A Trust-Based Adaptive Authentication Scheme for Space-Air-Ground Integrated NOMA System

Mingxuan He*, Tingyu Cui*, Junyu Xiong*, Shangwei Zhang*†, and Huixiang Zhang*
*School of Cybersecurity, Northwestern Polytechnical University, Xi’an, Shaanxi, 710072, China
†E-mail: swzhang@nwpu.edu.cn

***Abstract*—The emerging cooperative non-orthogonal multiple access (C-NOMA) technique is expected to be integrated in future space-air-ground integrated networks (SAGIN) to improve network capacity and achieve global coverage. However, the limited resource, untrust NOMA relays and openness channels of SAGIN would inevitably encounter severe data security and privacy violation problems. Many secure access authentication schemes have been developed for SAGIN either by employing physical layer authentication methods for NOMA communication or by utilizing post quantum cryptographic mechanism for orthogonal multiple access networks. In this regard, we propose an adaptive access authentication scheme by considering both trust and untrust NOMA relays to resist quantum attack for C-NOMA communication in SAGIN based on the Number Theory Research Unit scheme. Our scheme can achieve high performance in terms of communication latency and network capacity while guarantee communication security for SAGIN NOMA systems.**

***Index Terms*—space-air-ground, non-orthogonal multiple access, access authentication**

I. Introduction

To fulfill the requirements of huge traffic and seamless connection in the coming Internet of Everything (IoE) era, future sixth generation (6G) wireless communication systems will integrate space, air and ground networks to achieve global coverage [1]. Besides, emerging non-orthogonal multiple access (NOMA) is expected to further improve the radio access network coverage and spectrum efficiency in future space-air-ground integrated networks (SAGIN) [2]. By merging the cooperative communication and NOMA techniques, cooperative NOMA (C-NOMA) was proposed in literature [3] to enhance the NOMA communication by employing the near user as a relay in each NOMA group. Such a working way can obtain a higher performance gains than traditional NOMA in network capacity. Accordingly, the C-NOMA technique is supposed to be incorporated into SAGIN to achieve massive connectivity and higher spectrum efficiency for various potential IoE applications [4]. Although the utilization of NOMA technique holds great promise for future SAGIN, it will inevitably bring security issues like privacy information leakage, forgery attack, etc. One important fact is that the receiver in a typical NOMA group can decode the strongest signal of other user. Moreover, the NOMA group is generally determined based on channel quality rather than on security concerns. As such, confidential or sensitive data might be transmitted through insecure relays in each C-NOMA group.

To tackle this problem, one feasible direction is to explore advanced access authentication scheme, which is primarily design for orthogonal multiple access (OMA) communication systems. For space and ground integrated networks, Chen *et al* in [5] investigated an authentication scheme by utilizing a wireless security negotiation protection process. For NOMA communication systems, recent research works commonly focus on the physical layer authentication scheme. The authors in [6] proposed a group authentication mechanism by taking the advantages of NOMA and the irreversibility of hash operation for massive machine type communication systems. Considering the security problem caused by users colluding with the adversary, Xie *et al* in [7] proposed three physical layer authentication schemes with shared authentication tag, superimposed independent authentication tags and time division multiplexing authentication tags for NOMA systems. Based on this work, they further developed a privacy-preserving authentication scheme to improve the system authentication performance [8].

The above schemes have not paid attention to the secure access authentication against quantum attack, especially in SAGIN. Accordingly, researchers have tended to improve the access authentication scheme based on the post quantum cryptographic mechanism. For instance, the authors in [9] designed a lattice based anonymous access authentication scheme by utilizing the Number Theory Research Unit (NTRU) scheme into the key generation process. While Wang *et al* developed a NTRU based mutual authentication mechanism with the key generated based on a hash chain [10]. Note that the access authentication schemes to resist the quantum attacks are mostly designed for OMA communication systems. Therefore, how to design efficient schemes to take the merits of both C-NOMA and post quantum cryptographic mechanism in future SAGIN is still an open problem. In light of the above considerations, we in this paper investigate a NTRU based authentication scheme for C-NOMA communications in SAGIN. Our goal is to achieve secure communication with guaranteed performance by flexibly implementing security configuration within each C-NOMA group. Specifically, we design an adaptive authentication scheme for SAGIN C-NOMA communication by considering both trust and untrust relays. Besides, we also employ a Certification Authority (CA) and digital certificate technology to ensure the security of the NTRU public key.

II. System Models

As illustrated in Fig. 1, we consider a typical SAGIN downlink power-domain NOMA network systems containing satellites, unmanned aerial vehicles (UAV) and ground users. For sake of network capacity and spectrum efficiency, we assume both UAVs and ground users tend to connect to the satellite for data transmission via NOMA communication. If receivers in the same NOMA group can connect to each other, then C-NOMA communication will be constructed by establishing direct wireless links between them, so as to improve the data transmission capacity as much as possible. For ease of analysis, we consider the scenarios with each NOMA group having two users. Thereby, the tansmit power is divided into two parts, which are devoted to the near and far users in each NOMA group, respectively. It is worth noted that UAVs in the air commonly have much better channel quality (i.e., mostly line-of-sight channel) and higher computational capabilities than those of the ground users (i.e., probably non line-of-sight channel and IoT devices), so they might perform successive interference cancelation (SIC) if employ NOMA technique. Hence, we assume UAVs in the SAGIN can be employed as a relay in each C-NOMA group. For the public key authentication, we assume the public key for space , air and ground communication devices relies on a trusted Certification Authority (CA). The CA provides digital certificates for each devices, verifies the legitimacy of their identities, and facilitates the exchange of public keys.

For sake of threat model, we in this paper consider the Dolev-Yao model which is a commonly used framework for analyzing security protocols, assuming that attackers have full control over network communications with the capabilities to intercept, modify, forge, and replay messages. Specifically, this model defines the attackers abilities as follows:

- The attacker can capture all data transmitted over wireless links between UAVs and users.
- The attacker can send arbitrary data over wireless links to any UAV or satellite.
- The attacker is able to obtain certain users short-term keys used for secure communication.
- The attacker is unable to break specific cryptographic primitives, such as symmetric encryption and hash functions. That is, the attacker cannot recover plaintext from ciphertext or forge valid message authentication codes without getting the key.
- The attacker is unable to solve the shortest vector problem (SVP) in lattice-based cryptography.

III. The Proposed Scheme

In order to design an authentication scheme to resist quantum attacks while guarantee C-NOMA performance in SAGIN, we first propose a key negotiation mechanism in this part. Below is the symbol table for this authentication scheme.

A. System Establishment Phase

During the system initialization phase, the satellite, U_1 and U_2 generate their respective NTRU key pairs. Each user's key pair contains a public key and a private key, which are used for subsequent authentication and communication encryption. We denote the satellite's key pair as $(Pub_S, Priv_S)$, U_1's key pair as $(Pub_{U_1}, Priv_{U_1})$, and U_2's key pair as $(Pub_{U_2}, Priv_{U_2})$.

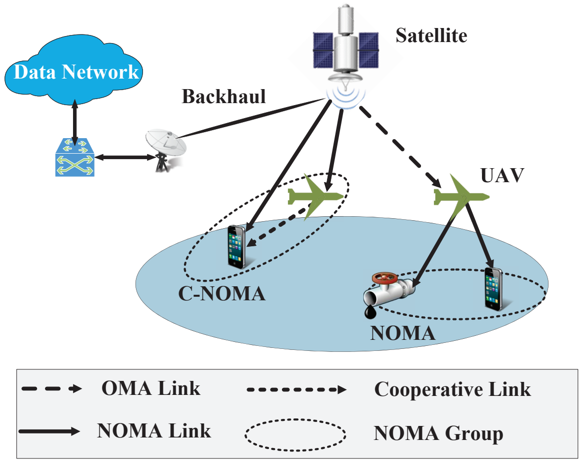

Fig. 1. Illustration of space-air-ground integrated NOMA system.

TABLE I
LIST OF VARIABLES

Variable	Description
$\|\|$	Connector
U_i	The ith user
C_i	The ith ciphertext
Pub_i	The public key of user i
$Priv_i$	The private key of user i
$SK_{i,j}$	Session key between user i and user j
$SK_{temp_{i,j}}$	Temporary session key between user i and user j
MK	MAC key
$MAC(key, msg)$	The message authentication code function
T	Validity time
$DF()$	Decode-and-Forward message
$rand()$	The random number generation fuction
$enc(key, data)$	Encryption of data using key
$dec(key, data)$	Decryption of data using key
$hash(key, data)$	Use private key to hash data
$Cert_i$	Digital certificates of user i
$HKDF()$	HMAC-based key derivation function

In the public key exchange process, all users submit their generated public keys (i.e., $Pub_S, Pub_{U_1}, Pub_{U_2}$) to the CA (Certificate Authority) for registration. After verifying the identity of each user, the CA uses its private key to generate digital certificates $Cert_S, Cert_{U_1}, Cert_{U_2}$. Each certificate includes a user identifier (ID), a public key, a validity period and a CA signature. Taking the public key exchange between satellite (SAT) and U_1 as an example, the SAT sends its certificate $Cert_S$ to U_1. U_1 uses the CA's public key to verify the legitimacy of $Cert_S$, then U_1 returns its certificate $Cert_{U_1}$ to the satellite. The SAT uses the CA's public key to verify the validity of $Cert_{U_1}$. If the verification is successful, the SAT and U_1 can securely share each others public keys Pub_S and Pub_{U_1}. Similarly, the public key exchange mechanism between the SAT and U_2, U_1 and U_2, can be realized by utilizing the above method.

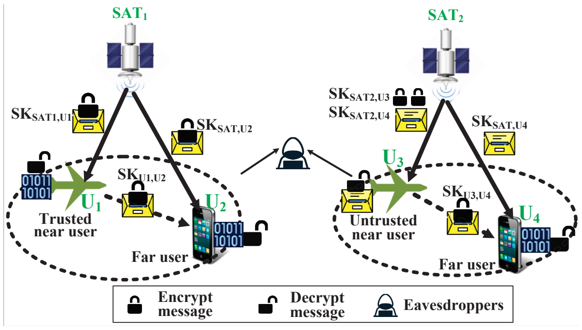

Fig. 2. Illustration of secure NOMA communication mechanism with trust and untrust relays.

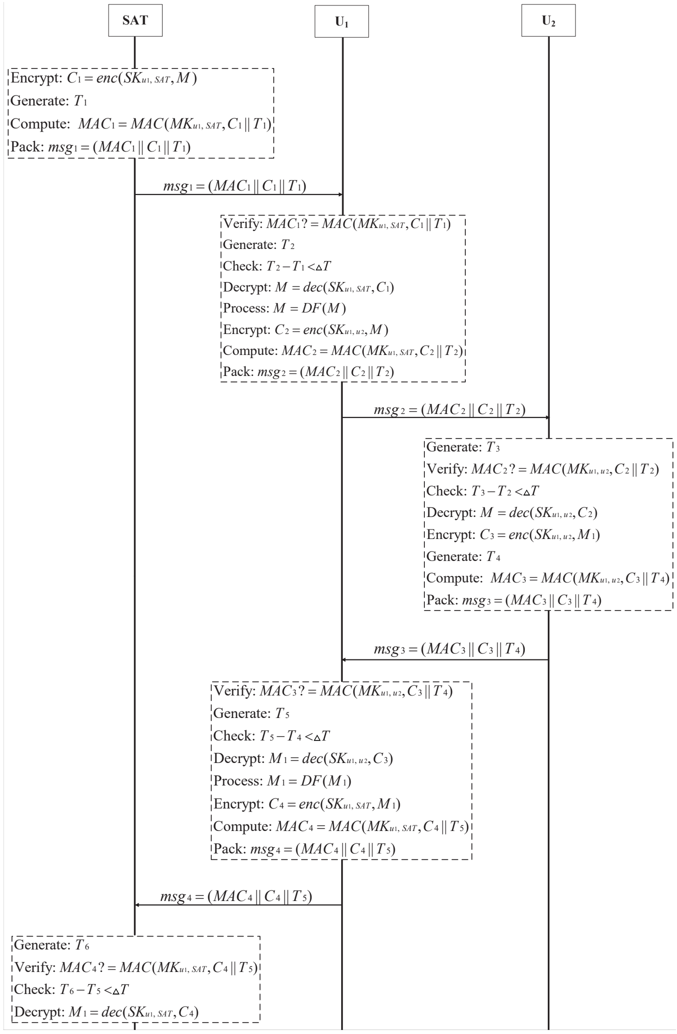
SAT
U$_1$
U$_2$
Encrypt: $C_1 = enc(SK_{u1,SAT}, M)$
Generate: T_1
Compute: $MAC_1 = MAC(MK_{u1,SAT}, C_1 \| T_1)$
Pack: $msg_1 = (MAC_1 \| C_1 \| T_1)$
$msg_1 = (MAC_1 \| C_1 \| T_1)$
Verify: $MAC_1 ?= MAC(MK_{u1,SAT}, C_1 \| T_1)$
Generate: T_2
Check: $T_2 - T_1 < \triangle T$
Decrypt: $M = dec(SK_{u1,SAT}, C_1)$
Process: $M = DF(M)$
Encrypt: $C_2 = enc(SK_{u1,u2}, M)$
Compute: $MAC_2 = MAC(MK_{u1,SAT}, C_2 \| T_2)$
Pack: $msg_2 = (MAC_2 \| C_2 \| T_2)$
$msg_2 = (MAC_2 \| C_2 \| T_2)$
Generate: T_3
Verify: $MAC_2 ?= MAC(MK_{u1,u2}, C_2 \| T_2)$
Check: $T_3 - T_2 < \triangle T$
Decrypt: $M = dec(SK_{u1,u2}, C_2)$
Encrypt: $C_3 = enc(SK_{u1,u2}, M_1)$
Generate: T_4
Compute: $MAC_3 = MAC(MK_{u1,u2}, C_3 \| T_4)$
Pack: $msg_3 = (MAC_3 \| C_3 \| T_4)$
$msg_3 = (MAC_3 \| C_3 \| T_4)$
Verify: $MAC_3 ?= MAC(MK_{u1,u2}, C_3 \| T_4)$
Generate: T_5
Check: $T_5 - T_4 < \triangle T$
Decrypt: $M_1 = dec(SK_{u1,u2}, C_3)$
Process: $M_1 = DF(M_1)$
Encrypt: $C_4 = enc(SK_{u1,SAT}, M_1)$
Compute: $MAC_4 = MAC(MK_{u1,SAT}, C_4 \| T_5)$
Pack: $msg_4 = (MAC_4 \| C_4 \| T_5)$
$msg_4 = (MAC_4 \| C_4 \| T_5)$
Generate: T_6
Verify: $MAC_4 ?= MAC(MK_{u1,SAT}, C_4 \| T_5)$
Check: $T_6 - T_5 < \triangle T$
Decrypt: $M_1 = dec(SK_{u1,SAT}, C_4)$

Fig. 3. Interaction diagram of mutual authentication between satellite and NOMA group for trust mode

B. Key Negotiation Phase

In this section, we give the key negotiation process among U_2, U_1 and SAT. Taking the key negotiation between U_1 and U_2 as an example, the U_2 firstly verifies U_1's certificate and obtains U_1's public key Pub_1. Then U_2 generates a random number r_1 which is further encrypted by using Pub_1. After creating the ciphertext C_1, U_2 generates a validity time T_1 and calculates a hash value $H_1 = H(Hello1||C_1||H(r_1)||T_1)$ to ensure data integrity. Finally, U_2 sends the message $(Hello1, C_1, Cert_{U_2}, H_1)$ to U_1. After receiving the U_2's message, U_1 first verifies U_2's certificate $Cert_{U_2}$ and obtains Pub_2 if the verification is passed. Then, U_1 uses its private key to decrypt the ciphertext C_1 to get the random number r_1', which is used to verify the hash value $(H1)'$. U_1 also checks the validity of T_1. If the verification fails, an error message will be sent to U_2 and the negotiation process is terminated. If the verification passes, U_1 will generate a random number r_2 which is further encrypted by using Pub_2 to create the ciphertext C_2. Then U_1 generates a validity time T_2, creates a session key SK_{U_1,U_2} and calculates a hash value $H(r_2) = H(Hello2||C_2||H(r_2)||T_2)$ to ensure data integrity. Finally, U_1 sends back $(Hello2, C_2, Cert_{U1}, H_2)$ to U_2.

U_2 decrypts C_2 by using its private key to obtain the random number r_2' after receiving U_1's message. Then U_2 verifies the validity of hash value H_2' and T_2. If H_2' and T_2 are valid, U_2 will calculate the session key SK_{U_2,U_1} and generate an integrity hash value $H(AllMessages_1)$ for all messages. Later, U_1 receives and verifies the hash value $H'(AllMessages_1)$ from U_2. If $H'(AllMessages_1)$ is valid, U_1 will send a hash value $H(AllMessages_2)$ back to U_2. Finally U_1 verifies the hash value $H'(AllMessages_2)$ from U_2. The key negotiation phase between the SAT and U_2, SAT and U_1, can also be executed by similarly employing the same method.

C. Secure NOMA Communication Process Phase

We in this part design a secure NOMA communication mechinsm for both trust and untrust relay scenarios. As illustrated in Fig. 2, the satellite establishes a secure communication link between the near user U_1 (resp. U_3) and the far user U2 (resp. U_4), where the U_1 (resp. U_3) acts as a relay in the NOMA group. At the beginning of the communication, U_2 (resp. U_4) can choose to use the trust mode or untrust mode if the corresponding relay is secure or not. In other words, different modes determine whether the relay can get the plaintext data during the relaying process.

Before the communication begins, the satellite, U_1, and U_2 first generate the MAC key $MK_{i,j}$ based on the session key $SK_{i,j}$ obtained from the previous handshake exchange. The MAC key is calculated as $MK_{i,j} = HKDF(SK_{i,j})$.

1) ***Trust Mode***

(1) $SAT \to U_1$

Step 1: As illustrate in Fig. 3, U_2 sends a trust mode communication request to the satellite. Upon receiving the request, the SAT encrypts the plaintext message M using the session key $SK_{U_1,SAT}$ to generate the ciphertext $C_1 = enc(SK_{SAT,U_1}, M)$.

Step 2: SAT generates the current valid time T_1 and calculates $MAC_1 = MAC(MK_{SAT,U_1}, C_1||T_1)$.

Step 3: The satellite packages the message $msg_1 = (MAC_1||C_1||T_1)$ and sends msg_1 to U_1.

(2) $U_1 \rightarrow U_2$

Step 1: U_1 first verifies the MAC value using $MK_{U_1,SAT}$ and verifies the validity of the timestamp T_1. If the timestamp is valid and the MAC verification passes, the message will be accepted. Otherwise, the message will be discarded.

Step 2: U_1 decrypts the ciphertext C_1 using the session key $SK_{U_1,SAT}$ to obtain the original plaintext data $M = dec(SK_{U_1,SAT}, C_1)$. Then U_1 generates the current valid timestamp T_2.

Step 3: U_1 encrypts the data M using the session key SK_{U_1,U_2} to generate the new ciphertext $C_2 = enc(SK_{U_1,SAT}, M)$.

Step 4: U_1 uses the MAC key MK_{U_1,U_2} to generate MAC_2, where $MAC_2 = (MK_{U_1,U_2}, C_2||T_2)$.

Step 5: U_1 packages the message $msg_2 = (MAC_2||C_2||T_2)$ and sends it to U_2.

(3) U_2 verifies the legitimacy of the message and decrypts it to obtain M.

Step 1: U_2 verifies the MAC value using MK_{U_2,U_1} and verifies the validity of the timestamp T_2. If the timestamp is valid and the MAC verification passes, the message will be accepted. Otherwise, the message will be discarded.

Step 2: U_2 uses the session key SK_{U_2,U_1} to decrypt the ciphertext C_2 and recover the original plaintext data $M = dec(SK_{U_2,U_1}, C_2)$.

At this point, the satellite has successfully transmitted a message to U_2 in the downlink. The mechanism for the uplink case is similar to the above steps and we omit it here.

2) ***Untrust Mode***

(1) $SAT \rightarrow U_1$

Step 1: As illustrate in Fig. 4, the SAT periodically generates a new random number $R = rand()$, which serves as a seed for mixing the temporary session key $SK_{temp_{SAT,U_2}} = HKDF(SK_{SAT,U_2}, R)$. $SK_{temp_{SAT,U_2}}$ is only used for encrypting communication in this session.

Step 2: The satellite encrypts the plaintext message using the temporary session key $SK_{temp_{SAT,U_2}}$ to produce ciphertext C_1 = $enc(SK_{temp_{SAT,u_2}}, M)$, and subsequently encrypts the ciphertext message C_1' using the key $SK_{U_1,SAT}$ to generate ciphertext C_1' =$enc(SK_{temp_{SAT,U_1}}, C_1)$.

Step 3: It then generates the current valid timestamp T_1 and calculates the message authentication code as $MAC_1 = MAC(MK_{U_1,SAT}, C_1'||R||T_1)$ using MK_{SAT,U_1}. Finally, the satellite packages the message $msg_1 = (MAC_1||C_1'||R||T_1)$ and sends it to U_1.

(2) $U_1 \rightarrow U_2$

Step 1: U_1 first verifies the MAC value using the shared MAC key $MK_{U_1,SAT}$ and verifies the validity of the timestamp T_1. If the timestamp is valid and the MAC verification passes, the message will be accepted. Otherwise, the message will be discarded.

Step 2: U_1 decrypts the ciphertext C_1' using the session key $SK_{U_1,SAT}$ to obtain the ciphertext C_1 = $dec(SK_{U_1,SAT}, C_1')$.

Step 3: U_1 generates the current valid timestamp T_2.

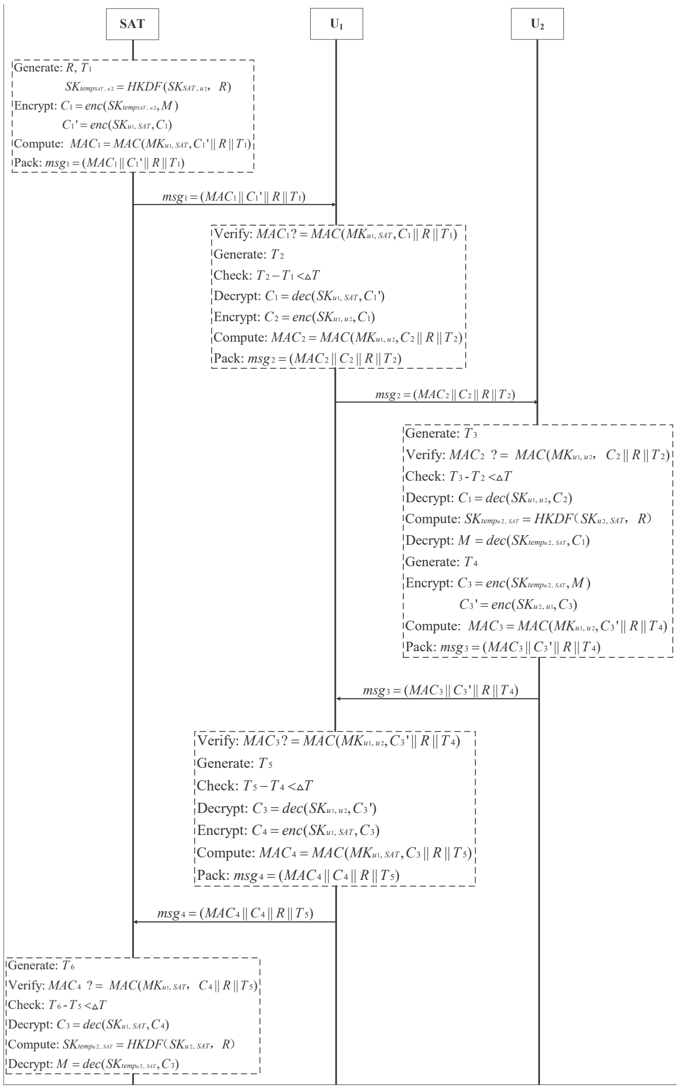
SAT
U_1
U_2
Generate: R, T_1
$SK_{temp_{SAT,u2}} = HKDF(SK_{SAT,u2}, R)$
Encrypt: $C_1 = enc(SK_{temp_{SAT,u2}}, M)$
$C_1' = enc(SK_{u1,SAT}, C_1)$
Compute: $MAC_1 = MAC(MK_{u1,SAT}, C_1'||R||T_1)$
Pack: $msg_1 = (MAC_1||C_1'||R||T_1)$
$msg_1 = (MAC_1||C_1'||R||T_1)$
Verify: $MAC_1 ?= MAC(MK_{u1,SAT}, C_1||R||T_1)$
Generate: T_2
Check: $T_2 - T_1 < \Delta T$
Decrypt: $C_1 = dec(SK_{u1,SAT}, C_1')$
Encrypt: $C_2 = enc(SK_{u1,u2}, C_1)$
Compute: $MAC_2 = MAC(MK_{u1,u2}, C_2||R||T_2)$
Pack: $msg_2 = (MAC_2||C_2||R||T_2)$
$msg_2 = (MAC_2||C_2||R||T_2)$
Generate: T_3
Verify: $MAC_2 ?= MAC(MK_{u1,u2}, C_2||R||T_2)$
Check: $T_3 - T_2 < \Delta T$
Decrypt: $C_1 = dec(SK_{u1,u2}, C_2)$
Compute: $SK_{temp_{u2,SAT}} = HKDF(SK_{u2,SAT}, R)$
Decrypt: $M = dec(SK_{temp_{u2,SAT}}, C_1)$
Generate: T_4
Encrypt: $C_3 = enc(SK_{temp_{u2,SAT}}, M)$
$C_3' = enc(SK_{u2,u1}, C_3)$
Compute: $MAC_3 = MAC(MK_{u1,u2}, C_3'||R||T_4)$
Pack: $msg_3 = (MAC_3||C_3'||R||T_4)$
$msg_3 = (MAC_3||C_3'||R||T_4)$
Verify: $MAC_3 ?= MAC(MK_{u1,u2}, C_3'||R||T_4)$
Generate: T_5
Check: $T_5 - T_4 < \Delta T$
Decrypt: $C_3 = dec(SK_{u1,u2}, C_3')$
Encrypt: $C_4 = enc(SK_{u1,SAT}, C_3)$
Compute: $MAC_4 = MAC(MK_{u1,SAT}, C_3||R||T_5)$
Pack: $msg_4 = (MAC_4||C_4||R||T_5)$
$msg_4 = (MAC_4||C_4||R||T_5)$
Generate: T_6
Verify: $MAC_4 ?= MAC(MK_{u1,SAT}, C_4||R||T_5)$
Check: $T_6 - T_5 < \Delta T$
Decrypt: $C_3 = dec(SK_{u1,SAT}, C_4)$
Compute: $SK_{temp_{u2,SAT}} = HKDF(SK_{u2,SAT}, R)$
Decrypt: $M = dec(SK_{temp_{u2,SAT}}, C_3)$

Fig. 4. Interaction diagram of mutual authentication between satellite and NOMA group for untrust mode

Step 4: U_1 encrypts the data using the session key SK_{U_1,U_2} to generate the new ciphertext C_2 = $enc(SK_{U_1,U_2}, C_1)$ and using the MAC key MK_{U_1,U_2} to generate MAC_2, where $MAC_2 = (MK_{U_1,SAT}, C_2||R||T_2)$.

Step 5: U_1 packages the message $msg_2 = (MAC_2||C_2||R||T_2)$ and sends it to U_2.

(3) U_2 verifies the legitimacy of the message and decrypts it to obtain M

Step 1: U_2 verifies the MAC value using the shared MAC key MK_{U_2,U_1}. If the timestamp is valid and the MAC verification passes, the message will be accepted. Otherwise, the message will be discarded.

Step 2: U_2 uses the session key SK_{U_1,U_2} to decrypt the ciphertext C_2 and obtains the ciphertext C_1= $dec(SK_{U_1,U_2}, C_2)$.

Step 3: U_2 calculates the temporary session key
$SK_{temp_{U_2,SAT}} = HKDF(SK_{U_2,SAT}, R)$

Step 4: U_2 uses $SK_{temp_{U_2,SAT}}$ to decrypt C_1 to retrieve the plaintext $M = dec(SK_{temp_{U_2,SAT}}, C_1)$.

At this point, the satellite has successfully transmitted a message to U_2 in the downlink. The mechanism for the uplink scenario is similar to these steps and will not be elaborated here.

IV. PERFORMANCE EVALUATION AND ANALYSIS

In this section, we analyze the effectiveness and security of the proposed scheme and demonstrate the performance advantages of our proposed scheme from two perspectives: communication overhead and computational overhead. Additionally, we employ the formal analysis tool, BAN Logic, to verify the correctness of the proposed scheme. All experiments are conducted on a personal computer equipped with an Intel Core i5-13400F CPU running on a Windows 11 operating system. Through analysis, our scheme can achieve secure authentication, key agreement functions, and supports message accessibility for near users in C-NOMA groups. It also ensures the confidentiality of key agreement, guarantees forward secrecy and backward secrecy.

A. Communication Overhead

For a complete end-to-end transmission, we use the metric STGD (satellite-to-ground transmission delay, around $4.833ms$ per STGD) raised in the literature [9] to measure the size of communication overhead. Table III is a comparison of the communication overhead of our scheme with other schemes. As shown in table II, the scheme in reference [11] and [10] require four STGD, [12] requires six STGD, and [9] requires two STGD. Compared to these schemes, our proposed scheme only requires two STGD, which is equal to the scheme in [9] and less than the others.

TABLE II
COMPARISON OF COMMUNICATION OVERHEAD

Scheme	Communication Overhead
Y. Zhong, *et al* [11]	4STGD
Y. Wang, *et al* [10]	4STGD
Y. Zhou, *et al* [12]	6STGD
S. Wang, *et al* [9]	2STGD
OURS	2STGD

B. Computation Overhead

We define the computational cost as the total time spent on various operations in the entire authentication scheme. The time cost of each type of operation is explained as follows: $T_h(0.186ms)$ is the time cost for Hash Operation, $T_r(0.51ms)$ is the time cost for Random Generation, $T_e(5.8ms)$ is the time cost for modular exponent, $T_{NE}(0.15ms)$ is the time cost for NTRU encryption, $T_{ND}(0.16ms)$ is the time cost for NTRU decryption, $T_{NM}(0.026ms)$ is the time cost for NTRU modulus multiplication, and $T_g(73ns)$ is the time cost for random sampling operation.

From Table III, it can be seen that the computational overheads of the schemes proposed in [11], [10], [12], and [9] are 23.944ms, 1.664ms, 1.266ms, and 1.427ms, respectively. While our scheme has a computational overhead of 3.838ms. Although the computational overhead of our scheme is higher than the other three schemes except for [11], [10] and [12] require at least 2 additional STGD operations compared to our scheme. [9] has the same number of STGD operations as our scheme, but it cannot complete satellite-ground communication when the users channel quality is poor. Therefore, our scheme can achieve better performance than the other four schemes.

TABLE III
COMPARISON OF COMPUTATION OVERHEAD

Scheme	Authentication delay
Y. Zhong, *et al* [11]	$3T_e + 4T_h$
Y. Wang, *et al* [10]	$8T_h + T_{NE} + T_{NM}$
Y. Zhou, *et al* [12]	$5T_h + T_{NE} + T_{ND} + T_{NM}$
S. Wang, *et al* [9]	$6T_h + 8T_g + T_{NE} + T_{ND}$
OURS	$4T_h + 4T_r + T_{NE} + T_{ND}$

C. Security Analysis

For security analysis and comparison, we employ the related research to analyze the security of our proposed scheme, as shown in Table IV. Specifically, Reference [11] demonstrates that their protocol is ECK-secure under the Computational Diffie-Hellman (CDH) assumption in the random oracle model. Reference [10] and Reference [12] use qualitative security analysis and BAN logic to evaluate the security properties of the proposed scheme. Reference [9] uses a mathematical formalization approach to verify the correctness of its designed protocol, ensuring that the protocol meets various security requirements under the expected security model and assumptions. We compare the security and functional features with four other schemes. To defend against **replay attacks**, our scheme uses timestamp, random numbers, and MAC to ensure each session key is unique, so that replayed messages cannot pass verification. To resist **DoS attacks**, Our scheme uses MAC and timestamp to authenticate legitimate requests, discarding expired or forged requests to mitigate the impact of DoS attacks. To defend against **man-in-the-middle attacks**, Our scheme uses public keys issued by the CA for bidirectional identity verification. This ensures the authenticity of the communication. To resist **quantum attacks**, our scheme adopts the NTRU encryption algorithm, which relies on a lattice-based design that quantum algorithms cannot efficiently break. Our scheme independently generates session keys through the HKDF function. Even if the current session key is compromised, the keys from previous or future sessions are not impacted. This ensures **forward and backward security**. Finally, our scheme achieves **message accessibility for weak-channel users**.

D. Correctness Proof

In this section, we use the Ban logic to prove the correctness and security of the scheme, which is based on a set of specific inference rules that abstract information about entities' beliefs, trust relationships, and the freshness of messages exchanged during protocol interactions to derive whether the

TABLE IV
COMPARISON OF SECURITY AND FUNCTIONAL FEATURES

Security Features	Y. Zhong, *et al* [11]	Y. Wang, *et al* [10]	Y. Zhou, *et al* [12]	S. Wang, *et al* [9]	**Ours**
Mutual authentication	✓	✓	✓	✓	✓
Key Agreement	✓	✓	✓	✓	✓
Forward Security	✓	✓	✓	✓	✓
Backward Security	✓	✓	✓	✓	✓
Man-in-the-middle Attack	✓	✓	✓	✓	✓
Replay Attack	✓	✓	✓	✓	✓
Security Formal Proof	✓	✓	✓	✓	✓
Resist Quantum Attack	×	×	✓	✓	✓
Message accessibility for near users	×	×	×	×	✓

final protocol can achieve its expected security goals [13]. First, we define the symbols used in Ban logic:

- A and B represent the communicating entities, and X represents the message transmitted in the protocol.
- $P| \equiv X$ denotes that entity P believes X to be true, meaning entity P considers X to be a valid assertion.
- $P \lhd X$ indicates that entity P has received the message containing X.
- $P| \sim X$ means that entity P once send message X.
- $P| \Rightarrow X$ means that P has jurisdiction over message X.
- $\#(X)$ denotes that X is fresh.
- X_K denotes that message X has been encrypted with key K.
- $P \overset{K}{\leftrightarrow} Q$ means that the K is shared between P and Q.
- SK_{U_1,U_2} represents the session key between U_1 and U_2, which is used for confidential communication between the two parties.
- $H(X)$ represents the hash value of X, which is used for data integrity verification."

Here are the definitions of inference rules:

- Message implication rule $\frac{P|\equiv Q \leftrightarrow K \leftrightarrow P, P \lhd X_K}{P|\equiv Q|\equiv X}$. This rule indicates that if P shares a secret key K with Q and receives a message X encrypted by K, then P believes that Q believes X is true.
- Nonce verification rule $\frac{P|\equiv \#(x), P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$. This rule states that if P believes X is fresh, and also believes that Q has sent X, then P can trust the authenticity of X.
- Jurisdiction rule $\frac{P|\equiv Q|\Rightarrow X, P|\equiv Q|\equiv X}{P|\equiv X}$. If P believes that Q has jurisdiction over X, and P trust Q's judgment on the authenticity of X, then P can trust X.
- Belief transmission rule $\frac{P|\equiv Q|\equiv X, P|\equiv Q|\sim X}{P|\equiv X}$. If P believes Q and P believes Q controls X, then P can consider X to be true.

Goals: Based on the above BAN symbols and logical definitions, our scheme should achieve the following security objectives.

G1: $U_1| \equiv U_1 \overset{SK_{U1,U2}}{\longleftrightarrow} U_2$

G2: $U_2| \equiv U_1 \overset{SK_{U1,U2}}{\longleftrightarrow} U_2$

G3: $U_1| \equiv U_2| \equiv U_1 \overset{SK_{U1,U2}}{\longleftrightarrow} U_2$

G4: $U_2| \equiv U_1| \equiv U_1 \overset{SK_{U1,U2}}{\longleftrightarrow} U_2$

Messages: According to the proposed authentication scheme, we transform the scheme into the following form:

Message1: $U_2 \rightarrow U_1$:
$U_1 \lhd \{T_1||\{r_1\}Pub_{U_1}||Hello1||H\{Hello1||\{r_1\}Pub_{U_1}||T_1\}\}$

Message2: $U_1 \rightarrow U_2$:
$U_2 \lhd \{T_2||\{r_2\}Pub_{U_2}||Hello2||H\{Hello2||\{r_2\}Pub_{U_2}||T_2\}\}$

Message3: $U_2 \rightarrow U_1$:
$U_1 \lhd \{H\{\{T_1||\{r_1\}Pub_{U_1}||Hello1||H\{Hello1||\{r_1\}Pub_{U_1}|| T_1\}\}, \{T_2||r_2Pub_{U_2}||Hello2||H\{Hello2||\{r_2\}Pub_{U_2}||T_2\}\}\}\}$

Message4: $U_1 \rightarrow U_2$:
$U_2 \lhd \{H\{\{H\{\{T1||\{r_1\}Pub_{U_1}||Hello1||H\{Hello1||\{r_1\} Pub_{U_1}||T_1\}\}, \{T_2||r_2Pub_{U_2}||Hello2||H\{Hello2||\{r_2\}Pub_{U_2} ||T_2\}\}\}\}\}\}$

States: According to the protocol description, the initial state is as follows

A1: $U_1 \lhd \{r_1\}Pub_{U_1}$

A2: $U_1| \equiv \#(r_1)$

A3: $U_1| \equiv \#(T_1)$

A4: $U_1| \equiv U_2| \Rightarrow U_1 \overset{r_1}{\longleftrightarrow} U_2$

A5: $U_2 \lhd \{r_2\}Pub_{U_2}$

A6: $U_2| \equiv \#(r_2)$

A7: $U_2| \equiv \#(T_2)$

A8: $U_2| \equiv U_1| \Rightarrow U_1 \overset{r_2}{\longleftrightarrow} U_2$

A9: $U_2| \equiv U_2 \overset{r_1}{\longleftrightarrow} U_1$

A10: $U_1| \equiv U_1 \overset{r_2}{\longleftrightarrow} U_2$

A11: $U_1 \lhd \{AllMessage\}SK_{U1,U2}$

A12: $U_1| \equiv \#(U_1 \overset{SK_{U1,U2}}{\longleftrightarrow} U_2)$

A13: $U_2 \lhd \{AllMessage\}SK_{U2,U1}$

A14: $U_2| \equiv \#(U_2 \overset{SK_{U2,U1}}{\longleftrightarrow} U_1)$

Proof:

With the message1, assumption A_1, and message-meaning rule $\frac{P|\equiv P \overset{K}{\leftrightarrow} Q, P \lhd \{x\}_K}{P|\equiv Q|\sim X}$, we get

S₁. $U_1| \equiv U_2| \sim message1$

With the assumption A_2 and the Nonce-verification rule $\frac{P|\equiv \#(X), P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$. we obtain

S₂. $U_1| \equiv U_2| \equiv U_2 \overset{r_1}{\longleftrightarrow} U_1$

According to assumption A_4, S_2, we apply the jurisdiction rule $\frac{P|\equiv Q|\Rightarrow X, P|\equiv Q|\equiv X}{P|\equiv X}$ to obtain

S₃. $U_1| \equiv U_2 \overset{r_1}{\longleftrightarrow} U_1$

With the message2, assumption A_5, and message-meaning rule $\frac{P|\equiv P \overset{K}{\leftrightarrow} Q, P \lhd \{x\}_K}{P|\equiv Q|\sim X}$, we get

S₄. $U_2| \equiv U_1| \sim message2$

With the assumption A_6 and the Nonce-verification rule $\frac{P|\equiv \#(x), P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$. we obtain

$\mathbf{S_5}$. $U_2| \equiv U_1| \equiv U_1 \overset{r_2}{\longleftrightarrow} U_2$

According to assumption A_8, S_5,we apply the jurisdiction rule $\frac{P|\equiv Q|\Rightarrow X, P|\equiv Q|\equiv X}{P|\equiv X}$ to obtain

$\mathbf{S_6}$. $U_2| \equiv U_2 \overset{r_2}{\longleftrightarrow} U_1$

With the assumption S_6, A_9, S_3, A_{10}, and $SK_{U_1,U_2} = HKDF(r_1, r_2)$, **we prove $\mathbf{G_1}, \mathbf{G_2}$.**

With the assumption G_1, A_{11}, we apply the message-meaning rule $\frac{P|\equiv P\overset{K}{\leftrightarrow}Q, P\lhd\{x\}_K}{P|\equiv Q|\sim X}$, we get

$\mathbf{S_7}$. $U_1| \equiv U_2| \sim (U_2 \overset{SK_{U_1,U_2}}{\longleftrightarrow} U_1)$

According to assumption S_7, A_{12}, we apply the Nonce-verification rule $\frac{P|\equiv \#(x), P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$, **we prove $\mathbf{G_3}$.**

With the assumption G_1, A_{13}, we apply the message-meaning rule $\frac{P|\equiv P\overset{K}{\leftrightarrow}Q, P\lhd\{x\}_K}{P|\equiv Q|\sim X}$, we get

$\mathbf{S_8}$. $U_2| \equiv U_1| \sim (U_1 \overset{SK_{U_1,U_2}}{\longleftrightarrow} U_2)$

According to assumption S_8, A_{14}, we apply the Nonce-verification rule $\frac{P|\equiv \#(x), P|\equiv Q|\sim X}{P|\equiv Q|\equiv X}$, **we prove $\mathbf{G_4}$.**

Finally, we finish the proof.

V. CONCLUSION

In this paper, we proposed a trust-based adaptive access authentication scheme to resist quantum attack while guarantee communication performance for C-NOMA communications in SAGIN. A dedicated NTRU key generation algorithm was designed by considering both trust and untrust C-NOMA relay scenarios. Moreover, we introduced both the temporary and long-term primary keys to solve the key leakage problem, while further enhancing the data security for the C-NOMA communications with untrusted relays. The security proof and performance evaluation results showed that the proposed scheme ensured security, while outperforming existing schemes in terms of communication, computation and data transmission security for all users in C-NOMA groups in SAGIN NOMA systems.

REFERENCES

[1] C.-X. Wang, X. You, X. Gao, X. Zhu, Z. Li, C. Zhang, H. Wang, Y. Huang, Y. Chen, H. Haas, J. S. Thompson, E. G. Larsson, M. D. Renzo, W. Tong, P. Zhu, X. Shen, H. V. Poor, and L. Hanzo, "On the road to 6g: Visions, requirements, key technologies, and testbeds," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 905–974, 2023.

[2] X. Wang, H. Chen, and F. Tan, "Hybrid oma/noma mode selection and resource allocation in space-air-ground integrated networks," *IEEE Transactions on Vehicular Technology*, pp. 1–17, 2024.

[3] Z. Ding, M. Peng, and H. V. Poor, "Cooperative non-orthogonal multiple access in 5g systems," *IEEE Communications Letters*, vol. 19, no. 8, pp. 1462–1465, 2015.

[4] P. Qin, M. Fu, Y. Fu, R. Ding, and X. Zhao, "Collaborative edge computing and program caching with routing plan in c-noma-enabled space-air-ground network," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2024.

[5] M. Chen, S. Guo, X. Huang, L. Su, and H. Du, "Research on secure access in converged satellite and terrestrial networks," in *2023 IEEE 31st International Conference on Network Protocols (ICNP)*, 2023, pp. 1–6.

[6] Y. Zhang, H. Wu, Z. Wei, Q. Gao, N. Zhang, and X. Tao, "Physical layer group authentication in mmtc networks with noma," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, 2021, pp. 1–6.

[7] N. Xie, S. Zhang, and A. X. Liu, "Physical-layer authentication in non-orthogonal multiple access systems," *IEEE/ACM Transactions on Networking*, vol. 28, no. 3, pp. 1144–1157, 2020.

[8] N. Xie, Q. Zhang, J. Chen, and H. Tan, "Privacy-preserving physical-layer authentication for non-orthogonal multiple access systems," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 4, pp. 1371–1385, 2022.

[9] S. Wang, G. Zhao, C. Xu, Z. Han, and S. Yu, "A ntru-based access authentication scheme for satellite terrestrial integrated network," in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, 2022, pp. 3629–3634.

[10] Y. Wang, W. Zhang, X. Wang, M. K. Khan, and P. Fan, "Security enhanced authentication protocol for space-ground integrated railway networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 1, pp. 370–385, 2024.

[11] Z. Yantao and M. Jianfeng, "A highly secure identity-based authenticated key-exchange protocol for satellite communication," *Journal of Communications and Networks*, vol. 12, no. 6, pp. 592–599, 2010.

[12] Y. Zhou and L. Wang, "A lattice-based authentication scheme for roaming service in ubiquitous networks with anonymity," *Security and Communication Networks*, vol. 2020, no. 3, pp. 1–19, 2020.

[13] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," vol. 8, no. 1, 1990. [Online]. Available: https://doi.org/10.1145/77648.77649

[14] X. Li, Q. Wang, M. Zeng, Y. Liu, S. Dang, T. A. Tsiftsis, and O. A. Dobre, "Physical-layer authentication for ambient backscatter-aided noma symbiotic systems," *IEEE Transactions on Communications*, vol. 71, no. 4, pp. 2288–2303, 2023.

[15] I. Altaf, M. A. Saleem, K. Mahmood, S. Kumari, P. Chaudhary, and C.-M. Chen, "A lightweight key agreement and authentication scheme for satellite-communication systems," *IEEE Access*, vol. 8, pp. 46 278–46 287, 2020.