUTN plaintext is transmitted, the value of the anonymity key (AK) can be obtained by the attacker by eavesdropping on the AUTN transmitted in the plaintext. However, because the AUTN transmitted in this scheme is encrypted by the public key, it can effectively resist the AK leak attack caused by SQN guessing.

3) Resisting Attacks Caused by Shared Key K Leaks: The shared key K value between the UE and the HSS is a key parameter of the original EPS-AKA protocol, and the CK, IK, and AK are generated by the K value and RAND via the $f3$, $f4$, and $f5$ functions. Because RAND is plaintext during the transmission, it can easily be intercepted by an attacker. Therefore, once the K value is leaked, the attacker can either pretend to be a user to initiate a request to the HSS, or can pretend to be the HSS to establish a connection with the user to further obtain the user's information or launch an attack. Moreover, because the frequent occurrence of K in the key agreement process will increase the probability of being attacked by the attacker, the K value exists as a shared key between the UE and the HSS for a long amount of time, and will not be updated.

In this scheme, due to the use of a public key system, even if the shared key K is exposed, the attacker cannot calculate the correct IDC and verify it through the UE if the attacker does not have the HSS private key to obtain the random number R_U. To some extent, this solves the security problem caused by the leakage of the shared key K value.

4) Resisting DoS Attacks: The server verifies the random value R_U and the IDC generated by the UE, which reduces the risk of DoS attacks to some extent.

C. Resisting Attacks of Forged Authentication Requests

Arapinis *et al.* analyzed the 3G-AKA protocol and found that there was a linkable attack in 3G networks [30], i.e., the attacker could judge whether the user is in a specific cell according to the message returned by the target UE's authentication response. The EPS-AKA protocol was analyzed, and it was found that the authentication response failure message of the EPS-AKA protocol is transmitted in plaintext. Therefore, an attacker can intercept the UE's return error message by eavesdropping and other attack methods, and can then judge whether the target UE is in a specific cell according to the error message. The specific steps of attack are as Fig. 10 shows.

In a 4G network, an attacker can set a malicious E-NodeB in the areas where the target UE may appear, and judges whether the target UE is in the certain area according to the attack method. The attack process will not be detected by the target UE.

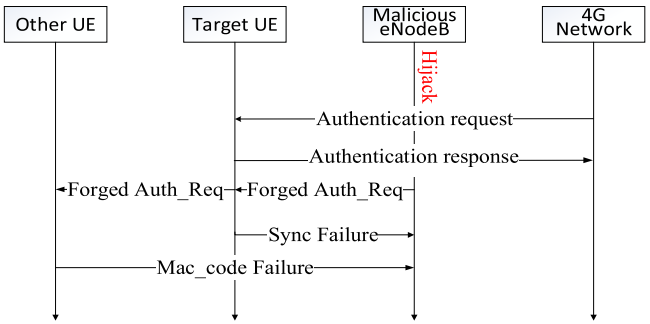

Fig. 10. The process of a forged authentication request attack.

TABLE III
SECURITY COMPARISON OF ESC-AKA PROTOCOL

	EPS-AKA	LTE-ID-AKA	5G-AKA	ESC-AKA
User identity confidentiality	No	Yes	Yes	Yes
Authentication vector privacy	No	Yes	Yes	Yes
Three way authentication	Incomplete	Incomplete	Yes	Yes
Shared key K leak risk	Yes	No	No	No
DDOS attack protection	No	No	Yes	Yes
User location confidentiality	No	No	No	Yes
Security level of public key system	Non public key	Level 1	Level 3	Level 3

The 5G-AKA protocol uses plaintext transmission to return error messages, which may leak the UE's location information and cause serious consequences [13], [14]. In the ESC-AKA scheme, error message handling is added. The UE returns the error message in the form of ciphertext. The attacker cannot get any useful information from the error message, so it cannot obtain the location of the target UE via this type of attack. Therefore, the ESC-AKA scheme is safe and effective in resisting this linkable attack.

D. Scheme Security Comparison

Table III compares the security performance of the original EPS-AKA, LTE-ID-AKA, and ESC-AKA.

It can be seen from Table III that the proposed scheme achieves the third level of public key system security due to the use of the self-certified public key encryption system. Moreover, the scheme realizes the encrypted transmission of the user identity and AV, and realizes mutual authentication among the UE, MME, and HSS. Due to the use of public key encryption, ESC-AKA avoids the security risk caused by the leakage of the shared key K, and has the ability to defend against DDoS attacks. Due to the handling of the returned error message, the confidentiality of the user's location is protected. Compared with the EPS-AKA, LTE-ID-AKA, and 5G-AKA schemes, ESC-AKA has better security performance.

TABLE IV
THE ENTITY LENGTH OF HSS/MME REGISTER PROTOCOL

Entity name	Length/bits
ID	64
x, V, k	128
w, SK_{TC}, SK	200
PK_{TC}, PK	400

VII. PERFORMANCE ANALYSIS OF THE ESC-AKA SCHEME

A. Bits Traffic Analysis

1) Selection of ECC Parameters and Implementation Process of Encryption: The selection of large prime number p: By the ECC principle, the larger the p, the safer the algorithm. However, considering the algorithm efficiency, the prime number of p is 200 bits.

The implementation of the encryption: because p is 200bits, the length of the plaintext should be less than 199bits, in addition, an identifier byte char (255) is required, the maximum text length of the plaintext at one time is 191bits. Therefore, consider to encrypt 20 bytes at a time. Because the cipher text of ECC is made up of dots, the plaintext can be divided into two segments. The first segment is used as the X coordinate, and the second segment is used as the Y coordinate. Since each coordinate can be encrypted with 20 bytes, 40 bytes of plaintext are encrypted at one time. When the remaining plaintext length is less than 20 bytes at the end of the file, the remaining plaintext is used as the X coordinate, and the Y coordinate is marked as 0, without identifying bytes. Thus, the program does not decrypt the contents of the Y without finding the identifying byte in the Y during the decrypting.

2) HSS/MME Registration Process Bits Traffic: Table IV shows the entities and their lengths involved in the HSS/MME registration process.

Parameter Description: ID: It is the unique identification information of each network element. To maintain its independence, it is the same as the identity information SNID in the original protocol, taking 64-bit integer.

x,k: They are random numbers selected by the protocol; they are the key elements for generating the private key. To ensure the complexity, x and k are 128-bit integer.

SK: It is the private key of each network element, which is a prime number between 1 and a 200-bit large prime number p-1. To ensure the complexity, SK is a 200-bit integer.

PK: It is the public key that is transferred in the form of coordinates. The X and Y coordinates are calculated according to the private key SK, which are both 200-bit integer. Therefore, the PK is a 400-bit integer.

Bit traffic for each message:

(1) Registration request $M_1 = \{\text{ID}||V\}_{\text{PKTC}}$, the length of the plaintext is 192bits and the length of the encrypted cipher text is 400bits.
(2) Public key and proof distribution $M_2 = \text{PK}||w$, there is no encryption process, the length is 600bits.
(3) Verify feedback $M_3 = \{w\}_{\text{SK}}$, the length of the plaintext is 200bits, and the length of the encrypted cipher text is 400bits.

TABLE V
THE ENTITY LENGTH OF UE REGISTER PROTOCOL

Entity name	Length/bits
SNID, ID_{HSS}, ID_{TC}	64
x_{UE}, V_{UE}, w_{UE}, IMSI, k	128
w_{UE}, SK_{TC}, SK_{UE}	200
PK_{TC}, PK_{MME}, PK_{UE}, PK_{HSS}	400

TABLE VI
THE ENTITY LENGTH OF UE ACCESS PROTOCOL

Entity name	Length/bits
S-TMSI	48
SNID, ID_{HSS}, RES, XRES	64
IMSI, R_U, IDC, ID_{HSS}, $RAND_M$, RAND, CK, IK, AUTN	128
w_{UE}, SK_{TC}, SK_{UE}	200
K_{ASME}	256
PK_{MME}, PK_{UE}, PK_{HSS}	400

According to the computation results, the total of bits transferred during the HSS/MME registration process is 1400bits.

3) UE Registration Process Bit Traffic: Table V shows the entities and their lengths involved in the UE registration process.

Bit traffic for each message:

(1) Registration request, M_1 ={IMSI||V_{UE}}$_{PKTC}$||ID_{HSS}||ID_{TC}, the length of IMSI||V_{UE} plaintext is 256bits, and the length of the encrypted cipher text is 400bits, the total message length is 400+64+64=528bits

(2) Public key request, M_2 ={SNID||PK_{MME}}$_{PKTC}$||{IMSI||-V_{UE}}$_{PKTC}$||ID_{HSS}, the total message length is 1264bits.

(3) Public key request response, M_3 ={PK_{UE}||w_{UE}||IMSI||SNI-D||PK_{HSS}}$_{PKMME}$, the total message length is 1600bits.

(4) Public key distribution M_4 =PK_{UE}||w_{UE}||PK_{HSS}, total message length is 1600bits.

According to the computation results, the total of bits transferred during the UE registration process is 4792bits.

4) UE Access Process Bit Traffic: Table VI shows the entities and their lengths involved in the UE access process [7].

Parameter description:

S-TMSI and K_{ASME} are the access layer entity that are existing in the original LTE access protocol, S-TMSI is 48-bit integer, K_{ASME} is 256-bit integer. The meaning and value of other parameters are the same as the original LTE access protocol.

Bit traffic for each message:

(1) Access request, M_1 ={IMSI||R_U||IDC||PK_{UE}}$_{PKHSS}$ || ID_{HSS}-||PK_{HSS}, the length of IMSI||R_U||IDC||PK_{UE} plaintext is 784 bits and the length of encrypted cipher text is 1200bits, the total message length is 1200+64+400=1664bits.

(2) Authentication data request M_2 ={SNID||PK_{MME}||$RAND_M$}$_{PKHSS}$|| {IMSI||R_U||IDC||PK_{UE}}$_{PKHSS}$, the total message length is 2000bits.

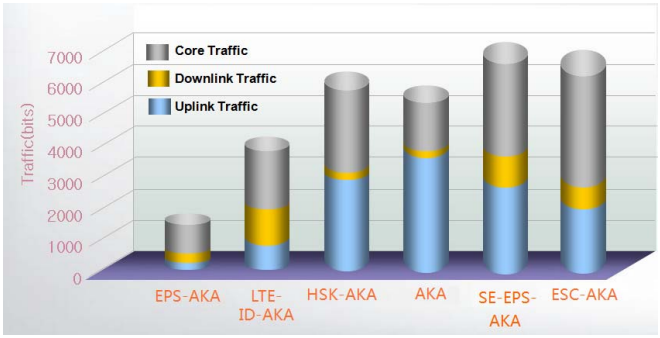

Fig. 11. The comparison of ESC-AKA bit traffic.

Authentication data response M3={AV(1,…,n)||IMSI||PKUE||RANDM||IDC} PKMME, suppose that the M group authentication vector AV is generated, the total message length is $400 \times \lceil \frac{608m+784}{320} \rceil$ bits.

(3) User authentication request M4={RAND(i)||AUTN(i)||STMSI||KSIASME(i)||IDC}$_{PKUE}$, the total message length is 800bits.

(4) user response RES, message length is 64bits.

(5) According to the computation results, the total bits transmission during the UE access is $4528 + 400 \times \lceil \frac{608m+784}{320} \rceil$ bits.

5) Bit Traffic Comparison: Fig. 11 presents the bit traffic comparison of the ESC-AKA and some other AKA protocols, including EPS-AKA, LTE-ID-AKA, HSK-AKA [20], AKA [24], and SE-EPS-AKA [31], when the AVs are the same.

As can be seen from the figure, the public key encryption system is adopted, and the proposed protocol consumes more bit traffic than the improved protocols that do not use public key cryptography, such as HSK-AKA and AKA. However, compared with the improved protocol using public key cryptography, such as SE-EPS-AKA, the uplink and downlink bit traffic used in the proposed protocol is reduced by 25%, and the total bit traffic is reduced by 2.7%. Moreover, the traffic of the proposed protocol is mostly core traffic. Compared with the uplink or downlink traffic, the information transmission in the core network is obviously more efficient and safer.

In practice, the re-access scenario is more frequent. Since this protocol can use the context information after the completion of the first access procedure when re-accessing, the traffic in the access process is greatly reduced. The traffic of the whole re-access process is only 918bits.

B. Efficiency Analysis

Due to the adoption of the public key cryptosystem and the limitation of the USIM computing power in the UE, ESC-AKA adds some encryption and decryption delays to the original EPS-AKA. Four metrics that represent the communication and computational overhead are used to measure the efficiency of authentication. Table VII exhibits the comparison between ESC-AKA and other protocols. The metric number of message transfers refers to the number of message transfer instances among the UE, MME, and HSS. The number of encryptions refers to the number of times the transmission data are encrypted with a public key at the UE, MME, and HSS. The

TABLE VII
EFFICIENCY COMPARISON OF ESC-AKA PROTOCOL

	Number of message transfers	Number of encryptions	Number of verifications	Number of Hash calculations
EPS-AKA	6	0	3	6
LTE-ID-AKA	6	5	3	6
SE-EPS-AKA	6	4	3	6
HSK-AKA	6	0	1	8
ESC-AKA (HSS/MME registration)	3	2	2	2
ESC-AKA (UE registration)	4	4	1	2
ESC-AKA (UE access)	5	4	4	7
ESC-AKA (HSS/MME and UE registration)	7	6	3	4
ESC-AKA (UE re-registration)	3	1	3	6

number of verifications refers to the number of times the UE, MME, and HSS verify the received data. Finally, the number of Hash calculations is the number of times the transmission data are computed with Hash functions.

Due to the use of the self-certified public key system, the HSS/MME registration and UE registration phases have been added. Consequently, the system efficiency of ESC-AKA is not as good as those of EPS-AKA and other solutions, but the security of the system is guaranteed and reaches the third level of the public key cryptosystem. None of other protocols reaches the third level of the public key cryptosystem. Moreover, the ECC algorithm used in this scheme has a faster encryption and decryption speed than the RSA and other algorithms, which reduces the impact on the system efficiency due to the increase of the numbers of encryption and decryption.

With the advancement of technology, the computing power of each network element is becoming increasingly more powerful. It is believed that the increase in the number of operations will not have a significant impact on the performance of the authentication process.

In addition, the UE can use the previous authentication information to conduct the authentication process when re-registering or re-calling authentication under the same MME. When the UE switches the MME for authentication, because there is a communication interface S10 between the previous MME and the new MME [7], the new MME can obtain the authentication information of the previous MME for UE identity authentication via the communication between the MMEs, without communicating with the HSS. It can also reduce the complicated work of AV generation and improves the system efficiency. These two scenarios are the most used scenarios in practical applications. Therefore, in general, ESC-AKA does not significantly affect the efficiency of the system.

C. Influence of Prime Number p

To study the influence of the prime number p on the ESC-AKA scheme, a program was developed to realize the registration of the ESC-AKA scheme on computer, including HSS/MME registration, UE registration, and UE access. For a small prime number p, 10 experiments were conducted, and the results were averaged over 10 times. It is not easy to find a 160-bit large prime number p, so one was found for experimentation. The computer used for the experiment was an Intel i5 CPU with 4G memory and a 1T hard disk.

TABLE VIII
TRANSMISSION DELAY AND PRIME p

Bit of prime p	Transmission delay of HSS/MME's registration (s)	Transmission delay of UE's registration(s)	Transmission delay of UE's access (s)	Total transmission delay (s)
8bits	0.048	0.057	0.012	0.117
16bits	0.054	0.059	0.016	0.129
64bits	0.071	0.064	0.032	0.167
160bits	0.083	0.073	0.065	0.221

Table VIII presents the influence of p on the transmission delay. The system transmission delay increased with the increase of the prime number p. The reason for this is that the increase of the number of bits of prime number p led to the increase of the complexity of encryption and decryption. However, the increase of the transmission delay was not large, and was within the acceptable range. Moreover, the transmission delay is believed to have been reduced if a better computer was used in the experiment.

The experimental results show that the increase of the number of bits of p did not significantly increase the transmission bytes and transmission delay in the system authentication process, but it could significantly improve the security of the system. However, too many bits of p will cause a greater time overhead in the elliptic curve generation and order calculation in the system initialization process. Therefore, it is recommended that the prime number p be between 64 and 200 bits. In the experiment, p was 64 bits.

VIII. DISCUSSION

Compared with the use of the EPS-AKA protocol in LTE networks, the TC in the ESC-AKA protocol is an additional node that brings an extra cost to the user authentication process, and the asymmetric encryption in the whole process of authentication introduces a large amount of computational overhead. However, these detractions are worthwhile. In terms of practical costs, compared with the original solution, the increased cost is mainly due to the increase of the TC. Because the TC is only responsible for the distribution of public and private keys during the registration process of each network element, it will not cause a lot of communication pressure. During actual deployment, only a small cost is required to deploy the TC server and the corresponding data link. The computational overhead of the encryption algorithm is also completely within the computing capability of the current equipment.

At present, the 5G network is gradually becoming commercial, and it can provide higher bandwidth and faster speeds. However, security issues remain, and some schemes have been

proposed as solutions [46]. The present research shows that although the 5G-AKA protocol can solve the problems of IMSI plaintext transmission, delegation authentication, etc., the use of plaintext transmission by 5G-AKA to return error messages may reveal the user's location information and bring serious consequences. ESC-AKA can avoid the reveal of the user's location privacy caused by the plaintext transmission of error messages.

Although the ESC-AKA protocol is characterized by increased computational overhead, it provides completely secure authentication. People pay more and more attention to network security and personal privacy [47], so improving LTE network security is completely meets people's requirements.

Compared with the high deployment cost of 5G network elements [15], the proposed scheme is based on the optimization of the LTE authentication protocol and has high practical operability. The ESC-AKA protocol could be used to improve the security of LTE networks before 5G is deployed in some countries and regions.

At present, the Internet of things (IoT) and edge computing are becoming increasingly more popular. IoT can be used for field data acquisition and remote equipment operation. However, it remains a problem for massive IoT devices to get access to networks with high security and low delay [49]. Some IoT applications require high security and can tolerate data transmission delays, such as the remote operation of oil pipeline valves. For this kind of high-security IoT application, it is necessary to use secure and reliable protocols to provide mutual authentication and effectively resist attacks. In this case, ESC-AKA can play a full role and provide completely secure message transmission.

IX. Conclusion

In this work, an improved LTE authentication protocol called ESC-AKA was proposed by employing a self-certified public key cryptosystem based on the elliptic curve. The proposed ESC-AKA protocol effectively protects transmission messages by introducing a TC that produces the public key of each element on the basis of the original LTE network. The correctness of the proposed ESC-AKA protocol was formally verified using the strand space model. The performance analysis demonstrated that the ESC-AKA protocol consumes more bit traffic than the standard EPS-AKA due to the adoption of the public key encryption system. However, the security analysis revealed that ESC-AKA can successfully solve many of the security problems of the original EPS-AKA, and can effectively protect the security of the communication between users.

Compared with the 5G-AKA protocol, ESC-AKA can avoid the reveal of the user's location privacy caused by the plaintext transmission of error messages. The ESC-AKA protocol could be used to improve the security of LTE networks before 5G is deployed. In the future, if the ESC-AKA protocol can be combined with 5G, a new completely secure authentication solution can be provided.

References

[1] H. Holma and A. Toskala, *LTE Advanced: 3GPP Solution for IMT-Advanced*, 1st ed. Hoboken, NJ, USA: Wiley, 2012.

[2] (2022). *ERICSSON*. [Online]. Available: https://www.ericsson.com/en/reports-and-papers/mobility-report

[3] X. Zhang, A. Kunz, and S. Schröder, "Overview of 5G security in 3GPP," in *Proc. IEEE Conf. Standards Commun. Netw.*, Sep. 2017, pp. 181–186.

[4] D. Forsberg, G. Horn, W.-D. Moeller, and V. Niemi, *LTE Security*. Hoboken, NJ, USA: Wiley, 2012, pp. 101–105.

[5] *Rationale and Track of Security Decisions in Long Term Evolution (LTE) RAN/3GPP System Architecture Evolution (SAE)*, document 3GPP TS 33.821, Release 9, 3rd Generation Partnership Project (3GPP), 2009.

[6] J. Liu, "A new lightweight key exchange protocol with provable security for securing cloud-assistec. Mobile communications," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Atlanta, GA, USA, May 2017, pp. 772–777, doi: 10.1109/INFCOMW.2017.8116474.

[7] *Security; Security Architecture*, document 3GPP TS 33.102.3G, Version 14.1.0, 3rd Generation Partnership Project (3GPP), 2017.

[8] K. S. Suvidha and S. S. Kamath, "Secure 4G SEPS-AKA protocol for UMTS networks," in *Proc. 5th Int. Conf. Comput., Commun. Secur. (ICCCS)*, 2020, pp. 1–6.

[9] M. Ouaissa and M. Ouaissa, "An improved privacy authentication protocol for 5G mobile networks," in *Proc. Int. Conf. Adv. Comput., Commun. Mater. (ICACCM)*, 2020, pp. 136–143.

[10] S. R. Hussain, M. Echeverria, O. Chowdhury, N. Li, and E. Bertino, "Privacy attacks to the 4G and 5G cellular paging protocols using side channel information," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2019, pp. 1–15.

[11] M. Girault, "Self-certificated public keys," in *Proc. 10th Annu. Int. Conf. Theory Appl. Cryptograph. Techn.*, 1991, pp. 490–497.

[12] Z. Zhao and Y. Liu, "An authenticated encryption scheme based on self-certified of public key," *Comput. Eng. Appl.*, vol. 20, pp. 152–155, Oct. 2005.

[13] J. Cao *et al.*, "A survey on security aspects for 3GPP 5G networks," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 170–195, 1st Quart., 2020, doi: 10.1109/COMST.2019.2951818.

[14] A. Koutsos, "The 5G-AKA authentication protocol privacy," in *Proc. IEEE Eur. Symp. Secur. Privacy (EuroSP)*, Stockholm, Sweden, Jun. 2019, pp. 464–479, doi: 10.1109/EuroSP.2019.00041.

[15] C. Manero and F. Pujol. *5G: Almost 2 Billion EUR Expected for European 5G Investment in the Next 5 Years*. Accessed: Jun. 2017. [Online]. Available: https://en.idate.org/5g-investment/

[16] G. M. Koien, "Mutual entity authentication for LTE," in *Proc. 7th Int. Wireless Commun. Mobile Comput. Conf.*, İstanbul, Turkey, Jul. 2011, pp. 689–694, doi: 10.1109/IWCMC.2011.5982630.

[17] Y. Bai, Q. Wang, Q. L. Jia, and H. B. Zhang, "An efficient and secured AKA for EPS networks," *J. Beijing Univ. Posts Telecommun.*, vol. 38, no. 1, pp. 10–14, 2015.

[18] L. Wang, "Research on security of access in LTE," Commun. Eng. College, Xi'an Electron. Sci. Univ., Xi'an, China, Tech. Rep. 0909420550, 2012.

[19] B. L. Parne, S. Gupta, and N. S. Chaudhari, "PSE-AKA: Performance and security enhanced authentication key agreement protocol for IoT enabled LTE/LTE-A networks," *Peer-to-Peer Netw. Appl.*, vol. 12, pp. 1156–1177, Jul. 2019.

[20] K. Hamandi, J. B. Abdo, I. H. Elhajj, A. Kayssi, and A. Chehab, "A privacy-enhanced computationally-efficient and comprehensive LTE-AKA," *Comput. Commun.*, vol. 98, pp. 20–30, Jan. 2017.

[21] K. Hamandi, I. Sarji, A. Chehab, I. H. Elhajj, and A. Kayssi, "Privacy enhanced and computationally efficient HSK-AKA LTE scheme," in *Proc. 27th Int. Conf. Adv. Inf. Netw. Appl. Workshops*, Barcelona, Spain, Mar. 2013, pp. 929–934, doi: 10.1109/WAINA.2013.133.

[22] M. Prasad and R. Manoharan, "A robust secure DS-AKA with mutual authentication for LTE-A," *Appl. Math. Sci.*, vol. 9, no. 47, pp. 2337–2349, 2015.

[23] M. A. Abdrabou, A. D. E. Elbayoumy, and E. A. El-Wanis, "LTE authentication protocol (EPS-AKA) weaknesses solution," in *Proc. IEEE 7th Int. Conf. Intell. Comput. Inf. Syst. (ICICIS)*, Cairo, Egypt, Dec. 2015, pp. 434–441, doi: 10.1109/IntelCIS.2015.7397256.

[24] M. Purkhiabani and A. Salahi, "Enhanced authentication and key agreement procedure of next generation evolved mobile networks," in *Proc. IEEE 3rd Int. Conf. Commun. Softw. Netw.*, Xi'an, China, May 2011, pp. 557–563, doi: 10.1109/ICCSN.2011.6014956.

[25] S. Zhu, C. Zhou, and C. Li, "LTE authentication and key agreement scheme with enhanced user identity privacy," *Ordnance Ind. Autom.*, vol. 35, no. 11, pp. 47–51, 2016.

[26] H. Choudhury, B. Roychoudhury, and D. K. Saikia, "Enhancing user identity privacy in LTE," in *Proc. IEEE 11th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Liverpool, U.K., Jun. 2012, pp. 949–957, doi: 10.1109/TrustCom.2012.148.

[27] C.-I. Fan, Y.-F. Tseng, C.-H. Cheng, H.-N. Kuo, J.-J. Huang, and Y.-T. Shih, "Anonymous authentication and key agreement protocol for LTE networks," in *Proc. 2nd Int. Conf. Commun. Eng. Technol. (ICCET)*, Nagoya, Japan, Apr. 2019, pp. 68–71, doi: 10.1109/ICCET.2019.8726911.
[28] N. Singh and M. S. Saini, "A robust 4G/LTE network authentication for realization of flexible and robust security scheme," in *Proc. 3rd Int. Conf. Comput. Sustain. Global Develop. (INDIACom)*, New Delhi, India, 2016, pp. 3211–3216.
[29] O. E. Ekene, R. Ruhl, and P. Zavarsky, "Enhanced user security and privacy protection in 4G LTE network," in *Proc. IEEE 40th Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Atlanta, GA, USA, Jun. 2016, pp. 443–448, doi: 10.1109/COMPSAC.2016.108.
[30] M. Arapinis *et al.*, "New privacy issues in mobile telephony: Fix and verification," in *Proc. ACM Conf. Comput. Commun. Secur.*, Raleigh, NC, USA, 2012, pp. 205–216.
[31] X. Li and Y. Wang, "Security enhanced authentication and key agreement protocol for LTE/SAE network," *Proc. 7th Int. Conf. Wireless Commun.*, Wuhan, China, 2011, pp. 1–4.
[32] J. B. Bou Abdo, H. Chaouchi, and M. Aoude, "Ensured confidentiality authentication and key agreement protocol for EPS," in *Proc. Symp. Broadband Netw. Fast Internet (RELABIRA)*, Baabda, Lebanon, May 2012, pp. 73–77, doi: 10.1109/RELABIRA.2012.6235098.
[33] P. K. Panda and S. Chattopadhyay, "An improved authentication and security scheme for LTE/LTE-A networks," *J. Ambient Intell. Hum. Comput.*, vol. 11, no. 5, pp. 2163–2185, 2020.
[34] R. Ma, J. Cao, D. Feng, H. Li, Y. Zhang, and X. Lv, "PPSHA: Privacy preserving secure handover authentication scheme for all application scenarios in LTE-A networks," *Ad Hoc Netw.*, vol. 87, pp. 49–60, May 2019.
[35] G. Indra and R. Taneja, "An ECC-time stamp based mutual authentication and key management scheme for WSNs," in *Proc. 27th Int. Conf. Adv. Inf. Netw. Appl. Workshops*, Barcelona, Spain, Mar. 2013, pp. 883–889, doi: 10.1109/WAINA.2013.52.
[36] A. P. Haripriya and K. Kulothungan, "ECC based self-certified key management scheme for mutual authentication in Internet of Things," in *Proc. Int. Conf. Emerg. Technol. Trends (ICETT)*, Kollam, India, Oct. 2016, pp. 1–6, doi: 10.1109/ICETT.2016.7873657.
[37] X. Yan, "Algorithm design and application research of the improved ECC-based self-certified public key," College Electron. Technol., PLA Inf. Eng. Univ., Zhengzhou, China, Tech. Rep. 21320064014, 2010.
[38] O. Nait-Hamoud, T. Kenaza, and Y. Challal, "Certificateless public key systems aggregation: An enabling technique for 5G multi-domain security management and delegation," *Comput. Netw.*, vol. 199, Nov. 2021, Art. no. 108443.
[39] A. Imghoure, A. El-Yahyaoui, and F. Omary, "ECDSA-based certificateless conditional privacy-preserving authentication scheme in vehicular ad hoc network," *Veh. Commun.*, vol. 37, Oct. 2022, Art. no. 100504.
[40] A. Braeken, J.-J. Chin, and S.-Y. Tan, "ECQV-IBI: Identity-based identification with implicit certification," *J. Inf. Secur. Appl.*, vol. 63, Dec. 2021, Art. no. 103027.
[41] Q. Yang and D. Li, "Provably secure lattice-based self-certified signature scheme," *Secur. Commun. Netw.*, vol. 2021, Dec. 2021, Art. no. 2459628.
[42] F. J. T. Fabrega, J. C. Herzog, and J. D. Guttman, "Strand spaces: Why is a security protocol correct?" *Proc. IEEE Symp. Secur. Privacy*, Oakland, CA, USA, May 1998, pp. 160–171, doi: 10.1109/SECPRI.1998.674832.
[43] F. J. T. Fabrega, J. C. Herzog, and J. D. Guttman, "Strand spaces: Proving security protocols correct," *J. Comput. Secur.*, vol. 7, nos. 2–3, pp. 191–230, 1999.
[44] N. Petrakos, P. Kotzanikolaou, and C. Douligeris, "Using strand space model to verify the privacy properties of a fair anonymous authentication scheme," in *Proc. 16th Panhellenic Conf. Informat.*, Pireas, Greece, Oct. 2012, pp. 105–110, doi: 10.1109/PCi.2012.73.
[45] J. Fang, "Research on LTE/SAE network security mechanism based on strand space model," School Inf. Sci. Eng., Southeast Univ., Dhaka, Bangladesh, Tech. Rep. 20110304, 2011.
[46] J. Cao, M. Ma, Y. Fu, H. Li, and Y. Zhang, "CPPHA: Capability-based privacy-protection handover authentication mechanism for SDN-based 5G HetNets," *IEEE Trans. Depend. Sec. Comput.*, vol. 18, no. 3, pp. 1182–1195, May/Jun. 2021, doi: 10.1109/TDSC.2019.2916593.
[47] J. Cao, M. Ma, and H. Li, "LPPA: Lightweight privacy-preservation access authentication scheme for massive devices in fifth generation (5G) cellular networks," *Int. J. Commun. Syst.*, vol. 32, no. 3, pp. e3860.1–e3860.24, Feb. 2019.

Xiaofeng Lu (Member, IEEE) received the Ph.D. degree from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2010.

During his Ph.D., he held a Visiting Scholar positions at the University of Cambridge. He is currently an Associate Professor with the National Engineering Research Center of Mobile Network Technologies, Beijing University of Posts and Telecommunications. His main research interests include cyberspace security, information security, and cryptography.

Fan Yang is currently with the China Unicom Network Technology Research Institute. His main research interests include information security and cryptography.

Luwen Zou is currently pursuing the degree with the Beijing University of Posts and Telecommunications. His main research interests include cyber security.

Pietro Lio is currently a Professor at the Computer Laboratory, University of Cambridge, U.K. He is also a Fellow and the Director of studies at the Fitzwilliam College, University of Cambridge. He is also modeling biological processes on networks; modeling stem cells; and developing transcription and phylogenetic applications on a grid environment. He is also interested in bio-inspired design of wireless networks and epidemiological networks.

Pan Hui (Fellow, IEEE) received the bachelor's and M.Phil. degrees from The University of Hong Kong, and the Ph.D. degree from the Computer Laboratory, University of Cambridge. During his Ph.D., he was also affiliated with Intel Research Cambridge. He is currently a Professor of computer science and engineering at The Hong Kong University of Science and Technology. He is also a Distinguished Scientist at the Deutsche Telekom Laboratories (T-Labs), Berlin. His research interests include delay tolerant networking, mobile networking and systems, planet-scale mobility measurement, social networks, and the application of complex network science in communication system design.