A Secure and Efficient Access and Handover Authentication Protocol for Internet of Things in Space Information Networks

Kaiping Xue, *Senior Member, IEEE*, Wei Meng, Shaohua Li, David S. L. Wei, *Senior Member, IEEE*, Huancheng Zhou, and Nenghai Yu

***Abstract*—Space information network (SIN) makes it possible for any object to be connected to the Internet anywhere, even in the areas with extreme conditions, where a cellular network is not easy to deploy. Access authentication is the key to secure users' access control in SIN, mainly to prevent illegal adversaries from getting access to SIN services. However, the highly complicated communication environment of SIN (e.g., exposed links, higher signal delay, etc.) poses a challenging issue in the design of a secure and efficient authentication scheme. Although some authentication schemes have been proposed for SIN, they are unsuitable for Internet of Things (IoT) in SIN due to the high signaling overhead and insufficient security properties. Therefore, in this paper, we design a provably secure and efficient authentication protocol, along with an efficient handover mechanism, for IoT in SIN. In our design, we introduce a new authentication system model, where the satellites are given the ability to authenticate users to avoid the online involvement of the network control center (NCC) when authenticating users, thereby reducing long authentication delay and avoiding a single point of bottleneck in NCC. Furthermore, the support of batch verification in our design can significantly enhance handover efficiency when a group of users switch to another satellite. Our further analysis shows that our scheme is secure against various attacks and can meet a variety of security requirements. In addition, performance evaluation shows the superiority of our scheme on both delay and handover efficiency compared with existing schemes.**

***Index Terms*—Authentication, batch verification, handover, Internet of Things (IoT), space information network (SIN).**

I. Introduction

THE CONCEPT of Internet of Things (IoT) is an emerging paradigm that points out one of the directions in the evolution of the Internet, where any object (e.g., sensor devices, tags, and smart objects) can be interconnected over the Internet [1], [2]. Therefore, the IoT is an important technology in the current connected world. There are many burgeoning applications based on IoT paradigm such as smart grid, environmental monitoring, geologic disaster forecasting, etc. However, in many application scenarios, IoT devices are distributed in remote areas (e.g., desert, forest, and ocean) where the construction of the terrestrial network infrastructure is infeasible or with enormous-cost. To solve the problem of covering and deployment, the space information network (SIN) is considered as an interesting choice for providing the extensive coverage and rapid deployment in some applications of IoT. The SIN is a heterogeneous network which takes satellite network as backbone, and generally consists of a variety of satellites and spacecrafts, deployed in different orbits, ground stations, and mobile terminals with the satellite communication capability [3]. As illustrated in Fig. 1, satellites can be mainly divided into three categories according to their operating orbit altitude: 1) geosynchronous earth orbit (GEO) satellites; 2) medium earth orbit (MEO) satellites; and 3) low earth orbit (LEO) satellites. As SIN employs the networking of space platforms to serve the users on land, at sea, and in the air, it makes up the shortage of the traditional ground-based wireless networks [4], and thus has become an important means of interconnecting everything [5], [6], and has been gaining increasingly attention to network service providers.

As being closer to the earth, LEO satellites have shorter transmission delay compared with GEO and MEO satellites, and they are thus more suitable for providing data communication and access service in developing regions. Through LEO satellites served as access points, SIN can provide mobile users (MUs) with rich services varying from telephone and broadcast to broadband and Internet. However, similar to other wireless access networks, the data transmission in SIN is also transmitted directly over the air, which makes it vulnerable for adversaries to launch various malicious attacks [7]–[9]. The security issue in SIN is extremely important because once the data in SIN was illegally accessed, eavesdropped, or tampered, serious consequences may threaten national security and cause social unrest [10], [11]. To prevent illegally accessing the network and consuming valuable resources from unauthorized users is thus one of the most concerned issues [12], where the confidentiality of communication sessions is also equally important. Moreover, privacy is also a key issue in the design of access authentication protocol for SIN.

Manuscript received January 22, 2019; revised February 17, 2019; accepted February 27, 2019. Date of publication March 5, 2019; date of current version June 19, 2019. This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB0800301, in part by the National Natural Science Foundation of China under Grant 61671420, in part by the Youth Innovation Promotion Association CAS under Grant 2016394, and in part by the Fundamental Research Funds for the Central Universities. *(Corresponding author: Kaiping Xue.)*

K. Xue, W. Meng, S. Li, H. Zhou, and N. Yu are with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China (e-mail: kpxue@ustc.edu.cn; mengwei2@mail.ustc.edu.cn; lshhsl@mail.ustc.edu.cn; ynh@ustc.edu.cn).

D. S. L. Wei is with the Computer and Information Science Department, Fordham University, Bronx, NY 10458 USA (e-mail: wei@cis.fordham.edu).

Digital Object Identifier 10.1109/JIOT.2019.2902907

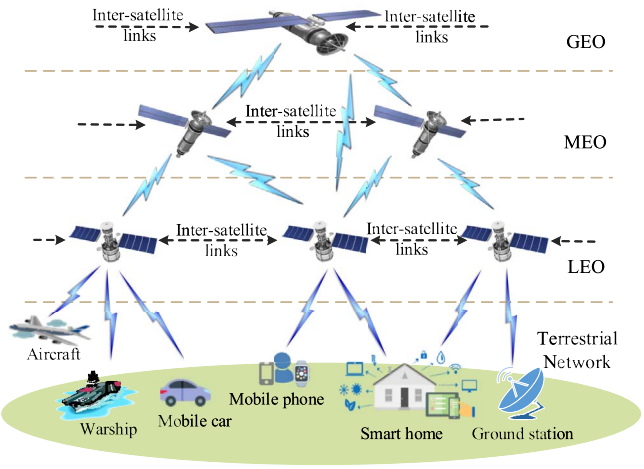


Fig. 1. Architecture of SIN.

Therefore, designing effective authentication scheme is very crucial for SIN to identify and reject any access request by an unauthorized user. Meanwhile, a secure channel should be established between an MU and SIN to provide confidentiality of the communication session. Besides, as the common security properties in traditional wireless networks [13]–[15], user anonymity, perfect forward/backward secrecy, and resisting the typical attacks are also required for SIN. In addition, compared with the traditional wireless networks, SIN has some special structural features, such as extremely long propagation delay, restricted computation ability, unstable and dynamic network topology, etc., which pose many technical challenges on the design of authentication mechanism [16]. Basically, the exceedingly unstable wireless links and long propagation delay will greatly enlarge the access latency when implementing the authentication schemes. Furthermore, the limited computation power and storage capacity of the involved entities in SIN make it unsuitable to implement algorithms with high complexity [17], and thus the proposed interaction protocols should try to avoid complex cryptographic procedures. Finally, the dynamic network topology makes the MUs to handover frequently during accessing to the SIN network. Meanwhile, nodes' large scale in IoT environment requires the proposed scheme to support bach authentication. Therefore, designing a secure and efficient access authentication protocol and handover authentication mechanism for SIN is obviously a nontrivial task.

In recent years, several access authentication schemes [7], [18]–[20] have been proposed to provide a secure and reliable SIN communication system. However, these schemes cannot be directly applied to IoT in SIN and none of them takes the long propagation delay of SIN into account. In these schemes, authentication is implemented between MUs and ground facilities [e.g., gateway and network control center (NCC)], whereas the satellite simply forwards the authentication signaling, rather than participating in practical authentication session. Consequently, an authentication protocol suffers at least four times of signaling transmission delay between the ground and the satellite (back and forth between user/satellite and NCC/satellite, respectively), which result in unacceptable access delay. Moreover, as an access point of SIN system, a satellite should be responsible for preventing unauthorized users from accessing SIN. However, these schemes cannot recognize the illegal access request until the signaling is forwarded to the ground facility. In addition, compared to access authentication, the secure handover is equally important, yet has not received much attention. How to ensure secure and efficient handover is undoubtedly an important issue to improve the QoS of SIN communications. In fact, with the advances of satellite hardware technology, satellite will carry larger computation ability, and be able to afford authentication interactions with users. Thus, it becomes a promising approach to let satellite, instead of the ground facilities, execute access control, thereby reducing network accessing delay. Motivated by these observations, this paper makes three main contributions.

1) We propose a new authentication system model that enables a more efficient authentication protocol to be developed. In our system, mutual authentication is conducted between a user and a satellite access point (SAP). The authentication scheme based on such system model reduces the latency of the implemented authentication process.
2) We identify some essential security requirements of a secure SIN authentication protocol and develop a secure and efficient authentication protocol for IoT based on the proposed authentication system model. In addition, we develop an efficient and secure handover mechanism for SIN to improve the QoS during handover, which can achieve batch handover authentication when a group of users handover to another satellite simultaneously.
3) In addition to the security analysis which demonstrates that the proposed authentication scheme indeed enforces its security guarantees, the experimental results also verify our scheme's efficiency compared with existing schemes.

The remainder of this paper is organized as follows. In Section II, we survey and analyze the related work, and then discuss their security issues and performance weakness. Section III introduces some mathematical preliminaries and Section IV introduces the system model and discusses the security requirements. In Section V, we present our proposed authentication protocol in detail. Then, informal and formal security analyses are presented in Sections VI and VII, respectively. The performance evaluation is presented in Section VIII. Finally, this paper is concluded in Section IX.

II. Related Work

In recently years, the use of satellite communication is found to be important in some applications of IoT due to the unique features of satellite communications. Many researchers have worked on the integration of the SIN and IoT to improve the performance of IoT applications (e.g., [21] and [22]).

In order to ensure the security for IoT users in SIN, it is extremely important to design an efficient access authentication scheme. Early in 1996, Cruickshank [20] first

presented an authentication system for satellite networks. In Cruichshank's [20] proposed scheme, MUs and NCC achieve mutual authentication by using public-key cryptosystem (PKC). Nevertheless, it is inefficient due to the high computation overhead and the complexity of the public-key management in a public-key infrastructure (PKI) system, and it is also insecure to reveal user's privacy. Then in 2003, Hwang *et al.* [7] proposed an improved authentication scheme based on secret-key cryptosystem (SKC) for mobile satellite communication systems. Their scheme has lower computation cost compared with the previous public-key-based authentication schemes (e.g., [20]). However, in 2005, Chang and Chang [23] found that Hwang *et al.*'s [7] scheme is still inefficient and insecure, because it lacks perfect forward secrecy and may suffer from stolen-verifier attacks. Consequently, in [23], they subsequently proposed a hash-chain-based authentication scheme to enhance the efficiency and security, in which, Diffie–Hellman [24] key exchange is used for the new session key generation. However, in Chang and Chang's [23] scheme, NCC will be a bottleneck of the secure communications in SIN when a large number of MUs are involved, as it must participate in each MU's authentication session. Moreover, Chang and Chang's [23] scheme may suffer from impersonation attacks and user's privacy is not kept confidential. Then, in 2009, a self-verification authentication protocol (CLC) based on PKC and SKC was first raised by Chen *et al.* [25], which can prevent heavy computation from MUs and eliminate the complexity of PKI. Besides, they also claimed that no sensitive information was involved in verification tables. Thereafter, many authentication schemes [26]–[29] based on CLC protocol were proposed for SIN.

There are some other schemes using different mechanisms [4], [30], [31], except the aforementioned authentication schemes. Chang *et al.* [4] proposed an authentication and key agreement protocol based on nonce mechanism and a three-party password-based authenticated key exchange protocol raised by Farash and Attari [30]. Chang *et al.*'s [4] scheme can resist replay attacks but under DoS attacks. While Farash and Attari's [30] scheme needs heavy computation overhead due to massive modular exponentiation computation.

Unfortunately, in all current access authentication schemes for SIN, it is commonly assumed that satellites act as relays between MUs and ground facilities, and only be responsible for forwarding messages. Thus the authentication is implemented between MUs and the ground facility, which will have a long access delay. Moreover, privacy is a serious concern for the authentication service in SIN whereas mobile privacy protection is a complicated issue. Users deeply concern about their privacy-related information such as the identity and position. But most of the current access authentication schemes (e.g., [7], [20], and [23]) for SIN ignore the consideration of user's privacy. Although some privacy-preserving authentication schemes based on group signature have been presented in [31] and [32] for some tradition networks in order to achieve the requirement of anonymity. Yang *et al.* [33] further proposed a group signature-based scheme for SIN. However, group signature introduces considerable computational complexity. Moreover, these techniques need to update the signing key or public-key once revocation occurs, which will result in unnecessary implemented delay. Thus, these schemes are time-consuming and unsuitable for mobile devices that are constrained by processing speed.

III. Mathematical Preliminaries

In this section, we will introduce the mathematical preliminaries needed for discussing our proposed scheme. We first describe the background of elliptic curve cryptography [34], [35], then briefly introduce the definition of ElGamal encryption [36].

A. Elliptic Curve Cryptography

We briefly introduce the elliptic curve defined on a prime field F_p, where p is a prime. In an elliptic curve cryptography system, the symbol E/F_p means elliptic curve E over a prime finite filed F_p, which can be defined as the form of $E_p(a, b) : y^2 = x^3 + ax + b \pmod p$, where $a, b \in F_p$, and with the discriminant $\Delta = 4a^3 + 27b^2 \neq 0 \pmod p$. The points on E/F_p together with a special point O, called zero point, form a group $\mathbb{G} = \{(x, y) : x, y \in F_q, E(x, y) = 0\} \cup \{O\}$. In elliptic curve cryptography, scalar multiplication over E/F_p is defined as $k \cdot G = G + G + \cdots + G$ (k times), where G is a base point on the elliptic curve. The following problems defined on $\mathbb{G}$ are assumed to be computational infeasible.

1) *Elliptic-Curve Discrete Logarithm Problem (ECDLP):* Let G be the generator of the additive cyclic group $\mathbb{G}$. It is computationally hard to compute integer x for given points P and $P = x \cdot G$.
2) *Elliptic-Curve Computational Diffie–Hellman Problem (ECDHP):* Given two random points $a \cdot G$ and $b \cdot G$ on elliptic curve, it is hard to compute $ab \cdot G$, where a and b are two unknown integers.

B. ElGamal Encryption

In cryptography, the ElGamal encryption system proposed by Elgamal [36] in 1985 is an asymmetric key encryption algorithm for public-key cryptography which is based on the Diffie–Hellman key exchange. In ElGamal encryption system, any one uses Alice's public-key to encrypt a message to assure confidentiality, and only Alice who possesses the corresponding private key can decrypt the ciphertext. We use ElGamal encryption technology to encrypt the tuple $(\mathrm{TID}_j^i \parallel \mathrm{ID}_j)$ and only the entity who has the private key can get real identity ID_j corresponding to the temporary identity TID_j^i. Therefore, the user accountability can be achieved. The ElGamal encryption algorithm with elliptic curve consists of following three processes [34].

1) *Key Generation:* Choose a secret key $\mathrm{sk} \in_R \mathbb{Z}_p^*$, and then publish the point $\mathrm{pk} = \mathrm{sk} \cdot G$.
2) *Encryption (Enc (pk, m)):* Choose a random $k \in_R \mathbb{Z}_p^*$ to compute $C_1 = k \cdot G$ and $C_1 = k \cdot \mathrm{pk}$. Then compute $P_m = f(m)$, where $f : m \mid\!\rightarrow P_m$ is a public known function, which maps the message m to the point P_m on the elliptic curve E. The ciphertext is $(C, D) = (C_1, C_2 + P_m)$.

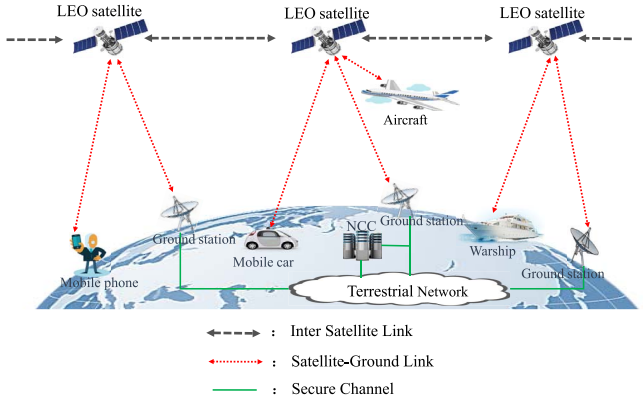


Fig. 2. System model of our proposed scheme.

3) *Decryption (Dec (sk, c)):* Compute $C' = \mathrm{sk} \cdot G$ and retrieve the point P_m with $P_m = D - C' = k(\mathrm{sk} \cdot G) + P_m - \mathrm{sk}(k \cdot G)$ upon receiving a ciphertext (C, D). Then compute message m as $f^{(-1)}(P_m)$.

IV. System and Security Model

A. System Model

The access authentication and handover interactions in this paper proceed in a communication system which mainly includes the following entities: MUs, SAPs, the ground station/gateway (G), and the NCC. Fig. 2 shows the distribution and interrelationship of the entities in the SIN structure. In order to join the network, an MU should first register with NCC, then subscribes services and connects to a SAP for accessing network resources. As illustrated in Fig. 2, in IoT environments, MUs can be any mobile object, such as cell phones, aircrafts, cars, and ships. In the traditional authentication scenario, LEO satellites are mainly responsible for forwarding messages between MUs and ground stations, and these authentication messages need to be forwarded to the ground station to verify their validity, which obviously brings in a higher latency for the authentication protocol implementation in SIN. Therefore, for lowering latency, our scheme enforces the authentication protocol between MU and SAP based on the fact that the capacity of computation and storage of satellites is gradually being enhanced with the rapid development of space technology. Ground stations around the world can provide an interface to the terrestrial network for MUs. NCC is responsible for the registration of network entities and user's authorization, and it can communicate with ground stations via secure channels; MUs cannot only be the user on the ground, but also the mobile nodes in the air or at sea, such as aircraft or warship, respectively; the LEO satellites mainly take the responsibilities of SAPs due to their lower latency compared with MEO and GEO satellites.

B. Security Model

A highly exposed communication system should face the threat of message eavesdropping and unauthorized network accessing, which is a common challenge for most of the wireless techniques. Especially, various attacks can threaten the network access legitimacy: *impersonation attacks* let malicious attacker obtain unauthorized access to the network and gain benefits by impersonating some authorized users to forge messages; *replay attacks* let the adversaries make use of historical messages to cheat the service; *tamper attacks* not only affect the networks, but also block authorized users' normal sessions, by tampering with the data during communication. Another concern for mobile networks is user's privacy, as user's real identity and location, such as in military environments, will sometimes be sensitive to adversaries [26]. However, traditional authentication (e.g., PKI-based) forces users to reveal their identities to other entities. Therefore, how to solve the contradiction between privacy preservation and authentication is still a big challenge in this research area.

In the system model mentioned in Section IV-A, we assume that NCC is trustworthy for all entities in our system and any adversary cannot compromise NCC. There is a secure channel between the network entity and NCC to protect the registration process. However, MUs and SAPs might be impersonated by attackers to launch replay attacks, impersonation attacks, and so on. In addition, for some purposes (e.g., tracking activity track), G, SAPs, and other wireless eavesdroppers may be curious about the real identity and location of users.

Frequent handover is another common problem needed to be solved in SIN due to the high-speed movement of satellites. When handover occurs, to implement full reauthentication may be unbearable since the long propagation link makes it far more difficult to ensure seamless session in SIN. Moreover, inappropriate mechanisms for handover scenarios will result in serious consequences if the later SAP can learn some sensitive and key information between the MU and the former SAP. Thus the authentication should not only be fast in order to allow MU to continue its applications with the required quality of service (QoS) when the handover occurred, but also be secure enough to keep the confidentiality of past/future communication session, which will pose great challenges on devising a secure handover mechanism.

For the security of SIN, it is important to have a clear understanding of the requirements that a secure and efficient authentication protocol should meet. The essential requirements are listed and briefly described as follows.

1) *Mutual Authentication:* An SAP must authenticate an MU to ensure their legitimacy, meanwhile users should be allowed to authenticate the accessed SAP to avoid potential deception and other malicious attacks.
2) *Key Establishment:* A session key should be negotiated between a user and an SAP to ensure the security of subsequent communication between them.
3) *User Privacy:* An MU's real identity and location are two major privacy concerns in SIN. For protecting user identity privacy, no one including other entities (e.g., G and SAP) in SIN and a third party can get the user's real identity from the intercepted message. Moreover, to protect the user's location privacy, the user's activities should be unlinkable, which means that adversaries should not link the communication activities of a particular MU together to monitor his/her behavior through obtaining current location of the linkable identity.

TABLE I
NOTATIONS USED IN OUR SCHEME

Nations	Definition
p	a k-bit prime
$\mathbb{Z}_p$	a prime finite field
$E_p(a,b)$	elliptic curve over a prime field $GF(p)$
$\mathbb{G}$	elliptic curve group
r_x	the random number generated by x
MU_j	the mobile user j
ID_x	the real identity of x
TID_x	the temporary identity of x
P_x^i	the i-th verification table of x
sk_x	the longtime private key of x
pk_x	the longtime public key of x
sk_x^i	the i-th lifetime limited private key of x
pk_x^i	the i-th lifetime limited public key of x
SK	the session key between user and ground station
$h_1()$	a secure hash function $h_1 : \{0,1\}^* \to \{0,1\}^n$
$h_2()$	a secure hash function $h_2 : \{0,1\}^* \times \mathbb{G} \to \mathbb{Z}_p$
$Enc(k,m)$	encrypting message m using key k
$\|$	concatenation operation

4) *Traceability:* We expect that a secure authentication protocol can achieve strong anonymity and unlinkability. However, it is essential to have ability to reveal the related private information of a user for SIN when misbehavior of an MU occurs.
5) *Backward/Forward Secrecy:* Considering the existence of handover in SIN, our protocol should achieve the backward and forward secrecy. The *forward secrecy* means that the knowledge of previous session keys cannot help an adversary to derive the future session keys, and thus the security of its future communication can be guaranteed during handover. Meanwhile, for the security of the previous interaction data, our proposed scheme should also provide the *backward secrecy* which means that the compromise of one secure channel will not compromise the security of previous channels.

V. PROPOSED SCHEME

In this section, we present our proposed authentication scheme in detail. Our scheme consists of five parts: 1) initial phase; 2) registration phase; 3) prenegotiation phase; 4) authentication and key agreement phase; and 5) handover phase. Table I lists some basic notations for the clarity of description.

A. Initial Phase

NCC chooses G and $E_p(a,b)$, where $E_p(a,b)$ is a nonsingular elliptic curve over a prime field GF_p and G is a base point over $E_p(a,b)$. Then NCC generates a long-term private key $\mathrm{sk_{NCC}}$, and the corresponding public-key $\mathrm{pk_{NCC}}$ can be computed as $\mathrm{sk_{NCC}} \cdot G$. Similarly, MU_j also generates his/her longtime private public-key sk_j and the corresponding public-key pk_j, which is defined as $\mathrm{sk}_j \cdot G$.

B. Registration Phase

Before a new MU accesses SIN, he/she must register with NCC to become a legal user. In order to provide identity anonymous protection, NCC will generate n temporary identities $\mathrm{TID}_i (i = 1, 2, \ldots, n)$ and send them to the MU through a secure channel during the registration phase. Meanwhile, the user will obtain the private key and corresponding public-key for each TID_i. Each satellite also needs to register with NCC, but only requires one permanent identity. The details of this phase are as follows.

1) User Registration: In the user registration phase, MU_j sends a registration request to NCC for registration to the SIN along with the real identity ID_j via a secure channel. Upon receiving the registration message, NCC first chooses n random numbers $q_j^i (i = 1, 2, \ldots, n)$, then NCC computes the ith public-key, temporary identity, proof, and encrypted private key as follows:

$$\begin{aligned}
\mathrm{pk}_j^i &= q_j^i \cdot G \\
\mathrm{TID}_j^i &= h_1\left(\mathrm{ID}_j \parallel pk_j^i\right) \\
P_j^i &= \mathrm{pk}_j^i \parallel \mathrm{TID}_j^i \parallel \mathrm{Enc}\left(\mathrm{pk_{NCC}}, \mathrm{TID}_j^i, \mathrm{ID}_j\right) \parallel \mathrm{LT}_j^i \\
\mathrm{sk}_j^i &= q_j^i + h_2\left(\mathrm{TID}_j^i \parallel P_j^i\right) \cdot \mathrm{sk_{NCC}}, \;\; Ed_j^i = \mathrm{Enc}\left(\mathrm{pk}_j, \mathrm{sk}_j^i\right)
\end{aligned}$$

where pk_j^i and sk_j^i are the ith temporary identity TID_j^i's public-key and private key, respectively. TID_j^i plays the role of pseudonym for the purpose of anonymity in the future authentication and key agreement phase. LT_j^i is the lifetime of ith temporary identity defined by NCC. The public-key, temporary identity, and the lifetime of ith temporary identity is included with the proof P_j^i. In order to protect key secret, the private key sk_j^i is encrypted with the long-term public-key of the MU. Finally, NCC sends all $< P_j^i, Ed_j^i >$ to MU_j.

2) Satellite Registration: All SAPs also need to register with NCC. Then, NCC will compute for each SAP as follows:

$$\begin{aligned}
\mathrm{pk_{SAP}} &= q_{\mathrm{SAP}} \cdot G \\
P_{\mathrm{SAP}} &= \mathrm{pk_{SAP}} \parallel \mathrm{ID_{SAP}} \\
\mathrm{sk_{SAP}} &= q_{\mathrm{SAP}} + h_2(\mathrm{ID_{SAP}} \parallel P_{\mathrm{SAP}}) \cdot \mathrm{sk_{NCC}}
\end{aligned}$$

where $\mathrm{pk_{SAP}}$ and $\mathrm{sk_{SAP}}$ are the SAP's public-key and private key, respectively. And the proof P_{SAP} contains the public-key and the identity of SAP. After registration, each SAP gains a pair of long-term public and private keys from NCC, which will be stored locally in a secure manner.

C. Prenegotiation Phase

We assume that the secure channel between satellites and gateways have been established. In this phase, each ground station on the earth needs to send prenegotiation message to SAPs via the secure channel. The prenegotiation message contains a timestamp and a key agreement parameter R_G, computed as $r_G \cdot G$, where r_G is a random number selected by the gateway, which will be utilized to generate the session key in authentication and key agreement phase. After receiving the message, the SAP first checks the timestamp to prevent the replay attacks, then stores this prenegotiation message. For the security of the further key agreement, the ground station will broadcast prenegotiation message periodically to update the key agreement parameter R_G. Although a specific recommended frequency

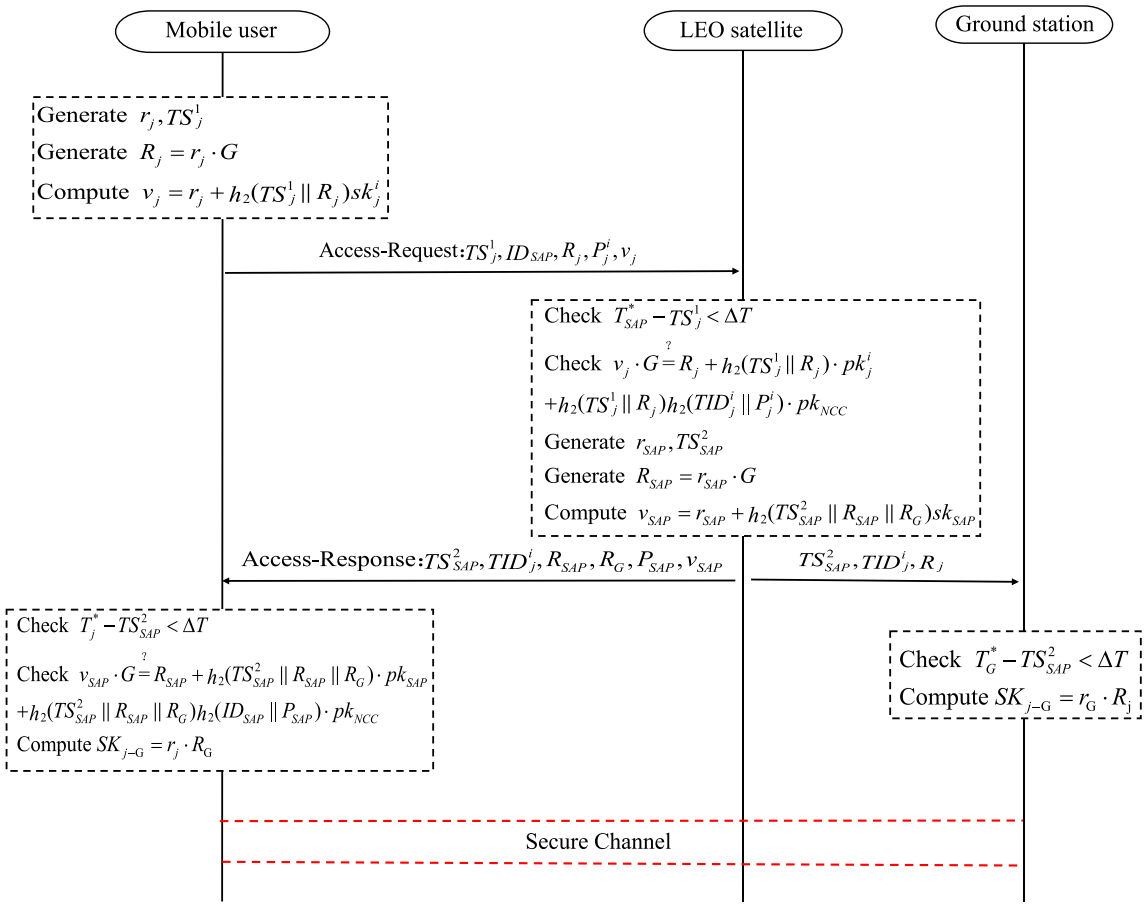


Fig. 3. Proposed authentication protocol.

for update needs not to be indicated, periodic R_G refreshment is a fundamental security technology that can help eliminate potential threats of key agreement [37].

D. Authentication and Key Agreement Phase

An authentication interaction proceeds when a user requests the access of the resources in SIN or communicating with other users. During the authentication phase, the MU and gateway will negotiate an session key at the same time. The proposed authentication scheme is illustrated in Fig. 3. For simplicity, we suppose that SAPs and ground stations have finished the mutual authentication and established a secure channel by adopting secure sockets layer protocol or transport layer security protocol [38]. The details of this phase are described as follows.

1) *Step A1 ($MU_j \rightarrow SAP$: $\{TS_j^l, ID_{SAP}, R_j, P_j^i, v_j\}$):* For the purpose of protecting the MU's location privacy, it is insecure to use an invariable temporary identity all the while in the SIN. Therefore, when MU_j generates the access-request message, he/she will randomly picks an unused P_j^i and the corresponding Ed_j^i, then gets the private key sk_j^i through decrypting Ed_j^i. MU_j generates a secret random number r_j and a timestamp value TS_j^1 to calculate $R_j = r_j \cdot G$, $v_j = r_j + h_2(TS_j^1 \| R_j)\mathrm{sk}_j^i$. Finally, MU_j sends the access-request message $\{TS_j^1, \mathrm{ID_{SAP}}, R_j, P_j^i, v_j\}$ to SAP.
2) *Step A2 ($SAP \rightarrow MU_j$: $\{TS_{SAP}^2, TID_j^i, R_{SAP}, R_G, P_{SAP}, v_{SAP}\}$):* Upon the receipt of $\{TS_j^1, \mathrm{ID_{SAP}}, R_j, P_j^i, v_j\}$, the SAP first checks whether the transmission delay is within the allowed time interval ΔT. Here, ΔT is an empirical value, which is adopted to prevent the message from being a replay one. If the time synchronization in the SIN can be accurate to several minutes ($< t_1$ min) and the measured round-trip time is t_2 min, we can set $\Delta T = (2t_1 + 3t_2)$ min. We assume that the current time is T_{SAP}^*. If $T_{\mathrm{SAP}}^* - TS_j^1 > \Delta T$, SAP stops here and sends REJ message back to MU_j. Otherwise, SAP continues to check whether $v_j \cdot G$ is equal to $R_j + h_2(TS_j^1 \| R_j) \cdot \mathrm{pk}_j^i + h_2(TS_j^1 \| R_j) h_2(\mathrm{TID}_j^i \| P_j^i) \cdot \mathrm{pk_{NCC}}$. If the equality does not hold, the authentication request is rejected to terminate this session. Otherwise, MU_j is authenticated and carries on the following procedures: SAP generates a random number r_{SAP} and a timestamp value TS_{SAP}^2, then computes $R_{\mathrm{SAP}} = r_{\mathrm{SAP}} \cdot G$, $v_{\mathrm{SAP}} = r_{\mathrm{SAP}} + h_2(TS_{\mathrm{SAP}}^2 \| R_{\mathrm{SAP}} \| R_G) \cdot \mathrm{sk_{SAP}}$. Finally, SAP sends the access-response $\{TS_{\mathrm{SAP}}^2, \mathrm{TID}_j^i, R_{\mathrm{SAP}}, R_G, P_{\mathrm{SAP}}, v_{\mathrm{SAP}}\}$ to MU_j and $\{TS_{\mathrm{SAP}}^2, \mathrm{TID}_j^i, R_j\}$ to the ground station via established secure channel simultaneously. These two steps are in parallel at the same time

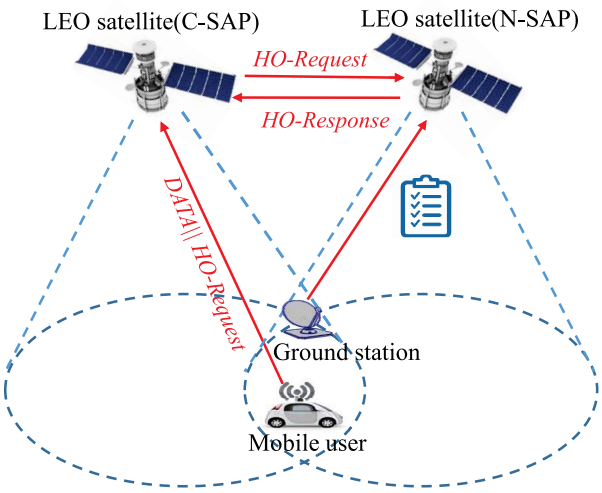


Fig. 4. Handover scenario (without ground station change).

in order to cut down some transmission delay. Here, $R_G = r_G \cdot G$ generated by the ground station is obtained after establishing a secure channel between SAP and ground station, which is stored in SAP's local-storage and updated periodically.

3) *Step A3:* After receiving the message from SAP, MU_j first verifies whether the transmission delay is within the allowed time interval ΔT. Assume that the current time is T_j^*. If $T_j^* - TS_{\mathrm{SAP}}^2 > \Delta T$, the MU rejects and stops here. Otherwise, MU_j checks whether $v_{\mathrm{SAP}} \cdot G$ is equal to $R_{\mathrm{SAP}} + h_2(TS_{\mathrm{SAP}}^2 \parallel R_{\mathrm{SAP}} \parallel R_G) \cdot \mathrm{pk}_{\mathrm{SAP}} + h_2(TS_{\mathrm{SAP}}^2 \parallel R_{\mathrm{SAP}} \parallel R_G) h_2(\mathrm{ID}_{\mathrm{SAP}} \parallel P_{\mathrm{SAP}}) \cdot \mathrm{pk}_{\mathrm{NCC}}$. If equality does not hold, the MU terminates the session and the authentication fails; otherwise, he/she is convinced that the SAP is legitimate and trusted, then computes the session key shared with the ground station: $SK_{j-G} = r_j \cdot R_G$. At the same time, after receiving message $\{TS_{\mathrm{SAP}}^2, \mathrm{TID}_j^i, R_j\}$, the ground station also needs to check whether the transmission delay is within the allowed time interval ΔT. Assume that the current time is T_G^*. If $T_G^* - TS_{\mathrm{SAP}}^2 > \Delta T$, the ground station rejects and stops here. Otherwise, the ground station computes the session key: $SK_{j-G} = r_G \cdot R_j$.

In the end, the session key $SK_{j-G} = r_j r_G \cdot G$ is securely shared between the ground station and the MU. Consequently, a secure channel between an MU and the ground station is established across the SAP: the communication packet in the session is encrypted by one of them (user/ground station), and is decrypted by the other when receiving the packet.

E. Handover Phase

In SIN, satellites move with higher speeds relative to the Earth's surface, which results in a high dynamic feature of the network topology. This topological feature brings a major challenge for continuous and secure communication. Therefore, it is essential to give a seamless and secure handover scheme to guarantee QoS for some services, especially, for the real-time ones. Although the network topology changes rapidly, the change is periodic and predictable because the satellite has strict orbital movements. Meanwhile, the users connected to the same satellite have strong similarities when switching, such as the same original SAP, the same new SAP (N-SAP) or G, and the same switching time. So it is reasonable to perform handover for these users in a group manner. Therefore, it is advisable to take above properties into account when designing handover scheme. Next, we will introduce two possible handover scenarios, and propose two handover mechanisms for these two scenarios, respectively.

1) Mobile User With Low Speed: This kind of user is stationary relative to the high speed satellite, so the transfer of satellite coverage is the major cause of handover. This kind of handover scenario, shown in Fig. 4, happens frequently, and we can see that the established connection is only handed over from the current SAP (C-SAP) to the N-SAP with the ground station unchanged. Therefore, the session key shared between the MU and the ground station need not to renegotiate. Considering this situation, we give an available solution to the problem of QoS guarantees when this kind of handover scenario occurs, which is described as follows.

1) *Pretransmission ($G \rightarrow$ N-SAP : White List):* The ground station has the satellite constellation topology and the movements of the satellite, so it can predict the coming satellite N-SAP. Then the ground station transfer a *white list* to N-SAP via secure channel in advance. The entry of *white list* contains (TID_j, $\mathrm{ID}_{\text{C-SAP}}$), where TID_j represents the temporary identity of the legal MU_j, who is connecting to the network currently, and $\mathrm{ID}_{\text{C-SAP}}$ is the identity of the C-SAP. To be noted, both of the two elements (TID_i, $\mathrm{ID}_{\text{C-SAP}}$) in each entry are 128 bits, so a two tuple of each entry in the whitelist occupies 256 bits of memory. Assuming that even though there are tens of thousands of devices connecting to the same satellite, the satellite only needs to store a whitelist of megabytes in size. It can be seen that the storage overhead consumed by the whitelist is small. Moreover, in the satellite-based IoT environment, IoT devices can be any mobile object, such as satellite phones, aircrafts, cars, ships, which are equipped with satellite communications capabilities. This requirement limits the number of IoT nodes connecting to the same satellite at the same time. Therefore, by adopting our scheme, the storage overhead for a satellite is acceptable.
2) *Step 1 ($\mathrm{MU}_j \rightarrow$ C-SAP : HO-Request (TID_j, $\mathrm{ID}_{\text{N-SAP}}$)):* When the MU is located in the overlap between the C-SAP and N-SAP, he/she decides whether to hand off according to the received signal strength. Before handover, MU_j sends HO-request message along with the data to C-SAP, the request message include the temporary identity of MU_j and the identity of N-SAP. Then, C-SAP forwards the request message to N-SAP.
3) *Step 2 (N-SAP $\rightarrow$ MU_j : HO-Response (ACC/REJ)):* Upon receiving the HO-request message, N-SAP verifies whether MU_j is legitimate by looking up TID_j in the *white list*. If TID_j is included in the *white list*, N-SAP accepts MU_j's handover request and sends "ACC" message back to MU_j through C-SAP. Otherwise, N-SAP rejects it and sends "REJ" message.

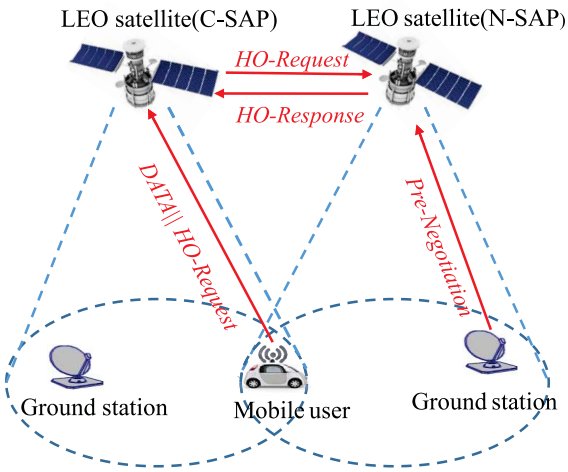


Fig. 5. Handover scenario (with ground station change).

2) Mobile User With High Speed: For the MU with high speed, such as various kinds of aircraft, reasons for handover are not only the transfer of satellite coverage, but also the movements of MUs. Therefore, as illustrated in Fig. 5, the MU not only switches from the C-SAP to a N-SAP, but also from the current ground station to a new ground station. Consequently, the session key must be changed for ensuring secure communication between MU and the new ground station. In such situation, MU and new satellite can use the preauthentication method to finish mutual authentication and key agreement.

1) *Prenegotiation:* When footprint of N-SAP enters the area of a new ground station, the prenegotiation phase will be executed between the new ground station and N-SAP. The details are similar to "prenegotiation phase" in Section V-C.
2) *Step 1 ($MU_j \rightarrow$ C-SAP : HO-Request (TS_j, $ID_{C\text{-}SAP}$, R_j, P_j^i, v_j)):* When the MU is located in the overlapping area, he/she decides whether to hand over according to the received signal strength. If the decision is to hand over, MU_j sends HO-request message along with the data to C-SAP, and the generation of HO-request message is similar to step A1. Then, C-SAP forwards all HO-request messages to N-SAP.
3) *Step 2 (N-SAP $\rightarrow$ MU_j : HO-Response (ACC/REJ)):* Upon receiving L HO-request messages generated by L MUs: $\{TS_j, \mathrm{ID}_{\text{C-SAP}}, R_j, P_j^i, v_j\}$ $(j = 1, 2, \ldots, L)$, N-SAP validates the legality of L HO-request messages by adopting batch verification mechanism, so it can verify the following equation:

$$G \cdot \sum_{j=1}^{L} v_j \stackrel{?}{=} \sum_{j=1}^{L} \left(R_j + h_2\left(TS_j \parallel R_j\right) \cdot \mathrm{pk}_j^i\right) + \mathrm{pk}_{\mathrm{NCC}} \times \sum_{j=1}^{L} \left(h_2\left(TS_j \parallel R_j\right) h_2\left(\mathrm{TID}_j^i \parallel P_j^i\right)\right).$$

If the above equation holds, the authentication is completed and N-SAP accepts all handover requests and sends "ACC" message back to these MUs. Otherwise, it detects the invalid HO-request message through the "divide-and-conquer" approach (BVDC) [39], and then rejects it and sends "REJ" message. Meanwhile, N-SAP sends the message $\{TS_{\mathrm{SAP}}, R_{\mathrm{SAP}}, R_G, P_{\mathrm{SAP}}, v_{\mathrm{SAP}}\}$ back in order to achieve mutual authentication and key agreement. Finally, MUs who have finished the preauthentication successfully share a new session key with the new ground station, and then the secure handover is achieved.

VI. Security Analysis

In this section, we give rigorous formal security analysis under the real-or-random mode and then analyze the security properties of the proposed authentication scheme with respect to the security requirements given in Section IV. The security analyses are presented as follows.

A. Formal Security Proof Using Random Oracle Model

We show that the proposed protocol can achieve user mutual authentication and key agreement in the random oracle mode. The proof includes the security model part and proof process part.

1) Security Model: In order to prove the security of our proposed scheme, we assume that there is a polynomial time adversary $\mathcal{A}$, who can access all the transmitted messages between the communicating parties, and also knows all the public parameters. We use the symbol $\prod_P^k$ to denote the kth instance of participation P, where P represents the participators in the protocol $\mathcal{P}$, i.e., U_j and SAP. Each instance $\prod_P^k$ can also be referred to as an oracle. For each oracle, there are three results: 1) *accept*; 2) *reject*; and 3) $\perp$. If an oracle obtains the right message, then outputs *accept*. If an oracle obtains the wrong message, then *reject* is the outcome. If an oracle does not receive any result, then $\perp$. The adversary $\mathcal{A}$ can use the simulator to conduct the following queries to break the security of the proposed protocol.

1) *Extract (ID_{U_j}):* In this query model, an adversary $\mathcal{A}$ can get the public/private key pair corresponding to user identity ID_{U_j}.
2) *Send ($\prod_P^k$, M):* $\mathcal{A}$ launches an active attack on $\prod_P^k$ by accessing this query. In this query model, an adversary, $\mathcal{A}$, can send a message M to the oracle $\prod_P^k$. When receives M, $\prod_P^k$ makes computations and returns the corresponding replay to $\mathcal{A}$ according to the proposed protocol.
3) *$h_1(m_1)$:* $\mathcal{A}$ makes this hash query with message m_1, and the oracle, $\prod_P^k$, checks whether m_1 already exists in list L_{h_1}. If it exists, $h_1(m_1)$ is returned. Otherwise, a random value $h_1 \in \{0, 1\}^n$ will be generated to replay $\mathcal{A}$, and the tuple (m_1, h_1) is put into list L_{h_1}.
4) *$h_2(m_2)$:* $\mathcal{A}$ makes this hash query with message m_2, and the oracle, $\prod_P^k$, checks whether m_2 already exists in list L_{h_2}. If it exists, $h_2(m_2)$ is returned. Otherwise, a random value $h_2 \in \mathbb{Z}_p$ will be generated to replay $\mathcal{A}$, and the tuple (m_2, h_2) is put into the list L_{h_2}.

5) *Reveal ($\prod_P^k$):* If the oracle accepts this query, the oracle return a session key to adversary $\mathcal{A}$; otherwise, the oracle returns a null result ($\perp$) to $\mathcal{A}$.
6) *Corrupt (ID_P):* In this query model, the adversary $\mathcal{A}$ can request the private key of participant P and gets back its private key.
7) *Test ($\prod_P^k$):* The *Test* query is used to measure the strength of semantic security of session key. $\mathcal{A}$ can send a single test query to oracle $\prod_P^k$. Upon receiving the query, oracle $\prod_P^k$ begins with the tossing of an unbiased coin $c \in \{0, 1\}$, and the outcome of which is kept secret. The real session key is returned to $\mathcal{A}$ if $c = 1$, or a random bit string is returned if $c = 0$.

a) Semantic security of session key: $\mathcal{A}$ needs to distinguish an instance's fresh session key from a random key as part of the experiment. $\mathcal{A}$ is allowed to execute many *Test* queries to either the MU instance or the LEO SAP instance. At the end of the game, $\mathcal{A}$ must output a guess bit c'. If $c' = c$, the adversary $\mathcal{A}$ wins the game. We denote $\Pr[\text{Succ}]$ as the probability that $\mathcal{A}$ wins the game, the advantage of $\mathcal{A}$ in breaking the semantic security of our proposed authentication and key agreement protocol $\mathcal{P}$ becomes $\text{Adv}_{\mathcal{P}}(\mathcal{A}) = |2\Pr[\text{Succ}] - 1|$. Then we call that our proposed scheme is secure under the random oracle if $\text{Adv}_{\mathcal{P}}(\mathcal{A})$ is negligible [i.e., $\text{Adv}_{\mathcal{P}}(\mathcal{A}) < \varepsilon$, for any sufficiently small $\varepsilon > 0$].

2) Security Proof:

Definition 1: The proposed scheme is said to be secure if:
1) the proposed scheme is anonymous, since no adversary can get the user's identity in polynomial time;
2) in the presence of a benign adversary on $\prod_{U_j}^n$ and $\prod_{\text{SAP}}^t$, both oracles always agree on the same session key, and this key is distributed uniformly at random. For any polynomial adversary, the probability of success $\text{Adv}_{\mathcal{P}}(\mathcal{A})$ is negligible.

Lemma 1: The proposed authentication and key agreement protocol is anonymous against any probabilistic polynomial-time adversary, $\mathcal{A}$, who has got all information in the communication channel.

Proof: Suppose that the adversary, $\mathcal{A}$, can get message $M_1 = \{\text{ID}_{\text{SAP}}, P_{U_j}^k, R_{U_j}, v_{U_j}, TS_{U_j}^1\}$ and $M_2 = \{\text{TID}_{1,U_j}^k, P_{\text{SAP}}, R_Z, v_{\text{SAP}}, TS_{\text{SAP}}^2\}$, where $P_{U_j}^k = \{\text{pk}_{U_j}^k \parallel h_{1,U_j}^k \parallel \text{Enc}(\text{pk}_{\text{NCC}}, h_{1,U_j}^k, \text{ID}_{U_j}) \parallel \text{LT}_{U_j}^k\}$, $\text{pk}_{U_j}^k = q_{U_j}^k \cdot G$, and $h_{1,U_j}^k = h_1(\text{TID}_{U_j}, \text{pk}_{U_j}^k)$. User's real identity ID_{U_j} is contained in $h_{1,U_j}^k = h_1(\text{TID}_{U_j}, \text{pk}_{U_j}^k)$ and $\text{Enc}(\text{pk}_{\text{NCC}}, h_{1,U_j}^k, \text{ID}_{U_j})$. Under the condition that the private key of NCC is secure, $\mathcal{A}$ cannot get the real identity of the user from $\text{Enc}(\text{pk}_{\text{NCC}}, h_{1,U_j}^k, \text{ID}_{U_j})$. If $\mathcal{A}$ can obtain the user's real identity from $h_{1,U_j}^k = h_1(\text{TID}_{U_j}, \text{pk}_{U_j}^k)$, it means that $\mathcal{A}$ can reconstruct x from its hash value $h_1(x)$ within polynomial time. Due to the one-way hash function, it is impossible to get ID_{U_j} from $h_{1,U_j}^k = h_1(\text{TID}_{U_j}, \text{pk}_{U_j}^k)$. Thus, none information on user's identity can be leaked, and our proposed scheme can achieve anonymity. ∎

Lemma 2: Assuming ECDHP is intractable, then the advantage of an adversary against our proposed protocol is negligible in the random oracle model.

Proof: Suppose that there is an adversary, $\mathcal{A}$, who can break the mutual authenticated key agreement semantic security of the proposed protocol with a non-negligible probability $\lambda(k)$ in polynomial-time t. We can construct another algorithm $\mathcal{F}$ from adversary $\mathcal{A}$ to solve the ECDHP with another non-negligible probability. The advantage of $\mathcal{F}$ in solving the ECDH problem is $\text{Adv}^{\text{ECDH}}(\mathcal{F})$.

1) *Setup:* First, algorithm $\mathcal{F}$ is given the system parameters $(F_p, E/F_p, G, h_1, h_2)$ and an instance $(G, a \cdot G, b \cdot G)$ of the ECDHP, and its task is to compute $ab \cdot G$. $\mathcal{F}$ randomly chooses $I \in [1, q_i]$, $Y \in [1, q_y]$, and $J \in [1, q_s]$. $\mathcal{F}$ chooses a random number $\text{sk}_{\text{NCC}} \in_R Z_q^*$ and computes the corresponding public-key $\text{pk}_{\text{NCC}} = \text{sk}_{\text{NCC}} \cdot G$. Then $\mathcal{F}$ sets the system parameters $Para = \{F_p, E/F_p, G, h_1, h_2, \text{pk}_{\text{NCC}}\}$ and sends it to adversary $\mathcal{A}$. $\mathcal{F}$ answers $\mathcal{A}$'s queries as follows.
2) *h_1(ID$_i$ $\parallel$ pk$_i^k$)-Query:* $\mathcal{F}$ keeps a list, L_{h_1}, with the form of $(\text{ID}_i, \text{pk}_i^k, h_{1,i} = \text{TID}_i^k)$. List L_{h_1} is initially empty. Upon receiving $\mathcal{A}$'s query with the message, $(\text{ID}_i, \text{pk}_i^k)$, $\mathcal{F}$ checks whether the tuple, $(\text{ID}_i, \text{pk}_i^k)$, is already in list L_{h_1}. If the tuple is in the list, $\mathcal{F}$ returns $h_{1,i}$ as corresponding. Otherwise, $\mathcal{F}$ randomly chooses $h_{1,i} \in \{0, 1\}^n$ and inserts tuple $(\text{ID}_i, \text{pk}_i^k, h_{1,i})$ into list L_{h_1}, and responses with $h_1(\text{ID}_i \parallel \text{pk}_i^k) = h_{1,i}$.
3) *h_2^1(TID$_i^k$ $\parallel$ P_i^k(= pk$_i^k$ $\parallel$ Randomstring))-Query:* $\mathcal{F}$ keeps a list, $L_{h_2^1}$, with the form of $(\text{TID}_i^k, P_i^k (= pk_i^k \parallel Randomstring), h_{2,i}^1)$. List $L_{h_2^1}$ is initially empty. Upon receiving $\mathcal{A}$'s query with message (TID_i^k, P_i^k), $\mathcal{F}$ checks whether tuple (TID_i^k, P_i^k) is already in list $L_{h_2^1}$. If the tuple is in the list, $\mathcal{F}$ returns the corresponding $h_{2,i}^1$. Otherwise, $\mathcal{F}$ randomly chooses $h_{2,i}^1 \in \mathbb{Z}_p$ and inserts tuple $(\text{TID}_i^k, P_i^k, h_{2,i}^1)$ into list $L_{h_2^1}$. Meanwhile, $\mathcal{F}$ returns it to $\mathcal{A}$.
4) *h_2^2(TS_i $\parallel$ R_i)-Query:* $\mathcal{F}$ keeps a list, $L_{h_2^2}$, with the form of $(TS_i, R_i, h_{2,i}^2, Z_i^j)$, where $Z_i^j = r_z \cdot R_i$. List $L_{h_2^2}$ is initially empty. Upon receiving $\mathcal{A}$'s query h_2^2-query, $\mathcal{F}$ checks whether tuple (TS_i, R_i) is already in list $L_{h_2^1}$. If the tuple is in the list, $\mathcal{F}$ returns the corresponding $h_{2,i}^2$. Otherwise, $\mathcal{F}$ randomly chooses $h_{2,i}^2 \in \mathbb{Z}_p$ and inserts tuple $(TS_i, R_i, h_{2,i}^2)$ into list $L_{h_2^2}$, and responses with $h_2^2(TS_i \parallel R_i) = h_{2,i}^2$.
5) *h_2^3(TS_y $\parallel$ R_y $\parallel$ R_z)-Query:* $\mathcal{F}$ keeps a list, $L_{h_2^2}$, with the form of $(TS_y, R_y, R_G, Z_j, h_{2,y}^3, Z_y^j)$, where $Z_y^j = r_i \cdot R_G$. List $L_{h_2^3}$ is initially empty. Upon receiving $\mathcal{A}$'s query with message (TS_y, R_y, R_G), $\mathcal{F}$ checks whether tuple (TS_y, R_y, R_G) is already in the list $L_{h_2^3}$. If the tuple is in the list, $\mathcal{F}$ returns the corresponding $h_{2,y}^3$. Otherwise, $\mathcal{F}$ randomly chooses $h_{2,y}^3 \in \mathbb{Z}_p$ and inserts tuple $(TS_y, R_y, R_G, h_{2,y}^3)$ into list $L_{h_2^3}$, and responses with $h_2^3(TS_y \parallel R_y \parallel R_z) = h_{2,y}^3$.
6) *Extract-Query:* $\mathcal{F}$ maintains an initially empty list, L_{Ex}, consisting of tuples of the form $(\text{ID}_i, \text{TID}_i^k, q_i^k, \text{sk}_i^k, \text{pk}_i^k)$. When adversary $\mathcal{A}$ queries the oracle with $(\text{ID}_i, \text{TID}_i^k)$, $\mathcal{F}$ searches list L_{Ex} by index $(\text{ID}_i, \text{TID}_i^k)$.

If $i = I$, $\mathcal{F}$ responds with $(\mathrm{ID}_I, \mathrm{TID}_I^k, q_I^k, \perp, \mathrm{pk}_I^k)$. Otherwise, if $i \neq I$, $\mathcal{F}$ chooses random number $h_{2,i}^1, \mathrm{sk}_i^k \in_R Z_q^*$, and computes the corresponding public-key $\mathrm{pk}_i^k = \mathrm{sk}_i^k \cdot G - h_{2,i}^1 \cdot \mathrm{pk}_{\mathrm{NCC}}$. Finally, $\mathcal{F}$ inserts tuples $(\mathrm{ID}_i, \mathrm{pk}_i^k, \mathrm{TID}_i^k)$, $(\mathrm{TID}_i^k, P_i^k (= \mathrm{pk}_i^k \parallel Randomstring), h_{2,i}^1)$, and $(\mathrm{TID}_i^k, \mathrm{ID}_i, q_i^k, \mathrm{sk}_i^k, \mathrm{pk}_i^k)$ into list L_{h_1}, $L_{h_2^1}$, and L_{Ex}, respectively.

7) *Corrupt-Query:* Upon receiving this query, if $i = I$, $\mathcal{F}$ aborts the current session. If $i \neq I$, $\mathcal{F}$ searches list L_{Ex} for ID_i, if ID_i is on L_{Ex}, $\mathcal{F}$ responses with sk_i^k; otherwise, $\mathcal{F}$ queries Extract-query, and computes sk_i^k and pk_i^k.
8) *Send-Query:*
   a) when adversary $\mathcal{A}$ makes a Send$(\prod_i^j$, "start") query, $\mathcal{F}$ selects uniformly random number $v_i, h_{2,i}^2 \in Z_q^*$, and computes $R_i = v_i \cdot G - h_{2,i}^2 \cdot pk_i^k$. $\mathcal{F}$ inserts tuples $(\prod_i^j, TS_i, \mathrm{pk}_i^k, R_i, v_i)$ and $(TS_i, R_i, h_{2,i}^2)$ into lists L_{Se} and $L_{h_{2,i}^2}$, respectively. $\mathcal{F}$ returns $(\prod_i^j, TS_i, pk_i^k, R_i, v_i)$;
   b) when adversary $\mathcal{A}$ makes a Send$(\prod_y^j, M_1)$ query, $\mathcal{F}$ verifies whether the timestamp is fresh. If not, $\mathcal{F}$ quits this session. Otherwise, $\mathcal{F}$ selects uniformly random number $v_y, h_{2,y}^2 \in Z_q^*$, and computes $R_y = v_y \cdot G - h_{2,y}^2 \cdot pk_y^k$. $\mathcal{F}$ inserts tuples $(\prod_y^j, TS_y, pk_y^k, R_y, R_z, v_y)$ and $(TS_y, R_y, R_z, h_{2,y}^2)$ into lists L_{Se} and $L_{h_{2,y}^2}$, respectively. Then $\mathcal{F}$ returns $(\prod_y^j, TS_y, \mathrm{pk}_y^k, R_y, R_z, v_y)$;
   c) when adversary $\mathcal{A}$ makes a Send$(\prod_i^j, M_2)$ query, $\mathcal{F}$ verifies whether the timestamp is fresh. If not, $\mathcal{F}$ quits the current session. Otherwise, $\mathcal{F}$ checks whether the tuple $(\prod_i^j, TS_i, \mathrm{pk}_i^k, R_i, v_i)$ is in the list L_{Se}: if not, $\mathcal{F}$ records session $(\prod_i^j, TS_i, \mathrm{pk}_i^k, R_i, R_y, R_z, v_i, v_y)$ in list L_{Se} and returns it.
9) *Reveal-Query:* $\mathcal{F}$ maintains an initially empty list L_{Re} consisting of a tuple of the form $(\prod_{i,y}^j, \mathrm{ID}_i, \mathrm{ID}_y, R_i, R_y, R_z, SK_{i,z}^j)$, where symbol $\prod_{i,y}^j$ represents jth session between entities i and y. When receiving this query, $\mathcal{F}$ answers as follows.
   a) If $j = J \wedge i = I \wedge y = Y$, $\mathcal{F}$ quits the game.
   b) Else if $i \neq I$, $\mathcal{F}$ looks up lists L_{Se}, $L_{h_2^2}$, and $L_{h_2^3}$ by the index (R_i, R_y, R_z) from list L_{Re} for obtaining session (Z_i^j, Z_y^j). Then $\mathcal{F}$ checks if $e(R_i, R_z) = e(G, Z_i^j)$ or $e(R_i, R_z) = e(G, Z_y^j)$, $\mathcal{F}$ selects Z_i^j or Z_y^j as the session key, $SK_{i,z}^j$, and returns it.
   c) Otherwise, $\mathcal{F}$ returns random value $SK_{i,z}^j$, and records it in list L_{Re}.
10) *Test-Query:* If $\mathcal{F}$ does not choose one of the oracle $\prod_{I,Y}^J$ to ask the Test-query, $\mathcal{F}$ quits the game. Otherwise, $\mathcal{F}$ simply outputs a random value $\xi \in \{0, 1\}^k$. ■

Analysis: If $\mathcal{F}$ is not aborted ($\mathcal{F}$ did not make corrupt-query or reveal-query queries), the probability that $\mathcal{F}$ chooses $\prod_{I,Y}^J$ as the test-query oracle is $[1/(q_i q_y q_s)]$. If $\mathcal{F}$ can win in such a game, then must $\mathcal{F}$ have made the corresponding h_2^2 and h_2^3 queries of the form $(TS_i, R_i, h_2^2, Z_i^j)$ and $(TS_y, R_y, R_G, Z_j, h_{2,y}^3, Z_y^j)$, respectively. h_2^2 and h_2^3 are random oracles, and thus $\mathcal{F}$ can find the corresponding item in $L_{h_2^2}$ or $L_{h_2^3}$ with the probability $[1/(q_{h_2^2} q_{h_2^3})]$ and output Z_i^j or Z_y^j as a solution to the ECDHP. The probability that $\mathcal{F}$ solves the ECDHP is: $\mathrm{Adv}^{\mathrm{ECDH}}(\mathcal{F}) \geq [(\lambda(k))/(q_i q_y q_s q_{h_2^2} q_{h_2^3})]$.

B. Other Discussions

1) Mutual Authentication: In the proposed authentication scheme, mutual authentication between SAP and the user is achieved. From step A2 of the authentication and key agreement phase described in Section V, SAP can verify the legitimacy of the MU by checking whether equation $v_j \cdot G = R_j + h_2(TS_j \parallel R_j) \cdot \mathrm{pk}_j^i + h_2(TS_j \parallel R_j) \cdot h_2(\mathrm{TID}_j^i \parallel P_j^i) \cdot \mathrm{pk}_{\mathrm{NCC}}$ holds. Due to the difficulty of the ECDLP and the properties of hash function, adversaries cannot get the knowledge of private key sk_j^i from public values. Thus, without knowing of private key sk_j^i, it is infeasible to forge a valid access request message. Therefore, only the legitimate user who owns the private key, sk_j^i, can generate the correct tuples $\{TS_j^1, \mathrm{ID}_{\mathrm{SAP}}, R_j, P_j^i, v_j\}$ to hold the authentication equality. Similarly, user can verify the legitimacy of satellite in step A3 by checking whether $v_{\mathrm{SAP}} \cdot G$ is equal to $R_{\mathrm{SAP}} + h_2(TS_{\mathrm{SAP}}^2 \parallel R_{\mathrm{SAP}}) \cdot \mathrm{pk}_{\mathrm{SAP}} + h_2(TS_{\mathrm{SAP}}^2 \parallel R_{\mathrm{SAP}}) \cdot h_2(\mathrm{ID}_{\mathrm{SAP}} \parallel P_{\mathrm{SAP}}) \cdot \mathrm{pk}_{\mathrm{NCC}}$ for the fact that only a legitimate SAP who has private key sk_{SAP} can generate the valid response, v_{SAP}. Therefore, our scheme could provide secure mutual authentication between SAP and MU_j.

2) Key Forward/Backward Secrecy: To provide key forward secrecy (KFS) and key backward secrecy (KBS) between MU_j and SIN, our protocol uses ECDH. In the proposed scheme, the session key is obtained as $SK_{j-G} = r_j \cdot R_G = r_G \cdot R_j = r_j \cdot r_G \cdot G$, where r_j and r_G are generated randomly. The adversary must derive r_j or r_G from R_j and R_G, respectively, if he/she wants to gain the session key, which is equivalent to the hardness of solving ECDLP. Thus, the proposed authentication protocol could provide security session key derivation. Obviously, while generating the current session key, our protocol uses $R_j = r_j \cdot G$ and $R_G = r_G \cdot G$ that are not related to the previous session key. Therefore, even with the disclosure of the current session key, attacker still cannot guess out the previously established session keys. Moreover, the compromise of current key will not reveal future session key either. Thus, the proposed scheme can achieve KFS/KBS, and thus guarantees session key security.

3) User Anonymity and Unlinkability: To ensure user privacy, the proposed scheme needs anonymity. This means that the real identity of a user ID_j should be protected for confidentiality. In our scheme, since the user's identity carried in all transmitted messages is expressed as a temporary identity, TID_j^i, instead of the real identity, ID_j, the user's real identity is not revealed in insecure wireless links. Also, the adversary cannot extract real identity ID_j from temporary identity TID_j^i due to the one-way property of hash function. Therefore, our scheme could provide user anonymity. However, privacy means not only protecting the real identity of a user, ID_j, from disclosing to adversaries, but also hiding the linkage from the transactions of the same unknown user [14]. In our scheme, the MU randomly picks an unused P_j^i that contains a new TID_j^i when generating access request message. Consequently, the

MU would not be identified by the malicious user/satellaite, neither be linked from its two transactions through all transmitted messages. In conclusion, it is clear that user anonymity and unlinkability requirements can be met in the proposed authentication scheme.

4) Accountability: Once the misbehavior of an MU, MU_j, is detected, the gateway station sends P_j^i and the evidence of malicious behaviors to NCC. In our scheme, we adopt ElGamal encryption technique to encrypt TID_j^i and ID_j during the registration phase, so only NCC has ability to obtain the user's real identity from P_j^i. Therefore, NCC can track the malicious user's real identity when a dispute occurs.

5) Resistance to Replay Attacks: The proposed scheme can withstand the replay attacks by introducing timestamp value. It is noted that the access request message contains a timestamp value TS_j^1, which is hashed to get $v_j = r_j + h_2(TS_j^1 \parallel R_j)sk_j^i$. If an adversary intercepts the message transmitted in the authentication procedure and replays it, the SAP could find this type of attacks by checking the validity of the timestamp value. What is more, since the one-way and collision-resistant properties of hash function, an adversary cannot fake the access request message by modifying the timestamp value in a new authentication procedure. Likewise, the MU could detect the replayed response message because the response message also contains a timestamp value TS_{SAP}^2 that is hashed to get $v_{\mathrm{SAP}} = r_{\mathrm{SAP}} + h_2(TS_{\mathrm{SAP}}^2 \parallel R_{\mathrm{SAP}}) \cdot \mathrm{sk}_{\mathrm{SAP}}$. Therefore, our protocol can resistant to replay attacks.

6) Resistance to Impersonation Attacks: Suppose adversary $\mathcal{B}$ intends to masquerade as legitimate user $\mathcal{A}$ to compromise the SIN user access authentication procedure of our scheme. For the reason, the adversary must forge an authentication request, which should satisfy this equation $v_{U_B} \cdot G = R_{U_B} + h_2(TS_{U_A} \parallel R_{U_B}) \cdot \mathrm{pk}_{U_A}^i + h_2(TS_{U_A} \parallel R_{U_B}) \cdot h_2(\mathrm{TID}_{U_A}^i \parallel P_{U_A}^i) \cdot \mathrm{pk}_{\mathrm{NCC}}$, where v_{U_B} is generated by himself/herself and the message $TS_{U_A}, P_{U_A}^i$ is obtained through eavesdropping; if it holds, $\mathcal{B}$ will succeed in cheating the SIN that he/she is the legal $\mathcal{A}$. However, $\mathcal{B}$ cannot compute a valid v_{U_B} without knowing the corresponding private key of $\mathcal{A}$. Let us consider the following impersonation attack method.

Assume that adversary $\mathcal{B}$ is a legal user in SIN, so he/she can use the private key $\mathrm{sk}_{U_B}^i$ to generate $v_{U_B} = r_{U_B} + h_2(TS_{U_A} \parallel R_{U_B}) \cdot \mathrm{sk}_{U_B}^i$. $\mathcal{B}$ needs to build $R'_{U_B} = R_{U_B} + \Delta$ for being authenticated successfully as another $\mathcal{A}$, where Δ is computed as follows: $\Delta = h_2(TS_{U_A} \parallel R_{U_B} + \Delta) \cdot (\mathrm{pk}_{U_B}^i - \mathrm{pk}_{U_A}^i) + h_2(TS_{U_A} \parallel R_{U_B} + \Delta) \cdot \mathrm{pk}_{\mathrm{NCC}} \cdot \{h_2(\mathrm{TID}_{U_B}^i \parallel P_{U_B}^i) - h_2(\mathrm{TID}_{U_A}^i \parallel P_{U_A}^i)\}$. However, it is computationally infeasible to build valid Δ due to the one-way and collision-resistant property of hash function. Therefore, our proposed scheme is secure to impersonation attacks.

7) Resistance to Man-in-the-Middle Attacks: A man-in-the-middle attacker tries to trick two parties into a three-party communication, which means that an attacker needs to have ability to impersonate a legitimate user to communicate with SAP and a legitimate SAP to communicate with a legitimate user simultaneously. However, from the proof of resistance to impersonation attacks, a man-in-the-middle attacker would fail to impersonate others. Therefore, our scheme would not be exposed to man-in-the-middle attacks.

```
role userA (A,S,NCC : agent,H : hash_func,SEND, RECV: channel(dy) )
played_by A
def=
local State: nat,
W: hash_func,
Ra, Rs, RAi, RS, RG,Va, Vs, T1, T2, SK : text,
IDa, IDs, IDncc, TIDa, PAi, PS, LTAi, LTS,  G, Qqai, Qqs, Dncc, SKai, SKs : text
const a_s_ra, s_a_rs, a_s_t1, s_a_t2, s1, s2, s3,s4: protocol_id
init State:= 0
transition
% Session key agreement phase
1. State=0 /\ RECV(start) =|>
% Send < M1 > to user S
State':= 2
/\ Qqai':= new()  /\ PAi':= W(Qqai'.G).
H(IDncc.IDa.W(Qqai'.G)).{H(IDncc.IDa.W(Qqai'.G)).IDa}_(Dncc).IDncc.LTAi
/\ Ra':= new() /\ RAi' := W(Ra'.G) /\ T1' := new()
/\ Va' := W(Ra'.W(H(T1'.W(Ra'.G)).SKai))
/\ secret({Dncc, Qqai'},s1, NCC)/\ secret({Ra',SKai,IDa}, s2, A) %/\ secret(SKs, s3, S)
/\ SEND(PAi'.RAi'.Va'.IDs.T1') /\ witness(A, S, a_s_ra, Ra')/\ witness(A, S, a_s_t1, T1')
% RECV < M2 > from user S
2. State =2
/\ RECV(TIDa.PS.RG.W(Rs'.G).W(Rs'.W(H(T2'.RS').SKs)).T2') =|>
% Send < M3 > to user S
State':= 4
/\ request(S, A, s_a_rs, Rs')/\ request(S, A, s_a_t2, T2')
end role
```

Fig. 6. Role specification for user in HLPSL.

VII. Simulation for Formal Security Analysis Using AVISPA Tool

In this section, we simulate the proposed protocol for the formal security verification using the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool [40]. AVISPA is a push-button tool for the automated validation of Internet security-sensitive protocols and applications. It provides a modular and expressive formal language for specifying protocols and their security properties, and integrates different back-ends that implement a variety of state-of-the-art automatic analysis techniques [41]. AVISPA measures whether a security protocol is SAFE or UNSAFE by looking for attacks on specified scenarios. In AVISPA, protocols are specified in high level protocols specification language (HLPSL). HLPSL is a role-oriented language in which each role is independent from other role and communicates with other roles through channels. In addition, AVISPA integrates four different back-ends, namely on-the-fly model-checker (OFMC), constraint-logic-based attack searcher (CL-AtSe), SAT-based model-checker (SATMC), and tree automata based on automatic approximation for the analysis of security protocols (TA4SP), in which the more details for how AVISPA works can be found in [40]. The intruder in AVISPA is modeled using the Dolev–Yao [42] model, which is a formal model used to prove properties of interactive cryptographic protocols. In this model, the intruder, in addition to having all the capabilities of an honest user such that all messages sent by the user will go to the intruder, may intercept, analyze, and synthesize messages, and send the new ones to impersonate other users during the protocol execution.

We implement the proposed protocol using HLPSL in the AVISPA tool to examine its security properties. In Figs. 6 and 7, we have implemented the role for user A and SAP S in HLPSL language, respectively. Meanwhile, in Fig. 8, we have implemented the role for the session, goal, and environment in HLPSL language. The simulation result presented in Fig. 9 shows that no attacks are found and the

```
role userS (A, S, NCC : agent, H: hash_func, SEND, RECV: channel(dy) )
played_by S
def=
local State: nat,
W: hash_func,
Ra, Rs, Va, Vs, T1, T2, SK : text,
IDa, IDs, IDncc, TIDa, PAi, PS, LTAi, LTS, RAi, RS, RG, G, Qqai, Qqs, Dncc, SKai, SKs :
text
const a_s_ra, s_a_rs, a_s_t1, s_a_t2, s1, s2, s3, s4: protocol_id
init State := 1
transition
% Receive < M1 > from user A
1.State = 1 /\ RECV(PAi'.W(Ra'.G).W(Ra'.W(H(T1'.RAi').SKai)).IDs.T1') =|>
% Send < M2 > to user A
State' := 3
/\ Qqs' := new()
/\ PS' := W(Qqs'.G).
H(IDncc.IDs.W(Qqs'.G)).{H(IDncc.IDs.W(Qqs'.G)).IDs}_(Dncc).IDncc.LTS
/\ Rs' := new() /\ RS' := W(Rs'.G)/\ T2' := new()
/\ Vs' := W(Rs'.W(H(T2'.RS').SKs))
/\ secret(Qqs', s4, NCC)/\ secret({Ra',SKai,IDa}, s2, A) /\ secret({Rs',SKs},s3, S)
/\ SEND(TIDa.RS'.RG.PS'.Vs'.T2')
/\ witness(S, A, s_a_rs, Rs')/\ witness(S, A, s_a_t2, T2')
end role
```

Fig. 7. Role specification for satellite in HLPSL.

```
role session(A, S, NCC : agent, H :
hash_func )
def=
local SN1, SN2, RV1, RV2 : channel(dy)
composition
userA (A, S, NCC, H, SN1, RV1)
/\ userS (A, S, NCC, H, SN2, RV2)
end role
```

```
role environment()
def=
const a, s, ncc : agent, h, w :hash_func,
idncc: text, a_s_ra, s_a_rs,a_s_t1, s_a_t2,s1,
s2, s3,s4: protocol_id
intruder_knowledge ={a, s, ncc, idncc, h, w}
composition
session(a, s, ncc, h)
/\ session(i, s, ncc, h)/\ session(a, i, ncc, h)
end role
goal
secrecy_of s1, s2, s3,s4
authentication_on a_s_ra, s_a_rs, a_s_t1,
s_a_t2
end goal
environment()
```

Fig. 8. Role specification for the session, goal, and environment in HLPSL.

security goals are achieved under the OFMC and CL-AtSe backends. From the OFMC analysis of the proposed protocol in Fig. 9(left subgraph), it is observed that the protocol is safe and runs with a bounded number of sessions with the number of visited nodes being 4. From the CL-AtSe results shown in Fig. 9(right subgraph), we can see that the proposed scheme executes a bounded number of sessions with three analyzed states and one reachable state.

VIII. Performance Analysis

In this section, we first analyze the performance of our scheme by comparing it with the existing authentication schemes, including Liu *et al.*'s [17] scheme, Chang *et al.*'s [4] scheme, and Lee *et al.*'s [28] scheme in terms of signaling overhead and authentication delay. Then, we analyze the advantage of the batch authentication and give the simulation results.

A. Signaling Overhead

On signaling overhead, we evaluate our scheme by comparing with the schemes of [4], [17], and [28] in terms of the number of signaling messages. Table II lists the comparison of different authentication schemes in terms of signaling overhead. In our scheme, MU and SAP need two signaling

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS

BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/
role_in_SIN.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.00s
  visitedNodes: 4 nodes
  depth: 2 plies
```

```
SUMMARY
  SAFE
DETAILS

BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
PROTOCOL
  /home/span/span/testsuite/results/
role_in_SIN.if
GOAL
  As Specified
BACKEND
  CL-AtSe
STATISTICS

  Analysed  : 3 states
  Reachable  : 1 states
  Translation: 0.00 seconds
  Computation: 0.00 seconds
```

Fig. 9. Simulation results of the proposed scheme using OFMC and CL-AtSe backends.

TABLE II
Comparison of Signaling Overhead

	User$\leftrightarrow SAP$	$SAP \leftrightarrow$ G	G$\leftrightarrow NCC$
Chang *et al.*'s [4]	2	2	2
Liu *et al.*'s [17]	2	2	2
Lee *et al.*'s [28]	2	2	2
Our scheme	2	1	0

messages to complete mutual authentication, just like other schemes. However, there is only one signaling message needed between SAP and G, and no signaling message is needed between gateway and NCC in the proposed scheme. It means that our scheme has a great advantage compared with other schemes since these schemes all need to exchange at least two signaling messages between SAP and G, and two signaling messages between G and NCC. Therefore, our scheme has better performance on signaling overhead than all existing schemes. Furthermore, our scheme can diminish the burden of NCC because it authenticates the legitimacy of an MU without the participation of NCC in each authentication process.

B. Authentication Delay

On authentication delay, we define it as the total time costs for the whole authentication process, including the time costs of computations and propagation delay. For the convenience of evaluating the computational cost, we define some notations as follows.

1) TG_{mul}: The time of executing one point multiplication over an elliptic curve.
2) TG_{add}: The time of executing one point addition over an elliptic curve.
3) TG_h: The time of executing a one-way hash function.

Since the speed of verification is mainly dominated by three operations, namely point multiplication, point addition, and hash operation, we only consider these three operations and neglect other operations such as additive. Here, we adopt the experiment in [43] for an MNT curve of embedding degree $k = 6$ and 160-bit q. The implementation was executed on an Intel P IV 3.0-GHz processor, and the following experiment

TABLE III
COMPARISON OF AUTHENTICATION DELAY

	Computational cost (ms)	Transmission delay (ms)	Authentication delay (ms)
Chang *et al.*'s [4]	$10TG_h$ (≈ 0.001)	$4T_{U-SAP} + 2T_{G-NCC}$ (≈ 60)	60.001
Liu *et al.*'s [17]	$9TG_h$ (≈ 0.0009)	$4T_{U-SAP} + 2T_{G-NCC}$ (≈ 60)	60.0009
Lee *et al.*'s [28]	$10TG_h$ (≈ 0.001)	$4T_{U-SAP} + 2T_{G-NCC}$ (≈ 60)	60.001
Our scheme	$9TG_{mul} + 6TG_h + 4TG_{add}$ (≈ 5.4046)	$2T_{U-SAP}$ (≈ 20)	25.4046

TABLE IV
VERIFICATION COST

	Computational cost of SAP
A single request	$3TG_{mul} + 2TG_h + 2TG_{add}$
n requests (without batch verification)	$3nTG_{mul} + 2nTG_h + 2nTG_{add}$
n requests (with batch verification)	$(n+2)TG_{mul} + 2nTG_h + 2nTG_{add}$

results are obtained: TG_{add} is 0.001 ms, TG_{mul} is 0.6 ms, and TG_h is 0.0001 ms. In this paper, we denote the time costs of signal propagation between the user and SAP, SAP and G, and G and NCC as $T_{U-\text{SAP}}$, $T_{\text{SAP}-G}$, $T_{G-\text{NCC}}$, respectively. Since the LEOs are 500–2000 km away from the ground [44], it is reasonable to set $T_{U-\text{SAP}} = T_{\text{SAP}-G} = 10$ ms. After we have performed enough experiments, and reference to many examples in Internet, we found that the time required to ping from wired or wireless PC to Google server generally rang from 20 to 50 ms. Thus, to facilitate the comparison of authentication delay, we assume that the propagation delay between G and NCC, $T_{G-\text{NCC}}$, is 10 ms.

In Table III, we demonstrate the comparisons among our protocol and the previously proposed access authentication schemes [4], [17], [28] in terms of computation cost, propagation delay, and authentication latency. From Table III, we can see that the computational cost of the proposed scheme is larger than that of other schemes since our scheme introduces expensive elliptic curve cryptography operations. In particular, the computational cost of our scheme is 5.4046 ms and the cost of others is less than 0.001 ms. However, our scheme can reduce propagation delay for a full authentication process compared with existing schemes due to fewer signaling messages needed between G and SAP/NCC. Thus, the authentication latency can be reduced drastically. As shown in Table III, we can see that the transmission delay of other schemes is 60 ms, while ours is only 20 ms. In summary, a successful authentication in the proposed protocol requires 25.4046 ms, while the authentication latency of the other schemes is generally more than 60 ms. Therefore, our proposed scheme is more suitable for SIN to provide MUs with better access service.

C. Evaluation of Batch Verification

When an SAP receives a large number of access requests from multiple users at the same time, batch authentication can be performed to significantly decrease the computational overhead of the satellite. Table IV, respectively, shows the computational cost of handling a single authentication, n authentication without batch verification, and n authentication with batch verification. As illustrated in Fig. 10, where the abscissa represents the number of access request ranging from 1 to 100, and the ordinate indicates the corresponding computational cost with/without batch verification. The result illustrates that the computational cost can be remarkably reduced through the operation of batch verification when n users forward their access authentication requests simultaneously.

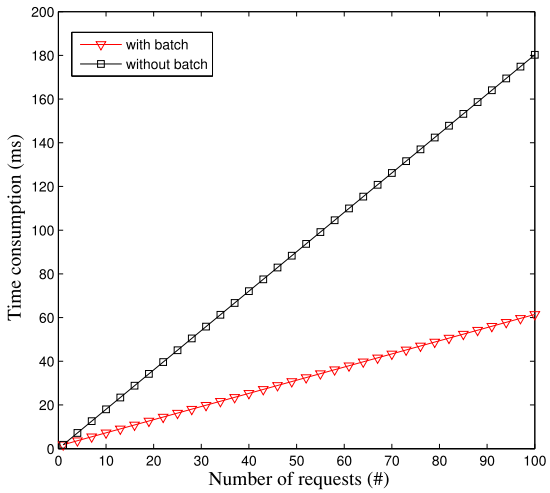


Fig. 10. Computational cost with/without batch verification.

IX. CONCLUSION

The security issues of SIN have gained a lot of attention with the development of satellite communication technology. As an important security issue, authentication protocol should be well studied in SIN system. In this paper, we outlined the security requirements of access authentication protocol for IoT in SIN and proposed a secure and fast access authentication protocol with an efficient handover mechanism based on elliptic curve cryptography. The proposed protocol provides some essential security properties and reduces user access latency to improve QoS of communications in SIN. Meanwhile, the support of batch verification in the scheme makes the handover more efficient when involving a large number of users.

The formal and informal security analyses demonstrate that our protocol is secure against various attacks. Moreover, the performance analysis shows that the proposed protocol is efficient compared with existing protocols in terms of signaling overhead and authentication delay, and is suitable for constructing a secure space communication system.

ACKNOWLEDGMENT

The authors would like to thank the anonymous referees for their invaluable suggestions that have led to the present improved version of this paper's original manuscript.

REFERENCES

[1] J. A. Stankovic, "Research directions for the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[3] J. Mukherjee and B. Ramamurthy, "Communication technologies and architectures for space network and interplanetary Internet," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 881–897, 2nd Quart., 2013.

[4] C.-C. Chang, T.-F. Cheng, and H.-L. Wu, "An authentication and key agreement protocol for satellite communications," *Int. J. Commun. Syst.*, vol. 27, no. 10, pp. 1994–2006, 2014.

[5] M. De Sanctis, E. Cianca, G. Araniti, I. Bisio, and R. Prasad, "Satellite communications supporting Internet of Remote Things," *IEEE Internet Things J.*, vol. 3, no. 1, pp. 113–123, Feb. 2016.

[6] Y. Qian, L. Ma, and X. Liang, "Symmetry chirp spread spectrum modulation used in LEO satellite Internet of Things," *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2230–2233, Nov. 2018.

[7] M.-S. Hwang, C.-C. Yang, and C.-Y. Shiu, "An authentication scheme for mobile satellite communication systems," *ACM Oper. Syst. Rev.*, vol. 37, no. 4, pp. 42–47, 2003.

[8] G. Zheng, P.-D. Arapoglou, and B. Ottersten, "Physical layer security in multibeam satellite systems," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 852–863, Feb. 2012.

[9] J. Lei, Z. Han, M. Á. Vazquez-Castro, and A. Hjorungnes, "Secure satellite communication systems design with individual secrecy rate constraints," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 661–671, Sep. 2011.

[10] D. Li, J. Liu, and W. Liu, "Secure and anonymous data transmission system for cluster organised space information network," in *Proc. IEEE Int. Conf. Smart Cloud (SmartCloud)*, 2016, pp. 228–233.

[11] M. Qiu, Z. Ming, J. Li, K. Gai, and Z. Zong, "Phase-change memory optimization for green cloud with genetic algorithm," *IEEE Trans. Comput.*, vol. 64, no. 12, pp. 3528–3540, Dec. 2015.

[12] D. He, C. Chen, S. Chan, and J. Bu, "Secure and efficient handover authentication based on bilinear pairing functions," *IEEE Trans. Wireless Commun.*, vol. 11, no. 1, pp. 48–53, Jan. 2012.

[13] K. Xue, C. Ma, P. Hong, and R. Ding, "A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 36, no. 1, pp. 316–323, 2013.

[14] D. He, S. Chan, and M. Guizani, "An accountable, privacy-preserving, and efficient authentication framework for wireless access networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1605–1614, Mar. 2016.

[15] C. Lai, H. Li, R. Lu, and X. S. Shen, "SE-AKA: A secure and efficient group authentication and key agreement protocol for LTE networks," *Comput. Netw.*, vol. 57, no. 17, pp. 3492–3510, 2013.

[16] C. Jiang, X. Wang, J. Wang, H.-H. Chen, and Y. Ren, "Security in space information networks," *IEEE Commun. Mag.*, vol. 53, no. 8, pp. 82–88, Aug. 2015.

[17] Y. Liu, A. Zhang, J. Li, and J. Wu, "An anonymous distributed key management system based on CL-PKC for space information network," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2016, pp. 1–7.

[18] Y. Zhang, J. Chen, and B. Huang, "An improved authentication scheme for mobile satellite communication systems," *Int. J. Satellite Commun. Netw.*, vol. 33, no. 2, pp. 135–146, 2015.

[19] W. Zhao, A. Zhang, J. Li, X. Wu, and Y. Liu, "Analysis and design of an authentication protocol for space information network," in *Proc. IEEE Military Commun. Conf. (MILCOM)*, 2016, pp. 43–48.

[20] H. S. Cruickshank, "A security system for satellite networks," in *Proc. IET Int. Conf. Satellite Syst. Mobile Commun. Navig.*, 1996, pp. 187–190.

[21] D. Hu, L. He, and J. Wu, "A novel forward-link multiplexed scheme in satellite-based Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1265–1274, Apr. 2018.

[22] A. Meloni and L. Atzori, "The role of satellite communications in the smart grid," *IEEE Wireless Commun.*, vol. 24, no. 2, pp. 50–56, Apr. 2017.

[23] Y.-F. Chang and C.-C. Chang, "An efficient authentication protocol for mobile satellite communication systems," *ACM SIGOPS Oper. Syst. Rev.*, vol. 39, no. 1, pp. 70–84, 2005.

[24] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. IT-22, no. 6, pp. 644–654, Nov. 1976.

[25] T.-H. Chen, W.-B. Lee, and H.-B. Chen, "A self-verification authentication mechanism for mobile satellite communication systems," *Comput. Elect. Eng.*, vol. 35, no. 1, pp. 41–48, 2009.

[26] E.-J. Yoon *et al.*, "An efficient and secure anonymous authentication scheme for mobile satellite communication systems," *EURASIP J. Wireless Commun. Netw.*, vol. 2011, no. 1, pp. 86–95, 2011.

[27] I. Lasc, R. Dojen, and T. Coffey, "Countering jamming attacks against an authentication and key agreement protocol for mobile satellite communications," *Comput. Elect. Eng.*, vol. 37, no. 2, pp. 160–168, 2011.

[28] C.-C. Lee, C.-T. Li, and R.-X. Chang, "A simple and efficient authentication scheme for mobile satellite communication systems," *Int. J. Satellite Commun. Netw.*, vol. 30, no. 1, pp. 29–38, 2012.

[29] G. Zheng, H.-T. Ma, C. Cheng, and Y.-C. Tu, "Design and logical analysis on the access authentication scheme for satellite mobile communication networks," *IET Inf. Security*, vol. 6, no. 1, pp. 6–13, Mar. 2012.

[30] M. S. Farash and M. A. Attari, "An efficient client–client password-based authentication scheme with provable security," *J. Supercomput.*, vol. 70, no. 2, pp. 1002–1022, 2014.

[31] X. Lin, X. Sun, P.-H. Ho, and X. Shen, "GSIS: A secure and privacy-preserving protocol for vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3442–3456, Nov. 2007.

[32] J. K. Liu *et al.*, "Time-bound anonymous authentication for roaming networks," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 1, pp. 178–189, Jan. 2015.

[33] Q. Yang *et al.*, "AnFRA: Anonymous and fast roaming authentication for space information network," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 2, pp. 486–497, Feb. 2019.

[34] N. Koblitz, "Elliptic curve cryptosystems," *Math. Comput.*, vol. 48, no. 177, pp. 203–209, 1987.

[35] V. S. Miller, "Use of elliptic curves in cryptography," in *Proc. Conf. Theory Appl. Cryptograph. Techn.*, 1985, pp. 417–426.

[36] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inf. Theory*, vol. IT-31, no. 4, pp. 469–472, Jul. 1985.

[37] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-hashing for message authentication," Internet Eng. Task Force, Fremont, CA, USA, Rep. RFC 2104, 1997.

[38] T. Dierks and E. Rescorla, "The transport layer security (TLS) protocol version 1.2," Internet Eng. Task Force, Fremont, CA, USA, RFC 5246, 2008.

[39] J.-L. Huang, L.-Y. Yeh, and H.-Y. Chien, "ABAKA: An anonymous batch authenticated and key agreement scheme for value-added services in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 1, pp. 248–262, Jan. 2011.

[40] A. Armando, D. Basin, J. Cuellar, M. Rusinowitch, and L. Viganò, "AVISPA: Automated validation of Internet security protocols and applications," in *Proc. ERCIM News*, vol. 64, 2006, pp. 66–67.

[41] P. Kumar, A. Braeken, A. Gurtov, J. Iinatti, and P. H. Ha, "Anonymous secure framework in connected smart home environments," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 4, pp. 968–979, Apr. 2017.

[42] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.

[43] M. Scott. *Efficient Implementation of Cryptographic Pairings*. Accessed: Mar. 14, 2019. [Online]. Available: http://ecrypt-ss07.isg.rhul.ac.uk/Slides/Thursday/mscott-samos07.pdf

[44] C. E. Fossa, R. A. Raines, G. H. Gunsch, and M. A. Temple, "An overview of the IRIDIUM (R) low earth orbit (LEO) satellite system," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON)*, 1998, pp. 152–159.

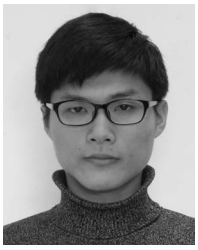
Kaiping Xue (M'09–SM'15) received the B.S. degree from the Department of Information Security, University of Science and Technology of China (USTC), Hefei, China, in 2003, and the Ph.D. degree from the Department of Electronic Engineering and Information Science, USTC, in 2007.

From 2012 to 2013, he was a Post-Doctoral Researcher with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA. He is currently an Associate Professor with the Department of Information Security and the Department of Electrical Engineering and Information Science, USTC. His current research interests include next-generation Internet, distributed networks, and network security.

Wei Meng received the B.S. degree from the Department of Information Security, School of Telecommunications Engineering, Xidian University, Xi'an, China, in 2016. She is currently pursuing the graduate degree in communication and information system at the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, China.

Her current research interest includes network security protocol design and analysis.

Ms. Meng was a recipient of the Best Paper Runner-Up Award of IEEE MASS 2018.



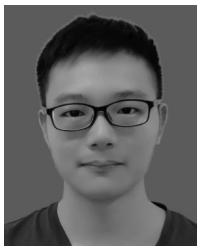
Shaohua Li received the B.S. degree from the Department of Information Security, University of Science and Technology of China (USTC), Hefei, China, in 2016, where he is currently pursuing the graduate degree in communication and information system at the Department of Electronic Engineering and Information Science.

His current research interests include network security and system security.

David S. L. Wei (SM'07) received the Ph.D. degree in computer and information science from the University of Pennsylvania, Philadelphia, PA, USA, in 1991.

He is currently a Full Professor with the Computer and Information Science Department, Fordham University, Bronx, NY, USA. From 1993 to 1997, he was as an Associate Professor and then a Full Professor with the Faculty of Computer Science and Engineering, University of Aizu, Aizuwakamatsu, Japan. He has authored of co-authored over 120 technical papers in various archival journals and conference proceedings. His current research interests include cloud and edge computing, Internet of Things, big data, machine learning, and cognitive radio networks.

Dr. Wei was a Lead Guest Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS and the IEEE TRANSACTIONS ON CLOUD COMPUTING for several special issues. He has also served as an Associate Editor for the IEEE TRANSACTIONS ON CLOUD COMPUTING from 2014 to 2018 and the *Journal of Circuits, Systems and Computers* from 2013 to 2018. He is currently an Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS for the Series on Network Softwarization and Enablers, a Lead Guest Editor of the IEEE TRANSACTIONS ON BIG DATA for the "Special Issue on Edge Analytics in the Internet of Things," and a Guest Editor of the IEEE TRANSACTIONS ON BIG DATA for the "Special Issue on Trustworthiness in Big Data and Cloud Computing Systems." He was the Chair of the Intelligent Transportation Forum of Globecom 2010, the General Chair of the Intelligent Transportation Workshop of ICC 2011, and the Cloud Security Forum and Intelligent Transportation Forum of Globecom 2011.



Huancheng Zhou received the B.S. degree from the Department of Information Security, University of Science and Technology of China, Hefei, China, in 2017, where he is currently pursuing the graduate degree in communication and information system at the Department of Electronic Engineering and Information Science.

His current research interest includes network security protocol design and analysis.

Nenghai Yu received the B.S. degree from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 1987, the M.E. degree from Tsinghua University, Beijing, China, in 1992, and the Ph.D. degree from the University of Science and Technology of China (USTC), Hefei, China, in 2004.

Since 1992, he has been a member of the faculty of the Department of Electronic Engineering and Information Science, USTC, where he is currently a Professor, the Executive Director of the Department of Electronic Engineering and Information Science, and the Director of the Information Processing Center. He has authored or co-authored over 130 papers in journals and international conferences. His current research interests include multimedia security, multimedia information retrieval, video processing, and information hiding.