2018 IEEE 15th International Conference on Mobile Ad-hoc and Sensor Systems

Low-Latency Authentication against Satellite Compromising for Space Information Network

Wei Meng, Kaiping Xue*, Jie Xu, Jianan Hong, Nenghai Yu
The Department of EEIS, University of Science and Technology of China, Hefei, Anhui 230027 China
*Corresponding author, kpxue@ustc.edu.cn

***Abstract*—With an advancement of mobile communication technology, the space information network (SIN) has been proposed to meet the increasing demands of mobile communication due to its advantage of providing great expanding access services. In SIN, authentication is significant for the security to prevent the network resource from unauthorized access. However, the features of highly exposed links and extremely high propagation delay make it difficult to design a secure and fast authentication scheme for SIN. Although some existing researches have tried to design authentication protocols for SIN, they haven't taken the intolerable authentication delay and the risk of satellite compromising into consideration. Faced with these problems, we design a proxy signature-based authentication scheme for SIN, in which, the interaction process of authentication can be only implemented between the mobile user and the satellite node, thus reducing the long authentication implementation delay. Furthermore, we utilize the proxy signature to mitigate the risk of satellites being attacked. The results of security and performance analysis show that the proposed scheme can provide the required security and largely reduce the authentication latency.**

***Index Terms*—Space information network, access authentication, handover, satellite compromising, proxy signature.**

I. Introduction

The high demands for mobile communications and the emergence of new wireless multimedia applications have accelerated the development of wireless access technologies in recent years. Under such situation, space information network (SIN) has been proposed, in which the artificial earth satellites are deployed as relay stations for transmitting radio waves to achieve a wider range of communications. It overcomes the shortage of geographic limitation in other traditional wireless networks (e.g., LTE-A networks, WiFi) to make communication more convenient [1]. Users will be more willing to access SIN to obtain network services due to its features of extensive applicability, great expanding access services, etc [2]. For providing secure network services, it is critical for SIN to deploy a secure access authentication protocol that makes resources to be protected and be accessible only by legitimate users.

It is of great necessity to design a secure and efficient access authentication protocol for SIN due to its special structural features compared with the other wireless networks. However, the features of the extremely long propagation delay, restricted computation power, etc., bring many technical challenges to the design of authentication mechanisms [3, 4]. In detail, the propagation delay between satellites and the ground is generally large (even for Low Earth Orbit (LEO) satellite, there is 10 to 40ms propagation delay because of the 500 to 2,000 kilometers transmission distance [5]). This long propagation delay will greatly enlarge the access latency if implementing the traditional authentication schemes, and further brings negative influence on the QoS of SIN access services. Furthermore, the limited computation and storage capacity of the satellites makes it unsuitable to implement algorithms with high complexity [6]. Besides, similar to other wireless access networks, highly exposed links in SIN make it vulnerable for adversaries to launch various malicious attacks (e.g., replay attacks and impersonation attacks) [7–9]. Moreover, the unintended disclosure of sensitive information from unwary and inexperienced users often occurs due to ease of wireless signal interception [10]. What's worse, satellite hijacking attacks as a unique type of attack on SIN may lead to serious consequences, even threaten social stability and national security.

Quite a lot of access authentication protocols have been proposed for traditional networks, and some of them have been standardized (e.g., EPS AKA in LTE-A networks). However, these schemes are not suitable to be implemented in SIN since they will bring some intolerable problems, such as the frequent signaling interaction and the heavy computation overhead. Therefore, a secure and fast access authentication scheme for SIN is required to ensure the security and QoS of the access. Until now, only a few researches have worked on the access authentication schemes for mobile users in SIN. In 1996, Cruickshank [11] first presented an authentication scheme based on PKC (public key cryptosystem) to achieve mutual authentication between the mobile user and the satellite network. Nevertheless, the involved public-key cryptographic operations are quite complex. Thus, Hwang *et al.* [7] proposed another authentication scheme based on SKC (secret-key cryptosystem). Unfortunately, it is still computationally inefficient and insecure against stolen-verifier attacks. Subsequently, in 2005, Chang *et al.* [12] proposed a hash-chain-based authentication scheme to enhance the computation efficiency and security properties, but it is vulnerable to impersonation attacks and users privacy can hardly be guaranteed. Then in 2009, a self-verification authentication protocol (CLC) based on PKC and SKC was first raised by Chen *et al.* [13]. Lee *et al.* [14] pointed out that Chen *et al.*'s scheme was insecure because the mobile users' secret keys and server's private key can be disclosed from verification table parameters through Euclidean algorithms, and then proposed an improved authentication

2155-6814/18/$31.00 ©2018 IEEE
DOI 10.1109/MASS.2018.00045

scheme. However, their protocol was found to be vulnerable to replay attacks and smart card loss attacks. Thus, in [9], the authors further proposed an enhanced authentication scheme based on [14], but it is proved to be vulnerable to the stolen-verifier attacks and requires higher storage capability [15].

In these existing schemes, authentication is implemented between mobile users and the ground facility (e.g., gateway, NCC), whereas, the satellite only simply forwards the authentication signals, rather than participating in the practical authentication session. Consequently, an authentication protocol at least needs four signal transmission delay between the earth and the satellite (back and forth between user/satellite and NCC/satellite, respectively), which will result in unacceptable access delay. Obviously, none of the existing schemes take the long propagation delay of SIN into consideration. And all of them ignore the situation that the satellite may be insecure. These schemes all suppose that the satellite is fully trusted to transmit signalling between users and the ground facilities. Moreover, in most existing schemes, the NCC is a bottleneck of performance and security since it needs to participate in every authentication session. Hence, the NCC is a potential vulnerability in the SIN communication system.

For the above mentioned issues, in this paper, we propose a proxy signature-based authentication scheme to achieve the security and high-quality communications of SIN. In our scheme, each satellite with certain computing power can execute access control instead of the ground facilities, which largely reduces the network accessing delay and the computation load of the NCC. The traditional authentication schemes will fail in SIN if the satellites are compromised by malicious entities since the implementation of these schemes relies on the satellites to forward authentication interaction messages honestly. Our scheme limits the satellite's authorization privilege and enables the legitimate satellite to securely obtain a temporal proxy delegation from the ground station to authenticate the validity of users' access requests. Besides, in our scheme, the ground station can manage and monitor the security state of the satellite more flexibly by setting the validity period of the proxy delegation. The main contributions of this work are listed as follows:

1) We give a new authentication system model to reduce the access latency, based on which, a secure and efficient access authentication scheme for SIN is proposed to enable the authentication process to be implemented securely between users and satellites.
2) By introducing the proxy signature, the risk of satellite being attacked will be mitigated to enhance security properties of the proposed protocol. The performance and security analysis demonstrate that our scheme is efficient and indeed enforces its security guarantees.

The rest of this paper is organized as follows: we first present the system and security model in Section II. After demonstrating some necessary preliminaries in Section III, we describe the proposed scheme in details in Section IV. The security analysis and evaluation are presented in Section V. Then, we provide the performance analysis in SectionVI. Finally, the paper is concluded in Section VII.

II. System and Security Model

A. System Model

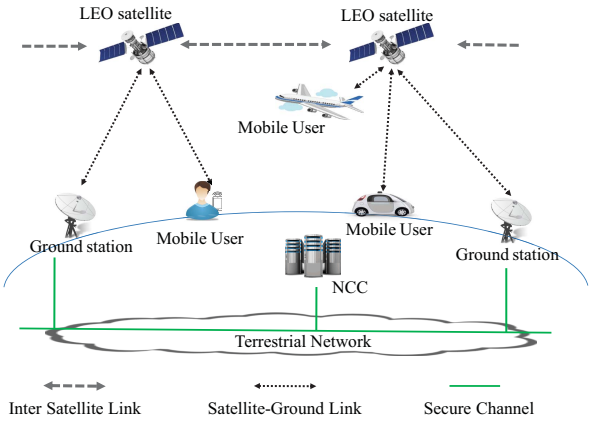


Fig. 1. System model

The increasing service demand for mobile communication makes it necessary for SIN to provide network access service for authorized users. The system model of access service in SIN is illustrated in Fig. 1, which shows the distribution and interrelationship of the entities in the space information network architecture. The system model in our scheme consists of the Low Earth Orbit satellites (LEO), the ground station/gateway (G), the Network Control Center (NCC) and mobile users (U).

- ***LEO*** is the access point for users to access the network. With the satellite manufacturing technology advancement, nowadays LEO satellites can have certain computing capacities to execute some complex functions. Therefore, in this paper, in addition to forwarding messages between U and G, LEO also needs to authenticate the legitimacy of the U.
- ***NCC*** is the management in the network domain. It is responsible for registration of network entities and user's authorization, and it communicates with ground stations via secure channels.
- ***G*** locates in the ground can provide users with an interface to the terrestrial network. It wirelessly communicates with users through LEO satellites and connects to the NCC via secure wired links.
- ***Users*** denote the legal users of the system, which need to register in the NCC for getting authentication information tuples. Then they utilize these authentication information tuples to connect to LEO for accessing network resources.

B. Security Model

We assume that the NCC is trustworthy for all entities in our system. It is infeasible for any adversary to compromise the NCC. There is a secure channel established between

the network entity and the NCC to protect the registration process. In addition to the above assumptions, we assume that a polynomial time adversary, who has the ability to modify, inject or interrupt the interaction messages over the air, and tries to corrupt the proposed authentication scheme to enjoy services for free. Each LEO may suffer from attacking, which would lead to the unsuccessful secure channel establishment between the user and the G. And the G is assumed to have the capability to monitor the security status of corresponding LEOs.

III. Preliminary

A. Proxy Signature

In this paper, we utilize the proxy signature scheme for partial delegation with warrant [16] to design the authentication scheme for SIN. The concept of proxy signature has firstly been introduced by Mambo *et al*. [17] in 1996, which allows an original signer to delegate his/her capability to proxy signer so that the proxy signer can sign on behalf of the original signer. A proxy signature scheme is generally specified by the following algorithms:

- *Setup*: The parameters generation algorithm takes the security parameter as input, and returns system parameters. This algorithm is performed by the key generation center (KGC), which is assumed as the trusted and secure party.
- *DGeneration*: The delegation-generation algorithm takes the original signer's secret key and a *warrant* as input, and outputs the delegation.
- *DVerify*: The delegation-verification algorithm takes the delegation as input and verifies whether the received delegation is a valid and legitimate one come from the original signer. The result of verification ("YES" or "NO") will be returned as the output.

The *warrant* mentioned in the *DGeneration* algorithm is an explicit description of the relative rights and information of the original signer and proxy signer. It usually contains the proxy information about the proxy signer, such as the identity of the proxy signer, the validity period of the delegation, and signature capability of the proxy signer such that a verifier can use it as a part of verification information.

B. Elliptic Curve Cryptography (ECC)

ECC is more efficient than RSA algorithm in terms of key size and computation overhead [18]. In an ECC system, the elliptic curve equation can be defined as the form of $E_p(a,b)$: $y^2 = x^3 + ax + b \ (mod\ p)$ over a prime finite F_p, where $a, b \in F_p$, and $4a^3 + 27b^2 \neq 0 \ (mod\ p)$. The security of ECC rests on the difficulty of the elliptic curve discrete logarithm problem (ECDLP): given two points P_1 and P_2 over the elliptic curve $E_p(a,b)$, it is hard within polynomial time to find an integer $x \in F_p^*$ that satisfies $P_1 = xP_2$. For more details, we refer the interested readers to [19].

IV. The Proposed Scheme

In this paper, we construct our authentication scheme based on the proxy signature introduced in [20]. The entire scheme consists of four phases: system initialization phase, registration phase, proxy delegation phase and user authentication phase.

A. System Initialization Phase

In the system initialization phase, the NCC can be regarded as a key generation center (KGC), and produces the system parameters and the system private key. The following steps are carried out by NCC:

1) Choose a k-bit prime p and determine an elliptic curve over a finite field: E/F_p. The points on E/F_p together with an infinity point O form a cyclic additive group $\mathbb{G}$ with the order q.
2) Choose a generator P of the group $\mathbb{G}$ and a master private key $x \in_R \mathbb{Z}_q^*$ then compute $P_{pub} = x \cdot P$.
3) Choose cryptographic secure hash functions $H : \{0,1\}^* \rightarrow Z_q^*$ and $h : \{0,1\}^* \rightarrow \{0,1\}^n$.
4) Publish the system parameter $Par = \{\mathbb{G}, P, P_{pub}, H, h\}$ and keep the private key x securely.

B. Registration Phase

1) Function Entity Registration: The entity j (i.e., G and LEO) in SIN system implements the following steps to register in the NCC:

(1) The entity j sends it own identity ID_j to the NCC in a secure manner.
(2) The NCC picks a random $r_j \in Z_q^*$, computes $K_j = r_j \cdot P$ and $\sigma_j = xH(K_j, ID_j) + r_j$. Then NCC sends $\{K_j, \sigma_j\}$ to the entity j securely.
(3) The entity j can validate the received message $\{K_j, \sigma_j\}$ by verifying the following equation:

$$\sigma_j \cdot P \stackrel{?}{=} H(K_j, ID_j) \cdot P_{pub} + K_j. \quad (1)$$

If the equation holds, the entity j stores the σ_j as its private key secretly, otherwise rejects.

2) User Registration: Before the user U_i wants to enjoy satellite access services, he/she should register with the NCC to be an authorized user. The NCC performs the following steps:

(1) The user U_i sends it real identity ID_{U_i} to the NCC in a secure manner.
(2) After receiving the user's real identity ID_{U_i}, the NCC first generates a temporary identity $TID_{U_i} = h(K_{U_i}, x) \oplus ID_{U_i}$ and generates a proxy warrant w_{U_i} for user U_i, where w_{U_i} is the proxy information of U_i including U_i's access permission on SIN, the expiration date of delegation, etc. And it can be used to validate the effectiveness of the proxy delegation. Then the NCC picks a random $r_{U_i} \in Z_q^*$, and computes $K_{U_i} = r_{U_i} \cdot P$ and $\sigma_{U_i} = xH(K_{U_i}, TID_{U_i}, w_{U_i}) + r_{U_i}$. Finally the NCC sends $\{K_{U_i}, \sigma_{U_i}, TID_{U_i}, w_{U_i}\}$ to U_i securely.
(3) The entity U_i can validate the received message $\{K_{U_i}, \sigma_{U_i}, TID_{U_i}, w_{U_i}\}$ by checking the following equation:

$$\sigma_{U_i} \cdot P \stackrel{?}{=} H(K_{U_i}, TID_{U_i}, w_{U_i}) \cdot P_{pub} + K_{U_i}. \quad (2)$$

If the equation holds, U_i computes his/her proxy private key as $psk_{U_i} = \sigma_{U_i}$ and stores it secretly, otherwise rejects.

C. Proxy Delegation Phase

In this phase, the ground station (e.g., G_s) will produce a temporal proxy delegation for the satellite (e.g., L_e) when it authenticates L_e successfully. For simplicity, in this paper, we mainly study the access authentication on the user side. And the concrete authentication process between the ground station and the satellite is beyond our scope. Therefore, we assume that G_s has successfully validated L_e when performing the proxy delegation phase. Then the G_s sends a temporary proxy delegation pair to L_e. The proxy delegation phase is presented as follows:

1) *DGeneration*: To prevent L_e from abusing the proxy delegation, G_s first needs to choose a proxy warrant $w_{G_sL_e}$ for L_e which can be used to validate the effectiveness of the proxy delegation. $w_{G_sL_e}$ contains the security status of the satellite L_e and the validity period of the temporal proxy delegation. The validity period can be set to different values to achieve different monitoring granularity for L_e's security status (e.g., $w_{G_sL_e}$ can be set to a smaller value to achieve a finer granularity). Then G_s picks random values $a_s, r_{G_sL_e} \in Z_q^*$, and computes:

$$\begin{aligned} &R_{G_s} = a_s \cdot P, \\ &K_{G_sL_e} = r_{G_sL_e} \cdot P, \\ &\sigma_{G_sL_e} = \sigma_{G_s} H(K_{G_sL_e}, ID_{L_e}, w_{G_sL_e}, R_{G_s}) + r_{G_sL_e}. \end{aligned} \tag{3}$$

Finally, G_s sends the delegation message $M_{G_sL_e} = ID_{G_s}||K_{G_s}||K_{G_sL_e}||\sigma_{G_sL_e}||w_{G_sL_e}||R_{G_s}$ to L_e securely.

2) *DVerify*: L_e validates the received delegation by checking whether the following equation holds:

$$\begin{aligned} &\sigma_{G_sL_e} \cdot P \stackrel{?}{=} K_{G_sL_e} + H(K_{G_sL_e}, ID_{L_e}, w_{G_sL_e}, R_{G_s}) \\ &(H(K_{G_s}, ID_{G_s}) \cdot P_{pub} + K_{G_s}). \end{aligned} \tag{4}$$

If the equation does not hold, L_e rejects the received message. Otherwise, L_e computes its proxy signing key $psk_{G_sL_e} = \sigma_{G_sL_e}$, and then stores the proxy delegation pair $(K_{G_sL_e}, \sigma_{G_sL_e}, w_{G_sL_e})$ and the corresponding key negotiation parameter R_{G_s}.

After the successful completion of this phase, L_e obtains a temporal proxy delegation from the ground station. The temporal proxy delegation allows L_e to authenticate the validity of U_i's access request and to assist G_s in establishing secure channels with U_i.

D. User Authentication Phase

In this phase, the mutual authentication and key agreement protocol will be executed between a mobile user (e.g., U_i), a LEO (e.g., L_e) and a G (e.g., G_s) for secure communication when L_e is within U_i's communication range. The detailed steps are given as follows.

1) When U_i wants to access SIN for service, he/she will generate the access request message and send it to a LEO (e.g., L_e). The details in access request generation are illustrated in **Algorithm 1**.

Algorithm 1: Access Request Generation

Input: generator P, U_i's signing key psk_{U_i};
Output: Access request $M_{U_i}||s_{U_i}$;
Select a random number b;
Compute $R_{U_i} = b_{U_i} \cdot P$;
Generate timestamp ts_1;
Set $M_1 = R_{U_i}||ts_1$;
Set $M_{U_i} = M_1||TID_{U_i}||K_{U_i}||w_{U_i}$;
Compute $s_{U_i} = psk_{U_i} - H(M_1)b$;
Return Access Request $M_{U_i}||s_{U_i}$;

2) Upon receiving the access request, L_e verifies the authentication signature s_{U_i} and generates the access response message by implementing **Algorithm 2**. According to w_{U_i}, L_e verifies whether U_i's access permission on SIN is legitimate and checks whether the proxy delegation is overdue. If illegitimate and overdue, L_e will reject the access request as well as terminating the authentication procedure. And U_i can renew the proxy delegation from the NCC if he/she wants to continue enjoying satellite access services. Otherwise, L_e verifies whether the authentication signature s_{U_i} is valid. If the verification fails, the access request will be rejected; otherwise, L_e generates the access response message and the details as shown in **Algorithm 2**. Finally, L_e sends the access response message to U_i and G_s in parallel for reducing some transmission delay.
3) Upon receiving the access response message from L_e, U_i and G_s respectively implement **Algorithm 3** to establish a secure channel between them.

After the authentication procedure is completed, G_s and U_i establish a sharing key sk with each other. Therefore, they can derive the authentication key AK_i and encryption key EK_i by applying the key derivation function (KDF) which takes sk as its master key. Then G_s sends AK_i to L_e securely. The parameters AK_i and EK_i are used to authenticate and encrypt the transmission data respectively in subsequent communications.

E. Extension Mechanism for Handover Authentication

For the handover scenario shown in Fig. 2, where the communication link between the user and the satellite remains unchanged, but the ground station connected to the satellite will change to another one. Noting that when L_e switches from the current ground station G_s to the new ground station G_s^*, the mutual authentication process between L_e and U_i doesn't need to be performed repeatedly as the trust relationship between U_i and G_s has been established. However, the proxy delegation process between L_e and G_s^* is necessary to perform in order to provide access service for new users in the future. Besides, U_i needs to establish a secure channel with G_s^*.

Algorithm 2: Access Response Generation

Input: Access Request $M_{U_i}||s_{U_i}$, delegation message $M_{G_sL_e}$, L_e's proxy signing key $psk_{G_sL_e}$;
Output: False or Access response $M_{L_e}||s_{L_e}$;

Check whether w_{U_i} and ts_1 are valid;
if *both* w_{U_i} *and* ts_1 *are invalid* **then**
  Reject the access request;
  return False;
else
  Compute
  $X_1 = s_{U_i} \cdot P$,
  $X_2 = H(R_{U_i}, ts_1) \cdot R_{U_i}$,
  $X_3 = H(K_{U_i}, TID_{U_i}, w_{U_i}) \cdot P_{pub}$;
  if $X_1 + X_2 \neq K_{U_i} + X_3$ **then**
    Reject the access request;
    return False;
  else
    Select a random number c_{L_e};
    Compute $R_{L_e} = c_{L_e} \cdot P$;
    Generate timestamp ts_2;
    Set $M_2 = R_{L_e}||R_{G_s}||ts_2$;
    Set $M_{L_e} = M_2||ID_{L_e}||ID_{G_s}||K_{G_sL_e}||K_{G_s}||w_{G_sL_e}$;
    Compute $s_{L_e} = psk_{G_sL_e} - H(M_2)c_{L_e}$;
    return Access Response $M_{L_e}||s_{L_e}$;
  end
end

Algorithm 3: Secure Channel Establishment

Input: Access Response $M_{L_e}||s_{L_e}$, generator P;
Output: sk;

Check whether $w_{G_sL_e}$ and ts_2 are valid;
if *both* $w_{G_sL_e}$ *and* ts_2 *are invalid* **then**
  Drop the access response message;
  return False;
else
  Compute
  $X_4 = s_{L_e} \cdot P$,
  $X_5 = H(R_{L_e}, R_{G_s}, ts_2) \cdot R_{L_e}$,
  $X_6 = H(K_{G_sL_e}, ID_{L_e}, w_{G_sL_e})$,
  $X_7 = K_{G_s} + H(K_{G_s}, ID_{G_s}) \cdot P_{pub}$;
  if $X_4 + X_5 \neq X_6 \cdot X_7$ **then**
    Drop the access response message;
    return False;
  else
    Set $sk = b_{U_i} \cdot R_{G_s}$ or $sk = a_s \cdot R_{U_i}$;
    return sk;
  end
end

1) *The proxy delegation process*: The proxy delegation process will be implemented between L_e and G_s^* when the coverage of L_e is just enough to cover both G_s and G_s^* but hasn't completely switched from the current ground station G_s to the new ground station G_s^*. Details can refer to the "*Proxy Delegation Phase*" in Section IV-C.
2) *The secure channel establishment*: L_e utilizes the proxy delegation message $M^*_{G_s^*L_e}$ and its proxy signing key $psk^*_{G_s^*L_e}$ obtained from G_s^* to generate the message $\{M^*_{L_e}||s^*_{L_e}\}$ and the details can refer to **Algorithm 2** in subsection IV-D. Then L_e sends $\{M^*_{L_e}||s^*_{L_e}\}$ to U_i.

Finally, U_i can build a new sharing key $sk^* = a_s^* \cdot R_{U_i} = b_{U_i} \cdot R_s^*$ with G_s^* by referring to **Algorithm 3** in Section IV-D.

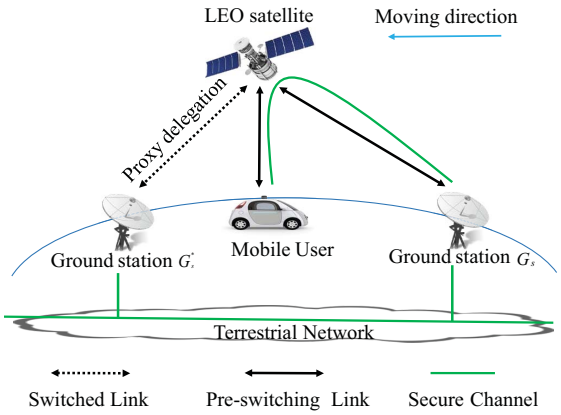


Fig. 2. Handover Scenario

V. Security Analysis and Evaluation

A. Security Analysis

Here, we give the security analysis of our proposed scheme.

1) Unforgeability: According to the proposed scheme mentioned above, only the user who holds a valid proxy delegation from the NCC can be authenticated by the LEO successfully, and then enjoys the network services of SIN. Hence, it motivates the unauthorized users to forge proxy delegations in order to access SIN legitimately. However, it is impossible for an unauthorized user to forge a proxy delegation as each valid proxy delegation is generated by using the private key x of the NCC in our proposed scheme. Therefore, forging proxy delegation is infeasible as long as x has not been leaked out.

2) Mutual Authentication: In the proposed scheme, mutual authentication between the LEO and user can be achieved. Since the NCC has uniquely chosen or generated the proxy delegation parameters for a given user and forging proxy delegation is infeasible according to the aforementioned security analysis of unforgeability, only the user holding a specific proxy private key can create a specific authentication signature s_{U_i}. The LEO can further verify s_{U_i} by using the system parameter Par and the proxy delegation parameters, i.e., w_{U_i} and K_{U_i}. Therefore, the user can be authenticated by submitting his/her authentication signature signed by his/her proxy private key to the LEO. Similarly, the LEO authentication can also be achieved. If the LEO L_e have been successfully authenticated by the ground station G_s, it can obtain a valid temporal proxy delegation from G_s and accordingly it is authorized to assist G_s in establishing a secure channel with the user. Therefore, authenticating the LEO can be done by verifying if it generates a valid authentication signature.

3) Conditional Anonymity: The temporary identity TID_{U_i} of U_i is transmitted to the LEO in the access request message $\{M_{U_i}, s_{U_i}\}$. A malicious LEO or general adversary can extract

TID_{U_i} from the access request message, but cannot extract real identity ID_{U_i} since they do not know the private key x. The real identity can be revealed only by the NCC by computing $ID_{U_i} = h(K_{U_i}, x) \oplus TID_{U_i}$. Therefore, the proposed scheme can provide the user anonymity.

4) Replay Attack: An adversary can intercept the messages and try to replay them. However, he/she cannot be authenticated successfully when the timestamp value in the message is not in a reasonable time window. Moreover, the timestamp cannot be modified as it is hashed to get the authentication signature. Thus, the replaying message can be detected by verifying the validation of the timestamp and signature.

5) Impersonation Attack and Man-in-the-Middle (MitM) Attack: A MitM attacker cannot derive the session key sk by eavesdropping the public values from the wireless communication channel since the key agreement is based on the Computational Diffie-Hellman (CDH). Besides, the key agreement parameters i.e., R_{U_i} and R_{L_e} can be confirmed by the authentication signature s_{U_i} and s_{L_e} respectively. For example, the MitM attacker tampers R_{L_e} to be $\tilde{R}_{L_e}$ and wants user to accept it. However, the user will refuse $\tilde{R}_{L_e}$, since the authentication signature s_{L_e} cannot be successfully verified. Furthermore, it is infeasible for the attacker to create a correct signature on $\tilde{R}_{L_e}$ without the proxy private key of L_e. And it is also infeasible for an adversary to launch the impersonation attack through modifying the public values.

6) Satellite Compromising Attack: The proposed scheme can mitigate the risk of satellite being compromised. In our scheme, the ground station G_s will produce the temporal proxy delegation to the LEO L_e when ensuring that L_e is legitimate. Thus, only L_e with a valid delegate pair can provide network access service and assist G_e to achieve key agreement with the user successfully. Once G_s detects that L_e behaves abnormally, it suspends issuing the updated delegate pair to L_e to make L_e unavailable. In addition, G_s can change the monitoring granularity by setting the value of $w_{G_sL_e}$. Therefore, benefiting from the proxy signature technology, our proposed scheme can reduce the impact of satellite hijacking attacks.

B. Simulation for Formal Security Verification Using AVISPA Tool

We simulate the proposed scheme for the formal security verification using AVISPA (Automated Validation of Internet Security Protocols and Applications) tool [21]. The AVISPA is a widely accepted formal security verification tool which provides a suit of applications for building and analysing formal models of the security protocols. The tool measures whether a security protocol is SAFE or UNSAFE by looking for attacks on specified scenarios. In AVISPA, protocols are specified in HLPSL (High Level Protocols Specification Language). HLPSL is a role-oriented language in which each role is independent from other role and communicate with other roles through channels. In addition, AVISPA tool integrates four different back-ends, namely OFMC (On-the-fly Model-Checker), CL-AtSe (Constraint- Logic-based Attack Searcher), SATMC (SAT-based Model-Checker) and TA4SP (Tree Automata-based Protocol Analyzer), the more details can be found in [21]. We implement the proposed protocol using HLPSL in the AVISPA tool to examine its security properties. For simplicity, we only present one of the basic roles "user" in HLPSL as shown in Fig. 3. The simulation result presented in Fig. 4 clearly ensures that the proposed protocol is safe under the OFMC and CL-AtSe models, which indicates it is secure against some active and passive attacks.

```
role user (Ui, L: agent, H, Union, Pred : hash_func, SND, RCV : channel (dy))
played_by Ui
def = local State: nat, T1, T2, Wui, Wgl, Sigui,Sigl, Nb, Nc, Rui, Rl, Rg, Pskui,
Pskgl, P, Ppub, IDui, IDl , K_ Ui : text
const u_l_nb, l_u_nc, u_l_t1, l_u_t2, subs1, subs2: protocol_id
init  State := 0
transition
1. State= 0/\ RCV(start)
=|>
State' := 2/\ Nb' := new( )
/\ T1' := new( )
/\ secret( {Nb',Pskui},subs1,Ui)
/\ Rui' := Pred(Nb',P)
/\ Sigui' := Union(Pskui,Pred(Nb',H(Rui',T1')))
/\ SND(Sigui'.T1'.Rui'.Wui)
/\ witness(Ui,L,u_l_nb,Nb')
/\ witness(Ui,L,u_l_t1,T1')
2. State = 2/\ RCV(Union(Pskgl,Pred(Nc',H(Rl',Rg,T1'))).T2'.Rl'.Rg.Wgl)
=|>
State' := 4
/\ request(L,Ui,l_u_nc,Nc')
/\ request(L,Ui,l_u_t2,T2')
end role
```

Fig. 3. Role specification for user in HLPSL

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/
role_user.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.02s
  visitedNodes: 18 nodes
  depth: 4 plies
```

```
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
PROTOCOL
  /home/span/span/testsuite/results/
role_user.if
GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS
  Analysed  : 0 states
  Reachable  : 0 states
  Translation: 0.00 seconds
  Computation: 0.00 seconds
```

Fig. 4. Simulation results using OFMC and CL-AtSe back-end

VI. Performance Analysis

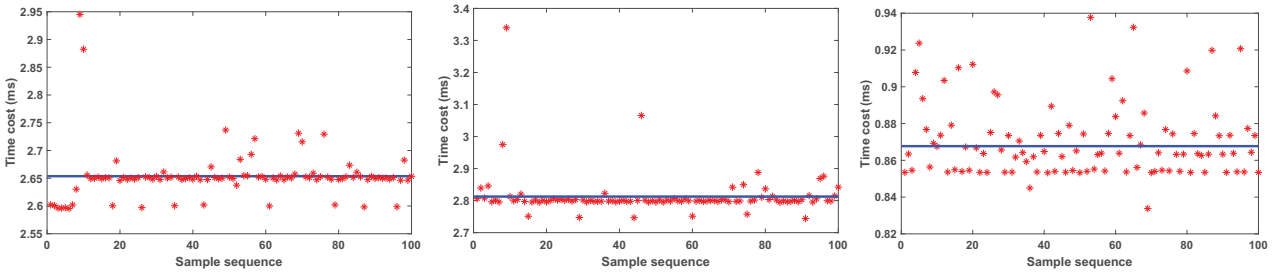
In this section, we analyze the performance of our scheme by comparing it with the existing authentication schemes. In addition, we implement the algorithm in our scheme to evaluate the algorithm runtime.

A. Performance Comparison

1) Signaling Overhead: On the signaling overhead, we evaluate our scheme by comparing with the existing schemes

TABLE I
PERFORMANCE COMPARISON

	Signing Cost				Propagation Delay	Computation Cost	Authentication Delay (approximation)
	U↔ L	L ↔ G	G↔ N	L↔ N			
Chang's[1]	2	2	2	-	$4T_{U\text{-}L} + 2T_{G\text{-}N}$	$10TG_h$	$4T_{U\text{-}L} + 2T_{G\text{-}N}$
Liu's[6]	2	2	2	-	$4T_{U\text{-}L} + 2T_{G\text{-}N}$	$9TG_h$	$4T_{U\text{-}L} + 2T_{G\text{-}N}$
Liu's[15]	2	-	-	2	$4T_{U\text{-}L}$	$9TG_h$	$4T_{U\text{-}L}$
Lee's[14]	2	2	2	-	$4T_{U\text{-}L} + 2T_{G\text{-}N}$	$10TG_h$	$4T_{U\text{-}L} + 2T_{G\text{-}N}$
Ours	2	1	0	-	$2T_{U\text{-}L}$	$10TG_{mul} + 7TG_h + 4TG_{add}$	$10TG_{mul} + 2T_{U\text{-}L} + 4TG_{add}$



(a) Access Request Generation Time (b) Access Response Generation Time (c) Secure Channel Establishment Time

Fig. 5. Time Cost for Different Algorithm

[1, 6, 14, 15] in terms of the number of signaling messages. TABLE I lists the comparison of signing cost for different schemes. As can be seen from the table, as the same as other 4 schemes, our scheme also needs two signaling messages between mobile user and the LEO. However, there is only one signaling message needed between the LEO and the G, and no signaling message needed between the G and the NCC in the proposed scheme. Therefore, our scheme has better performance on signaling overhead than other schemes.

2) Authentication Delay: We define the authentication delay as the total time costs for the whole authentication process, including the computation cost and the signal propagation delay. Although our scheme introduces extra computation cost during the proxy delegation phase, it enhances the security of the authentication system. Furthermore, this phase has been implemented before the user accesses SIN, so it is transparent to the user during the authentication process. Therefore, it is reasonable to ignore the computation cost of the proxy delegation phase when analyzing user authentication delay. In our evaluation, TG_{mul} and TG_{add} denote the time required to perform a point multiplication operation and a point addition operation respectively, and TG_h denotes the time required to perform an one-way hash operation. Besides, we denote the time costs of signal propagation between the user and LEO, LEO and G, G and NCC as $T_{U\text{-}L}$, $T_{L\text{-}G}$, $T_{G\text{-}N}$, respectively. The value of $T_{U\text{-}L}/T_{L\text{-}G}$ is considered not less than $10ms$ due to the fact that the LEOs are 500-2000 kilometers away from the ground. Obviously, the value of $T_{L\text{-}G}$ is equal to $T_{U\text{-}L}$, so we use $T_{U\text{-}L}$ to represent the propagation delay between the user and the LEO, and between the LEO and the G. And we ignore the hash function computations as it is light computations compared with the propagation delay when analyzing the authentication delay. The last column of the TABLE I demonstrates the comparisons of our protocol and other schemes in terms of authentication delay. From the table, it can be seen that the computational cost of the proposed scheme is larger than other schemes due to the expensive elliptic curve cryptography operations: 10 point multiplication operations, 4 point addition operations. However, our scheme only needs 2 signal propagation time, while other related schemes need 6, which makes the proposed scheme faster than others. Therefor, our proposed scheme is more suitable for SIN to provide mobile users with fast access authentication service.

B. Algorithm Implementation

We implement the algorithms in our scheme on the Banana Pi R1 with 1.2GHz CPU speed and 1GB RAM using C language with Pairing-Based Cryptography (PBC) library [22]. For the reliability of the experiment, we conducted the experiments 100 times, and in each execution, the algorithm in our scheme was run 100 times. Consequently, 100 experimental data are obtained and each one represents the time cost of a particular algorithm running 100 times. Then we can get 100 values from the experimental data, each of which representing the time cost of a single algorithm running. Further, the scatter plot is utilized to describe these 100 values. Finally, we can obtain three scatter plots for the three algorithms in our scheme i.e., the access request generation algorithm, the access response generation algorithm and the secure channel establishment algorithm, as shown in Fig. 5. The access request and response generation time is about $2.654ms$ and $2.813ms$ on average respectively. And the average time of the secure channel establishment is about $0.868ms$. Therefore, our proposed protocol is efficient in terms of computation cost and is feasible in practical implementation.

VII. Conclusion

In this paper, we have proposed a fast access authentication scheme for SIN based on a proxy signature algorithm. In the proposed scheme, we emphasize the authentication function of the LEO, which means the LEO can directly authenticate whether the mobile users can access SIN without the realtime involvement of the NCC. Moreover, in order to reduce the risk of satellite hijacking attacks, we use proxy signatures to ensure that only the secure and authorized satellites can obtain permissions from the gateway to authenticate users. Our security analysis and simulation show that our scheme satisfies a series of essential security features and achieves the security preservation in a more rigorous way. Meanwhile, the performance analysis indicates that the proposed scheme can largely reduce the authentication delay and signaling cost and is feasible in practical implementation.

Acknowledgment

This work is supported in part by the National Key Research and Development Program of China under Grant No. 2016YFB0800301, the National Natural Science Foundation of China under Grant No. 91538203, and Youth Innovation Promotion Association CAS under Grant No. 2016394.

References

[1] C.-C. Chang, T.-F. Cheng, and H.-L. Wu, "An authentication and key agreement protocol for satellite communications," *International Journal of Communication Systems*, vol. 27, no. 10, pp. 1994–2006, 2014.

[2] Y. Hu and V. O. Li, "Satellite-based Internet: a tutorial," *IEEE Communications Magazine*, vol. 39, no. 3, pp. 154–162, 2001.

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

[4] F. Li, L. Yang, W. Wu, L. Zhang, and Z. Shi, "Research status and development trends of security assurance for space-ground integration information network," *Journal on Communications*, vol. 37, no. 11, pp. 156–168, 2016.

[5] I. F. Akyildiz, H. Uzunalioğlu, and M. D. Bender, "Handover management in low earth orbit (LEO) satellite networks," *Mobile Networks and Applications*, vol. 4, no. 4, pp. 301–310, 1999.

[6] Y. Liu, A. Zhang, J. Li, and J. Wu, "An anonymous distributed key management system based on CL-PKC for space information network," in *Proceedings of the 2016 IEEE International Conference on Communications (ICC)*, pp. 1–7, IEEE, 2016.

[7] M.-S. Hwang, C.-C. Yang, and C.-Y. Shiu, "An authentication scheme for mobile satellite communication systems," *ACM SIGOPS Operating Systems Review*, 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 Transactions on Wireless Communications*, vol. 11, no. 2, pp. 852–863, 2012.

[9] Y. Zhang, J. Chen, and B. Huang, "An improved authentication scheme for mobile satellite communication systems," *International Journal of Satellite Communications and Networking*, vol. 33, no. 2, pp. 135–146, 2015.

[10] H. Lu, J. Li, and M. Guizani, "Secure and efficient data transmission for cluster-based wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 3, pp. 750–761, 2014.

[11] H. S. Cruickshank, "A security system for satellite networks," in *International Conference on Satellite Systems for Mobile Communications and Navigation*, pp. 187–190, IET, 1996.

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

[13] T.-H. Chen, W.-B. Lee, and H.-B. Chen, "A self-verification authentication mechanism for mobile satellite communication systems," *Computers & Electrical Engineering*, vol. 35, no. 1, pp. 41–48, 2009.

[14] C.-C. Lee, C.-T. Li, and R.-X. Chang, "A simple and efficient authentication scheme for mobile satellite communication systems," *International Journal of Satellite Communications and Networking*, vol. 30, no. 1, pp. 29–38, 2012.

[15] Y. Liu, A. Zhang, S. Li, J. Tang, and J. Li, "A lightweight authentication scheme based on self-updating strategy for space information network," *International Journal of Satellite Communications and Networking*, vol. 35, no. 3, pp. 231–248, 2017.

[16] S. Kim, S. Park, and D. Won, "Proxy signatures, revisited," in *International Conference on Information and Communications Security*, pp. 223–232, Springer, 1997.

[17] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures: Delegation of the power to sign messages," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 79, no. 9, pp. 1338–1354, 1996.

[18] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz, "Comparing elliptic curve cryptography and RSA on 8-bit CPUs," in *International workshop on cryptographic hardware and embedded systems*, pp. 119–132, Springer, 2004.

[19] D. Hankerson, A. J. Menezes, and S. Vanstone, *Guide to elliptic curve cryptography*. Springer Science & Business Media, 2006.

[20] C. Ma, K. Xue, and P. Hong, "A Proxy Signature Based Re-authentication Scheme for Secure Fast Handoff in Wireless Mesh Networks," *International Journal of Network Security*, vol. 15, no. 2, pp. 122–132, 2013.

[21] A. Armando, D. Basin, J. Cuellar, M. Rusinowitch, and L. Viganò, "AVISPA: Automated validation of internet security protocols and applications," *ERCIM News-Online Edition*, vol. 64, 2006.

[22] "The pairing-based cryptography (PBC) library," [Online]. Available: https://crypto.stanford.edu/pbc.