QuDPas-FHA: Quantum-Defended Privacy-Preserved Fast Handover Authentication in Space Information Networks

Arijit Karati [a], Ting-Yu Chen [b] and Kai-Yao Lin
Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan

Keywords: Authentication, Space Information Networks, Post-Quantum Security, Key Agreement, Handover Mechanism, Access Identity Authentication, Anonymity.

Abstract: The Space Information Network (SIN) has evolved from a terrestrial network to an extension, enhancing communication capabilities and enabling augmented intelligence research. However, communication security is crucial due to potential risks like inadequate implementation and high access latency. This could allow malicious organizations to access gateways and compromise the system's safety and privacy. This work proposes a novel framework and authentication protocol to streamline the incorporation of security measures into unencrypted wireless communication within the SIN. The proposed authentication protocol is based on signcryption and HMAC, ensuring communication confidentiality, access identity validation, and anonymity. The protocol utilizes lattice cryptography and demonstrates resilience against quantum attacks. Besides, the protocol ensures user anonymity while safeguarding identity management by considering a suitable approach to overseeing revocable keys. The evaluated protocol satisfies message authentication, unlinkability, traceability, and identity privacy criteria, thwarting several security risks, including replay attacks, man-in-the-middle attacks, node impersonation, and quantum attacks. Compared to existing works, our protocol exhibits significant promise in enabling secure communication with adequate functional overheads within the SIN framework.

1 INTRODUCTION

With the acceleration of communication technologies anywhere and anytime, the space industry is reviving services like earth observation, space-based cloud, re mote sensing, and Internet of Things (IoT) data collection. Standardization organizations like the 3rd Generation Partnership Project (3GPP), International Telecommunication Union (ITU), and European Telecommunications Standards Institute (ETSI) are exploring satellite communications to integrate space and terrestrial networks, supporting future wireless ecosystems (Liberg et al., 2020).

The Space Information Network (SIN) offers global coverage and on-demand bandwidth and are not limited by geographical conditions, unlike traditional wireless communication like road and cellular networks (Chen et al., 2021). Improved technologies like satellite miniaturization, reusable rocket launch, and semiconductor technology can integrate low-orbit satellites, drones, and airships into the SIN for extended connectivity. However, data can be received if they pass exact detection ranges, as ground stations are sometimes not feasible in specific locations.

Orbit Specifications. As shown in Figure 1, the SIN combines the capabilities of ground-based wireless networks with satellites. It has three categories: *geo- synchronous Earth orbit* (GEO), *medium Earth orbit* (MEO), and *low Earth orbit* (LEO) satellites. GEO satellites appear stationary over the Earth's surface, while LEO and MEO satellites are classified as non-GSO. LEO satellites conduct communication and scientific data sharing, while MEO spacecraft navigate and share space data. Typically, GEO satellites have a round-trip delay of about 550 milliseconds (ms), while 240 ms for LEO satellites indicates a significant benefit in real-time applications. SIN allows LEOs to interact with terrestrial networks via satellite-ground (SG) links, inter-satellite links (ISL), and a network control center (NCC). A ground station (GS) connects LEOs to other Internet endpoints and resources, serving as a ground interface for LEOs and connecting to the NCC via terrestrial networks. However, there are various cyberthreats while communicating with LEO.

[a] https://orcid.org/0000-0001-5605-7354
[b] https://orcid.org/0009-0009-6135-3508

Karati, A., Chen, T. and Lin, K.
QuDPas-FHA: Quantum-Defended Privacy-Preserved Fast Handover Authentication in Space Information Networks.
DOI: 10.5220/0012845600003767

In *Proceedings of the 21st International Conference on Security and Cryptography (SECRYPT 2024)*, pages 298-309
ISBN: 978-989-758-709-2; ISSN: 2184-7711

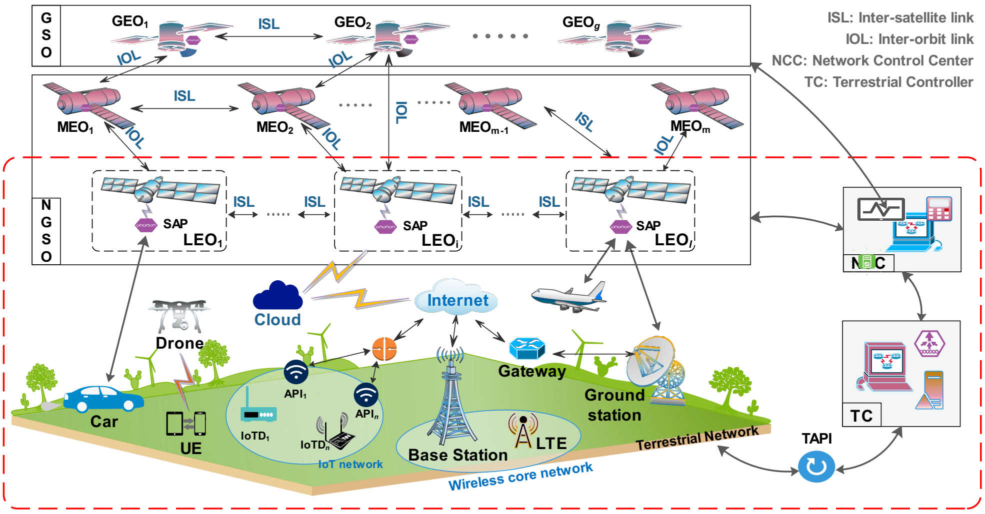

Figure 1: A schematic diagram of the Space Information Network (SIN) where attackers compromise satellite services by exploiting public parameters and information-sharing flaws. Robust handover authentication (in red box) prevents data invasion.

Handover Between SIN Entities. Transmission delays are less for LEO spacecraft than for GEO and MEO satellites, making it better suited to delivering data communication and access services in underdeveloped countries. Communication between the user, satellite, and GS is usually ongoing during a successful authentication session where a user connects to a satellite via a satellite access point (SAP). Typically,

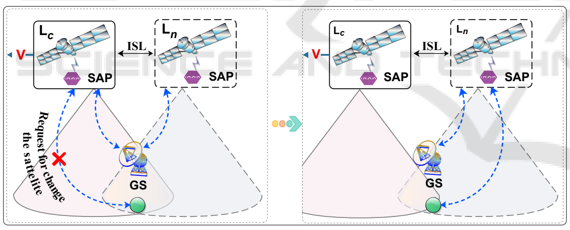

Figure 2: Satellite only handover.

satellites move faster than Earth's surface, creating a dynamic network where GS offers users an interface to the terrestrial network as a ground station. Two wireless connections are commonly used to communicate with the user: a) *between the user and SAP*, and b) *between SAP and GS*. These two links may fail due to the mobility of the satellite and its users. Figure 2 defines a typical handover scenario for user U_x.

- **Satellite *only* Handover** ($S_c \nleftrightarrow U_x$)**:** The satellite S_c moves to the far end with speed v, and the user's distance from S_c increases, interrupting signal. Before the signal is interrupted, U_x applies to S_c for the user-ground station link to be relayed by a new S_n with improved communication.

Thus, it is important to provide communication safety for U_x when S_c hands over session to S_n.

1.1 Security and Privacy Requirements

Fast Handover Authentication (FHA) in SIN should meet critical safety traits between U_i and the ground station GS_k relaying S_j for an attacker $\mathcal{A}$:

F1. (*Qu-safety*): Quantum computers may break classic public-key cryptosystems. So, SIN needs strong authentication to stem quantum attacks.

F2. (*Level-II Security*): The key generation requires user and NCC participation, due to the fact that the NCC may be compromised (Girault, 1991).

F3. (*Mutual Authentication (Karati et al., 2023)*): Enabling U to discern GS and vice versa is a crucial trait for end-to-end authentication.

F4. (*Session Key Agreement*): It enables U_i and GS_k to convince a specific key for a single session.

F5. (*Forward/Backward Safety*): The former avoids future data theft while the latter fixes past breach.

F6. (*User Privacy*): Accessing GS_k makes U_i untraceable and communication unlinkable for $\mathcal{A}$.

F7. (*Fast Handover*): U_i does not need to be reauthenticated to continue services from GS_k if its session doesn't timeout when S_j leaves the range.

Further, an authentication scheme must resist attacks:

A1. (*Entity Impersonation*): One must repel $\mathcal{A}$ to mimic a valid U_i to link with GS_k to send fake data.

A2. (*Replay Attack*): One must repel $\mathcal{A}$ to purposely and deceptively send past data for system access.

A3. (*Man-in-the-Middle Attack*): One must repel $\mathcal{A}$ to modify data between two parties (assuming they covertly find a session key).

A4. (*Ephemeral Secret Leakage*): One must stop $\mathcal{A}$ to get ephemeral secrets to reveal sessions keys.

A5. (*GPS Spoofing*): Robust authentication prevents U_i from sending a fake location to S_j or GS_k.

1.2 Related Works

The surge in quantum computation has resulted in a substantial upswing in designing cryptosystems that protect against quantum attacks. The notion of lattice-based signcryption under the Fiat-Shamir framework was proposed in 2009 (Lyubashevsky, 2009), which later was enhanced by refining the signing procedure (Lyubashevsky, 2012). Following this, the authors in (Bai and Galbraith, 2014) proposed a lattice-based signcryption based on the learning with errors (LWE) assumption. Next, the authors in (Ma et al., 2015) designed public key encryption with delegated equality test with flexible authorization, strengthening privacy protection. In 2018, the authors in (Sato and Shikata, 2018) devised a lattice-based signcryption approach without the random oracle model (ROM) under the hardness of the LWE and small integer solution (SIS). After that, the authors in (Yang et al., 2018) proposed a roaming authentication for the SIN. It utilizes the group signature to ensure user anonymity. Following that, the authors in (Gérard and Merckx, 2018) presented a signcryption based on the work in (Bai and Galbraith, 2014) under the Fiat-Shamir framework. Shortly thereafter, the authors in (Duong et al., 2019) developed a new public key encryption with equality test (PKEET) to verify if two ciphertexts are from the same message. Next, the authors in (Ma et al., 2019) introduced a lattice-based access authentication for massive IoT devices. After that, the authors in (Guo and Du, 2020) designed an RLWE-based anonymous mutual authentication and key agreement (AKA) protocol to support a lower cost without compromising security. Based on the work in (Duong et al., 2019), the authors in (Le et al., 2021) found critical issues in (Sato and Shikata, 2018) and devised a new scheme. Next, the authors in (Le et al., 2021) proposed a lattice-based signcryption with equality test resistant to internal attacks. However, it fail to support the authorization model of the designated tester. After that, two quantum-safe PKEET constructions were devised that provide anonymity and secure in the standard model based on integer and ideal lattices (Roy et al., 2022). Subsequently, the authors in (Guo et al., 2022) designed a secure authentication protocol based on the randomized RLWE, which decreases the authentication delay.

Recently, the authors in (Dharminder et al., 2023) introduced an authentication protocol to enhance the security of the satellite's communication based on the RLWE assumption. After that, the authors in (Al-Mekhlafi et al., 2023) proposed a lattice-based lightweight cryptosystem in 5G-enabled vehicular networks. Most of the above works emphasize quantum safety without delving into any specific application, particularly in the context of SIN. Thus, despite resisting quantum attacks, they might not possess the comprehensive security attributes mentioned earlier.

1.3 Motivation and Our Contributions

Despite extensive use cases, most authentication systems in SIN do not consider modern attacks, including quantum attacks. Some systems employ timestamps with cumbersome time synchronization; others ignore effective handover scenarios. Besides, many methods are secure yet inefficient privacy measures with inadequate cryptographic operations. Motivated by these security and privacy challenges, we pose a question: *Could we design a fast, anonymous authentication protocol for the SIN, allowing secure handover authentication while resisting quantum attacks?*

To address the aforementioned question, we propose a novel protocol called QuDPas-FHA employing the *nonce-based challenge-response pairs* (CRPs) for authentication. We utilize the hash-based message authentication code (HMAC), encryption, and singcryption to achieve specific security benefits as mentioned in Section 1.1. Within the context of the authentication framework, we contribute the following:

- The QuDPas-FHA utilizes lattice-based operations and HMAC to achieve mutual authentication and robust session key agreement. Besides, it attains Girault's Level-II security for cryptographic keys by generating keys that consider the user's and NCC's participation.
- It guarantees user privacy by enabling anonymous communication during authentication, which is untraceable and unlinkable for $\mathcal{A}$.
- It satisfies the necessary security properties F1-F7 and A1-A5 based on the decisional Ring-LWE and HMAC assumptions.
- We assess the efficacy of QuDPas-FHA in terms of authentication delay, handover overhead, transmission, and storage costs. It achieves more security functionalities compared to existing related works with adequate functional overheads.

The remaining paper is organized as follows. Section 2 mentions the preliminaries. Section 3 illustrates details of the QuDPas-FHA. Next, Section 4 provides security discussion and Section 5 lists performance comparison. Finally, Section 6 concludes our work.

2 PRELIMINARIES

This section describes the necessary backgrounds for comprehending the proposed work.

2.1 Cryptographic Hash Function

For some t, a cryptographic hash $H \in \mathcal{H}$ maps $m(t)$-bit binary strings to $l(t)$-bit binary strings (Ramos-Calderer et al., 2021). It holds the following traits:

- The length $m(t)$ is greater than the length of $l(t)$.
- $H(\cdot)$ can be computed in polynomial time t, and
- A polynomial-time algorithm $\mathcal{A}$ gets advantage $\Pr[\mathcal{A}(x_1,x_2) \mid x_1 \neq x_2$ and $H(x_1) = H(x_2)] \leq \varepsilon(n)$ where $\varepsilon(n)$ is the negligible function for some n.

Definition 1 (HMAC Safety)**.** *A hash-based message authentication code defined as* $\mathsf{HMAC}(K,M) = H(K \oplus \mathsf{opad}, H(K \oplus \mathsf{ipad}, M))$ *outputs a* tag *using a symmetric secret K for data M, where*

- *ipad=64 times 0x36 and opad=64 times 0x5c*

It holds the following properties:

- *It is infeasible for any* $\mathcal{A}$ *to retrieve* M_1 *from* $tag_1 = \mathsf{HMAC}(K, M_1)$ *in polynomial time t.*
- *For any* $M_1 \neq M_2 \bigwedge |M_1| = |M_2|$, *we have* $\varepsilon(n) \geq \Pr[\mathcal{A}(M_1,M_2)|\mathsf{HMAC}(K,M_1) = \mathsf{HMAC}(K,M_2)]$.

2.2 Notion of Lattice and Assumptions

Let $\boldsymbol{x} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\}$ be a set of linearly independent vectors in m-dimensional Euclidean space $\mathcal{R}^m$ that forms a lattice $\mathcal{X}(\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n) = \sum_{i=1}^{n} \alpha_i \boldsymbol{x}_i$ such that $\alpha_{i\in[1,n]} \in Z$ where $\boldsymbol{x}_{i\in[1,n]} \in \mathcal{R}^m$ are known as basis vectors, and (m,n) be the (dimension,rank) of $\mathcal{X}$, respectively (Lyubashevsky and Micciancio, 2018). The lowest distance of $\mathcal{X}$ is $d_{\min}(\mathcal{X})$=$\min_{\boldsymbol{x}\in\mathcal{X}\setminus\{0\}} \|\boldsymbol{x}\|$. A linearly independent vector set outputs $\mathcal{X}$, forming its basis. The basis vectors support: a) *every* $\mathcal{X}$ *has at least one basis*, and b) *basis of* $\mathcal{X}$ *is not unique*.

Definition 2. *Let* $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in Z^{m\times n}$ *be a basis matrix of a lattice where basis vectors are placed in the column of* $\boldsymbol{X}$. *Lattice* $\mathcal{X}$ *in a m-dimensional Euclidean space* $\mathcal{R}^m$ *is denoted as* $\mathcal{X}(\boldsymbol{X}) = [\boldsymbol{X}\boldsymbol{t} : \boldsymbol{t} \in Z^n]$ *where* $\boldsymbol{X}\boldsymbol{t}$ *represents a matrix-vector multiplication.*

Post-quantum structures often use *q-ary* lattices on implementation. For integer q, a lattice $\mathcal{X}$ ($Z_q^n \subseteq \mathcal{X} \subseteq Z^n$) supports modular arithmetic.

Definition 3. *For* $\boldsymbol{X} \in Z_q^{m\times n}$ *under modulo q, two different q-ary lattices* $\mathcal{X}_q^{\perp}$ *and* $\mathcal{X}_q$ *can be defined:*

a) $\mathcal{X}_q^{\perp} = \{\boldsymbol{t} \in Z^n \mid \boldsymbol{X}\boldsymbol{t} = \boldsymbol{0} \mod q\}$,

b) $\mathcal{X}_q = \{\boldsymbol{t} \in Z^n,\ \boldsymbol{u} \in Z^m \mid \boldsymbol{t} = \boldsymbol{X}^T\boldsymbol{u} \mod q\}$

Definition 4 (Inhomogeneous Small Integer Solution (ISIS))**.** *Given* $\boldsymbol{X} \in Z_q^{m\times n}$, *an integer constant* α *and a random vector* $\boldsymbol{u} \in Z_q^m$, *finding a vector* $\boldsymbol{t} \in Z^n \setminus \{0\}$ *where* $\|\ \boldsymbol{t}\ \| < \alpha$ *and* $\boldsymbol{X}\boldsymbol{t} = \boldsymbol{u} \mod q$ *is infeasible.*

Definition 5 (Discrete Gaussians)**.** *Given standard deviation* σ *and* $c \in Z_q^n$, *the discrete Gaussian distribution is* $\mathcal{D}_{\mathcal{X},\sigma,c}(x) = \rho_{\sigma,c}(x)/\rho_{\sigma,c}(\mathcal{X}), \forall x \in \mathcal{X}$, *where* $\rho_{\sigma,c}(x)$ *is a Gaussian function on* Z^n *centered at c.*

Ring Learning with Error. The ring learning with errors (RLWE) is built on the polynomials arithmetic with coefficients in a finite field. A polynomial ring is defined as $\mathcal{P}_q = Z_q[x]/\langle p(x)\rangle$ with a narrow $\mathcal{D}$ of zero mean over Z, where $p(x) = \sum_{i=0}^{n} p_i x^i$ is an irreducible polynomial. For fast execution, we set $p(x) = x^n + 1$ and $q \equiv 1 \bmod 2n$. Assume $a_i \in \mathcal{P}_q$ is a set of random but *known* polynomials with coefficients in Z_q and $e_i \in \mathcal{P}$ is a set of random but *unknown* small polynomials with coefficients in $\mathcal{D}$. Let $s \in \mathcal{P}_q$ be a *unknown* polynomial where $b_i \in \mathcal{P}_q$ such that $b_i = a_i \cdot s + e_i$.

Definition 6 (Search-RLWE)**.** *Given* (a_i, b_i), *finding unknown s for any polynomial-time bounded adversary* $\mathcal{A}$ *is computationally infeasible. The advantage of* $\mathcal{A}$ *in breaching the Search-RLWE is defined as*

$$\Pr\left[a_i \in \mathcal{P}_q, e_i \in \mathcal{P}; s \leftarrow \mathcal{A}(a_i,b_i) | b_i = a_i \cdot s + e_i\right] \geq \varepsilon$$

Definition 7 (Decisional-RLWE)**.** *Given* (a_i, b_i), *deciding whether* $b_i = a_i \cdot s + e_i$ *or a random* $b_i \in \mathcal{P}_q$ *for any* $\mathcal{A}$ *is infeasible. The advantage of* $\mathcal{A}$ *in breaching the Decisional-RLWE is defined as*

$$|\Pr[\mathcal{A}(a_i, a_i \cdot s + e_i)] - \Pr[\mathcal{A}(a_i, b_i)]| \geq \varepsilon$$

2.3 Details of SETLA Scheme

The proposed QuDPas-FHA protocol uses some functions, such as $ENC[\cdot,\cdot]$, $DEC[\cdot,\cdot]$, $SetlaSC[\cdot,\cdot,\cdot]$ and $SetlaUSC[\cdot,\cdot,\cdot]$ under SETLA specification (Gérard and Merckx, 2018) which is discussed as below.

- **KeyGen[$\mathbf{a}_1, \mathbf{a}_2$]:** It declares two parameters $\mathbf{a}_1$ and $\mathbf{a}_2$ as public variables. Then, it generates $\mathbf{s}$, $\mathbf{e}_1$, and $\mathbf{e}_2$ at random from $\mathcal{P}_{q,[1]}$, where $\mathbf{s}$ represents the secret parameter, and $\mathbf{e}_1$ and $\mathbf{e}_2$ denote the noise parameters. Next, it computes $\mathbf{t}_1 \leftarrow \mathbf{a}_1 \cdot \mathbf{s} + \mathbf{e}_1$ and $\mathbf{t}_2 \leftarrow \mathbf{a}_2 \cdot \mathbf{s} + \mathbf{e}_2$ and outputs the key pair $(\mathrm{PK} = (\mathbf{t}_1, \mathbf{t}_2), \mathrm{SK} = (\mathbf{s}, \mathbf{e}_1, \mathbf{e}_2))$.
- **SetlaSC[$\mathbf{PK_b}, \mathbf{SK_a}, \mathbf{m}$]:** It takes input receiver's public key pk_b, sender's keys $(\mathbf{s}_a, \mathbf{e}_{\mathbf{a},1}, \mathbf{e}_{\mathbf{a},2}, \mathrm{PK}_a)$, a message m, a random oracle $H : * \rightarrow \{\mathbf{v} \mid \mathbf{v} \in \mathcal{P}_{q,[1]}, \|\mathbf{v}\|_1 = \omega\}$, and a symmetric encryption algorithm $SE(\cdot)$. First, it chooses $K \xleftarrow{r} \{0,1\}^{256}$ and $\mathbf{y} \xleftarrow{r} \mathcal{P}_{q\cdot[B]}$. Next, it using the random oracle to compute $\mathbf{c} \leftarrow H(\lfloor \mathbf{a}_1 \cdot \mathbf{y} \rceil_d, \lfloor \mathbf{a}_2 \cdot \mathbf{y} \rceil_d, m, K, \mathbf{pk}_a, \mathbf{pk}_b)$. Besides it calculates $\mathbf{z} \leftarrow$

$\mathbf{s}_a \cdot \mathbf{c} + \mathbf{y}$. After that, it generates $\mathbf{w}_1 = \mathbf{a}_1 \cdot \mathbf{y} - \mathbf{e}_{\mathbf{a},1} \cdot \mathbf{c}$ and $\mathbf{w}_2 = \mathbf{a}_2 \cdot \mathbf{y} - \mathbf{e}_{\mathbf{a},2} \cdot \mathbf{c}$. It verifies whether $\mathbf{z} \notin \mathcal{P}_{q,[B-\omega]}$ and $\lfloor \mathbf{a}_1 \cdot \mathbf{y} \rfloor_d \neq \lfloor \mathbf{w}_1 \rfloor_d$ and $\lfloor \mathbf{a}_2 \cdot \mathbf{y} \rfloor_d \neq \lfloor \mathbf{w}_2 \rfloor_d$. Now, it selects $\mathbf{y}' \xleftarrow{r} \mathcal{P}_{q,[B]}$ and derives $\mathbf{x} \leftarrow \mathbf{t}_{b,1} \cdot \mathbf{y} + \mathbf{y}' + \mathsf{Encode}(K)$. Finally, it computes $\varepsilon = SE(K,m)$. The output is $C = (\mathbf{z}, \mathbf{c}, \mathbf{x}, \varepsilon)$.

- **SetlaUSC[$\mathbf{SK_b}, \mathbf{PK_a}, \mathbf{m}$]:** It takes input receiver's key $(\mathbf{s}_b, \text{PK}_b)$, sender's public key PK_a, the signcryptext $C = (\mathbf{z}, \mathbf{c}, \mathbf{x}, \varepsilon)$, a random oracle $H : * \to \{\mathbf{v} \mid \mathbf{v} \in \mathcal{P}_{q,[1]}, \|\mathbf{v}\|_1 = \omega\}$, and a symmetric decryption SD. It calculates $\mathbf{w}_1 \leftarrow \mathbf{a}_1 \cdot \mathbf{z} - \mathbf{t}_{\mathbf{a},1} \cdot \mathbf{c}$ and $\mathbf{w}_2 \leftarrow \mathbf{a}_2 \cdot \mathbf{z} - \mathbf{t}_{\mathbf{a},2} \cdot \mathbf{c}$. Next, it decodes ε as $K = \mathsf{Decode}(\mathbf{x} - \mathbf{w}_1 \cdot \mathbf{s}_b)$. Finally, it returns $m = SD(K, \varepsilon)$ and if $\mathbf{c} = H(\mathbf{v}, m, K, \text{PK}_a, \text{PK}_b)$ and $\mathbf{z} \in \mathcal{P}_{q,[k-\omega]}$ hold; otherwise, returns $\perp$.
- **ENC[$\mathbf{PK}, m$]:** Define the public key $\text{PK} = \mathbf{a} \cdot \mathbf{s} + \mathbf{e}$, where $\mathbf{s}, \mathbf{e} \xleftarrow{r} \mathcal{D}$. Then, it samples $\mathbf{y}_1$, $\mathbf{y}_2$, and $\mathbf{y}_3$ from $\mathcal{D}$ and computes $\mathbf{c}_1 \leftarrow \mathbf{a} \cdot \mathbf{y}_1 + \mathbf{y}_2$ and $\mathbf{c}_2 \leftarrow \text{PK} \cdot \mathbf{y}_1 + \mathbf{y}_3 + \mathsf{Encode}(m)$. The output is $(\mathbf{c}_1, \mathbf{c}_2)$.
- **Encode[m]:** Given message $m = [m_1, \cdots, m_n]$, it element-wise encodes as $\mathbf{m}[i] = m[i] \cdot \lfloor \frac{q-1}{2} \rfloor$.
- **DEC[$\mathbf{c}_1, \mathbf{c}_2, \mathbf{SK}$]:** Define secret key $\mathbf{SK} = \mathbf{s}$, where $\mathbf{s} \xleftarrow{r} \mathcal{D}$. It computes $\mathbf{m} = \mathbf{c}_2 - \mathbf{c}_1 \cdot \mathbf{s} \approx \mathsf{Encode}(m)$. Next, it applies $\text{Decode}(\mathbf{m})$ to recover original m.
- **Decode[m]:** It checks whether each $\mathbf{m}[i]$ lies within the interval $[-\lceil \frac{q}{4} \rceil, \lceil \frac{q}{4} \rceil - 1]$. If yes, it sets $m[i] = 1$. Otherwise, declares $m[i] = 0$.

Now, we introduce our QuDPas-FHA to address, "Could we design a fast, anonymous authentication protocol for the SIN, allowing secure handover authentication while resisting quantum attacks?"

3 OUR CONSTRUCTION

This section introduces the system model and our QuDPas-FHA with essential notations in Table 1.

Table 1: List of useful notations.

Notation	Description
ID_i	Identity of entity i
PK_{x_i}, SK_{x_i}	Public and secret key of the SIN entity x_i.
q	Rational integer modulus ≥ 2
$\mathcal{D}$	Discrete Gaussian distribution
$\mathcal{P}_{q,[1]}$	A polynomial ring $Z_q[X]/\langle X^n+1\rangle$ with a narrow $\mathcal{D}$, $q \equiv 1 \bmod 2n$, and coefficients in the range $[-1,1]$.
$v \xleftarrow{r} \mathcal{D}$	v is sampled from $\mathcal{D}$
$F_{ms}(\cdot)$	Secure trapdoor: $\{0,1\} \times \mathcal{P}_{q,[1]} \to \mathcal{P}_{q,[1]}$
$H(\cdot)$	Hash oracle $\{0,1\}^* \to \{\mathbf{v} \in \mathcal{P}_{q,[1]}, \|\mathbf{v}\|_1 = \omega\}$
$\mathcal{L}_{s_j}$	List of users receive services via satellite S_j
$A \oplus B$	Bit-wise XOR operation A and B
$A\|B$	Concatenation between A and B
HMAC$[x,M]$	Outputs a *tag* for input secret x and data M
SE$[K,M]$, SD$[K,C]$	Symmetric encryption and decryption of M and C for secret x
ENC$[PK_{x_i},M]$, DEC$[SK_{x_i},C]$	Public-key Encryption and decryption of M and C for key PK_{x_i} and SK_{x_i}, respectively
SetlaSC $[SK_{x_i}, PK_{y_j}, M]$	SETLA signcryption of M with keys SK_{x_i} and PK_{y_j}
SetlaUSC $[SK_{y_i}, PK_{x_j}, C]$	SETLA unsigncryption of C with keys SK_{x_i} and PK_{y_j}

3.1 System Model Description

Our QuDPas-FHA ensures uninterrupted service to users within a single SIN network, ensuring generality while roaming within the network. The system model comprises five entities, as explained below.

- *Trusted Third Party (TTP)*: The TTP is crucial in managing public-private key pairs for GSs and users across multiple domains, ensuring identity verification and securing information exchange channels through the use of cryptographic keys.
- *Network Control Center (NCC)*: The operator segment network domain is managed by this entity, which allows users to access the network through registration and certification processes.
- *Ground Station (GS)*: It establishes connectivity via terrestrial networks and provides an interface for ground-based LEO satellite access.
- *Low Earth Orbit (LEO) Satellites*: LEO satellites are the endpoints for users connecting to the network, and recent improvements in satellite manufacturing technology help execute complex tasks.
- *Users*: It accesses the network via LEO satellites, enabling data sharing and exchange, thereby fostering a resilient digital communication platform.

As depicted in Figure 3, the TTP oversees system policy, maintaining exclusive access to its primary secret. It gives a high-entropy secret key to NCC, which is crucial for securely registering SIN entities. A mutual authentication process is initiated when a user wants to access services from ground station GS_k via a nearby satellite S_j. The user (U_i) initiates anonymous communication by connecting to the proximate S_j via a predetermined protocol. S_j assists U_i generating session-dependent tokens, building a secure communication with U_i and GS_k. Next, GS_k verifies U_i and negotiates a session key $UGSK$. Once $UGSK$ is negotiated, S_j acts as a relay for data exchange between U_i and GS_k. For effective service management, each S_j maintains its list $\mathcal{L}_{s_j}$ of active users. Due to its roaming nature, S_j may move out of the communication range of U_i or GS_k during an active session c. If so, and c still is active, then S_j forwards its latest $\mathcal{L}_{s_j}$ to a new S_j^{new} via ISL (protected with a pre-negotiated key) to continue service without re-authenticating U_i. Next, S_j^{new} updates its list as $\mathcal{L}_{s_j}^{new} = \mathcal{L}_{s_j}^{new} \bigcup \mathcal{L}_{s_j}$. When S_j^{new} comes in the range, U_i sends ESL-free tokens to prove its authenticity. If

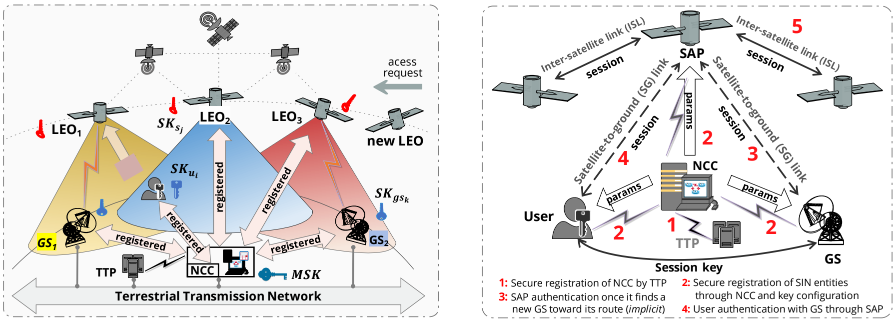

Figure 3: QuDPas-FHA system model overview showing crypto keys distribution for SIN entities (first part) and the communication for secure authentication (second part).

S_j^{new} finds U_i as an already authenticated user through $\mathcal{L}_{s_j}^{new}$, it act as a relay between U_i and GS_k.

3.2 The QuDPas-FHA Protocol

Figure 4 depicts tokens exchange between U_i, S_j, and GS_k. Our protocol comprises five phases: *NCC setup*, *entity registration*, *authentication and key agreement*, *user management as LEO constellation*, and *fast handover authentication*, each of which is discussed now.

3.2.1 NCC Setup

On input security parameter 1^γ, the TA selects $\mathbf{ms} = \mathbf{s} \in \mathcal{P}_{q,[1]}$ and chooses two polynomials $\mathbf{a}_1, \mathbf{a}_2 \in \mathbb{Z}_q[X]$. Next, it chooses a trapdoor function secured with $\mathbf{ms}$ as $F_{ms} : \{0,1\} \times \mathcal{P}_{q,[1]} \rightarrow \mathcal{P}_{q,[1]}$ and sets a random oracle as $H : \{0,1\}^* \rightarrow \{\mathbf{v} \mid \mathbf{v} \in \mathcal{P}_{q,[1]}, \|\mathbf{v}\|_1 = \omega\}$. Next, it declares the symmetric encryption and decryption as $\mathsf{SE}[\cdot,\cdot]$ and $\mathsf{SD}[\cdot,\cdot]$. Besides, it considers the Public-key encryption and decryption $\mathsf{ENC}[\cdot,\cdot]$ and $\mathsf{DEC}[\cdot,\cdot]$, and the singcryption and unsigncryption $\mathsf{SetlaSC}[\cdot,\cdot,\cdot]$ and $\mathsf{SetlaUSC}[\cdot,\cdot,\cdot]$, which works as shown in Section 2.3. Finally, it declares the global public parameter as $params = (\mathbf{a}_1, \mathbf{a}_2, H)$ while save the master secret key $MSK = (\mathbf{ms}, F)$ securely.

3.2.2 Entity Registration

Upon obtaining user U_i details such as identity ID_i and other essential proofs, *NCC* runs $\mathsf{UKeyGen}$ which returns a partial-private key $\mathbf{s}_{u_i} \leftarrow F(\mathbf{ms}, ID_i)$. Upon receiving $\mathbf{psk} = \mathbf{s}_{u_i}$, U_i runs $\mathsf{FullKEY}$ process that generates noise parameters $\mathbf{e}_{u_i,1}, \mathbf{e}_{u_i,2} \in \mathcal{P}_{q,[1]}$ at random. Next, it computes $\mathbf{t}_{u_i,1} \leftarrow \mathbf{a}_1 \cdot \mathbf{s}_{u_i} + \mathbf{e}_{u_i,1}$ and $\mathbf{t}_{u_i,2} \leftarrow \mathbf{a}_2 \cdot \mathbf{s}_{u_i} + \mathbf{e}_{u_i,1}$. Finally, It sets its full-public-key $PK_{u_i} = \langle \mathbf{t}_{u_i,1}, \mathbf{t}_{u_i,2} \rangle$ and full-secret-key $SK_{u_i} = \langle \mathbf{s}_{u_i}, \mathbf{e}_{u_i,1}, \mathbf{e}_{u_i,2} \rangle$. U_i stores SK_{u_i} in the private space securely.

Following the similar tasks, satellite S_j processes its keys $PK_{s_j} = \langle \mathbf{t}_{s_j,1}, \mathbf{t}_{s_j,2} \rangle$ and $SK_{s_j} = \langle \mathbf{s}_{s_j}, \mathbf{e}_{s_i,1}, \mathbf{e}_{s_i,2} \rangle$.

Besides, ground station GS_k ensures its key-pair as $PK_{gs_k} = \langle \mathbf{t}_{gs_k,1}, \mathbf{t}_{gs_k,2} \rangle$ and $SK_{gs_k} = \langle \mathbf{s}_{gs_k}, \mathbf{e}_{gs_k,1}, \mathbf{e}_{gs_k,2} \rangle$.

On successful registration, the credentials of SIN entities are $U_i : (PK_{u_i}, SK_{u_i})$, $S_j : (PK_{s_j}, SK_{s_j}, \mathcal{L}_{s_j})$ and $GS_k : (PK_{GS_k}, SK_{GS_k})$, where S_j maintains a list $\mathcal{L}_{s_j}$ of active users receive satellite services through S_j.

3.2.3 Entity Authentication and Key Agreement

Entities U_i, S_j, and GS_k interact among themselves to generate a session key following the steps below:

1. **User to Satellite** $\left(\mathbf{U_i} \xrightarrow{\mathbf{C_2}} \mathbf{S_j}\right)$**:** To initiate a secure communication, U_i proves the authenticity. For this, U_i selects a nonce n_u as ephemeral secret and signcrypts n_u with its secret key SK_{u_i} and GS_k's PK_{GS_k} as $C_1 = \mathsf{SetlaSC}[PK_{GS_k}, SK_{u_i}, n_u]$. Then, it sends $C_2 = \mathsf{ENC}[PK_{S_j}, C_1 \| PK_{u_i}]$ to S_j.

2. **Satellite to Ground Station** $\left(\mathbf{S_j} \xrightarrow{\mathbf{C_3}} \mathbf{GS_k}\right)$: On receiving a request from U_i, satellite S_j decrypts with its secret SK_j as $(C_1' \| PK_{u_i}') = \mathsf{DEC}[SK_{S_j}, C_2]$. Next, it selects a nonce n_s and generates a trapdoor C_3 for GS_k with PK_{GS_k} as
$$C_3 = \mathsf{ENC}[PK_{GS_k}, (n_S \| C_1' \| PK_{u_i}')] \quad (1)$$
and sends C_3 to GS_k.

3. **Ground Station to Satellite** $\left(\mathbf{GS_k} \xrightarrow[\langle \mathbf{I_1}, \mathbf{I_2} \rangle, \mathbf{I_{gu}}]{\mathbf{C_4}} \mathbf{S_j}\right)$**:** On receiving C_3, GS_k decrypts C_3 with its secret key SK_{GS_k} and retrieves certain parameters as
$$(n_S' \| C_1' \| PK_u') = \mathsf{DEC}[SK_{GS_k}, C_3] \quad (2)$$
$$n_u = \mathsf{SetlaUSC}[SK_{GS_k}, PK_{u_i}', C_1'] \quad (3)$$
Note that function $\mathsf{SetlaUSC}[\cdot,\cdot,\cdot]$ fails indicates a tampered communication by a potential foe. Else, it confirms the user authenticity via relay S_j. To do this, it chooses nonce n_{gs} and performs
$$I_1 = n_{gs} \oplus n_u \oplus n_S' \quad (4)$$
$$C_4 = \mathsf{SetlaSC}[PK_{u_i}', SK_{GS_k}, n_{gs}] \quad (5)$$
$$I_2 = \mathsf{HMAC}[n_s', C_4] \quad (6)$$

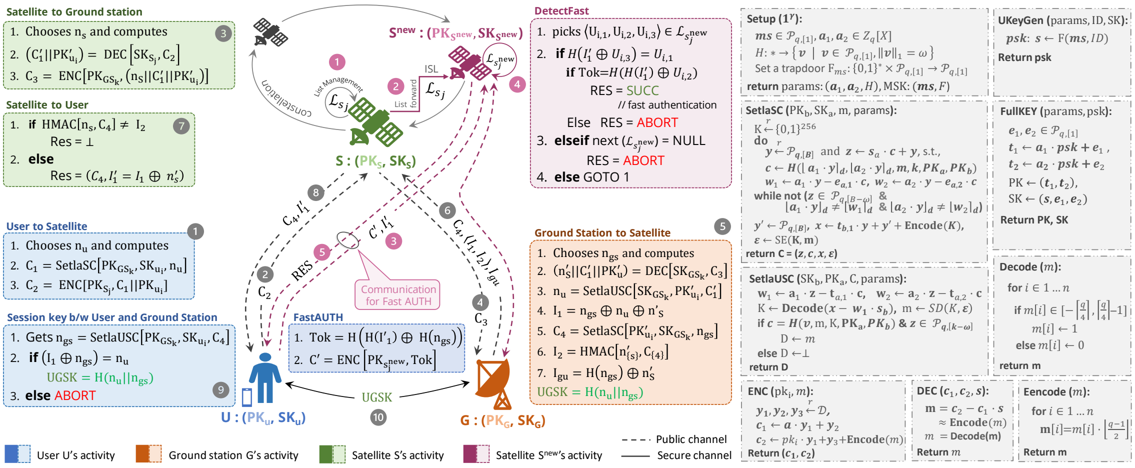

Figure 4: SIN entity authentication and fast handover details in the QuDPas-FHA protocol.

Finally, it sends $(C_4, \langle I_1, I_2\rangle, I_{gu} = H(n_{gs}) \oplus n'_S)$ and an acknowledgment $SUCC_{gsu}$ to S_j. The GS_k considers the session key as $UGSK = H(n_u || n_{gs})$.

4. **Satellite to User** $(\mathbf{S_j} \xrightarrow{\mathbf{C_4, I'_1}} \mathbf{U_i})$**:** On receiving $(C_4, \langle I_1, I_2\rangle)$, S_j recalls n_s. If $\mathsf{HMAC}[n_s, C_4] = I_2$, then S_j forwards $(C_4, I'_1 = I_1 \oplus n'_s)$ to U_i.
5. **Session key between User and Ground Station** $(\mathbf{U_i} \xleftrightarrow{\mathbf{UGSK}} \mathbf{GS_k})$**:** Upon receiving C_4 from S_j, user U_i unsigncrypts nonce n_{gs} as
$$n_{gs} = \mathsf{SetlaUSC}[PK_{GS_k}, SK_{u_i}, C_4] \quad (7)$$
Next, it checks whether $I_1 \oplus n_{gs} \stackrel{?}{=} n_u$. If it hods, the authenticity of GS_k via relay S_j is confirmed, thus, U_i sets the session key $UGSK = H(n_u || n_{gs})$. Finally, it sends an acknowledgment $SUCC_{ugs}$ to S_j while holding (I'_1, n_{gs}, SK) for this session.

3.2.4 User Management at LEO Constellation

Effective handover eliminates the requirement to reauthenticate with GS_k through below steps.

1. **User Management** $(S_j \Leftarrow \circlearrowleft \mathcal{L}_{s_j})$**:** If S_j receives both $SUCC_{gsu}$ and $SUCC_{ugs}$, it adds U_i in its list $\mathcal{L}_{s_j} = \mathcal{L}_{s_j} \cup \{U_{i,1}, U_{i,2}, U_{i,3}\}$ where $U_{i,1} = H(I_1)$, $U_{i,2} = I_{gu} \oplus n_s$, $U_{i,3} = n_s$. S_j periodically checks for the user's session expiration. If it is expired based on a threshold time limit, S_j may remove the added entry for specific U_i from its list $\mathcal{L}_{s_j}$.
2. **List Forward** $(\mathbf{S_j} \xrightarrow{\mathcal{L}_{s_j}} \mathbf{S_j^{new}})$**:** When S_j cannot serve U_i owing to departing its range, it checks whether the connected users $U = \{U_i\}$ still want to connect with GS_k. If it does not receive any willingness (say, "YES") from some users, say $U' \subseteq U$, it removes each $U_j \in U'$ entries form $\mathcal{L}_{s_j}$. For $\mathcal{L}_{s_j} \neq$ null, S_j sends $\mathcal{L}_{s_j}$ to the next satellite S_j^{new} of its LEO constellation via ISL transmission. Finally, S_j^{new} adds those active users' details in its list $\mathcal{L}_{s_j}^{new} = \mathcal{L}_{s_j}^{new} \bigcup \mathcal{L}_{s_j}$. A confirmed addition disrupts service for all $U_i \in U \setminus U'$ from S_j.

3.2.5 Fast Handover Authentication

This phase begins when S_j cannot serve U_i owing to its communication range. A new S_j^{new} from its LEO constellation will provide services to U_i. For our case, $U_i \in U \setminus U'$. In a typical scenario, U_i must reauthenticate to continue services from GS_k. However, the QuDPas-FHA does not reauthenticate users. Note that S_j must execute List Forward since U_i already verified its authenticity to S_j and its session has not timed out. To avoid reauthentication, we enable a fast authentication. Recall, U_i holds (I'_1, n_{gs}, SK) and sets
$$Tok = H(H(I'_1) \oplus H(n_{gs})) \quad (8)$$
$$C' = \mathsf{ENC}[PK_{s_j^{new}}, Tok] \quad (9)$$
and sends (C', I'_1) to S_j^{new}. On receiving it, SK_j^{new} decrypts as $Tok = \mathsf{DEC}[SK_j^{new}, C']$. Then, it executes

DetectFast $(\mathcal{L}_{s_j}^{new})$**:**

S1: picks a tuple $\langle U_{i,1}, U_{i,2}, U_{i,3}\rangle \in \mathcal{L}_{s_j}^{new}$.
S2: If $H(I'_1 \oplus U_{i,3}) = U_{i,1}$, then
  S2.1: Retrieves corresponding $U_{i,2}$ from $\mathcal{L}_{s_j}^{new}$.
  S2.2: If $Tok = H(H(I'_1) \oplus U_{i,2})$, then
    – fast authentication "SUCC".
  S2.3: Else, "ABORT" session.
S3: Else, i.e., $H(I'_1 \oplus U_{i,3}) \neq U_{i,1}$,
  S3.1: If no tuple left in $\mathcal{L}_{s_j}^{new}$, S_j^{new} unlinks U_i.
  S3.2: Else, GOTO step S1.

This completes the description of the QudPas-FHA protocol. Now, we provide the security discussion.

4 SECURITY ANALYSIS

The proposed QudPas-FHA protocol achieves several security properties as mentioned in Section 1.1.

4.1 Threat Model and Assumptions

Satellite communications over unsecured links are exposed to cyber threats by hostile users and active foes ($\mathcal{A}$). $\mathcal{A}$ can be broadly classified into two categories: a) *insiders* with authorized data access, difficult to trace, and b) *outsiders* with a lesser consequence. The QuDPas-FHA targets both of these foes. Besides, *existential unforgeability* (EUF) assures that no one can impersonate a legitimate SIN entity to obtain satellite services without the respective secret key. Further, $\mathcal{A}$ employs a *chosen-ciphertext attack* (CCA) to breach the system's integrity. Our protocol meets each aspect above based on the following premises:

- *Reduction capacity:* QuDPas-FHA algorithms are public. $\mathcal{A}$ with quantum analysis must determine the prequisite to solve RLWE and HMAC.
- *Channel security:* Data over the private channel is impenetrable by $\mathcal{A}$. Conversely, $\mathcal{A}$ can intercept, delete, and modify data through an open channel.
- *Key safety:* The user U_i maintains the key security and ensures that stored data retains its integrity in the presence of computational capabilities.
- *Forward/Backward secrecy:* $\mathcal{A}$ may read keys from a device but must be traced or denied altering session key building to avoid full invasion.
- *Security level:* NCC's activity is susceptible to monitoring by $\mathcal{A}$, who may also be able to manipulate the credibility of user key generation under security Level-1 (Girault, 1991).
- *Safety basics:* All the conventional cryptographic primitives are secure, thus ensuring that none acquires a non-negligible advantage.

4.2 Security Properties

Theorem 1. *The SETLA is $(\varepsilon',t')-$IND-SC-CCA safe against $\mathcal{A}=(\mathcal{A}_1,\mathcal{A}_2)$ in the ROM under the $(\varepsilon,t)-$Decisional-RLWE assumption, where $\varepsilon'<\varepsilon$ and $t'>t$.*

The confidentiality of SETLA can be shown with a sequence of games $\mathsf{Game}_0 \sim \mathsf{Game}_3$ showing semantic security as mentioned in (Gérard and Merckx, 2018).

Theorem 2. *Under the RLWE and HMAC assumptions, the QuDPas-FHA protocol meets critical security traits, including quantum safe authentication.*

Proof. The theorem follows when Lemmas 1-6 according to Definitions 1-6 and Theorem 1 are hold. □

Lemma 1 (F1: Quantum Safety). *The QudPas-FHA resists quantum attack under the random oracle model based on the RLWE and HMAC assumptions.*

Proof. The QuDPas-FHA resists quantum attacks under the SETLA and HMAC framework. Based on Theorem 1, a quantum capable $\mathcal{A}$ fails launching IND-CCA for nonces $n_{u,0}$ and $n_{u,1}$. Under Game_0, the simulator $\mathcal{S}$ chooses a binary coin b and signcrypt $n_{u,b}$ as the original signcryption process. In Game_1, $\mathcal{S}$ just change $\mathbf{z} \xleftarrow{r} \mathcal{P}_{q,[B-\omega]}$ rather $\mathbf{z} \leftarrow \mathbf{s}_a \cdot \mathbf{c} + \mathbf{y}$. In this *rejection sampling* ($\mathsf{Game}_0 \rightarrow \mathsf{Game}_1$), $\mathcal{A}$ achieves a negligible advantage ε_{01}. Now, in Game_2, $\mathcal{S}$ changes $\mathbf{x} \leftarrow \mathbf{a}' \cdot \mathbf{y} + \mathbf{y}' + \mathsf{Encode}(K)$ for $\mathbf{a}' \xleftarrow{r} \mathcal{P}_q$ instead of $\mathbf{x} \leftarrow \mathbf{t}_1 \cdot \mathbf{y} + \mathbf{y}' + \mathsf{Encode}(K)$ for public $\mathbf{t}_1$. In this decisional-RLWE ($\mathsf{Game}_1 \rightarrow \mathsf{Game}_2$), $\mathcal{A}$ gets a negligible advantage ε_{12}. Similarly, in Game_3, $\mathcal{S}$ updates $\mathbf{x} \xleftarrow{r} \mathcal{P}_q$ instead of $\mathbf{x} \leftarrow \mathbf{a}' \cdot \mathbf{y} + \mathbf{y}' + \mathsf{Encode}(K)$. Like before, $\mathcal{A}$ has ε_{12} in $\mathsf{Game}_2 \rightarrow \mathsf{Game}_3$ due to the DRLWE. Besides, $\mathcal{A}$ has a negligible advantage for HMAC due to its strong collision-resistance trait with secret K. Note that combining these operations does not provide a non-negligible advantage for $\mathcal{A}$. Thus, the QuDPas-FHA ensures quantum-safety trait. □

Lemma 2 (F2: Level-II Security). *The QudPas-FHA ensures Girault's Level-II safety, avoiding key-escrow issues in satellite communication.*

Proof. Most authentications in satellite communication require a fully-trusted server to generate key-pair (PK_{u_i}, SK_{u_i}). Modern approaches can avoid this issue by choosing the public PK_{u_i} and a fully-trusted third party (TP) yielding the secret SK_{u_i}. However, finding SK_{u_i} allows TP to act as a genuine user without being traced. Thus, TP can decrypt the cipher by providing the user's SK_{u_i}. Thus, relying on a fully-trusted TP for full-key computation could grant $\mathcal{A}$ accesses to a backdoor. QuDPas-FHA views NCC a semi-trusted entity. For each SIN entity, NCC generates a partial-private-key $\mathbf{psk} = \mathbf{s}_{u_i}$ as $\mathbf{s}_{u_i} \leftarrow F(\mathbf{ms}, ID_i)$. Afterward, U_i finds the full-secret key as $SK_{u_i} = \langle \mathbf{s}_{u_i}, \mathbf{e}_{u_i,1}, \mathbf{e}_{u_i,2} \rangle$ and full-public key $PK_{u_i} = \langle \mathbf{t}_{u_i,1} \leftarrow \mathbf{a}_1 \cdot \mathbf{s}_{u_i} + \mathbf{e}_{u_i,1}, \mathbf{t}_{u_i,2} \leftarrow \mathbf{a}_2 \cdot \mathbf{s}_{u_i} + \mathbf{e}_{u_i,1} \rangle$ using $\mathbf{psk}$. Thus, even if $\mathcal{A}$ gets control of NCC, it cannot find SK_{u_i} needed for transmission, ensuring Girault's Level-II security. □

Lemma 3 (F3: Mutual Authentication). *QudPas-FHA ensures robust mutual authentication between users and the ground station relying satellites.*

Proof. Initially, U_i sends C_2=$\mathsf{ENC}[PK_{S_j}, C_1 \| PK_{u_i}]$ where C_1=$\mathsf{SetlaSC}[PK_{GS_k}, SK_{u_i}, n_u]$ for nonce n_u. Due to hardness of RLWE according to Definition 6, $\mathcal{A}$ cannot reveal C_1 and PK_{u_i} from C_2. Later, GS_k gets C_3=$\mathsf{ENC}[PK_{GS_k}, (n_S \| C_1' \| PK_{u_i}')]$ from S_j. Now, GS_k

retrieves n_u if $\mathsf{SetlaUSC}[SK_{GS_k}, PK'_{u_i}, C'_1]$ is successful. Indeed, successful unsigncryption requires input PK'_{u_i} to confirm the authenticity of C'_1. Similarly, when $\mathsf{SetlaUSC}[PK_{GS_k}, SK_{u_i}, C_4] \neq \perp$, U_i confirms GS_k's authenticity. Note, the underlying signature of $\mathsf{SetlaSC}$ is unforgeable (Bai and Galbraith, 2014). Under the *forking lemma*, $\mathcal{A}$ outputs two forgeries $(\mathbf{z}, \mathbf{c})$ and $(\mathbf{z}', \mathbf{c}')$ for distinct random oracles but the same random tape (thus, same q). Now (for simplicity, arguing for one RLWE sample instead of two in signature) $\lfloor \mathbf{a}_1 \cdot \mathbf{z} - \mathbf{t}_{u_i,1} \cdot \mathbf{c} \rceil_d = \lfloor \mathbf{a}_1 \cdot \mathbf{z}' - \mathbf{t}_{u_i,1} \cdot \mathbf{c}' \rceil_d = \lfloor \mathbf{a}_1 \cdot \mathbf{y} \rceil_d$. For small $\mathbf{e}$, $\mathbf{a}_1 \cdot \mathbf{z} - \mathbf{t}_{u_i,1} \cdot \mathbf{c} = \mathbf{a}_1 \cdot \mathbf{z}' - \mathbf{t}_{u_i,1} \cdot \mathbf{c}$. For $\mathbf{t}_{u_i,1} = \mathbf{a}_1 \cdot \mathbf{s}_{u_i} + \mathbf{e}_{u_i,1}$, we have $\mathbf{a}_1 \cdot (\mathbf{z} - \mathbf{z}' - \mathbf{s}_{u_i,1} \cdot \mathbf{c} + \mathbf{s}_{u_i,1} \cdot \mathbf{c}') + (\mathbf{e}_{u_i,1} \cdot (\mathbf{c}' - \mathbf{c}) + \mathbf{e}) = 0$. As shown in Section 4.2 (Bai and Galbraith, 2014), if $\mathbf{z} - \mathbf{z}' - \mathbf{s}_{u_i,1} \cdot \mathbf{c} + \mathbf{s}_{u_i,1} \cdot \mathbf{c}'$ and $\mathbf{e}_{u_i,1} \cdot (\mathbf{c}' - \mathbf{c}) + \mathbf{e}$ are non-zero, a SIS solution can be found. Thus, $\mathcal{A}$ cannot forge any signature, and (U_i, GS_k) uniquely identify each other with their keys and nonce. Hence, the QuDPas-FHA meets strong mutual authentication. □

Lemma 4 (F4: Session Key Agreement). *The QudPas-FHA ensures strong session key agreement.*

Proof. The robust key agreement when U_i and GS_k agree on session-specific random tokens never communicated in plaintext during authentication. On a valid authentication, both U_i and GS_k agree on the key $UGSK = H(n_u || n_{gs})$, where n_u and n_{gs} are ephemeral nonce generated by U_i and GS_k, respectively. Besides, GS_k agrees on valid $n_u \leftarrow \mathsf{SetlaUSC}[SK_{GS_k}, PK_{u_i}, C_1]$. Note, $\mathcal{A}$ (even S_j) cannot forge n_u on behalf of U_i as per Lemma 3 where $\mathbf{a}_1 \cdot (\mathbf{z} - \mathbf{z}' - \mathbf{s}_{u_i,1} \cdot \mathbf{c} + \mathbf{s}_{u_i,1} \cdot \mathbf{c}') + (\mathbf{e}_{u_i,1} \cdot (\mathbf{c}' - \mathbf{c}) + \mathbf{e}) = 0$ for two forgeries $(\mathbf{z}, \mathbf{c})$ and $(\mathbf{z}', \mathbf{c}')$. For valid $n_{gs_k} \leftarrow \mathsf{SetlaUSC}[PK_{GS_k}, SK_{u_i}, C_4]$, U_i agrees on n_{gs_k}. Note, when considering rapid authentication during handover, $UGSK = H(n_u || n_g s)$ does not consider n_s of S_j due to the satellite's role as a relay for U_i and GS_k authentication. Thus, QuDPas-FHA ensures strong session key agreement. □

Lemma 5 (F5: Forward and Backward Secrecy). *The QudPas-FHA supports essential forward secrecy and backward secrecy on the session key.*

Proof. Forward secrecy adopted system regularly and automatically updates encryption and decryption keys. It safeguards essential data through session keys even if the server's private key is revealed. Moreover, every user-initiated session has a unique session key; thus, only the disclosed key is vulnerable. Note, the compromised NCC keys cannot be used to recover user keys according to Lemma 2. Now, consider in past session p, U_i and GS_k agree on a session key $UGSK^{(p)} = H(n_u^{(p)} || n_{gs}^{(p)})$, where $n_u^{(p)}$ and $n_{gs}^{(p)}$ are the ephemeral secrets chosen by U_i and GS_k, respectively. Similarly, one considers $UGSK^{(c)} = H(n_u^{(c)} || n_{gs}^{(c)})$ for the current session $c \geq p+1$. Although $\mathcal{A}$ finds $UGSK^{(c)}$, it does not breach past sessions, as $\mathcal{A}$ cannot find $UGSK^{(p)}$, i.e., $UGSK^{(c)} \not\rightarrow UGSK^{(p)}$. Note, if $\mathcal{A}$ still find $UGSK^{(p)}$ in c irrespective of $UGSK^{(c)}$, then it must breach IND-SC-CCA and EUF-SC-CMA safety; however, it is infeasible due to Theorem 1 and Lemma 3. Thus, QuDPas-FHA meets forward secrecy on session keys. Similarly for backward safety, one finds $UGSK^{(c)} \not\rightarrow UGSK^{(f)}$ where $f \geq c+1$. □

Lemma 6 (F6: User Privacy). *QudPas-FHA ensures anonymity, untraceability, and unlinkability traits.*

Proof. Ensuring user privacy is a critical component of SIN communication. It indicates $\mathcal{A}$ cannot trace the user's footprint during communication. Note that user information, such as PK_{u_i}, is not disclosed publicly while $C_2 \sim C_4$ is transmitted. Besides, $C_2 \sim C_4$ are safeguarded under the RLWE assumption. Thus, tracing source as PK_{u_i} from $C_2 \sim C_4$ results in discovering an RLWE solution. Moreover, by employing nonces n_u, n_s, and n_{gs}, $C_2 \sim C_4$ is rendered random for each session. Thus, $\mathcal{A}$ cannot determine which data in $\{C_2^{(i)} \sim C_4^{(i)}\}$ is associated with the same anonymous U_i. Hence, user privacy is preserved . □

Theorem 3. *The QuDPas-FHA protocol withstands important security attacks for the SIN.*

Proof. The theorem follows when Lemmas 7-11 under Definitions 1-6 and Theorem 1 are hold. □

Lemma 7 (A1: Entity Impersonation). *An attacker cannot impersonate any entity to send fake data.*

Proof. Suppose $\mathcal{A}$ impersonates U_i and sends erroneous C_2 to S_j during the authentication. Upon receiving C_2, S_j gets $(C'_1 || PK_{u_i}) = \mathrm{DEC}[SK_{s_j}, C_2]$ where $C'_1 = \mathsf{SetlaSC}[PK_{gs_k}, SK_{u_i}, n_u]$. After receiving C_3 from S_j, GS_k decrypts $(n_s || C'_1 || PK_{u_i})$ but cannot get $n_u \neq \mathsf{SetlaUSC}[SK_{gs_k}, PK_{u_i}, C'_1]$ with the derived PK_{u_i}. This is due to not generating C'_1 with SK_{u_i}. Thus, $\mathcal{A}$ cannot impersonate U_i successfully. Now, consider $\mathcal{A}$ impersonates GS_k and sends $(C_4, \langle I_1, I_2 \rangle)$ to S_j where $I_1 = n_{gs} \oplus n_u \oplus n'_s$ and $I_2 = \mathsf{HMAC}[n'_s, C_4]$. On getting it, S_j finds $\mathsf{HMAC}[n_s, C_4] \neq I_2$. Thus, it aborts communication. Even if $\mathcal{A}$ breaches it, U_i finds $n_{gs} \neq \mathsf{SetlaUSC}[PK_{GS_k}, SK_{u_i}, C_4]$. Therefore, QuDPas-FHA resists U_i and GS_k impersonation attempts. □

Lemma 8 (A2: Replay Attack). *An attacker cannot replay challenge-response pairs for system access.*

Proof. During authentication, if $\mathcal{A}$ attempts to replay any of $(C_2 \sim C_4)$ as U_i, it will be promptly detected, and the session key agreement will be terminated. Suppose, $\mathcal{A}$ replay $C_2 = \mathsf{ENC}[PK_{S_j}, C_1 || PK_{u_i}]$

Table 2: Security functionalities comparisons of the existing schemes.

Scheme	Assumption	Security Functionalities												
		F1	F2	F3	F4	F5	F6	F7	A1	A2	A3	A4	A5	(in %)
W1 (Yang et al., 2018)	ECDSA	×	×	✓	×	✓	✓	×	×	×	×	×	×	25
W2 (Ma et al., 2019)	ISIS	✓	×	✓	×	✓	×	×	✓	✓	✓	×	×	50
W3 (Guo and Du, 2020)	RLWE	✓	×	✓	×	✓	✓	✓	✓	✓	×	×	×	58
W4 (Guo et al., 2022)	RLWE	✓	×	✓	×	✓	✓	×	✓	✓	✓	×	×	58
W5 (Dharminder et al., 2023)	RLWE	×	×	✓	✓	✓	✓	×	✓	×	✓	×	×	50
W6 (Al-Mekhlafi et al., 2023)	SIS and ISIS	✓	×	×	×	×	✓	×	×	✓	✓	×	×	33
The QuDPas-FHA	RLWE and HMAC	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	**100**

F1: Quantum-safe, F2: Level-II Safety, F3: Mutual Authentication, F4: Session Key Agreement, F5: Forward and Backward Secrecy, F6: User Privacy, F7: Authentication Handover, A1: Withstand Impersonation Attack, A2: Anti-Replay, A3: Withstand MitM Attack, A4: ESL-free, A5: No GPS spoofing

where $C_1 = \mathsf{SetlaSC}[PK_{GS_k}, SK_{u_i}, n_u]$ and sends C_2 to S_j. While S_j sends $C_3 = \mathsf{ENC}[PK_{GS_k}, (n_S \| C_1 \| PK_{u_i})]$, GS_k reveals $(n'_S \| C'_1 \| PK'_u) = \mathsf{DEC}[SK_{GS_k}, C_3]$, $n_u = \mathsf{SetlaUSC}[SK_{GS_k}, PK'_{u_i}, C'_1]$ from C_3. Although, after several processes, $\mathcal{A}$ retrieves (C_4, I'_1), it cannot reveal n_{gs} due to unavailability of SK_{u_i}. Now, consider $\mathcal{A}$ replays old $C_3 = \mathsf{ENC}[PK_{GS_k}, (n_S^{old} \| C'_1 \| PK'_{u_i})]$, then GS_k can compute and submits $X = (C_4, \langle I_1, I_2 \rangle, I_{gu})$. Note that $\mathcal{A}$ cannot alter X due to unknown n_S^{old}. On getting X, S_j confirms $\mathsf{HMAC}[n_s^{new}, C_4] \neq I_2$, thus, $\perp$. Similarly in the new session, if $\mathcal{A}$ replays $C_4 = \mathsf{SetlaSC}[PK'_{u_i}, SK_{GS_k}, n_{gs}^{old}]$, then S_j confirms $I_2 \neq \mathsf{HMAC}[n_s^{new}, C_4]$. Moreover, the QuDPas-FHA flags for replaying any communication by $\mathcal{A}$ between U_i, S_j and GS_k. Thus, it resists replay attacks. □

Lemma 9 (A3: Man-in-the-Middle Attack). *An attacker cannot tamper with communication to obtain session or secret key(s) in the QuDPas-FHA protocol.*

Proof. A man-in-the-middle (MitM) attack occurs when $\mathcal{A}$ intercepts communication between U_i and GS_k to steal the session key $UGSK$ or produces two keys for both without being traced. Note $UGSK = H(n_u || n_{gs})$. When U_i sends C_2 to S_j, $\mathcal{A}$ alters it to $C'_2 = \mathsf{ENC}[PK_{S_j}, C_1'^{\mathcal{A}} || PK_{u_i}]$ for some $C_1'^{\mathcal{A}}$ and sends C'_2 instead of C_2 to S_j. On valid decryption, S_j retrieves $(C_1'^{\mathcal{A}} || PK_{u_i})$ and use it with a nonce n_s to send C_3. On receiving C_3, GS_k cannot retrieve user nonce n_u as $\mathsf{SetlaUSC}[SK_{GS_k}, PK_{u_i}, C_1'^{\mathcal{A}}] \rightarrow \perp$. This is because $\mathcal{A}$ chose random $C_1'^{\mathcal{A}}$ as it cannot execute $\mathsf{SetlaSC}[\cdot, \cdot, \cdot]$ for n_u due to unavailability of SK_{u_i}. It is also possible that $\mathcal{A}$ replays old C_2^{old} during current authentication; however, a replay is infeasible based on Lemma 8. Hence, QuDPas-FHA resists MitM attack. □

Lemma 10 (A4: Ephemeral Secret Leakage). *An attacker cannot disclose ephemeral secrets during entity authentication in the QuDPas-FHA protocol.*

Proof. If ephemeral secrets are leaked, $\mathcal{A}$ can reveal the session key. ESL attacks leak session keys through eavesdropped messages. In QuDPas-FHA, the ephemeral secrets in every session are n_u and n_{gs} where the session key is $UGSK = H(n_u \| n_{gs})$. Note, n_u is signcrypted with PK_{gs_k}. Thus, $\mathcal{A}$ or S_j cannot reveal n_u due to the hardness of RLWE assumption. Similarly, for n_{gs}. Thus, unauthorized disclosure of n_u and n_{gs} is maintained. Beside, when authentication is delegated to S_j^{new} via secure forwarding, S_j^{new}'s list is updated as $\mathcal{L}_j^{new} \leftarrow \mathcal{L}_j^{new} \cup \mathcal{L}_j$. Note that $\mathcal{L}_j^{new}$ holds the ephemeral secret of GS_k as $H(n_{gs})$, therefore only U_i and GS_k know it, preventing leaking. Thus, QuDPas-FHA resists ESL attacks. □

Lemma 11 (A5: GPS Spoofing). *A GPS location spoofing attempt is unsuccessful during entity authentication in the related protocols.*

Proof. Insufficient authentication lets U_i spoof location data while being traced as a valid user. After authentication, QuDPas-FHA prevents users from sending fake locations due to strong key agreement. Else, an RLWE solution results from its security breaches. Note that user privacy may be impaired while sending location data. However, QuDPas-FHA fulfills this by making communication anonymous while identifying U_i uniquely. Thus, QuDPas-FHA withstands GPS location spoofing once a user is authenticated. □

5 PERFORMANCE EVALUATION

Table 2 compares the security attributes where the QuDPas-FHA outperforms existing approaches. Now, we examine various overheads incurred in different phases of the QuDPas-FHA.

Security Specification: For a minimum of 128 bits of classical security, we use $n = 1024, m = 2048, \omega = 16, d = 15, B = 2^{15}, q = 2^{25}, \kappa = 131$. Besides, we use SHA-256 underlining hash function and 128-bit AES for symmetric encryption and deciphering.

5.1 Computation Cost

U_i executes various operations to verify its legitimacy to GS_k via S_j. For this, U_i utilizes public-key encryp-

Table 3: Computation overheads of various entities in the related protocols.

Scheme	AUTH Type	Computation Cost			
		C_U (at U_i side)	C_S (at S_j side)	C_{GS} (at GS_k side)	Total $(C_U+C_S+C_{GS})$
W1 (Yang et al., 2018)	Regular	$T_h+T_{ex}+2T_{bp}$	$2T_h+4T_{ex}+5T_{bp}$	T_h+5T_{ex}	$4T_h+10T_{ex}+7T_{bp}$
	Handover	$T_h+T_{ex}+2T_{ecc}$	$2T_h+4T_{ex}+5T_{ecc}$	T_h+5T_{ex}	$4T_h+10T_{ex}+7T_{ecc}$
W2 (Ma et al., 2019)	Regular	$3T_h+3T_{la}+3T_{lm}$	$2(\|\mathcal{L}\|+1)T_h+2\|\mathcal{L}\|T_{la}+(3\|\mathcal{L}\|+2)T_{lm}$	$T_h+T_{la}+T_{lm}$	$2(\|\mathcal{L}\|+3)T_h+2(\|\mathcal{L}\|+2)T_{la}+3(\|\mathcal{L}\|+2)T_{lm}$
	Handover	$3T_h+3T_{la}+3T_{lm}$	$2(\|\mathcal{L}\|+1)T_h+2\|\mathcal{L}\|T_{la}+(3\|\mathcal{L}\|+2)T_{lm}$	$T_h+T_{la}+T_{lm}$	$2(\|\mathcal{L}\|+3)T_h+2(\|\mathcal{L}\|+2)T_{la}+3(\|\mathcal{L}\|+2)T_{lm}$
W3 (Guo and Du, 2020)	Regular	$7T_h+T_{la}+4T_{lm}$	$5T_h+T_{la}+4T_{lm}$	–	$14T_h+2T_{la}+8T_{lm}$
	Handover	$7T_h+T_{la}+4T_{lm}$	$5T_h+T_{la}+4T_{lm}$	–	$14T_h+2T_{la}+8T_{lm}$
W4 (Guo et al., 2022)	Regular	$5T_h+T_{ed}+2T_{ra}+4T_{rm}$	$3T_h$	$2T_h+T_{ed}+2T_{ra}+4T_{rm}$	$10T_h+2T_{ed}+4T_{ra}+8T_{rm}$
	Handover	$3T_h$	$4T_h$	T_h	$8T_h$
W5 (Dharminder et al., 2023)	Regular	$6T_h+T_{la}+4T_{lm}$	–	$5T_h+T_{la}+4T_{lm}$	$11T_h+2T_{la}+8T_{lm}$
	Handover	$6T_h+T_{la}+4T_{lm}$	–	$5T_h+T_{la}+4T_{lm}$	$11T_h+2T_{la}+8T_{lm}$
The QuDPas-FHA (ours)	Regular	$T_h+2T_{ed}+7T_{la}+6T_{lm}$	$T_{hm}+2T_{ed}+5T_{la}+3T_{lm}$	$2T_h+3T_{ed}+T_{hm}$ $8T_{la}+11T_{lm}$	$3T_h+2T_{hm}+7T_{ed}+20T_{la}+20T_{lm}$
	Handover	$3T_h+T_{ed}+3T_{la}+2T_{lm}$	$(\|\mathcal{L}\|+2)T_h$	–	$(\|\mathcal{L}\|+5)T_h+2T_{ed}+4T_{la}+3T_{lm}$

T_h: Cost for one hash operation; T_{ex}: Time to run modular exponentiation; T_{bp}: Cost for one bilinear pairing; T_{lm}: Time for matrix multiplication; T_{hm}: Time for HMAC operation; T_{la}: Time for matrix addition; T_{rm}: Time for ring multiplication; T_{ra}: Time for ring addition; T_{ed}: Time to execute Encode/Decode

tion (E), signcrypt (SC), unsigncrypt (USC), and hash operations (H). The overhead of U_i is C_U=$(E+SC+USC+H)$. Besides, S_j runs E, D, and HMAC (HM) operations. The burden on S_j is C_S=$(E+D+HM)$. Further, GS_k runs one from each of D, SC, USC, HM, and H operations. Thus, GS_k's overhead is C_{GS}=$(D+SC+USC+HM+2H)$. The authentication handover needs U_i to run $C_U^{FA}=(E+3H)$ operations, while S_j has at least C_S^{FA}=$2H$ computations. Note that the our handover authentication is much faster than re-authentication. Table 3 exhibits a detailed comparison between related works (W1∼W5) and QuDPas-FHA. In this comparison, the symbols E, D, SC, and USC are broken down into low-level operations, excluding lightweight cryptographic operations like XOR and concatenation, as our primary emphasis is on time-consuming operations. It is worth noting that if any scheme does not provide any handover authentication, the cost is equivalent to standard authentication.

5.2 Token Exchange and Keys Storage

In QuDPas-FHA, several data are transmitted as authentication tokens. For the comparison purpose, we assume the length of various parameters as $|ID_i|$=16 Bytes (B), $|\mathbb{G}|$=128 B, $|\mathbb{Z}_q^*|$=20 B, random $|r|$=16 B, hash $|h|$=32 B, $|Z^m|$=256 B, and $|Z^{m\times n}|$=32768 B. Communication cost focuses on open channel data size. For authentication, U_i sends D_U=288 B while satellite S_j sends $D_S=960$ B. The ground station GS_k transfers D_{GS}=800 B. Besides, U_i sends D_U^{FA}=288 B to validate its pre-authentication status for fast authentication. To store crypto keys, U_i, S_j, and GS_k consider 1208 B keys to store in the private space.

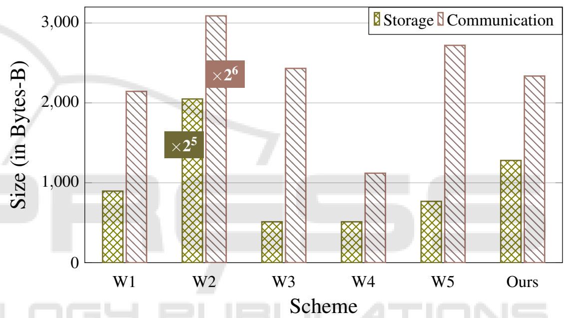

Figure 5: Comparison of transmission and storage costs.

Figure 5 shows detailed storage and transmission cost comparisons between related works.

Further Discussion. Table 2 shows that the QuDPas-FHA achieves critical security attributes. Besides, it avoids the need for synchronizing data for each authentication session which withstand desynchronization attacks launched by prospective foes. It ensures robust mutual authentication through unique self-authentication of both the user and ground station, generating a session key as $UGSK=H(n_u||n_GS)$. Service undeniability is maintained by using lattice-based signcryption with the secret keys of the respective entities. User revocation involves creating an explicit list with the revoked user's public key stored in a publicly available storage, which can be updated by the NCC. The QuDPas-FHA distributes diverse duties among multiple entities, reducing the number of crypto operations at lightweight devices compared to high-end ground stations. Although W4 requires fewer costs, as shown in Figure 5, it achieves

58% of total F1-F7 and A1-A5 traits, making it a feasible solution with adequate functional overheads.

6 CONCLUSION AND FUTURE RESEARCH DIRECTION

The paper introduces a robust privacy-preserved authentication and key agreement for the space information network. It offers various safety traits, including mutual authentication, session key agreement, forward/backward secrecy, and user anonymity. Under the decisional-RLWE assumption, it withstands several attacks, including quantum, user impersonation, replay and man-in-the-middle attacks. Compared to existing works, the suggested protocol provides ample operational safety (at least 40% more) with adequate computation, transmission, and storage costs.

Although, our protocol has comprehensive traits, it needs more storage and processing power due to SETLA-based approach. In the future, we will design a more efficient authentication for undeniable services in a *zero-trust region-based multi-NCC* framework.

ACKNOWLEDGEMENTS

This work was supported in part by the National Science and Technology Council (NSTC) under grants 112-2221-E-110-027 and 112-2634-F-110-001-MBK and by the CANSEC-LAB@NSYSU in Taiwan.

REFERENCES

Al-Mekhlafi, Z. G., Al-Shareeda, M. A., Manickam, S., Mohammed, B. A., and Qtaish, A. (2023). Lattice-based lightweight quantum resistant scheme in 5G-enabled vehicular networks. *Mathematics*, 11(2):399.

Bai, S. and Galbraith, S. D. (2014). An improved compression technique for signatures based on learning with errors. In *Topics in Cryptology – CT-RSA 2014*, pages 28–47, Cham. Springer International Publishing.

Chen, Y.-A., Zhang, Q., Chen, T.-Y., Cai, W.-Q., Liao, S.-K., Zhang, J., Chen, K., Yin, J., Ren, J.-G., Chen, Z., et al. (2021). An integrated space-to-ground quantum communication network over 4,600 kilometres. *Nature*, 589(7841):214–219.

Dharminder, D., Dadsena, P. K., Gupta, P., and Sankaran, S. (2023). A post quantum secure construction of an authentication protocol for satellite communication. *Int. J. Satell. Comm. N.*, 41(1):14–28.

Duong, D. H., Fukushima, K., Kiyomoto, S., Roy, P. S., and Susilo, W. (2019). A lattice-based public key encryption with equality test in standard model. In *Information Security and Privacy–ACISP 2019*, pages 138–155, Cham. Springer International Publishing.

Gérard, F. and Merckx, K. (2018). SETLA: Signature and encryption from lattices. In *Intl. Conf. on Cryptology and Network Security*, pages 299–320. Springer.

Girault, M. (1991). Self-certified public keys. In *Advances in Cryptology – EUROCRYPT'91*, pages 490–497, Berlin, Heidelberg. Springer.

Guo, J. and Du, Y. (2020). A novel RLWE-based anonymous mutual authentication protocol for space information network. *Secur. commun. netw.*, 2020:1–12.

Guo, J., Du, Y., Wu, X., Li, M., Wu, R., and Sun, Z. (2022). PSAA: Provable secure and anti-quantum authentication based on randomized RLWE for space information network. *arXiv preprint arXiv:2208.00901*.

Karati, A., Chang, Y.-S., and Chen, T.-Y. (2023). Robust three-factor lightweight authentication based on extended chaotic maps for portable resource-constrained devices. In *Proc. of the 20th International Conference on Security and Cryptography - SECRYPT*, pages 673–682. INSTICC, SciTePress.

Le, H. Q., Duong, D. H., Roy, P. S., Susilo, W., Fukushima, K., and Kiyomoto, S. (2021). Lattice-based signcryption with equality test in standard model. *Computer Standards & Interfaces*, 76:103515.

Liberg, O., Löwenmark, S. E., and Euler et al., S. (2020). Narrowband internet of things for non-terrestrial networks. *IEEE Commun. Mag.*, 4(4):49–55.

Lyubashevsky, V. (2009). Fiat-shamir with aborts: Applications to lattice and factoring-based signatures. In *Intl. Conf. on the Theory and Application of Cryptology and Information Security*, pages 598–616. Springer.

Lyubashevsky, V. (2012). Lattice signatures without trapdoors. In *Advances in Cryptology – EUROCRYPT 2012*, pages 738–755, Berlin, Heidelberg. Springer.

Lyubashevsky, V. and Micciancio, D. (2018). Asymptotically efficient lattice-based digital signatures. *Journal of Cryptology*, 31(3):774–797.

Ma, R., Cao, J., Feng, D., and Li, H. (2019). LAA: lattice-based access authentication scheme for iot in space information networks. *IEEE Internet Things J.*, 7(4):2791–2805.

Ma, S., Zhang, M., Huang, Q., and Yang, B. (2015). Public key encryption with delegated equality test in a multi-user setting. *The Computer Journal*, 58(4):986–1002.

Ramos-Calderer, S., Bellini, E., Latorre, J. I., Manzano, M., and Mateu, V. (2021). Quantum search for scaled hash function preimages. *Quantum Inf. Process.*, 20(5):180.

Roy, P. S., Duong, D. H., Susilo, W., Sipasseuth, A., Fukushima, K., and Kiyomoto, S. (2022). Lattice-based public-key encryption with equality test supporting flexible authorization in standard model. *Theoretical Computer Science*, 929:124–139.

Sato, S. and Shikata, J. (2018). Lattice-based signcryption without random oracles. In *Post-Quantum Cryptography*, pages 331–351, Cham. Springer International Publishing.

Yang, Q., Xue, K., Xu, J., Wang, J., Li, F., and Yu, N. (2018). AnFRA: Anonymous and fast roaming authentication for space information network. *IEEE Trans. Inf. Forensics Secur.*, 14(2):486–497.