Identity-Based Ring Signature from Quantum Token

Nabanita Chakraborty and Ratna Dutta

Department of Mathematics, Indian Institute of Technology of Kharagpur,
Kharagpur, 721302, India
nabanita@kgpian.iitkgp.ac.in, ratna@maths.iitkgp.ac.in

Abstract. An *identity-based ring signature* (IRS) has numerous applications including e-cash, online voting and whistle-blowing. This is an attractive cryptographic primitive that enables a user to sign sensitive documents or approve important statements on behalf of a ring of user identities chosen by the signer and provides *anonymity* and em unforgeability. Anonymity ensures that a distinguisher cannot identify the actual signer identity inside the ring of user identities with probability non-negligibly better than a random guess while unforgeability ensures that no user can sign a message on behalf of a ring that does not contain his/her identity. An IRS scheme uses a trusted setup to generate the secret keys of the users whose public keys can be calculated by everyone from their identities. However, the existing IRS schemes do not support signature delegation. We enforce secret key leasing in IRS by integrating the technique proposed in a recently emerging notion of quantum signing token to make the signing key quantum. We introduce the *first quantum-tokenized identity-based ring signature* (QTIRS) scheme qtIRS and its variant D-qtIRS with signature delegation assuming the existence of obfuscation of membership checking circuits and a computationally sound *non-interactive zero-knowledge* (NIZK) proof system. We emphasize that our schemes are dynamic in the sense that a user can join or leave the system without disturbing others and allow unrestricted ring size with signature size independent of the ring size. Our constructions need two one-way quantum communication channels to transmit the secret token from the trusted authority to the user and from the lessor to the lessee. We prove the security of our proposed schemes in standard security model and provide rigorous security analysis for unforgeability as well as anonymity. We enhance our security framework by minimizing restrictions for the adversary and allowing a broader domain to forge and introduce the notion of strong unforgeability and strong anonymity. More precisely, our scheme qtIRS achieves a *strong existential unforgeability against chosen message and identity attack* (sEU-ACMI) and *strong anonymity* (sAnon) while our scheme D-qtIRS attains *strong existential unforgeability against chosen message and identity attack with signature delegation* (sEU-ACMID) and sAnon. The most interesting feature of D-qtIRS is that it enables a lessor to lease her secret signing token to a lessee and grant the lessee limited power to sign a document of the lessee's choice on behalf of a ring of user identities formed by the lessor. Moreover, the lessor can revoke the delegated signing authority at her convenience. Additionally, we have shown the existence of obfuscation of membership-checking circuits from [20] of Hofheinz et al., which is of independent interest.

Keywords: Quantum token, identity-based ring signature, non-interactive zero-knowledge proof system, obfuscation, key-leasing, key-revocation.

1 Introduction

RS. *Ring Signature* (RS) is a well-known cryptographic primitive that allows users to authenticate sensitive documents, messages or decisions without revealing their identities. In an RS scheme, a user forms a ring of users using their corresponding public keys and signs a message using its secret signing key which can be efficiently verified with respect to that ring. More interestingly, the RS provides anonymity, ensuring infeasibility to identify the actual signer within that concerned ring and guarantees unforgeabilty, ensuring no user can sign a message on behalf of a ring that does not contain its public key.

RS plays a significant role in privacy protection and has recently found a wide range of applications including whistle-blowing, online voting, e-cash, smart grid security, ad-hoc anonymous key agreement and ad-hoc networks [11]. For instance, consider the following scenario of whistle-blowing. Let there be a confidential meeting between several world leaders who are not supposed to leak any sensitive information outside since it can lead to disharmony. Suppose one leader gets information that would significantly impact public life and wants to share it with a journalist. If that leader uses a fake or anonymous identity to communicate with the journalist, then the journalist perhaps would not trust her due to lack of authenticity. In contrast, if the leader uses her original identity, it may pose a serious risk to that leader. In such scenarios, a ring signature is an ideal candidate that allows the leader to form a ring of leaders who are present in that meeting and are trustworthy to the journalist. She can sign the sensitive message on behalf of the ring so that the journalist be convinced that the information is authentic and the leader remains anonymous and safe.

RS schemes are widely designed in *public key infrastructure* (PKI). In a PKI system, the public key of a user is bound with its identity by a digital certificate. The major drawbacks to implement an efficient PKI-based RS scheme are managing, storing, transmitting and verifying the digital certificates corresponding to each member of the concerned ring and the computation and time complexity grow with the ring size. To avoid public key authentication and digital certification in PKI-based cryptographic systems, Shamir [3], in 1985, proposed a novel concept of identity-based encryption and signature which simplifies key management. In an identity-based setup, anyone can calculate or access a user's corresponding public key using its identity whereas a trusted party generates the secret key of each user and sends it to that specific user via a secure communication channel.

RS was first proposed by Rivest et al. [4], in 2001. Later, in 2006, Bender et al. [5] formalized the two main security notions of RS scheme, namely unforgeability and anonymity. They have formally defined unforgeability against chosen-subring attacks, unforgeability against insider corruption, anonymity against adversarially chosen keys and anonymity against full key exposure. In 2017, an efficient RS scheme was developed by Malavolta and Schröder [6] in the standard model using rerandomizable keys and NIZK under generalization of the knowledge of exponent assumption. Recently in 2024, Hu et al. [15] designed a *linkable ring signature* (LRS) scheme where given two signatures, one can be efficiently determined whether both the signatures are generated by one user or not. Their proposed LRS scheme is based on standard lattice assumptions of *short integer solution* (SIS) and *learning with error* (LWE) and achieves a stronger notion of unforgeability which provides the adversary a broader space to forge by minimizing the restriction on the forged tuple.

IRS. Based on *identity-based infrastructure* (IBI), Zhang and Kim [7], in 2002, proposed the first *identity-based ring signature* (IRS) scheme from bilinear pairing, equipped with similar features of anonymity and unforgeability where the public keys of the users can be derived by anyone using their respective identities while the secret keys are issued by a trusted authority. IRS is applicable in similar scenarios as RS. Consider one case of online voting where several people are required to cast their votes online. To keep the identity secret, a voter signs the name his preferred candidate using his secret key with respect to a ring of identities containing few valid voters chosen by him. The vote i.e., the signature can be verified correctly to be generated by a member of that ring containing only legitimate voters. However, the feature of IRS makes sure that the actual signer/voter cannot be identified.

In 2005, Chow et al. [8] proposed an IRS protocol with bilinear pairings. This scheme attains existential unforgeability under adaptive chosen message and identity attack where the adversary is allowed to make polynomially many signing queries on chosen messages and rings and polynomially many corruption queries on identities. Additionally, it achieves unconditional signer's ambiguity, meaning the adversary finds the actual signer within a ring with probability inverse of ring size. In 2006, Au et al. [9] proposed two IRS schemes, one with a constant signature size in the number of users in the ring which is secure in the standard model under the *computational Diffie-Hellman* (CDH) assumption and the other scheme achieving constant signature size with restricted ring size which is secure in the weaker model of selective identity and selectively chosen message model under the *Diffie-Hellman inversion* (DHI) assumption where the adversary has to submit the challenge identity at the beginning of the unforgeability game,

and the query of the challenge identity to the corruption oracle is rejected. In 2013, Au et al. [11] proposed an IRS scheme based on *hierarchical identity-based encryption* (HIBE) to ensure strong anonymity against an unbounded adversary i.e., an adversary whose run-time is unbounded. In 2019, Deng et al. [12] proposed an *identity-based linkable ring signature* (ILRS) which relies on the CDH and the *decisional bilinear Diffie Hellman* (DBDH) problem in the *random oracle model* (ROM). Since bilinear pairing is an expensive tool to implement, Herranz [10] proposed an IRS scheme based on RSA in 2007. Later in 2021, Le et al. [14] designed the first ILRS in integer lattice and ideal lattice setting, assuming hardness of SIS and *ring short integer solution* (rSIS). Recently, in 2021, Tang et al. [17] introduce a quantum-secure ILRS scheme in the ROM based on *n-th degree truncated polynomial ring units* (NTRU) lattices which provide unconditional anonymity, unforgeability and likability. However, their ring signature is small only when the ring size is very small. In 2022, Buser et al. [29] proposed two post-quantum IRS schemes, one from Picnic digital signature and the other from a stateful digital signature. In 2024, Gao et al. [26] came up with a post-quantum IRS using ROM based on lattice assumption of *module small integer solution* (MSIS). In the same year, Gao et al. [27], proposed another mSIS-based IRS. Though the IRS has gained a lot of interest from cryptographers, there is no work on IRS in the quantum world. Due to the unique features of quantum information and computation, it has the potential to add many more properties to an IRS scheme.

Quantum token: With the advent of fast and remarkable growth of quantum computation and information theory, it is a continuous endeavor to employ the power of quantum computing in cryptography. Quantum computing avails novel features including no-cloning, measurements, etc. which are impossible in the classical world. No-cloning ensures that quantum information cannot be arbitrarily copied. These properties open new avenues to cryptography such as secret key-leasing, secure software-leasing, certifiable deletion, provable revocation of cryptographic objects etc. In 1983, Wiesner [1] proposed the concept of quantum money based on the uncloneability of quantum states for the first time which prepared the ground for quantum cryptography in future. However, this quantum money scheme is not information-theoretically secure since it can be broken in exponential time. Later in 2012, Aaronson and Christiano [21] proposed a public key quantum money based on hidden subspaces where the quantum (money) state can be generated only by the bank but verified by anyone using the related public key. Moreover, they have developed unconditionally secure private-key quantum money against adaptive interaction between the bank and the adversary. Later, in 2017, Ben-David and Sattath [16] proposed a concept of quantum signing tokens for digital signatures. Following this, Shmueli [18] presented a public-key semi-quantum money which relies on strong assumptions like *indistinguishability obfuscation* (iO) secure against the quantum adversary and sub-exponential hardness of the *learning-with-error* (LWE) problem in 2022. In the same year, a similar concept of semi-quantum tokenized signature was introduced by Shmueli [19] based on the same assumptions where a classical bank issues a quantum signing token to a user via a classical communication channel. Recently, Coladangelo et al. [23] have generalized the idea of hidden subspaces and proposed hidden coset states. They have proved that hidden coset states satisfy the direct product hardness property which implies the existence of a tokenized signature from iO and *one-way function* (OWF). They have given instantiations of obfuscated membership-checking circuits namely, subspace obfuscation and a compute-and-compare function. Hofheinz et al. [20] have shown the existence of obfuscation for point functions in the ROM.

1.1 Our contribution

Existing IRS schemes are not up to the mark due to restricted ring size, parameter selection or insufficient security guarantees. Most of the existing works in IRS provide weaker security notions by restricting the power of the adversary. There is no IRS in the quantum world. Besides, existing RS and IRS schemes do not support signature delegation where a user (lessor) can lease its secret signing key to a third party (lessee) and delegate the signing authority for a limited period.

As the security models of existing IRS schemes are not well organized and adversarial powers in the security games are very restricted in most of the works [5, 7, 8, 9, 11], one of our major goals in this work is to design IRS by providing the adversary with as much power as possible. Apart from enhancing security, we also concentrate on designing an IRS in the quantum setting to enforce features like key-leasing and revocation that cannot be attained in classical. In this paper, we introduce more formalized and *stronger* notions of security for the IRS, construct IRS schemes in the quantum world and prove their security in our proposed security frameworks. To be more concrete, we have constructed a quantum-tokenized IRS scheme qtIRS that provides security in our proposed stronger security model. We exploit quantum signing token to make the signing key quantum and couple an obfuscation of membership-checking circuit with a computationally sound *non-interactive zero-knowledge* (NIZK) proof system. To the best of our knowledge, this is the *first* IRS scheme in the quantum world where the signing key is made quantum. Further, we have extended qtIRS to D-qtIRS, a quantum-tokenized IRS with signature delegation. The scheme D-qtIRS not

only supports the delegation of signing authority by a lessor to a lessee for a limited period but also facilitates the revocation of that delegated authority from the lessee by the lessor.

The main contributions of this paper are two-fold
a) qtIRS: quantum-tokenized identity-based ring signature and
b) D-qtIRS: quantum-tokenized identity-based ring signature with signature delegation.
The salient features of qtIRS and D-qtIRS are detailed below.

a) qtIRS**: quantum-tokenized identity-based ring signature**. Inspired by the work of [6], we propose the *first* quantum-tokenized IRS scheme qtIRS based on the existence of secure obfuscation $\mathcal{O}$ of membership checking circuits and computationally sound NIZK. Informally speaking, the verification keys in our qtIRS contain obfuscation of membership-checking circuits which are generated and published by a trusted authority on receiving identity from a user and are tamper-resistant. The approach of quantum-tokens is adapted from the work of Coladangelo et al. [23] to make the signing key (token) a quantum state which essentially guarantees uncloneability to the signing key. To generate a ring signature $\boldsymbol{\sigma}$ on a message $m \in \{0,1\}$ with respect to a ring of user identities ID, the signer with its identity id in the ring ID, measures its quantum signing token or its Hadamard transform. Then it calculates a randomized verification key rvk depending on m by choosing a secret randomness. Finally, it employs NIZK to generate a proof on the public statement $\mathsf{ID}||\mathsf{rvk}$ consisting of the ring ID of user identities and the randomized verification key rvk with secret witness $(\mathsf{id}, \hat{\mathbf{x}})$ consisting of the signer identity id and the corresponding secret randomness $\hat{\mathbf{x}}$. The verification of the signature is done classically by verifying the respective NIZK proof and a membership-checking of a string. We have established that secure obfuscation of membership-checking circuits used in qtIRS, can be easily constructed from the work of Hofheinz et al. [20] which is of independent interest by proving the following lemma.

Lemma 1 (*Informal*). *If point functions can be obfuscated efficiently, the membership-checking circuits for finite sets can be obfuscated efficiently.*

The NIZK can be instantiated with SNARK [22]. The interesting features of qtIRS are listed below.

-*Stronger unforgeability*: We have introduced strong existential unforgeability against chosen message and identity attack sEU-ACMI, a stronger notion of existential unforgeability by modifying the signing query and forgery in the security definitions of the unforgeability game in contrast to the existing works [5, 7, 8, 9, 10, 11, 17]. Fewer the criteria for the adversary to forge, wider the coverage for attack scenarios. Thus, if one decreases the restrictions on the adversary to produce a valid forgery, he addresses more attack scenarios. In the forgery phase, we expand the criteria for the adversary to succeed in order to encompass a wider range of attacks. The challenger maintains a list $\mathcal{L}_S$ of previous signing queries and related responses. The condition that our security model imposes on the forged tuple of the form $(m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*)$ submitted by the adversary is $(m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*) \notin \mathcal{L}_S$ where m^* is a message, $\boldsymbol{\sigma}^*$ is a signature and ID^* is a ring of user identities. This is a much-relaxed requirement imposed on the forged signature than requiring $(m^*, \cdot, \mathsf{ID}^*) \notin \mathcal{L}_S$ and $(\cdot, \boldsymbol{\sigma}^*, \cdot) \notin \mathcal{L}_S$ in the existing RS (for RS the ring ID^* consists of public verification keys of users instead of their identities) and IRS schemes [5, 7, 8, 9, 10, 11, 17], allowing no signing query to be made for the same message and ring pair $(m^*, \cdot, \mathsf{ID}^*)$ by the adversary and no signature $\boldsymbol{\sigma}^*$ that was ever returned for any signing query by the challenger.

In the previous literature on RS and IRS schemes [7, 8, 9, 11], the adversary sends a signing query of the form (m, ID) where m is a message and ID is a ring of user identities in IRS schemes and a ring of public verification keys in RS schemes. The challenger chooses a signer with identity $\mathsf{id} \in \mathsf{ID}$ and returns the signature for the tuple $(\mathsf{id}, m, \mathsf{ID})$. However, in our proposed security game of unforgeability, the adversary asks a signing query of the form $(\mathsf{id}, m, \mathsf{ID})$ and can specify the signer by whom it wants to generate the signature. Thus, our security model provides the adversary with additional power that strengthens the framework. We proved the following theorem.

Theorem 1 (*Informal*). *If there exists a secure obfuscator of membership checking circuits and a computationally sound NIZK proof system* NIZK*, then our* qtIRS *attains* sEU-ACMI*.*

-*Stronger anonymity*: We have formally defined strong anonymity (sAnon), a stronger anonymity game for an IRS scheme in contrast to the existing literature on IRS [7, 8, 9] and given the adversary the power to make polynomially many signing and corruption queries unlike [11], thus providing the adversary more power than the existing IRS schemes. Furthermore, similar to our unforgeability game, our anonymity game also allows the adversary to specify the signer identity id during the signing query $(\mathsf{id}, m, \mathsf{ID})$ unlike [5, 7, 8, 9, 10] which allow signing query of the form

(m, ID) without specifying the signer identity id_i. For RS schemes, the identities of users are replaced with their public verification keys. More formally, we proved the following theorem.

Theorem 2 (Informal). *Assuming the existence of obfuscation of membership-checking circuits and computationally sound NIZK proof system* NIZK*, our* qtIRS *achieves* sAnon*.*

Dynamic membership and automatic membership revocation: Our qtIRS is dynamic in the sense that any member can join or leave the protocol at any point of time of the protocol execution. Most importantly, the joining and leaving of any member never affect any other members in the system. The joining of a member is ensured by issuing a secret signing token. Membership revocation is automatic as the signing token is destroyed once it (or its Hadamard transform) is measured during signature generation. Several signing tokens can be issued to a user at prior by the trusted authority to allow the user to sign a certain number of messages. Revocation for a classical IRS is non-trivial as the secret key holder can use that key for a lifetime whereas in a quantum-tokenized signature, the quantum token, once measured, gets destroyed and the user holding that token automatically gets revoked. More importantly, in our qtIRS, given only one secret quantum token, a member can sign either 0 or 1 but not both simultaneously since after signing one message the token gets destroyed and useless to the signer.

-*Efficiency*: We analyze the efficiency of our qtIRS scheme in Table 1. In qtIRS, the signing token is a λ-qubit state. The verification key size of a user is $2l$ as it contains two obfuscated membership-checking circuits, each of size l. The signature $\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi)$ contains a λ-bit string $\boldsymbol{\beta}$, a randomized verification key rvk which is an obfuscated membership-checking circuit of length l and an NIZK proof π of length $|\pi|$. Hence, our signature size is $\lambda + |\pi| + l$. Since the computation cost for creating a superposition over a finite set and the quantum one-time pad algorithm QOTP are constants, the algorithm key-generation has cost $T_1 + c$ for some constant c where T_1 is the computation cost for the obfuscated membership-checking circuit. The measurement of quantum state in superposition and an M-circuit calculation take constant time. Consequently, our sign algorithm has computation cost $T_2 + d$ for some constant d where T_2 is the computation cost of the generation of an NIZK proof. Let the cost of the algorithm $\mathsf{NIZK}.\mathcal{V}$ be T_3. The membership-checking takes constant time, so the cost of our verificaton algorithm is $T_3 + e$ when T_3 is the cost of verifying a NIZK proof.

The communication cost from the trusted authority to a user is λ in the quantum channel to issue the signing token. The communication cost of the trusted authority to publish the verification key of each user in a public list is $2l$. The cost to send the signature is $\lambda + |\pi| + l$ in the classical channel.

Table 1: Efficiency in terms of key size, signature size, computation cost, security

Scheme	STS	VKS	SS	GC	SC	DGC	VC	Security
qtIRS	λ-qubit	$2l$	$\lambda + \|\pi\| + l$	$T_1 + c$	$T_2 + d$	-	$T_3 + e$	sEU-ACMI, sAnon
D-qtIRS	λ-qubit	$2l$	$\lambda + \|\pi\| + l$	$T_1 + c$	$T_2 + d$	$2T_2 + f$	$T_3 + e$	sEU-ACMID, sAnon

STS: secret token size, VKS: verification key size, SS: signature size, GC: computation cost of key generation, SC: signing cost, DGC: cost for generation of delegated signing token VC: Verification cost, λ: security parameter, $|\pi|$: NIZK proof size, l: size of an obfuscated membership checking circuit for a set of order $2^{\lambda/2}$, T_1: computation of obfuscated membership checking circuit, T_2: computation cost of the algorithm $\mathsf{NIZK}.\mathcal{P}$, T_3: Computation cost of $\mathsf{NIZK}.\mathcal{V}$, c, d, e, f: real constants, sEU-ACMI: strong existential unforgeability against chosen message and identity attack, sEU-ACMID: strong existential unforgeability against chosen message and identity attack with delegation of signature, sAnon: strong anonymity.

-*Comparison*: We have compared our proposed schemes qtIRS with existing post-quantum IRS schemes in Table 2. To the best of our knowledge, [17, 29, 26, 27] are the only post-quantum candidates of IRS none of which provide secret key leasing and revocation. The signature size in [17] is short only when the ring size is very small precisely ≤ 8, otherwise, the signature size increases significantly and signing cost is high due to employment of expensive tools such as discrete Gaussian sampling and rejection sampling. Also, verification takes significant time since it involves polynomial multiplications. In [27], the signature size grows linearly with the ring size and the secret key size is significantly large. In [26] gives an IRS using random oracles and collision resistant one-way function where the signature size grows in logarithmic to the ring size and the signature size is significantly large and the signing key size is small ($\sim$ 4 kb). [29] gave a generic construction and two applicable constructions based on NIZK and Markle accumulator. Their first construction is based on a Picnic digital signature that allows a ring of unrestricted size with constant-time setup and relatively large secret key size ($\sim$ 167 kb) and the second one is based on a stateful digital signature that allows a restricted ring size of order 2^{20} and setup time exponential in the tree hight

Table 2: Comparison with existing post-quantum IRS schemes

Ref.	sEU-ACMI	sAnon	Assumption	Signature Delegation
[17]	No	Yes	NTRU lattice (ROM)	No
[29]	No	No	SKE	No
[26]	No	Yes	mSIS (ROM)	No
[27]	No	No	mSIS	No
Our qtIRS	Yes	Yes	$\mathcal{O}$, csNIZK	No
Our D-qtIRS	Yes	Yes	$\mathcal{O}$, csNIZK	Yes

sEU-ACMI: strong existential unforgeability against chosen message and identity attack, sAnon: strong anonymity, sEU-ACMID: strong existential unforgeability against chosen message and identity attack with signature delegation,NTRU: N-th degree truncated polynomial ring, ROM: random oracle model, SKE: secret key encryption, mSIS: module short integer solution, $\mathcal{O}$: secure obfuscator, csNIZK: computationally sound non-interactive zero-knowledge proof system

with secret key size relatively small (~ 5 kb). In both of their constructions, signature size grows with logarithmic ring size which is comparatively better.

In contrast, the secret key size in our scheme is only λ-qubit. The signature size in our qtIRS does not depend on the ring size. Despite signature size being not optimal, our proposed scheme qtIRS features unrestricted ring size, dynamic joining, key-leasing, automatic membership revocation and stronger security than all existing post-quantum IRS candidates. More importantly, the NIZK proof size and obfuscated membership-checking circuit size can be significantly improved in near future to decrease the signature size.

b) D-qtIRS**: quantum-tokenized identity-based ring signature with signature delegation**: We employ signature delegation in qtIRS and proposed D-qtIRS facilitating its users the power of delegation of the signing authority. In D-qtIRS, the lessor is a user who owns a valid identity and has her secret quantum signing token and public verification key. To lease her signing authority to a lessee, the lessor generates a randomized version of her signing token using QOTP together with the corresponding necessary information γ consisting of two randomized verification keys rvk_0, rvk_1 formed by using two obfuscated membership checking circuits along with M-circuit and two NIZK proofs on statement $(\mathsf{ID}||\mathsf{rvk}_i)$-witness $(\mathsf{id}, \hat{\mathbf{x}}_i)$ pair similar to that in qtIRS for $i \in \{0,1\}$ where ID is a ring of user identities and $\hat{\mathbf{x}}_i \in \{0,1\}^\lambda$ is a random string. The lessor sends the randomized signing token through a secure quantum channel and the additional information γ through a secure classical channel. The lessee uses that token along with the additional information to generate a valid signature on any message of his choice concerning a ring of identities pre-selected by the lessee. We make the signing token quantum and design the additional information in such a way that the lessee can sign a message only once. The signature generation involves measuring the quantum token or its Hadamard transform. Once measured, the signing token gets destroyed. However, to allow the lessee to sign a certain number of messages, the lessor can provide a fixed number of randomized signing tokens and corresponding classical information. Apart from secret key leasing, D-qtIRS achieves the following interesting properties.

-*Revocation of delegated signing authority*: In D-qtIRS scheme, the lessor has an additional power to revoke the delegated signing authority from the lessee at any time which information-theoretically ensures that the lessee cannot use the same delegated authority in future. If the lessor wants to revoke the delegated signing authority leased to the lessee, then it sends a dummy message $m \in \{0,1\}$ and asks the lessor to sign m using the delegated authority. If the lessee signs the message and the signature is verified correctly, then from the unforgeability of D-qtIRS, it follows that the delegated token of the lessee is destroyed and it cannot sign any message in the future.

-*Security*: We have formalized the security notion for D-qtIRS. We have defined the strong existential unforgeability against chosen message and identity attacks with signature delegation for D-qtIRS in which the adversary is given the extra power to ask for signature delegation authority. More formally, here the adversary can get the leased key of a specific user and supporting information corresponding to a ring chosen by the challenger which enables him to generate a valid signature on any message of his choice with respect to that ring. It is similar to the signing query in the unforgeability game in the case of the qtIRS. The only difference is that the signature delegation query allows the adversary to decide the message later with some adaptive knowledge. We insist that the adversary wins if either it can generate two distinct valid tuples $(m_1^*, \boldsymbol{\sigma}_1^*, \mathsf{ID})$, $(m_2^*, \boldsymbol{\sigma}_2^*, \mathsf{ID})$ such that ID contains no corrupted user and only one user with delegated key and the supporting information received by the adversary corresponds to the same ring ID or it can produce one valid tuple $(m^*, \boldsymbol{\sigma}, \mathsf{ID}^*)$ such that ID^* contains no corrupted user and is a different ring than ID specified by the challenger. More precisely, we prove the following theorem.

Theorem 3 (Informal). *Our* qtIRS *attains* sEU-ACMID *assuming the existence of obfuscation of membership checking circuits and computational soundness of the NIZK proof system* NIZK.

Since the anonymity of D-qtIRS is the same as the anonymity of the scheme qtIRS, it does not require a separate proof.

-*Efficiency*: We analyze the efficiency of our D-qtIRS scheme in Table 1. Generation of delegated signature tokens involves the generation of two NIZK proofs and the generation of two obfuscated membership-checking circuits yielding the computation cost $2T_2 + f$ for some constant f where T_2 is the computation cost of the generation of an NIZK proof. There is only one measurement on the delegated quantum token in the signature delegation algorithm which costs only constant time. The other costs are similar to that of qtIRS as presented in Table 1.

The communication cost from the lessor to the lessee is $2|\pi| + 2l$ in the classical channel (containing two NIZK proofs, each of length $|\pi|$ and two obfuscated circuits, each of length l) and λ in the quantum channel to issue the λ-qubit randomized signing token. Other communication costs are similar to that of qtIRS.

-*Comparison*: We have compared our scheme D-qtIRS with existing post-quantum IRS schemes in Table 2. In contrast to existing post-quantum IRS schemes [17, 29, 26, 27], our D-qtIRS is the first IRS scheme to enable signature delegation and delegated signing authority revocation along with all other features of qtIRS.

1.2 Technical overview

Obfuscation of membership checking circuit. We use obfuscation $\mathcal{O}$ of membership-checking circuits of [23] as one of the building blocks in our construction for qtIRS and D-qtIRS. We have proved the existence of secure obfuscation of membership-checking circuits by breaking the membership-checking function into a union of a finite number of point functions, based on the work of Hofheinz et al. [20] where they proved the existence of secure obfuscation of point functions. The obfuscation function $\mathcal{O}$ takes as input a circuit C and outputs a circuit $\mathcal{O}(C)$ with the same functionality as C such that from the circuit $\mathcal{O}(C)$, one gets no information about the circuit C which cannot be learned from oracle access to C. In our qtIRS scheme, the public key of a user is a pair of obfuscated circuits the form $(O_{S+\mathbf{x}}, O_{S^\perp+\mathbf{z}})$ where $O_{S+\mathbf{x}} = \mathcal{O}(\mathbf{C}[\mathsf{B}_S, \mathbf{x}])$ and $O_{S^\perp+\mathbf{z}} = \mathcal{O}(\mathbf{C}[\mathsf{B}_S^\perp, \mathbf{z}])$ and $\mathbf{C}[\mathsf{B}_S, \mathbf{x}]$, $\mathbf{C}[\mathsf{B}_S^\perp, \mathbf{z}]$ are membership checking circuits for the cosets $S + \mathbf{x}$ and $S^\perp + \mathbf{z}$ respectively for $S \subseteq \{0,1\}^\lambda$, $\mathbf{x}, \mathbf{z} \in \{0,1\}$. Therefore, $O_{S+\mathbf{x}}$ and $O_{S^\perp+\mathbf{z}}$ takes a λ-bit string $\mathbf{y}$ as input and does the following

$$O_{S+\mathbf{x}}(\mathbf{y}) = \begin{cases} 1 \text{ if } \mathbf{y} \in S + \mathbf{x} \\ 0 \text{ if } \mathbf{y} \notin S + \mathbf{x} \end{cases}$$

$$O_{S^\perp+\mathbf{z}}(\mathbf{y}) = \begin{cases} 1 \text{ if } \mathbf{y} \in S^\perp + \mathbf{z} \\ 0 \text{ if } \mathbf{y} \notin S^\perp + \mathbf{z} \end{cases}$$

To hide the descriptions about the cosets $S + \mathbf{x}$ and $S^\perp + \mathbf{z}$ and more importantly the master secret key S and the secret strings $\mathbf{x}$ and $\mathbf{z}$, we use obfuscated circuits instead of original simple membership checking circuits.
If an adversary can recover the strings from the coset $S + \mathbf{x}$, then it needs to add two distinct strings at a time to get one member of S. For example, suppose the adversary gets two distinct strings $\mathbf{s}_1 + \mathbf{x}, \mathbf{s}_2 + \mathbf{x} \in S + \mathbf{x}$ for $\mathbf{s}_1 \neq \mathbf{s}_2$, then it can calculate

$$(\mathbf{s}_1 + \mathbf{x}) + (\mathbf{s}_2 + \mathbf{x}) = \mathbf{s}_1 + \mathbf{s}_2 \in S$$

Thus, the adversary can recover a member of the master secret key S. Repeating this step for $O(\lambda)$ times with different combinations of elements of $S + \mathbf{x}$, the adversary can get the set S. Then using the descriptions of S and $S + \mathbf{x}$, the adversary can recover $\mathbf{x}$ efficiently. Similarly, it can get $\mathbf{z}$. To avoid this attack, we take obfuscated circuits which hide the circuit and secret coset description.

Usage of computationally sound non-interactive zero-knowledge proof system. Our construction qtIRS and D-qtIRS integrates a computationally sound NIZK proof system with obfuscation $\mathcal{O}$ of membership-checking circuits [23]. For a language $\mathcal{L}$ and its corresponding relation $\mathcal{R}$, a computationally sound NIZK proof system $\mathsf{NIZK} = (\mathsf{Setup}, \mathcal{P}, \mathcal{V})$ consists of three *probabilistic polynomial-time* (PPT) algorithms. The algorithm NIZK.Setup takes a security parameter λ as input and outputs a common reference string crs. The NIZK.$\mathcal{P}$, on input crs and a pair of a statement x and a witness w satisfying the relation $\mathcal{R}$, outputs a ZK proof π. The algorithm NIZK.$\mathcal{V}$ takes as input crs, a statement x and a proof π and outputs 1 (accept) or 0 (reject).

The computational soundness of NIZK implies that if a computationally bounded polynomial time adversary produces a proof π for a statement x which passes the verification, then the probability that $x \notin \mathcal{L}$ is negligible. In

other words, if a statement $x \notin \mathcal{L}$, the adversary can hardly generate a valid proof on x. In our scheme, NIZK plays a vital role in ensuring unforgeability and anonymity. If we use SNARK [22] as NIZK in our qtIRS and D-qtIRS, it will improve the proof size significantly.

Making the secret signing key quantum. In our proposed schemes qtIRS and D-qtIRS, we make the signing key (token) a (pure) quantum state. First, the trusted setup prepares the uniform superposition $|S\rangle = 2^{-\lambda/2} \sum_{\mathbf{s}\in S} |\mathbf{s}\rangle$ over the $\frac{\lambda}{2}$-dimensional subspace $S \subseteq \{0,1\}^\lambda$, chosen uniformly at random. Then it uses the QOTP algorithm to prepare the signing token

$$|S\rangle^{x,z} = 2^{-\lambda/2} \sum_{\mathbf{s}\in S} (-1)^{(\mathbf{s},\mathbf{z})} |\mathbf{s}+\mathbf{x}\rangle = \mathsf{QOTP}\big(|S\rangle, (\mathbf{x},\mathbf{z})\big)$$

for a user with identity id where $\mathbf{x}, \mathbf{z}$ are two uniformly random λ-bit strings and $(\cdot,\cdot)$ denotes inner product. Due to uncloneability of quantum states, users in our scheme cannot arbitrarily copy their signing tokens. Our scheme restricts a user from completely learning its signing token, although allowing the user to sign on an arbitrary message of its choice. Measurement of the state $|S\rangle^{\mathbf{x},\mathbf{z}}$ in the computational basis will return an element from the coset $S+\mathbf{x}$ with probability $2^{-\lambda/2}$. Measurement of the Hadamard transform of the state $|S\rangle^{\mathbf{x},\mathbf{z}}$ will return an element from the coset $S^\perp + \mathbf{z}$ with probability $2^{-\lambda/2}$. Also, the fact that measurement permanently transforms a quantum state is utilized in our scheme.

Quantum tokenized identity-based ring signature. We provide below an overview of our quantum tokenized IRS scheme $\mathsf{qtIRS} = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ employing obfuscation $\mathcal{O}$ for membership-checking circuits and NIZK where qtIRS.Gen, qtIRS.Sign are *quantum polynomial-time algorithms* (QPT) and qtIRS.Setup, qtIRS.Verify are PPT algorithms.

The qtIRS.Setup algorithm is run by a trusted authority who generates system parameters (made public) and master secret key (kept secret). The algorithm qtIRS.Gen is run by the trusted authority that takes an identity id of a user, the master secret key and public parameters and generates a quantum signing token and a verification key. The trusted authority sends the quantum token to that specific user via a secure quantum communication channel. The verification key of the identity holder is published through a classical channel in a public list $\mathcal{L}_{\mathsf{vk}}$ which cannot be tampered by anyone. Any user can join the system at any time without disturbing other users in the system since qtIRS.Gen algorithm does not require any involvement of any other users. This makes our scheme qtIRS dynamic. The signer signs a message using its signing token by calling qtIRS.Sign algorithm. The algorithm qtIRS.Verify can be run by anybody to verify the signature.

To be more specific, the qtIRS.Setup algorithm takes as input the security parameter λ and outputs public parameters $\mathsf{params} = \mathsf{crs}$ and master secret key $\mathsf{msk} = S$ where $\mathsf{crs} \leftarrow \mathsf{NIZK}.\mathcal{P}(\lambda)$ and $S \subseteq \{0,1\}^\lambda$ is a $\frac{\lambda}{2}$-dimensional uniformly random subspace. The algorithm qtIRS.Gen takes params, msk and user identity id as input and outputs the secret quantum signing token $\mathsf{sk}_{\mathsf{id}_i} = |S\rangle^{\mathbf{x},\mathbf{z}}$ and public classical verification key $\mathsf{vk}_{\mathsf{id}_i} = (O_{S+\mathbf{x}_i}, O_{S^\perp+\mathbf{z}_i})$ for each user where $\mathbf{x}_i, \mathbf{z}_i$ are uniformly random λ-bit strings. Then the algorithm qtIRS.Sign takes params, $\mathsf{sk}_{\mathsf{id}_i}$, a message $m \in \{0,1\}$ and a ring of users ID as input such that $\mathsf{id}_i \in \mathsf{ID}$ and outputs a signature $\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi)$ where

$$\boldsymbol{\beta} \in \begin{cases} S + \mathbf{x}_i + \hat{\mathbf{x}} & \text{if } m = 0 \\ S^\perp + \mathbf{z}_i + \hat{\mathbf{x}} & \text{if } m = 1 \end{cases}$$

$$\mathsf{rvk} = \begin{cases} O_{S+\mathbf{x}_i+\hat{\mathbf{x}}} = M[O_{S+\mathbf{x}_i}, \hat{\mathbf{x}}] & \text{if } m = 0 \\ O_{S^\perp+\mathbf{z}_i+\hat{\mathbf{x}}} = M[O_{S^\perp+\mathbf{z}_i}, \hat{\mathbf{x}}] & \text{if } m = 1 \end{cases}$$

$$\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_i, \hat{\mathbf{x}}), \mathsf{ID}||\mathsf{rvk})$$

where rvk is randomized verification key obtained by randomizing $\mathsf{vk}_{\mathsf{id}}$ by choosing a random λ-bit string $\hat{\mathbf{x}}$ and using the M-circuit. The algorithm qtIRS.Verify takes as input params, a message m, a signature $\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi)$ and a ring of user identities ID and outputs 1 (accept) if $(\mathsf{rvk}(\boldsymbol{\beta}) = 1 \wedge 1 \leftarrow \mathsf{NIZK}.\mathcal{V}(\mathsf{crs}, \mathsf{ID}||\mathsf{rvk}, \pi))$ and 0 (reject) otherwise.

For our scheme, we let the language set corresponding to the NIZK to be the set

$$\mathcal{L} = \{\mathsf{ID}||\mathsf{rvk} : \quad \mathsf{ID} = \mathsf{id}_{i_1}||\mathsf{id}_{i_2}||\ldots||\mathsf{id}_{i_n}, \quad \mathsf{rvk} \text{ is a randomized verification key}\}$$

Thus, the statement must be of the specific form ID||rvk i.e., the concatenation of a polynomial number of valid identities and a randomized verification key.

Delegation of signature in qtIRS. Our proposed quantum-tokenized IRS scheme with signature delegation D-qtIRS = (Setup, Gen, Sign, DGen, DSign, Verify) consists of six algorithms where Setup, Gen, Sign, Verify are same as the scheme qtIRS and Dgen and DSign are two additional QPT algorithms.

The lessor runs the algorithm DGen that takes as input public parameters params, secret quantum signing token $\mathsf{sk}_{\mathsf{id}_i}$ and a ring of user identities ID and returns a randomized signing token rsk and additional information γ consisting of two randomized verification keys rvk_0, rvk_1 and two NIZK-proofs π_0,π_1 where

$$\mathsf{rsk} = |S\rangle^{\mathbf{x}_i+\hat{\mathbf{x}}_0,\mathbf{z}_i+\hat{\mathbf{x}}_1}, \mathsf{rvk}_0 = O_{S+\mathbf{x}_i+\hat{\mathbf{x}}_0}, \mathsf{rvk}_1 = O_{S^\perp+\mathbf{z}_i+\hat{\mathbf{x}}_1},$$

$$\pi_0 \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_i, \hat{\mathbf{x}}_0), \mathsf{ID}||\mathsf{rvk}_0), \pi_1 \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_i, \hat{\mathbf{x}}_1), \mathsf{ID}||\mathsf{rvk}_1)$$

for two random λ-bit strings $\hat{\mathbf{x}}_0$ and $\hat{\mathbf{x}}_1$.

The lessor sends the randomized token rsk through a secure quantum communication channel and the additional information $\gamma = (\mathsf{rvk}_0, \mathsf{rvk}_1, \pi_0, \pi_1)$ via a secure classical communication channel to the lessee and keeps $(\hat{\mathbf{x}}_0, \hat{\mathbf{x}}_1)$ secret. The lessee runs the QPT algorithm DSign on input rsk, γ and a message $m \in \{0,1\}$ and outputs a signature $\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi)$ where

$$\boldsymbol{\beta} \in \begin{cases} S + \mathbf{x}_i + \hat{\mathbf{x}}_0 & \text{if } m = 0 \\ S^\perp + \mathbf{z}_i + \hat{\mathbf{x}}_1 & \text{if } m = 1 \end{cases}$$

$$\mathsf{rvk} = \begin{cases} \mathsf{rvk}_0 \text{ if } m = 0 \\ \mathsf{rvk}_1 \text{ if } m = 1 \end{cases}$$

$$\pi = \begin{cases} \pi_0 \text{ if } m = 0 \\ \pi_1 \text{ if } m = 1 \end{cases}$$

Due to uncloneability of the delegated quantum token rsk, the lessee cannot sign two distinct messages since it can either measure rsk or its Hadamard transformation but not both together. Moreover, the ring of user identities ID is chosen by the lessor in DGen and cannot be manipulated by the lessee during DSign. If the lessee chooses a new ring of users ID′, he cannot generate a new NIZK proof π' for the new statement $\mathbf{y}' = \mathsf{ID}'||\mathsf{rvk}$ where m is the message since the does not know the random string $\hat{\mathbf{x}}_m$ in the witness $(\mathsf{id}_i, \hat{\mathbf{x}}_m)$. For the same reason, the lessee cannot generate two different signatures on the same message when provided with only one delegated token. With this technicality, the lessee, endowed with a single leased token and corresponding additional information, can generate at most one signature.

If the lessee does not use the delegated authority and the lessor wants to revoke that authority from the lessee, then the lessor sends a dummy message to the lessee and asks to sign it. Once the lessee signs the message and the signature is correctly verified, it loses the delegation authority. In this way D-qtIRS enables revocation of delegated authority.

1.3 Organization

This paper is organized as follows. In Section 2, we recall all the basic definitions and theorems with proposed security enhancement in the security games of unforgeability and anonymity. In Section 3.1, we explicitly propose our quantum-tokenized identity-based ring signature scheme qtIRS and prove its security in the standard model. In Section 4, we extend qtIRS to a quantum-tokenized IRS scheme with signature delegation D-qtIRS, analyzed the unforgeability of D-qtIRS in the standard model and discuss the revocation of delegated signing authority. Finally, we concluded in Section 5.

2 Preliminaries

Definition 1 (Tensor product of matrices). *[2] Let* $P_{n\times m} = \begin{bmatrix} p_{11} & \cdots & p_{1m} \\ \cdots & \cdots & \cdots \\ p_{n1} & \cdots & p_{nm} \end{bmatrix}$ *and* $Q_{r\times s} = \begin{bmatrix} q_{11} & \cdots & q_{1s} \\ \cdots & \cdots & \cdots \\ q_{r1} & \cdots & q_{rs} \end{bmatrix}$ *be two matrices of order* $n \times m$ *and* $r \times s$ *respectively over a field* F*. Then the tensor product or Kronecker product of* P *and* Q *is defined as*

$$P \otimes Q = \begin{bmatrix} p_{11} & \cdots & p_{1m} \\ \cdots & \cdots & \cdots \\ p_{n1} & \cdots & p_{nm} \end{bmatrix} \otimes Q = \begin{bmatrix} p_{11}Q & \cdots & p_{1m}Q \\ \cdots & \cdots & \cdots \\ p_{n1}Q & \cdots & p_{nm}Q \end{bmatrix}$$

Table 3: Notations

Notations	Name
$\mathbb{R}$	set of all real numbers
$\mathbb{R}^+$	set of all positive real numbers
$\mathbb{N}$	set of all natural numbers
λ	security parameter
$[n]$	$\{1, 2, \cdots, n\}$
$\lvert x\rvert$	the length of a bit-string $x \in \{0,1\}^*$
$\lvert\cdot\rangle$	quantum state (Ket)
$\lvert S\rangle$	uniform superposition of all elements of a set S i.e., $\frac{1}{\sqrt{\lvert S\rvert}} \sum_{s\in S} \lvert s\rangle$
$\otimes$	tensor product or Kronecker product
$\oplus$	bitwise-XOR
H	Hadamard transformation $\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$ for 1-qubit quantum state
$H^{\otimes\lambda}$	Hadamard transformation on λ-qubit quantum state i.e., $H \otimes \cdots \otimes H$ (λ times)
$t \xleftarrow{R} T$	the element t is randomly chosen from the set T

$$= \begin{bmatrix} p_{11}q_{11} & \cdots & p_{11}q_{1s} & \cdots & p_{1m}q_{11} & \cdots & p_{1m}q_{1s} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ p_{11}q_{r1} & \cdots & p_{11}q_{rs} & \cdots & p_{1m}q_{r1} & \cdots & p_{1m}q_{rs} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ p_{n1}q_{11} & \cdots & p_{n1}q_{1s} & \cdots & p_{nm}q_{11} & \cdots & p_{nm}q_{1s} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ p_{n1}q_{r1} & \cdots & p_{n1}q_{rs} & \cdots & p_{nm}q_{r1} & \cdots & p_{nm}q_{rs} \end{bmatrix}.$$

Definition 2 (Tensor product of quantum states). *[2] Let A and B be two 1-qubits which are described by quantum states $\lvert\phi\rangle_A$ and $\lvert\psi\rangle_B$ respectively with the following computational basis representations:*

$$\lvert\phi\rangle_A = \alpha_A \lvert 0\rangle + \beta_A \lvert 1\rangle = \begin{bmatrix}\alpha_A\\ \beta_A\end{bmatrix}, \lvert\psi\rangle_B = \alpha_B \lvert 0\rangle + \beta_B \lvert 1\rangle = \begin{bmatrix}\alpha_B\\ \beta_B\end{bmatrix}$$

where $\alpha_A, \beta_A, \alpha_B, \beta_B \in \mathbb{C}$. The joint quantum state of 2-qubit AB is computed as the tensor product or Kronecker product $\lvert\theta\rangle_{AB}$ of $\lvert\phi\rangle_A$ and $\lvert\psi\rangle_B$ is defined as

$$\lvert\theta\rangle_{AB} = \lvert\phi\rangle_A \otimes \lvert\psi\rangle_B = \begin{bmatrix}\alpha_A\\ \beta_A\end{bmatrix} \otimes \lvert\psi\rangle_B = \begin{bmatrix}\alpha_A \otimes \lvert\psi\rangle_B\\ \beta_A \otimes \lvert\psi\rangle_B\end{bmatrix} = \begin{bmatrix}\alpha_A\alpha_B\\ \alpha_A\beta_B\\ \beta_A\alpha_B\\ \beta_A\beta_B\end{bmatrix}$$

$\lvert\theta\rangle_{AB} \in \mathbb{C}^4$ is the tensor product of the two states $\lvert\phi\rangle_A$ and $\lvert\psi\rangle_B$.
Similarly, if A_1, A_2, $\cdots$, A_n are 1-qubit each represented by states $\lvert\phi_1\rangle$, $\lvert\phi_2\rangle$, $\cdots$, $\lvert\phi_n\rangle$ respectively, then the joint quantum state $\lvert\theta\rangle_{A_1A_2\cdots A_n} \in \mathbb{C}^{2n}$ of n-qubits $A_1A_2\cdots A_n$ corresponds to the tensor product $\lvert\phi_1\rangle \otimes \lvert\phi_2\rangle \otimes \cdots \otimes \lvert\phi_n\rangle$.

Remark 1. Note that

$$H^{\otimes n}\big(\lvert\phi_1\rangle \otimes \cdots \otimes \lvert\phi_n\rangle\big) = \big(H\lvert\phi_1\rangle\big) \otimes \cdots \otimes \big(H\lvert\phi_n\rangle\big)$$

where the Hadamard matrix H transforms the computational basis $\left\{\lvert 0\rangle = \begin{bmatrix}1\\0\end{bmatrix}, \lvert 1\rangle = \begin{bmatrix}0\\1\end{bmatrix}\right\}$ of 1-qubit states in $\mathbb{C}^2$ to the Hadamard basis $\{\lvert+\rangle, \lvert-\rangle\}$ with $H(\lvert 0\rangle) = \lvert+\rangle = \frac{1}{\sqrt{2}}(\lvert 0\rangle + \lvert 1\rangle) = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}$ and $H(\lvert 1\rangle) = \lvert-\rangle = \frac{1}{\sqrt{2}}(\lvert 0\rangle - \lvert 1\rangle) = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}$. For instance

$$\begin{aligned} H^{\otimes 2}\lvert 00\rangle &= H^{\otimes 2}(\lvert 0\rangle \otimes \lvert 0\rangle) = (H\lvert 0\rangle) \otimes (H\lvert 0\rangle) = \lvert+\rangle \otimes \lvert+\rangle = \lvert++\rangle \\ &= \frac{1}{\sqrt{2}}(\lvert 0\rangle + \lvert 1\rangle) \otimes \frac{1}{\sqrt{2}}(\lvert 0\rangle + \lvert 1\rangle) = \frac{1}{2}(\lvert 00\rangle + \lvert 01\rangle + \lvert 10\rangle + \lvert 11\rangle) \\ &= \frac{1}{2}\begin{bmatrix}1 & 1 & 1 & 1\end{bmatrix}^T, \end{aligned}$$

as

$$|00\rangle = |0\rangle \otimes |0\rangle = \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}1\,0\,0\,0\end{bmatrix}^T, |01\rangle = |0\rangle \otimes |1\rangle = \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}0\,1\,0\,0\end{bmatrix}^T,$$
$$|10\rangle = |1\rangle \otimes |0\rangle = \begin{bmatrix}0\\1\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}0\,0\,1\,0\end{bmatrix}^T, |11\rangle = |1\rangle \otimes |1\rangle = \begin{bmatrix}0\\1\end{bmatrix} \otimes \begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}0\,0\,0\,1\end{bmatrix}^T.$$

Alternatively, one can calculate $H^{\otimes 2}(|00\rangle)$ by first computing

$$H^{\otimes 2} = H \otimes H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \otimes \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} = \frac{1}{2}\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & -1 & 1 & -1\\ 1 & 1 & -1 & -1\\ 1 & -1 & -1 & 1\end{bmatrix}$$

followed by $H^{\otimes 2}(|00\rangle) = \frac{1}{2}\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & -1 & 1 & -1\\ 1 & 1 & -1 & -1\\ 1 & -1 & -1 & 1\end{bmatrix}\begin{bmatrix}1\\0\\0\\0\end{bmatrix} = \frac{1}{2}\begin{bmatrix}1\\1\\1\\1\end{bmatrix}$.

We use the abbreviations PPT for probabilistic polynomial time, DPT for deterministic polynomial time and QPT for quantum polynomial time algorithms.

Definition 3 (Measurement of a quantum state in the computational basis). *[2] Let* $|\phi\rangle = |\phi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix}\alpha\\\beta\end{bmatrix} \in \mathbb{C}^2$ *be a 1-qubit quantum state satisfying* $|\alpha|^2 + |\beta|^2 = 1$. *Then the measurement of the state* $|\phi\rangle$ *in the computational basis* $\{|0\rangle, |1\rangle\}$ *returns the basis element* $|0\rangle$ *with probability* $|\alpha|^2$ *and* $|1\rangle$ *with probability* $|\beta|^2$. *For a n-qubit quantum state* $|\Phi\rangle = \alpha_1|\mathbf{b}_1\rangle + \alpha_2|\mathbf{b}_2\rangle + \ldots + \alpha_{2^n}|\mathbf{b}_{2^n}\rangle = \begin{bmatrix}\alpha_1\,\alpha_2\,\ldots\,\alpha_{2^n}\end{bmatrix}^T \in \mathbb{C}^{2^n}$ *with* $\sum_{i=1}^{2^n}|\alpha_i|^2 = 1$, *the measurement in the computational basis* $\{|\mathbf{b}_1\rangle, |\mathbf{b}_2\rangle, \ldots, |\mathbf{b}_{2^n}\rangle\}$ *returns the i-th basis element with probability* $|\alpha_i|^2$ *where* $\mathbf{b}_i \in \{0,1\}^n$ *with i-th entry* 1 *and remaining 0. We denote the function of measurement in the computational basis by* Measure.

Measurement of a quantum state is a one-way transformation as the state gets permanently changed once measurement is done.

$$\mathsf{Measure}(|\phi\rangle) \to |\mathbf{b}_i\rangle \text{ for some } i \in [2^n] \text{ with probability } |\alpha_i|^2.$$
$$\mathsf{Measure}(H^{\otimes n}|\phi\rangle) \to |\hat{\mathbf{b}_i}\rangle \text{ for some } i \in [2^n] \text{ with probability } |\alpha_i|^2 \text{ where } |\hat{\mathbf{b}_i}\rangle \in \{|+\rangle, |-\rangle\}^n.$$

Definition 4 (Quantum one-time pad). *[19] A quantum one-time pad (QOTP) is a QPT algorithm that allows a user to take an n-qubit quantum state* $|\psi\rangle$ *and quantum one-time pads* $\boldsymbol{\alpha} = \alpha_1 \cdots \alpha_n \in \{0,1\}^n, \boldsymbol{\beta} = \beta_1 \cdots \beta_n \in \{0,1\}^n$ *as input and generates the state*

$$|\psi\rangle^{\boldsymbol{\alpha},\boldsymbol{\beta}} = (\otimes_{i\in[n]} X^{\alpha_i}) \cdot (\otimes_{i\in[n]} Z^{\beta_i}) \cdot |\psi\rangle \leftarrow \mathsf{QOTP}\big(|\psi\rangle, (\boldsymbol{\alpha},\boldsymbol{\beta})\big)$$

where $X = \begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$ *and* $Z = \begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}$ *are Pauli matrices with standard notation.*

From the quantum one-time padded state $|\psi\rangle^{\boldsymbol{\alpha},\boldsymbol{\beta}}$, *no one can extract the pads* $(\boldsymbol{\alpha},\boldsymbol{\beta})$ *or the original state* $|\psi\rangle$.

Definition 5 (Negligible, non-negligible, noticeable, overwhelming functions). *[19] A function* $f : \mathbb{N} \to \mathbb{R}^+$ *is said to be*

i. negligible if for all constant $d \geq 0$, $\exists N \in \mathbb{N}$ *such that* $\forall n \geq N$, $f(n) \leq \frac{1}{n^d}$.

ii. non-negligible if $\exists$ *a constant* $d \in \mathbb{N}$ *such that for infinitely many* $n \in \mathbb{N}$, $f(n) \geq \frac{1}{n^d}$.

iii. noticeable if $\exists d, N \in \mathbb{N}$ *such that* $\forall n \geq N$, $f(n) \geq \frac{1}{n^d}$.

iv. overwhelming if $\forall n \in \mathbb{N}$, $f(n) = 1 - \mu(n)$ *for some negligible function* $\mu : \mathbb{N} \to \mathbb{R}^+$.

Definition 6 (Adversarial model). *[19] A polynomial-size adversary $\mathcal{A} = \{\mathcal{A}_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ is a polynomial-size non-uniform sequence of quantum circuits $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$ along with a sequence of polynomial-size auxiliary input states $\{\rho_\lambda\}_{\lambda\in\mathbb{N}}$.*

Definition 7 (Obfuscation). *[20] An obfuscation $\mathcal{O}$ is defined to take as input a function or circuit $C : \{0,1\}^k \to R$ and output an obfuscated circuit $\mathcal{O}(C) : \{0,1\}^k \to R$ such that C and $\mathcal{O}(C)$ are functionally equivalent i.e., $\forall \mathbf{x} \in \{0,1\}^k$, $C(\mathbf{x}) = (\mathcal{O}(C))(\mathbf{x})$ and $\mathcal{O}(C)$ reveals no information about the circuit C which cannot be learnt from the oracle access to C. Obfuscated circuit can be assumed as an oracle for the concerned circuit.*

2.1 Quantum tokenized identity-based ring signature

A quantum tokenized identity-based ring signature scheme $\mathsf{qtIRS} = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ has the same syntax as an identity-based ring signature scheme. However, unlike IRS, the algorithms Gen and Sign are QPT in qtIRS and the secret signing key $\mathsf{sk_{id}}$ is a quantum token instead of a classical key.

Definition 8 (Quantum tokenized identity-based ring signature). *A quantum tokenized identity-based ring signature scheme $\mathsf{qtIRS} = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ works as follows*

$(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$*: This is run by a trusted authority that takes as input a security parameter λ and outputs a master secret key msk and common public parameters params. The trusted authority makes params public and keeps msk secret to itself.*

$(\mathsf{sk_{id}}, \mathsf{vk_{id}}) \leftarrow \mathsf{Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id})$*: This is run by the trusted authority who on input the public parameters params, the master secret key msk and the identity of a user id, outputs a secret quantum signing token $\mathsf{sk_{id}}$ and a public verification key $\mathsf{vk_{id}}$ for the user. The trusted authority sends the quantum signing token $\mathsf{sk_{id}}$ to the user with identity id through a secure quantum communication channel and adds the verification key $\mathsf{vk_{id}}$ corresponding to the identity id into the public verification key list $\mathcal{L}_{\mathsf{vk}}$. A user can fetch the public key $\mathsf{vk_{id}}$ of any user with identity id from the published list $\mathcal{L}_{\mathsf{vk}}$.*

$\sigma \leftarrow \mathsf{Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, m, \mathsf{ID})$*: This is run by the signer with identity id_i who takes as input a message m, its secret quantum signing token $\mathsf{sk}_{\mathsf{id}_i}$ and a ring $\mathsf{ID} = \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_n\}$ consisting of identities of n users chosen by the signer such that $\mathsf{id}_i \in \mathsf{ID}$ and outputs a signature σ.*

$1/0 \leftarrow \mathsf{Verify}(\mathsf{params}, m, \sigma, \mathsf{ID})$*: This is run by the verifier that takes as input the public parameters params, a message m, a signature σ and a ring of identities $\mathsf{ID} = \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_n\}$ and outputs 1 if σ is a valid signature on message m generated by a certain member in the ring ID and 0 otherwise.*

Correctness: We call a quantum-tokenized identity-based ring signature scheme qtIRS correct if for all $\lambda \in \mathbb{N}$ and polynomial $p(\cdot)$,

$$\Pr\left\{ 1 \leftarrow \mathsf{qtIRS.Verify}(\mathsf{params}, m, \sigma, \mathsf{ID}) \;\middle|\; \begin{array}{c} (\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{qtIRS.Setup}(1^\lambda), \\ (\mathsf{sk}_{\mathsf{id}_i}, \mathsf{vk}_{\mathsf{id}_i}) \leftarrow \mathsf{qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_i) \;\; \forall i \in [p(\lambda)] \\ \sigma \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, m, \mathsf{ID}) \\ \mathsf{id}_i \in \mathsf{ID} \subseteq \{\mathsf{id}_1, \ldots, \mathsf{id}_{p(\lambda)}\} \end{array} \right\} = 1$$

Security: We call a quantum-tokenized IRS scheme qtIRS secure if it provides the strong existential unforgeability against adaptive chosen-message-and-identity attack (Def. 9) and strong anonymity (Def. 10).

Definition 9 (Strong existential unforgeability of quantum tokenized identity-based ring signature against adaptive chosen-message-and-identity attack). *A quantum tokenized identity-based ring signature scheme qtIRS attains existential unforgeability against adaptive chosen-message-and-identity attack if for all polynomials $p(\cdot)$, any PPT adversary $\mathcal{A}$ wins the experiment $\mathsf{Exp}^{\mathsf{sEU\text{-}ACMI}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda)$ presented in Fig. 1 only with negligible probability i.e., $\mathsf{Adv}^{\mathsf{sEU\text{-}ACMI}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda) = \Pr[\mathsf{Exp}^{\mathsf{sEU\text{-}ACMI}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda) = 1] < \epsilon$ for some negligible function ϵ.*

Definition 10 (Strong anonymity). *A quantum tokenized identity-based ring signature scheme qtIRS attains anonymity if for all polynomials $p(\cdot)$, any PPT adversary $\mathcal{A}$ wins the experiment $\mathsf{Exp}^{\mathsf{sAnon}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda)$ presented in Fig. 2 only with probability negligibly close to $1/2$ i.e., $\mathsf{Adv}^{\mathsf{sAnon}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda) = \left|\Pr[\mathsf{Exp}^{\mathsf{sAnon}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda) = 1] - \frac{1}{2}\right| < \epsilon$ for some negligible function ϵ.*

Fig. 1: Strong existential unforgeability experiment $\mathsf{Exp}^{\mathsf{sEU\text{-}ACMI}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda)$ for scheme qtIRS against adaptive chosen-message-and-identity attack

Setup phase: The challenger $\mathcal{C}$ first executes the setup algorithm $(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{qtIRS.Setup}(1^\lambda)$, sends the common public parameters params to the adversary $\mathcal{A}$ and keeps msk secret to itself. Further, $\mathcal{C}$ creates a list $\mathcal{L}_{\mathsf{vk}}$ of verification keys corresponding to user identities, a list $\mathcal{L}_C$ of corrupted user identities and a list $\mathcal{L}_S$ of signing queries made by the adversary which are initially empty.
Key generation: The challenger $\mathcal{C}$ runs the key generation algorithm $(\mathsf{sk}_{\mathsf{id}_i}, \mathsf{vk}_{\mathsf{id}_i}) \leftarrow \mathsf{qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_i)\ \forall i \in [p(\lambda)]$ and adds the identity and verification key pair $(\mathsf{id}_i, \mathsf{vk}_{\mathsf{id}_i})$ to the list $\mathcal{L}_{\mathsf{vk}}$. Then the challenger $\mathcal{C}$ publishes the list $\mathcal{L}_{\mathsf{vk}}$ and keeps the secret keys $\{\mathsf{sk}_{\mathsf{id}_1}, \mathsf{sk}_{\mathsf{id}_2}, \ldots, \mathsf{sk}_{\mathsf{id}_{p(\lambda)}}\}$ secret to itself.
Corruption query: The adversary $\mathcal{A}$ makes polynomially many adaptive corruption queries (id_j) with $j \in [p(\lambda)]$ to $\mathcal{C}$. The challenger $\mathcal{C}$ returns the secret key $\mathsf{sk}_{\mathsf{id}_j}$ of the j-th user to $\mathcal{A}$ and adds id_j to the list $\mathcal{L}_C$.
Signing query: The adversary $\mathcal{A}$ makes polynomially many adaptive signing queries $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to $\mathcal{C}$ where m is the message, $\mathsf{ID}_j \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ is a set of identities chosen by $\mathcal{A}$ and $\mathsf{id}_k \in \mathsf{ID}_j$. The challenger $\mathcal{C}$ responds with $\boldsymbol{\sigma}_j \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_j, \mathsf{ID}_j)$ and adds $(m_j, \boldsymbol{\sigma}_j, \mathsf{ID}_j)$ to the list $\mathcal{L}_S$.
Forge: The adversary $\mathcal{A}$ outputs a three tuple $(m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*)$. Then $\mathcal{C}$ returns 1 as output of $\mathsf{Exp}^{\mathsf{sEU\text{-}ACMI}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda)$ if

1. $(m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*) \notin \mathcal{L}_S$,
2. $(\mathsf{id}_j) \notin \mathcal{L}_C\ \forall \mathsf{id}_j \in \mathsf{ID}^*$,
3. $1 \leftarrow \mathsf{qtIRS.Verify}(\mathsf{params}, m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*)$

otherwise, it returns 0.

2.2 Delegation of quantum tokenized identity-based ring signature

The delegation of the quantum tokenized identity-based ring signature scheme $\mathsf{D\text{-}qtIRS} = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Sign}, \mathsf{DGen}, \mathsf{DSign}, \mathsf{Verify})$ is a variant of quantum-tokenized IRS scheme where the algorithms Gen, Sign, DGen and DSign are QPT and Setup, Verify are PPT.

Definition 11 (Quantum tokenized identity-based ring signature). *A quantum delegation of the tokenized identity-based ring signature scheme* $\mathsf{D\text{-}qtIRS} = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Sign}, \mathsf{DGen}, \mathsf{DSign}, \mathsf{Verify})$ *works as follows.*

$(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$*: This is run by a trusted authority that takes as input a security parameter* λ *and outputs a master secret key* msk *and common public parameters* params*. The trusted authority makes* params *public and keeps* msk *secret to itself.*

$(\mathsf{sk}_{\mathsf{id}}, \mathsf{vk}_{\mathsf{id}}) \leftarrow \mathsf{Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id})$*: This is run by the trusted authority who on input the public parameters* params*, the master secret key* msk *and the identity of a user* id*, outputs a secret quantum signing token* $\mathsf{sk}_{\mathsf{id}}$ *and a public verification key* $\mathsf{vk}_{\mathsf{id}}$ *for the user. The trusted authority sends the quantum signing token* $\mathsf{sk}_{\mathsf{id}}$ *to the user with identity* id *through a secure quantum communication channel and adds the verification key* $\mathsf{vk}_{\mathsf{id}}$ *corresponding to the identity* id *to the public verification key list* $\mathcal{L}_{\mathsf{vk}}$*. A user can fetch the public key* $\mathsf{vk}_{\mathsf{id}}$ *of any user with identity* id *from the published list* $\mathcal{L}_{\mathsf{vk}}$*.*

$\sigma \leftarrow \mathsf{Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, m, \mathsf{ID})$*: This is run by the signer with identity* id_i *who takes as input a message* m*, its secret quantum signing token* $\mathsf{sk}_{\mathsf{id}_i}$ *and a ring* $\mathsf{ID} = \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_n\}$ *consisting of identities of* n *users chosen by the signer such that* $\mathsf{id}_i \in \mathsf{ID}$ *and outputs a signature* σ*.*

$(\mathsf{rsk}, \gamma) \leftarrow \mathsf{DGen}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, \mathsf{ID})$*: This is run by the lessor with identity* id_i *who takes as input its secret quantum signing token* $\mathsf{sk}_{\mathsf{id}_i}$ *and a ring* $\mathsf{ID} = \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_n\}$ *consisting of identities of* n *users chosen by the signer such that* $\mathsf{id}_i \in \mathsf{ID}$ *and outputs a rerandomized signing token* rsk *and additional information* γ*. The signer with identity* id_i *or the lessor sends the quantum token* rsk *through a secure quantum communication channel and* γ *through a secure classical communication channel to the lessee. Using the new quantum token* rsk *and the additional information, the lessee can sign a message.*

$\sigma \leftarrow \mathsf{DSign}(m, \mathsf{rsk}, \gamma)$*: This is run by the lessee on the input of the quantum token* rsk *and the additional information* γ *received from the lessor and a message* m*. It outputs the signature* σ*.*

Fig. 2: Strong anonymity experiment $\mathsf{Exp}^{\mathsf{sAnon}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda)$ for scheme qtIRS

Setup phase: The challenger $\mathcal{C}$ first executes the setup algorithm $(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{qtIRS.Setup}(1^\lambda)$, sends the common public parameters params to the adversary $\mathcal{A}$ and keeps msk secret to itself. Further, $\mathcal{C}$ creates a list $\mathcal{L}_{\mathsf{vk}}$ of verification keys corresponding to user identities, a list $\mathcal{L}_C$ of corrupted user identities and a list $\mathcal{L}_S$ of signing queries made by the adversary which are initially empty.
Key generation: The challenger $\mathcal{C}$ runs the key generation algorithm $(\mathsf{sk}_{\mathsf{id}_i}, \mathsf{vk}_{\mathsf{id}_i}) \leftarrow \mathsf{qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_i)$ $\forall i \in [p(\lambda)]$ and adds the identity and verification key pair $(\mathsf{id}_i, \mathsf{vk}_{\mathsf{id}_i})$ to the list $\mathcal{L}_{\mathsf{vk}}$. Then the challenger $\mathcal{C}$ publishes the list $\mathcal{L}_{\mathsf{vk}}$ and keeps the secret keys $\{\mathsf{sk}_{\mathsf{id}_1}, \mathsf{sk}_{\mathsf{id}_2}, \ldots, \mathsf{sk}_{\mathsf{id}_{p(\lambda)}}\}$ secret to itself.
Corruption query: The adversary $\mathcal{A}$ makes polynomially many adaptive corruption queries (id_j) with $j \in [p(\lambda)]$ to $\mathcal{C}$. The challenger $\mathcal{C}$ returns the secret key $\mathsf{sk}_{\mathsf{id}_j}$ of the j-th user to $\mathcal{A}$ and adds id_j to the list $\mathcal{L}_C$.
Signing query: The adversary $\mathcal{A}$ makes polynomially many adaptive signing queries $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to $\mathcal{C}$ where m is the message, $\mathsf{ID}_j \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ is a set of identities chosen by $\mathcal{A}$ and $\mathsf{id}_k \in \mathsf{ID}_j$. The challenger $\mathcal{C}$ responds with $\mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_j, \mathsf{ID}_j)$ and adds $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to the list $\mathcal{L}_S$.
Forge: The adversary $\mathcal{A}$ makes a query of the form $(\mathsf{id}_{i_0}, \mathsf{id}_{i_1}, m^*, \mathsf{ID}^*)$ to $\mathcal{C}$ for message m^* where $\mathsf{ID}^* \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ is ring of identities chosen by $\mathcal{A}$ and $\mathsf{id}_{i_0}, \mathsf{id}_{i_1} \in \mathsf{ID}^*$ are two distinct identities. The challenger $\mathcal{C}$ uniformly samples a bit $b \leftarrow \{0,1\}$ and returns $\mathsf{qtIRS.Sign}(\mathsf{sk}_{\mathsf{id}_{i_b}}, m^*, \mathsf{ID}^*)$ to $\mathcal{A}$.
Finally, $\mathcal{A}$ outputs a guess bit b' for b. If

1. $\mathsf{id}_{i_0}, \mathsf{id}_{i_1} \notin \mathcal{L}_C$,
2. $(\mathsf{id}_{i_0}, m^*, \mathsf{ID}^*), (\mathsf{id}_{i_1}, m^*, \mathsf{ID}^*) \notin \mathcal{L}_S$,
3. $b' = b$

then $\mathcal{C}$ returns 1 as output of $\mathsf{Exp}^{\mathsf{sAnon}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda)$; otherwise, it returns 0.

$1/0 \leftarrow \mathsf{Verify}(\mathsf{params}, m, \sigma, \mathsf{ID})$*: This is run by the verifier that takes as input the public parameters* params*, a message* m*, a signature* σ *and a ring of identities* $\mathsf{ID} = \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_n\}$ *and outputs* 1 *if* σ *is a valid signature on message* m *generated by a certain member in the ring* ID *and* 0 *otherwise.*

Correctness: We call a delegation of quantum tokenized identity-based ring signature scheme $\mathsf{del\text{-}qtIRS}$ correct if for all $\lambda \in \mathbb{N}$ and polynomial $p(\cdot)$, the following conditions hold.

$$\Pr\left\{1 \leftarrow \mathsf{D\text{-}qtIRS.Verify}(\mathsf{params}, m, \sigma, \mathsf{ID}) \;\middle|\; \begin{array}{c}(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{D\text{-}qtIRS.Setup}(1^\lambda),\\ (\mathsf{sk}_{\mathsf{id}_i}, \mathsf{vk}_{\mathsf{id}_i}) \leftarrow \mathsf{D\text{-}qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_i)\\ \forall i \in [p(\lambda)]\\ \sigma \leftarrow \mathsf{D\text{-}qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, m, \mathsf{ID})\\ \mathsf{id}_i \in \mathsf{ID} \subseteq \{\mathsf{id}_1, \ldots, \mathsf{id}_{p(\lambda)}\}\end{array}\right\} = 1$$

$$\Pr\left\{1 \leftarrow \mathsf{D\text{-}qtIRS.Verify}(\mathsf{params}, m, \sigma, \mathsf{ID}) \;\middle|\; \begin{array}{c}(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{D\text{-}qtIRS.Setup}(1^\lambda),\\ (\mathsf{sk}_{\mathsf{id}_i}, \mathsf{vk}_{\mathsf{id}_i}) \leftarrow \mathsf{D\text{-}qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_i)\\ \forall i \in [p(\lambda)],\\ (\mathsf{rsk}, \gamma) \leftarrow \mathsf{D\text{-}qtIRS.DGen}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, \mathsf{ID}),\\ \sigma \leftarrow \mathsf{D\text{-}qtIRS.DSign}(m, \mathsf{rsk}, \gamma),\\ \mathsf{id}_i \in \mathsf{ID} \subseteq \{\mathsf{id}_1, \ldots, \mathsf{id}_{p(\lambda)}\}\end{array}\right\} = 1$$

Security: We call a quantum-tokenized IRS scheme with signature delegation $\mathsf{D\text{-}qtIRS}$ secure if it provides srong existential unforgeability against adaptive chosen-message-and-identity attack allowing signature delegation (Def. 12).

Definition 12 (Strong existential unforgeability of delegation of quantum tokenized identity-based ring signature against adaptive chosen-message-and-identity attack). *A delegation of quantum tokenized identity-based ring signature scheme* $\mathsf{D\text{-}qtIRS}$ *attains existential unforgeability against adaptive chosen-message-and-identity attack if for all polynomials* $p(\cdot)$*, any PPT adversary* $\mathcal{A}$ *wins the experiment* $\mathsf{Exp}^{\mathsf{sEU\text{-}ACMID}}_{\mathcal{A},\mathsf{D\text{-}qtIRS}}(\lambda)$ *presented in Fig. 3 only with negligible probability i.e.,* $\mathsf{Adv}^{\mathsf{sEU\text{-}ACMID}}_{\mathcal{A},\mathsf{D\text{-}qtIRS}}(\lambda) = \Pr[\mathsf{Exp}^{\mathsf{sEU\text{-}ACMID}}_{\mathcal{A},\mathsf{D\text{-}qtIRS}}(\lambda) = 1] < \epsilon$ *for some negligible function* ϵ*.*

Fig. 3: Strong existential unforgeability experiment $\mathsf{Exp}^{\mathsf{sEU\text{-}ACMID}}_{\mathcal{A},\mathsf{D\text{-}qtIRS}}(\lambda)$ for scheme D-qtIRS against adaptive chosen-message-and-identity attack with signature delegation

Setup phase: The challenger $\mathcal{C}$ first executes the setup algorithm $(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{D\text{-}qtIRS.Setup}(1^\lambda)$, sends the common public parameters params to the adversary $\mathcal{A}$ and keeps msk secret to itself. Further, $\mathcal{C}$ creates a list $\mathcal{L}_{\mathsf{vk}}$ of verification keys corresponding to user identities, a list $\mathcal{L}_C$ of corrupted user identities and a list $\mathcal{L}_{\mathsf{del}}$ of signing token delegation queries made by the adversary which are initially empty.
Key generation: The challenger $\mathcal{C}$ runs the key generation algorithm $(\mathsf{sk}_{\mathsf{id}_i}, \mathsf{vk}_{\mathsf{id}_i}) \leftarrow \mathsf{D\text{-}qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_i)$ $\forall i \in [p(\lambda)]$ and adds the identity and verification key pair $(\mathsf{id}_i, \mathsf{vk}_{\mathsf{id}_i})$ to the list $\mathcal{L}_{\mathsf{vk}}$. Then the challenger $\mathcal{C}$ publishes the list $\mathcal{L}_{\mathsf{vk}}$ and keeps the secret keys $\{\mathsf{sk}_{\mathsf{id}_1}, \mathsf{sk}_{\mathsf{id}_2}, \ldots, \mathsf{sk}_{\mathsf{id}_{p(\lambda)}}\}$ secret to itself.
Corruption query: The adversary $\mathcal{A}$ makes polynomially many adaptive corruption queries (id_j) with $j \in [p(\lambda)]$ to $\mathcal{C}$. The challenger $\mathcal{C}$ returns the secret key $\mathsf{sk}_{\mathsf{id}_j}$ of the j-th user to $\mathcal{A}$ and adds id_j to the list $\mathcal{L}_C$.
Signing query: The adversary $\mathcal{A}$ makes polynomially many adaptive signing queries $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to $\mathcal{C}$ where m is the message, $\mathsf{ID}_j \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ is a set of identities chosen by $\mathcal{A}$ and $\mathsf{id}_k \in \mathsf{ID}_j$. The challenger $\mathcal{C}$ responds with $\boldsymbol{\sigma}_j \leftarrow \mathsf{D\text{-}qtIDRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_j, \mathsf{ID}_j)$ and adds $(m_j, \boldsymbol{\sigma}_j, \mathsf{ID}_j)$ to the list $\mathcal{L}_S$.
Delegation query: The adversary sends signing token delegation request of the form (id_j) with $j \in [p(\lambda)]$. The challenger chooses a ring of identities $\mathsf{ID} \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ such that $\mathsf{id}_j \in \mathsf{ID}$ and responds with $(\mathsf{rsk}, \gamma) \leftarrow \mathsf{D\text{-}qtIRS.DGen}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_j}, \mathsf{ID})$ and adds (id_j) to the list $\mathcal{L}_{\mathsf{del}}$.
Forge: Suppose $\mathcal{L}_{\mathsf{del}} \cap \mathsf{ID} = \{\mathsf{id}_j\}$. The challenger $\mathcal{C}$ returns 1 as output of $\mathsf{Exp}^{\mathsf{sEU\text{-}ACMID}}_{\mathcal{A},\mathsf{D\text{-}qtIRS}}(\lambda)$ if any one of the following two holds; otherwise, it returns 0.

- The adversary outputs two valid tuples $(m_1^*, \boldsymbol{\sigma}_1^*, \mathsf{ID})$ and $(m_2^*, \boldsymbol{\sigma}_2^*, \mathsf{ID})$ such that
  1. $(m_1^*, \boldsymbol{\sigma}_1^*, \mathsf{ID}), (m_2^*, \boldsymbol{\sigma}_2^*, \mathsf{ID}) \notin \mathcal{L}_S$,
  2. $(m_1^*, \boldsymbol{\sigma}_1^*) \neq (m_2^*, \boldsymbol{\sigma}_2^*)$,
  3. $(\mathsf{id}) \notin \mathcal{L}_C$ $\forall \mathsf{id} \in \mathsf{ID}$,
  4. $1 \leftarrow \mathsf{D\text{-}qtIRS.Verify}(\mathsf{params}, m_1^*, \boldsymbol{\sigma}_1^*, \mathsf{ID})$ and $1 \leftarrow \mathsf{D\text{-}qtIRS.Verify}(\mathsf{params}, m_2^*, \boldsymbol{\sigma}_2^*, \mathsf{ID})$.
- The adversary outputs one valid tuple $(m, \boldsymbol{\sigma}, \mathsf{ID}^*)$ such that
  1. $(m, \boldsymbol{\sigma}, \mathsf{ID}^*) \notin \mathcal{L}_S$,
  2. $(\mathsf{id}) \notin \mathcal{L}_C$ $\forall \mathsf{id} \in \mathsf{ID}^*$,
  3. $\mathsf{ID} \neq \mathsf{ID}^*$,
  4. $1 \leftarrow \mathsf{D\text{-}qtIRS.Verify}(\mathsf{params}, m, \boldsymbol{\sigma}, \mathsf{ID}^*)$.

2.3 Non-interactive zero-knowledge proof system

Definition 13. *[6] Let $\mathcal{L}$ be an NP language and its corresponding relation be $\mathcal{R}$ where a statement-witness pair $(x, w) \in \mathcal{R}$ if $\mathcal{R}(x, w) = 1$ for $x \in \mathcal{L}$. (A public statement $x \in \mathcal{L}$ may have more than one witness satisfying the relation $\mathcal{R}$.) A non-interactive zero-knowledge proof system $\mathsf{NIZK} = (\mathsf{Setup}, \mathcal{P}, \mathcal{V})$ is a tuple of three PPT algorithms satisfying the following requirements where Setup generates common reference string and trapdoor, $\mathcal{P}$ is the non-interactive zero-knowledge proof generating algorithm and $\mathcal{V}$ is the proof verification algorithm.*

$\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda)$*: This is run by a trusted party who takes as input a security parameter $\lambda \in \mathbb{N}$ and outputs a common reference string crs and publishes it publicly.*

$\pi \leftarrow \mathcal{P}(\mathsf{crs}, w, x)$*: The prover runs this non-interactive algorithm on input crs, a public statement $x \in \mathcal{L}$ and a secret witness w satisfying $\mathcal{R}(x, w) = 1$ and generates a zero-knowledge (ZK)-proof π.*

$b \leftarrow \mathcal{V}(\mathsf{crs}, x, \pi)$*: The verifier runs this algorithm that takes crs, a public statement $x \in \mathcal{L}$ and a ZK-proof π as input and outputs a bit $b \in \{0, 1\}$, indicating success if $b = 1$ and failure if $b = 0$.*

Completeness: A proof system $\mathsf{NIZK} = (\mathsf{Setup}, \mathcal{P}, \mathcal{V})$ is complete if

$$\mathsf{Pr}[1 \leftarrow \mathsf{NIZK}.\mathcal{V}(\mathsf{crs}, x, \pi) \big| \mathsf{crs} \leftarrow \mathsf{NIZK.Setup}(1^\lambda), \pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, w, x)] = 1 \ \ \forall \lambda \in \mathbb{N}, \forall x \in \mathcal{L}.$$

Soundness: A proof system $\mathsf{NIZK} = (\mathsf{Setup}, \mathcal{P}, \mathcal{V})$ is computationally sound if

$$\mathsf{Pr}[x \notin \mathcal{L} \big| \mathsf{crs} \leftarrow \mathsf{NIZK.Setup}(1^\lambda), (x, \pi) \leftarrow \mathcal{A}(\mathsf{crs}), 1 \leftarrow \mathsf{NIZK}.\mathcal{V}(\mathsf{crs}, x, \pi)] \leq \mathsf{neg}(\lambda) \ \ \forall \lambda \in \mathbb{N}$$

and all PPT adversary $\mathcal{A}$ where $\mathsf{neg}(\cdot)$ is a negligible function.

Definition 14. *[6] A proof system* NIZK *is said to be statistically zero-knowledge if* $\exists$ *a simulator* $\mathcal{S}$ *such that* $\mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, w, x) \sim \mathcal{S}(\mathsf{crs}, \mathsf{td}, x)$ $\forall \lambda \in \mathbb{N}$, $\forall x \in \mathcal{L}$, *witness* w *and* $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{NIZK.Setup}(1^\lambda)$ *where* td *is the trapdoor,* $\sim$ *denotes statistical indistinguishability.*

Moreover, if the above indistinguishability holds for all possible choices of crs*, then the system* NIZK *is called unconditionally zero-knowledge.*

3 qtIRS: Quantum-tokenized IRS

3.1 Description of the protocol

In this section, we propose a quantum tokenized ID-based ring signature scheme $\mathsf{qtIRS} = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ that uses the obfuscation $\mathcal{O}$ of membership checking circuits (Def. 7), a non-interactive zero-knowledge proof system $\mathsf{NIZK} = (\mathsf{Setup}, \mathcal{P}, \mathcal{V})$ (Def. 13) and a quantum one-time padding algorithm QOTP (Fig. 4) and a randomized hash-function $h : \{0,1\}^* \Rightarrow \{0,1\}^{2\lambda}$.

Fig. 4: Quantum one-time pad for $|S\rangle$

From Def. 4, we have

$$X|0\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = |1\rangle, X|1\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle$$

$$Z|0\rangle = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle, Z|1\rangle = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} = -|1\rangle$$

Therefore, for $s, x, z \in \{0,1\}$,

$$X^1|s\rangle = |s+1\rangle, \quad X^0|s\rangle = |s\rangle = |s+0\rangle, \quad Z^1|s\rangle = (-1)^{s\cdot 1}|s\rangle \text{ and } Z^0|s\rangle = |s\rangle = (-1)^{s\cdot 0}|s\rangle$$

$$\implies X^x|s\rangle = |s+x\rangle \text{ and } Z^z|s\rangle = (-1)^{s\cdot z}|s\rangle$$
$$\implies X^x \cdot Z^z|s\rangle = (-1)^{s\cdot z}|s+x\rangle$$

Hence, for the $\frac{\lambda}{2}$-dimensional subspace $S \subseteq \{0,1\}^\lambda$ and $\mathbf{x} = x_1 \cdots x_\lambda \in \{0,1\}^\lambda$, $\mathbf{z} = z_1 \cdots z_\lambda \in \{0,1\}^\lambda$, we have

$$\begin{aligned}
\mathsf{QOTP}(|S\rangle, (\mathbf{x}, \mathbf{z})) &= (\otimes_{i\in[\lambda]} X^{x_i}) \cdot (\otimes_{i\in[\lambda]} Z^{z_i}) \sum_{\mathbf{s}\in S} |\mathbf{s}\rangle = \sum_{\mathbf{s}\in S} (\otimes_{i\in[\lambda]} X^{x_i}) \cdot (\otimes_{i\in[\lambda]} Z^{z_i}) |\mathbf{s}\rangle \\
&= \sum_{s_1\cdots s_\lambda \in S} (\otimes_{i\in[\lambda]} X^{x_i} \cdot Z^{z_i}) \otimes_{i\in[\lambda]} |s_i\rangle = \sum_{s_1\cdots s_\lambda \in S} \otimes_{i\in[\lambda]} X^{x_i} \cdot Z^{z_i} |s_i\rangle \\
&= \sum_{s_1\cdots s_\lambda \in S} \otimes_{i\in[\lambda]} (-1)^{s_i\cdot z_i} |s_i + x_i\rangle = \sum_{s_1\cdots s_\lambda \in S} (-1)^{\sum_{i\in[\lambda]} s_i\cdot z_i} \otimes_{i\in[\lambda]} |s_i + x_i\rangle \\
&= \sum_{\mathbf{s}\in S} (-1)^{(\mathbf{s},\mathbf{z})} |\mathbf{s} + \mathbf{x}\rangle
\end{aligned}$$

Our qtIRS works as follows.

$(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$: This algorithm is run by a trusted authority who takes as input a security parameter $\lambda \in \mathbb{N}$ and randomly samples a $\frac{\lambda}{2}$-dimensional subspace $S \subset \{0,1\}^\lambda$. By running $\mathsf{crs} \leftarrow \mathsf{NIZK.Setup}(1^\lambda)$, it generates a common reference string crs. It sets the master secret key $\mathsf{msk} = S$ and the common public parameters $\mathsf{params} = \mathsf{crs}$ which includes λ. It keeps msk secret to itself and publishes params.

$(\mathsf{sk}_{\mathsf{id}}, \mathsf{vk}_{\mathsf{id}}) \leftarrow \mathsf{Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id})$: This QPT algorithm is run by the trusted authority who takes as input params, the master secret key msk and an identity $\mathsf{id} \in \{0,1\}^*$ of a user. The following notations are used.

- $(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{i=1}^{\lambda} (\alpha_i \cdot \beta_i) (\text{mod } 2)$ denotes the inner product of $\boldsymbol{\alpha} = \alpha_1 \alpha_2 \cdots \alpha_\lambda \in \{0,1\}^\lambda$ and $\boldsymbol{\beta} = \beta_1 \beta_2 \cdots \beta_\lambda \in \{0,1\}^\lambda$.
- B_S and $\mathsf{B}_{S^\perp}$ are two matrices of order $\frac{\lambda}{2} \times \lambda$ whose row sets form bases of the respective subspaces S and $S^\perp$ where $S^\perp = \{\mathbf{u} \in \{0,1\} | (\mathbf{u}, \mathbf{s}) = 0 \ \ \forall \mathbf{s} \in S\}$.
- $\mathbf{C}[\mathsf{B}_S, \boldsymbol{\alpha}]$ is the circuit embedded with the description of the matrix B_S and the bit-string $\boldsymbol{\alpha} \in \{0,1\}^\lambda$ that checks membership of a bit-string into the coset $S + \boldsymbol{\alpha}$ i.e.,

$$\mathbf{C}[\mathsf{B}_S, \boldsymbol{\alpha}](\mathbf{y}) = \begin{cases} 1 \text{ if } \mathbf{y} \in S + \boldsymbol{\alpha} \\ 0 \text{ if } \mathbf{y} \notin S + \boldsymbol{\alpha} \end{cases}$$

- The circuit $\mathbf{C}[\mathsf{B}_{S^\perp}, \boldsymbol{\beta}]$ is defined in a similar manner for some $\boldsymbol{\beta} \in \{0,1\}^\lambda$.

The following steps are executed by the trusted authority.

i. It extracts the security parameter λ from params and the $\frac{\lambda}{2}$-dimensional subspace S from msk.

ii. It samples uniformly random $\mathbf{r} \xleftarrow{R} \{0,1\}^{2\lambda}$ and computes $(\mathbf{x}, \mathbf{z}) \leftarrow h(\mathbf{r}, \mathsf{id})$.

iii. It computes $|S\rangle^{\mathbf{x},\mathbf{z}} = \sum_{\mathbf{s} \in S} (-1)^{(\mathbf{s},\mathbf{z})} |\mathbf{s} + \mathbf{x}\rangle \leftarrow \mathsf{QOTP}\big(|S\rangle, (\mathbf{x}, \mathbf{z})\big)$ (see Fig. 4) and sets its quantum signing token $\mathsf{sk}_{\mathsf{id}} = |S\rangle^{\mathbf{x},\mathbf{z}}$. Note that $H^{\otimes\lambda} |S\rangle^{\mathbf{x},\mathbf{z}} = |S^\perp\rangle^{\mathbf{z},\mathbf{x}} = \sum_{\hat{\mathbf{s}} \in S^\perp} (-1)^{(\hat{\mathbf{s}},\mathbf{x})} |\hat{\mathbf{s}} + \mathbf{z}\rangle$.

iv. It computes obfuscated membership checking circuits $O_{S+\mathbf{x}} \leftarrow \mathcal{O}(\mathbf{C}[\mathsf{B}_S, \mathbf{x}])$ and $O_{S^\perp+\mathbf{z}} \leftarrow \mathcal{O}(\mathbf{C}[\mathsf{B}_{S^\perp}, \mathbf{z}])$ and sets the verification key $\mathsf{vk}_{\mathsf{id}} = (O_{S+\mathbf{x}}, O_{S^\perp+\mathbf{z}})$.

v. It sends $\mathsf{sk}_{\mathsf{id}} = |S\rangle^{\mathbf{x},\mathbf{z}}$ to the user with identity id via a secure quantum communication channel and publishes the user idenity and corresponding verification key $(\mathsf{id}, \mathsf{vk}_{\mathsf{id}}) = (\mathsf{id}, (O_{S+\mathbf{x}}, O_{S^\perp+\mathbf{z}}))$ in the public verification key list $\mathcal{L}_{\mathsf{vk}}$. Note that this QPT interactive protocol normally needs quantum communication between the trusted authority and the user with identity id. But it is possible to enable the same key generation with classical communication between the trusted authority and the users supported with only local quantum computation using the concept of '3-message protocol' proposed by Shmueli [19].

$\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi) \leftarrow \mathsf{Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, m, \mathsf{ID})$: The signer with identity id_i takes as input the public parameters params, its quantum signing token $\mathsf{sk}_{\mathsf{id}_i} = |S\rangle^{\mathbf{x}_i,\mathbf{z}_i}$, a message $m \in \{0,1\}$ and a ring of identities $\mathsf{ID} = \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_n\}$ of n users of its choice with $\mathsf{id}_i \in \mathsf{ID}$. The signer accesses the list $\mathcal{L}_{\mathsf{vk}}$ to extract the verification key $\mathsf{vk}_{\mathsf{id}_i} = (O_{S+\mathbf{x}_i}, O_{S^\perp+\mathbf{z}_i})$, makes use of M-circuit that works as described in Fig. 5 and proceeds as follows.

Fig. 5: M-circuit $M[O_T, \mathbf{x}]$

- $T \subseteq \{0,1\}^\lambda$, $\mathbf{x} \in \{0,1\}^\lambda$.
- O_T be an obfuscated membership checking circuit that checks membership into T.
- O_T and the string $\mathbf{x}$ are embedded in M-circuit $M[O_T, \mathbf{x}]$.
- $M[O_T, \mathbf{x}]$ checks the membership of any string $\mathbf{y} \in \{0,1\}^\lambda$ into the set $T + \mathbf{x}$ by outputting $O_T(\mathbf{y} - \mathbf{x})$.
- $M[O_T, \mathbf{x}](\mathbf{y}) = O_T(\mathbf{y} - \mathbf{x})$.
- Note that $M[O_T, \mathbf{x}](\mathbf{y}) = 1 \iff O_T(\mathbf{y} - \mathbf{x}) = 1 \iff \mathbf{y} - \mathbf{x} \in T \iff \mathbf{y} \in T + \mathbf{x}$.

i. It extracts the security parameter λ from the common reference string crs in the public parameter params.

ii. It samples uniformly $\hat{\mathbf{x}} \xleftarrow{R} \{0,1\}^\lambda$ and computes

$$\mathsf{rvk} = \begin{cases} O_{S+\mathbf{x}_i+\hat{\mathbf{x}}} = M[O_{S+\mathbf{x}_i}, \hat{\mathbf{x}}] & \text{if } m = 0 \\ O_{S^\perp+\mathbf{z}_i+\hat{\mathbf{x}}} = M[O_{S^\perp+\mathbf{z}_i}, \hat{\mathbf{x}}] & \text{if } m = 1 \end{cases}$$

We represent an obfuscated circuit as a binary string.

iii. It sets $\mathbf{y} = \mathsf{ID}||\mathsf{rvk} \in \{0,1\}^*$ and computes a non-interactive zero-knowledge proof of the randomization of the corresponding verification key $\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_i, \hat{\mathbf{x}}), \mathbf{y})$.

iv. It sets $\boldsymbol{\beta} = \boldsymbol{\alpha} + \hat{\mathbf{x}}$ where

$$\boldsymbol{\alpha} = \begin{cases} \mathsf{Measure}\big(|S\rangle^{\mathbf{x}_i,\mathbf{z}_i}\big) & \text{if } m = 0 \\ \mathsf{Measure}\big(H^{\otimes\lambda}(|S\rangle^{\mathbf{x}_i,\mathbf{z}_i})\big) & \text{if } m = 1 \end{cases}$$

and $\mathsf{Measure}$ is the quantum transformation that measures a quantum state in the computational basis (see Def. 3) and H is the Hadamard transformation (see Remark 1). More concretely, $\mathsf{Measure}\big(|S\rangle^{\mathbf{x},\mathbf{z}}\big)$ returns a string from the set $S + \mathbf{x}$ if $m = 0$ and $\mathsf{Measure}\big(H^{\otimes\lambda}(|S\rangle^{\mathbf{x},\mathbf{z}})\big) = \mathsf{Measure}\big(|S^{\perp}\rangle^{\mathbf{z},\mathbf{x}}\big)$ returns a string from the set $S^{\perp} + \mathbf{z}$ if $m = 1$. Consequently, $\boldsymbol{\beta} \in S + \mathbf{x}_i + \hat{\mathbf{x}}$ if $m = 0$ and $\boldsymbol{\beta} \in S^{\perp} + \mathbf{z}_i + \hat{\mathbf{x}}$ if $m = 1$.

vi. It outputs $\boldsymbol{\sigma} = (\boldsymbol{\beta}, \hat{\mathsf{vk}}, \pi)$ as a signature on the message m with respect to the ring ID.

$b \leftarrow \mathsf{Verify}(\mathsf{params}, m, \boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi), \mathsf{ID})$: On input of $\mathsf{params} = \mathsf{crs}$, a message $m \in \{0,1\}$, a signature $\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi)$ including a randomized verification key rvk where

$$\mathsf{rvk} = O = \begin{cases} O_{S+\mathbf{x}_i+\hat{\mathbf{x}}} & \text{if } m = 0 \\ O_{S^{\perp}+\mathbf{z}_i+\hat{\mathbf{x}}} & \text{if } m = 1 \end{cases}$$

and non-interactive ZK-proof π of the randomization of the corresponding verification key, the verifier performs the following.

i. It sets $\mathbf{y} = \mathsf{ID}||\mathsf{rvk} \in \{0,1\}^*$.

ii. It returns

$$b = \begin{cases} 1 \text{ if } \ 1 \leftarrow \mathsf{NIZK}.\mathcal{V}(\mathsf{crs}, \mathbf{y}, \pi) \wedge O(\boldsymbol{\beta}) = 1 \\ 0 \text{ otherwise} \end{cases}$$

Correctness: Let $\lambda \in \mathbb{N}$, $(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{qtIRS.Setup}(1^{\lambda})$, $(\mathsf{sk}_{\mathsf{id}_i}, \mathsf{vk}_{\mathsf{id}_i}) \leftarrow \mathsf{qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_i)$ $\forall i \in [p(\lambda)]$ for some polynomial $p(\cdot)$ and $\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, m, \mathsf{ID})$ where $\mathsf{id}_i \in \mathsf{ID} \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ and $\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_i, \hat{\mathbf{x}}), \mathsf{ID}||\mathsf{rvk})$.

By correctness of the non-interactive ZK-proof system NIZK, it holds that $1 \leftarrow \mathsf{NIZK}.\mathcal{V}(\mathsf{crs}, \mathsf{ID}||\mathsf{rvk}, \pi)$. Note that if $m = 0$, then $\boldsymbol{\beta} \in S + \mathbf{x}_i + \hat{\mathbf{x}}$ and $O = \mathsf{rvk} = O_{S+\mathbf{x}_i+\hat{\mathbf{x}}}$. Hence, $O(\boldsymbol{\beta}) = O_{S+\mathbf{x}_i+\hat{\mathbf{x}}}(\boldsymbol{\beta}) = 1$. On the other hand, if $m = 1$, then $\boldsymbol{\beta} \in S^{\perp} + \mathbf{z}_i + \hat{\mathbf{x}}$ and $O = \mathsf{rvk} = O_{S^{\perp}+\mathbf{z}_i+\hat{\mathbf{x}}}$. Hence, $O(\boldsymbol{\beta}) = O_{S^{\perp}+\mathbf{z}_i+\hat{\mathbf{x}}}(\boldsymbol{\beta}) = 1$.

Thus, $1 \leftarrow \mathsf{qtIRS.Verify}(\mathsf{params}, m, \boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi), \mathsf{ID})$.

Remark 2. Our proposed scheme qtIRS is dynamic since any user with a certain identity id can join the system anytime by requesting the trusted authority to generate its key pair $(\mathsf{sk}_{\mathsf{id}}, \mathsf{vk}_{\mathsf{id}})$ without disturbing other users. As soon as the trusted party runs the key generation algorithm for the user with identity id, the trusted party publishes the corresponding public verification key in an existing public list $\mathcal{L}_{\mathsf{vk}}$. Consequently, anyone can fetch public verification key $\mathsf{vk}_{\mathsf{id}}$ from the list $\mathcal{L}_{\mathsf{vk}}$ by searching with identity id.

Remark 3. If the trusted party wants to revoke the membership of a user enrolled in our proposed qtIRS scheme, it removes the corresponding identity and verification key of that user from the public list $\mathcal{L}_{\mathsf{vk}}$.

Remark 4. Any user with identity id can have multiple signature tokens simultaneously with different serial numbers. In that case, each public key corresponds to an identity and a certain serial number. For simplicity, we have assumed in our scheme that the identity is enough to query for a public key. Suppose each user is endowed with multiple key pairs with different serial numbers. In that case, we replace id by $\mathsf{id}' = (\mathsf{id}||\mathsf{sn})$ where id is the identity of that user and sn is the serial number of the public key which is taken to form the ring.

Instantiations. Few recent works on lattice-inspired indistinguishability obfuscation (iO) candidates [24, 25] have shown that indistinguishability obfuscation for classical circuits has security against quantum polynomial-time attacks. We can use these iO candidates to generate obfuscated circuits for membership-checking programs. In 2019, Zhandry [13] proposed the concept of a subspace-hiding obfuscator. In 2019, Coladengelo et al. [23] used the concept of subspace-hiding obfuscation and also a compute-and-compare function to obfuscate membership checking circuits. These types of existing functions can be used as an obfuscator $\mathcal{O}$ in our scheme. In 2007, Hofheinz et al.

[20] showed that a uniformly random point function can easily be obfuscated. This result eventually implies that membership-checking circuits for finite sets can also be easily obfuscated which validates our assumption of the existence of obfuscation $\mathcal{O}$ of membership-checking circuits.

Lemma 2. *If point functions can easily be obfuscated, the membership-checking circuits for a finite set can be obfuscated easily.*

Proof. Suppose $S = \{a_1, \ldots, a_n\} \subset \mathcal{D}$ be a finite set in a possibly large domain $\mathcal{D}$. Let $f_a : \mathcal{D} \to \{0,1\}$ is the point function for the point $a \in S$ defined as $f_a(x) = \begin{cases} 1 \text{ if } x = a \\ 0 \text{ if } x \neq a \end{cases}$. Let the membership-checking circuit for the set S be $C[S]$ works as follows $C[S](x) = \begin{cases} 1 \text{ if } x \in S \\ 0 \text{ if } x \notin S \end{cases}$. Then $C[S] = f_{a_1} \vee f_{a_2} \vee \ldots \vee f_{a_n}$.

Hence, if f_{a_i} can be obfuscated easily $\forall i \in [n]$ then $C[S]$ can also be obfuscated easily.

3.2 Security analysis

Theorem 4. *Our quantum-tokenized identity-based ring signature scheme* qtIRS *presented in Section 3.1 attains strong existential unforgeability against adaptive chosen-message-and-identity attacks as per Def. 9 in Section 2 assuming the existence of obfuscation of membership checking circuits defined in Def. 7 in Section 2 and computationally sound non-interactive ZK proof system* NIZK *defined in Section 2.3.*

Proof. We assume that an adversary $\mathcal{A}$ exists that wins the existential unforgeability game against adaptive chosen-message-and-identity attack given in Fig. 1 against the challenger $\mathcal{C}$ with non-negligible probability.

The challenger and the adversary proceed as follows.

Setup phase: The challenger $\mathcal{C}$ takes as input a security parameter λ which is constant for the entire game and runs the algorithm $\mathsf{qtIRS.Setup}$, to generate the master secret key $\mathsf{msk} = S$ and public parameter set $\mathsf{params} = \mathsf{crs}$ where $S \subseteq \{0,1\}^\lambda$ is a uniformly random $\frac{\lambda}{2}$-dimensional subspace and crs is the common reference string output by $\mathsf{NIZK.Setup}$ i.e., $\mathsf{crs} \leftarrow \mathsf{NIZK.Setup}(\lambda)$. The crs contains λ in it. The challenger sends the params to the adversary $\mathcal{A}$ and keeps msk secret to itself. Further, $\mathcal{C}$ creates a list $\mathcal{L}_{\mathsf{vk}}$ of user verification keys, a list $\mathcal{L}_C$ of corrupted users identities and a list $\mathcal{L}_S$ of signing queries made by the adversary which are initially empty.

Key generation: The challenger $\mathcal{C}$ runs the key generation algorithm $\mathsf{qtIRS.Gen}$ on input the public parameter $\mathsf{params} = \mathsf{crs}$, the master secret key $\mathsf{msk} = S$ and the identity of a user $\mathsf{id}_i \in \{0,1\}^*$ $\forall i \in [p(\lambda)]$ for some polynomial $p(\cdot)$ and generates the secret and public key pair $(\mathsf{sk}_{\mathsf{id}_i}, \mathsf{vk}_{\mathsf{id}_i})$ corresponding to the user with identity id_i where $\mathsf{sk}_{\mathsf{id}_i} = |S\rangle^{\mathbf{x}_i, \mathbf{z}_i}$ is the quantum signature token and $\mathsf{vk}_{\mathsf{id}_i} = (O_{S+\mathbf{x}_i}, O_{S^\perp + \mathbf{z}_i})$ is the verification key consisting of two obfuscated membership checking circuits where $\mathbf{x}_i, \mathbf{z}_i \leftarrow h(\mathbf{r}_i, \mathsf{id}_i)$ for $\mathbf{r}_i \in \{0,1\}^{2\lambda}$ $\forall i \in [p(\lambda)]$ being randomly chosen 2λ-bit binary string. The challenger publishes all the verification keys corresponding to user identities i.e., $(\mathsf{id}_i, \mathsf{vk}_{\mathsf{id}_i})$ $\forall i \in [p(\lambda)]$ on the public verification key list $\mathcal{L}_{\mathsf{vk}}$. After generating key pairs for all users, the challenger $\mathcal{C}$ sends the verification key list $\mathcal{L}_{\mathsf{vk}}$ to the adversary $\mathcal{A}$ and keeps the secret keys corresponding to specific user identities $\{(\mathsf{id}_1, \mathsf{sk}_{\mathsf{id}_1}), (\mathsf{id}_2, \mathsf{sk}_{\mathsf{id}_2}), \ldots, (\mathsf{id}_{p(\lambda)}, \mathsf{sk}_{\mathsf{id}_{p(\lambda)}})\}$ and random strings $\{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_{p(\lambda)}\}$ secret to itself.

Corruption query: The adversary $\mathcal{A}$ makes polynomially many corruption queries (id_j) with $j \in [p(\lambda)]$ to $\mathcal{C}$. The challenger $\mathcal{C}$ returns the secret quantum signing token $\mathsf{sk}_{\mathsf{id}_j} = |S\rangle^{\mathbf{x}_j, \mathbf{z}_j}$ of the j-th user to $\mathcal{A}$ and adds id_j to the list $\mathcal{L}_C$ of corrupted users.

Claim 1: Since the signing key $\mathsf{sk}_{\mathsf{id}_j}$ is a quantum state, the adversary can neither copy it nor learn any information about the master secret key S or the secret strings $\mathbf{x}_j, \mathbf{z}_j$.

Proof of the claim: The adversary can either measure $\mathsf{sk}_{\mathsf{id}_j} = |S\rangle^{\mathbf{x}_j, \mathbf{z}_j}$ in the standard basis or perform the Hadamard transform and then measure the resulting state $H^{\otimes\lambda}(\mathsf{sk}_{\mathsf{id}_j}) = H^{\otimes\lambda}(|S\rangle^{\mathbf{x}_j, \mathbf{z}_j}) = |S^\perp\rangle^{\mathbf{z}_j, \mathbf{x}_j}$. So, the adversary can produce a vector from $S + \mathbf{x}_j$ or $S^\perp + \mathbf{z}_j$.

Let the number of corruption queries be $l(\lambda)$ for a polynomial $l(\cdot)$ such that $l(\lambda) < p(\lambda)$. Suppose that the adversary measures t number of original quantum signing tokens $\mathsf{sk}_{\mathsf{id}_j}$ for $\mathsf{id}_j \in \{\mathsf{id}_{i_1}, \mathsf{id}_{i_2}, \ldots, \mathsf{id}_{i_t}\} \subseteq \mathcal{L}_C$ and measures $l(\lambda) - t$ number of quantum signing tokens after Hadamard transformation $H^{\otimes}(\mathsf{sk}_{\mathsf{id}_j})$ for $\mathsf{id}_j \in \{\mathsf{id}_{i_{t+1}}, \mathsf{id}_{i_{t+2}}, \ldots, \mathsf{id}_{i_{l(\lambda)}}\} \subseteq \mathcal{L}_C$. Therefore, the adversary gets $\mathbf{s}_{i_1} + \mathbf{x}_{i_1}, \mathbf{s}_{i_2} + \mathbf{x}_{i_2}, \ldots, \mathbf{s}_{i_t} + \mathbf{x}_{i_t}$ and $\hat{\mathbf{s}}_{i_{t+1}} + \mathbf{z}_{i_{t+1}}, \hat{\mathbf{s}}_{i_{t+2}} + \mathbf{z}_{i_{t+2}}, \ldots, \hat{\mathbf{s}}_{i_{l(\lambda)}} + \mathbf{z}_{i_{l(\lambda)}}$

where $\mathbf{s}_j \in S\ \forall j \in \{i_1, i_2, \ldots, i_t\}$, $\hat{\mathbf{s}}_j \in S^{\perp}\ \forall j \in \{i_{t+1}, i_{t+2}, \ldots, i_{l(\lambda)}\}$, $\mathbf{x}_{i_j} \in \{0,1\}^{\lambda}\ \forall j \in \{1, 2, \ldots t\}$ and $\mathbf{z}_{i_j} \in \{0,1\}^{\lambda}$ $\forall j \in \{t+1, t+2, \ldots l(\lambda)\}$. Note that, given $\mathbf{a}+\mathbf{b}$ for $\mathbf{a}, \mathbf{b} \xleftarrow{R} \{0,1\}^{\lambda}$, any unbounded adversary can determine $\mathbf{a}$, $\mathbf{b}$ only with negligible probability. Consequently, the adversary $\mathcal{A}$ cannot extract any information about $\mathbf{s}_j$, $\hat{\mathbf{s}}_j$, $\mathbf{x}_j$ and $\mathbf{z}_j\ \forall j \in [l(\lambda)]$ by measuring either the signing tokens $\mathsf{sk}_{\mathsf{id}_j}$ or their Hadamard transforms.

Signing query: The adversary $\mathcal{A}$ makes polynomially many signing queries $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to $\mathcal{C}$ where $m_j \in \{0,1\}$ is the message, $\mathsf{ID}_j \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ is a set of identities chosen by $\mathcal{A}$ and $\mathsf{id}_k \in \mathsf{ID}_j$ is the identity of a user chosen by $\mathcal{A}$. The challenger $\mathcal{C}$ responds with $\boldsymbol{\sigma}_j = (\boldsymbol{\beta}_j, \mathsf{rvk}_j, \pi_j)$ such that $\boldsymbol{\sigma}_j = (\boldsymbol{\beta}_j, \mathsf{rvk}_j, \pi_j) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_j, \mathsf{ID}_j)$ using $\mathsf{sk}_{\mathsf{id}_k} = |S\rangle^{\mathbf{x}_k, \mathbf{z}_k}$ with $\mathbf{x}_k, \mathbf{z}_k \in \{0,1\}^{\lambda}$ where

$$\boldsymbol{\beta}_j = \begin{cases} \mathbf{s}_j + \mathbf{x}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 0 \\ \hat{\mathbf{s}}_j + \mathbf{z}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 1 \end{cases}$$

$$\mathsf{rvk}_j = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 0 \\ O_{S^{\perp}+\mathbf{z}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 1 \end{cases}$$

$$\pi_j \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_j), \mathsf{ID}_j || \mathsf{rvk}_j)$$

for $\hat{\mathbf{x}}_j \xleftarrow{R} \{0,1\}^{\lambda}$, $\mathbf{s}_j \xleftarrow{R} S$ and $\hat{\mathbf{s}}_j \xleftarrow{R} S^{\perp}$. The challenger adds $(m_j, \boldsymbol{\sigma}_j, \mathsf{ID}_j)$ to the list $\mathcal{L}_S$ of signing queries.

Note that the adversary $\mathcal{A}$ can make the same signing query $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ polynomially many times to $\mathcal{C}$. Suppose the adversary makes the query $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to the challenger $q(\lambda)$ times. The the responses by the challenger would be $\boldsymbol{\sigma}_j^{(i)} = (\boldsymbol{\beta}_j^{(i)}, \mathsf{rvk}_j^{(i)}, \pi_j^{(i)})$ where

$$\boldsymbol{\beta}_j^{(i)} = \begin{cases} \mathbf{s}_j^{(i)} + \mathbf{x}_k + \hat{\mathbf{x}}_j^{(i)} \text{ if } m_j = 0 \\ \hat{\mathbf{s}}_j^{(i)} + \mathbf{z}_k + \hat{\mathbf{x}}_j^{(i)} \text{ if } m_j = 1 \end{cases}$$

$$\mathsf{rvk}_j^{(i)} = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_j^{(i)}} & \text{if } m_j = 0 \\ O_{S^{\perp}+\mathbf{z}_k+\hat{\mathbf{x}}_j^{(i)}} & \text{if } m_j = 1 \end{cases}$$

$$\pi_j^{(i)} \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_j^{(i)}), \mathsf{ID}_j || \hat{\mathsf{vk}}_j^{(i)})$$

for $\hat{\mathbf{x}}_j^{(i)} \xleftarrow{R} \{0,1\}^{\lambda}$, $\mathbf{s}_j^{(i)} \xleftarrow{R} S$ and $\hat{\mathbf{s}}_j^{(i)} \xleftarrow{R} S^{\perp}\ \forall i \in [q(\lambda)]$. Hence, each time the adversary gets different responses and the XORs with the random strings $\hat{\mathbf{x}}_j^{(i)}$ makes it difficult to manipulate the results to obtain elements of S or the strings $\mathbf{x}_k, \mathbf{z}_k \in \{0,1\}^{\lambda}$.

View: After $l(\lambda)$ corruption queries and $r(\lambda)$ signing queires such that $1 \leq l(\lambda), r(\lambda) \leq p(\lambda)$.

Table 4: $\mathsf{View}_{\mathcal{A}}^{\mathsf{sEU\text{-}ACMI}}$

$$\mathsf{crs}, \quad \mathcal{L}_S = \{(m_{j_i}, \boldsymbol{\sigma}_{ji}, \mathsf{ID}_{j_i}) \mid\ i \in r(\lambda)\}, \quad \mathcal{L}_{\mathsf{vk}} = \{(\mathsf{id}_j, \mathsf{vk}_{\mathsf{id}_j}) \mid\ j \in [p(\lambda)]\},$$
$$\mathbf{s}_{i_1} + \mathbf{x}_{i_1}, \ldots, \mathbf{s}_{i_t} + \mathbf{x}_{i_t}, \hat{\mathbf{s}}_{i_{t+1}} + \mathbf{z}_{i_{t+1}}, \ldots, \hat{\mathbf{s}}_{i_{l(\lambda)}} + \mathbf{z}_{i_{l(\lambda)}} \quad \forall \mathsf{id}_{i_k} \in \mathcal{L}_C$$
$$\text{where } \{\mathbf{s}_{i_1}, \ldots, \mathbf{s}_{i_t}\} \subseteq S\ \ \&\ \ \{\mathbf{s}_{i_{t+1}}, \ldots, \mathbf{s}_{i_{l(\lambda)}}\} \subseteq S^{\perp}$$

$(\mathsf{params} = \mathsf{crs}, \mathsf{msk} = S) \leftarrow \mathsf{qtIRS.Setup}(1^{\lambda})$;
$\left(\mathsf{sk}_{\mathsf{id}_j} = |S\rangle^{\mathbf{x}_j, \mathbf{z}_j},\ \ \mathsf{vk}_{\mathsf{id}_j} = (O_{S+\mathbf{x}_j}, O_{S^{\perp}+\mathbf{z}_j})\right) \leftarrow \mathsf{qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_j), \quad \mathbf{x}_j, \mathbf{z}_j \leftarrow h(\mathbf{r}_j, \mathsf{id}_j),\ \mathbf{r}_j \xleftarrow{R} \{0,1\}^{\lambda},\ \forall j \in [p(\lambda)]$;
$\boldsymbol{\sigma}_j = (\boldsymbol{\beta}_j, \mathsf{rvk}_j, \pi_j) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_j, \mathsf{ID}_j),\ m_j \in \{0,1\}, \mathsf{id}_k \in \mathsf{ID}_j \subseteq \{\mathsf{id}_i\}_{i \in p(\lambda)}$;
$\boldsymbol{\beta}_j = \begin{cases} \mathbf{s}_j + \mathbf{x}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 0 \\ \hat{\mathbf{s}}_j + \mathbf{z}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 1 \end{cases}$, $\mathsf{rvk}_j = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 0 \\ O_{S^{\perp}+\mathbf{z}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 1 \end{cases}$, $\pi_j \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_j), \mathsf{ID}_j || \mathsf{rvk}_j)$

For $\mathcal{L}_C$, let the adversary $\mathcal{A}$ measures the first t signing tokens $\mathsf{sk}_{\mathsf{id}_j}$ for $j \in \{i_1, \ldots, i_t\} \subseteq [l(\lambda)]$ directly yielding $\mathcal{A}$ to have the knowledge $\{\mathbf{s}_{i_1}+\mathbf{x}_{i_1}, \ldots, \mathbf{s}_{i_t}+\mathbf{x}_{i_t}\}$ where $\{\mathbf{s}_{i_1}, \ldots, \mathbf{s}_{i_t}\} \subseteq S$ and remaining $(l(\lambda)-t)$ many tokens $\mathsf{sk}_{\mathsf{id}_j}$ for $j \in \{i_{(t+1)}, \ldots, i_{l(\lambda)}\} \subseteq [l(\lambda)]$, after Hadamard transform resulting $\mathcal{A}$ to have the knowledge of $\{\hat{\mathbf{s}}_{i_{t+1}} + \mathbf{z}_{i_{t+1}}, \ldots, \hat{\mathbf{s}}_{i_{l(\lambda)}} + \mathbf{z}_{i_{l(\lambda)}}\}$ where $\{\mathbf{s}_{i_{t+1}}, \ldots, \mathbf{s}_{i_{l(\lambda)}}\} \subseteq S^{\perp}$. Finally, $\mathcal{A}$ has the information view as presented in $\mathsf{View}_{\mathcal{A}}^{\mathsf{sEU\text{-}ACMI}}$ in Fig. 4.

Forge: The adversary $\mathcal{A}$ outputs a three tuple $(m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*)$ where $\boldsymbol{\sigma}^* = (\boldsymbol{\beta}^*, \mathsf{rvk}^*, \pi^*)$. If $(\mathsf{id}_j) \notin \mathcal{L}_C\ \forall \mathsf{id}_j \in \mathsf{ID}^*$, $(m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*) \notin \mathcal{L}_S$ and $1 \leftarrow \mathsf{qtIRS.Verify}(\mathsf{params}, m^*, \sigma^*, \mathsf{ID}^*)$, then $\mathcal{C}$ returns 1 as output of $\mathsf{Exp}_{\mathcal{A},\mathsf{qtIRS}}^{\mathsf{EUnforge}}(\lambda)$; otherwise, it returns 0.

Towards contradiction: Suppose the adversary wins the unforgeability game $\mathsf{Exp}^{\mathsf{EUnforge}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda)$ with non-negligible probability i.e., the advantage of $\mathcal{A}$ in the game is

$$\mathsf{Adv}^{\mathsf{EUnforge}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda) = \mathsf{Pr}[\mathsf{Exp}^{\mathsf{EUnforge}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda) = 1] = \mu(\lambda)$$

where $\mu(\lambda)$ is a non-negligible function with $0 \leq \mu(\cdot) \leq 1$. then we show a contradiction to our assumption of the existence of obfuscation of membership checking circuits or computationally sound non-inteactive ZK proof system NIZK.

First, the adversary $\mathcal{A}$ fixes $m^* \in \{0,1\}$ and $\mathsf{ID}^* \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ as per its choice. Let $(m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*)$ be a valid forgery, $\mathsf{id} \notin \mathcal{L}_C \ \forall \mathsf{id} \in \mathsf{ID}^*$ and $(m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*) \notin \mathcal{L}_S$.

Now, consider the following three possible cases how the adversary $\mathcal{A}$ can proceed in order to produce the valid forged signature $\boldsymbol{\sigma}^* = (\boldsymbol{\beta}^*, \mathsf{rvk}^*, \pi^*)$.

Case 1: The adversary prepares valid rvk^* and π^*. Then it can produce $\boldsymbol{\beta}^*$ only with negligible probability by exploiting the security of obfuscation of circuits.

The adversary chooses $\mathsf{id} \in \mathsf{ID}^*$ and searches the corresponding verification key $\mathsf{vk}_{\mathsf{id}} = (O_{S+\mathbf{x}}, O_{S^\perp+\mathbf{z}})$ in the the list $\mathcal{L}_{\mathsf{vk}}$. Then it chooses a random λ-bit string $\hat{\mathbf{x}}^*$ and calculates

$$\mathsf{rvk}^* = \begin{cases} O_{S+\mathbf{x}+\hat{\mathbf{x}}^*} & \text{if } m^* = 0 \\ O_{S^\perp+\mathbf{z}+\hat{\mathbf{x}}^*} & \text{if } m^* = 1 \end{cases}$$

where $O_{S+\mathbf{x}+\hat{\mathbf{x}}^*} = M[O_{S+\mathbf{x}}, \hat{\mathbf{x}}^*]$ and $O_{S^\perp+\mathbf{z}+\hat{\mathbf{x}}^*} = M[O_{S^\perp+\mathbf{z}}, \hat{\mathbf{x}}^*]$.

It sets $\mathbf{y} = \mathsf{ID}^*||\mathsf{rvk}^*$ and computes $\pi^* \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}, \hat{\mathbf{x}}^*), \mathsf{ID}^*||\mathsf{rvk}^*)$.

Then in order to pass the verification, the adversary $\mathcal{A}$ needs to produce $\boldsymbol{\beta}^* \in S + \mathbf{x} + \hat{\mathbf{x}}^*$ if $m^* = 0$ and $\boldsymbol{\beta}^* \in S^\perp + \mathbf{z} + \hat{\mathbf{x}}^*$ if $m^* = 1$. Hence, the adversary guesses a λ-bit string from $S + \mathbf{x} + \hat{\mathbf{x}}^*$ (or $S^\perp + \mathbf{z} + \hat{\mathbf{x}}^*$) by seeing the obfuscated circuit $O_{S^\perp+\mathbf{z}+\hat{\mathbf{x}}^*}$ (or $O_{S+\mathbf{x}+\hat{\mathbf{x}}^*}$). Similarly, given $O_{S+\mathbf{x}}$ (or $O_{S^\perp+\mathbf{z}}$), the adversary $\mathcal{A}$ can return a string from $S + \mathbf{x}$ or $S^\perp + \mathbf{z}$ since $\hat{\mathbf{x}}^*$ is known to $\mathcal{A}$. Since, $\mathsf{id} \in \mathsf{ID}^*$, $\mathsf{id} \notin \mathcal{L}_C$. So, the adversary does not have the secret quantum token $\mathsf{sk}_{\mathsf{id}} = |S\rangle^{\mathbf{x},\mathbf{z}}$.

If the adversary $\mathcal{A}$ wins with non-negligible probability, then it will contradict the security of obfuscation O_T of a membership checking circuit for some subset $T \subseteq \{0,1\}^\lambda$ with non-negligible probability. Hence, the adversary $\mathcal{A}$ can forge $\boldsymbol{\beta}^*$ only with negligible probability in this case.

Case 2: The adversary prepares valid $\boldsymbol{\beta}^*$ and rvk^*. Then we show that it can produce π^* only with negligible probability by exploiting the security of the computationally sound non-interactive ZK proof system NIZK as shown below.

The adversary chooses any previous signature $\boldsymbol{\sigma}_i = (\boldsymbol{\beta}_i, \mathsf{rvk}_i, \pi_i) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_i, \mathsf{ID}_i)$ where

$$\boldsymbol{\beta}_i \in \begin{cases} S + \mathbf{x}_k + \hat{\mathbf{x}}_k & \text{if } m_i = 0 \\ S^\perp + \mathbf{z}_k + \hat{\mathbf{x}}_k & \text{if } m_i = 1 \end{cases}$$

and chooses a random string $\hat{\mathbf{x}}$. It sets $\boldsymbol{\beta}^* = \boldsymbol{\beta}_i + \hat{\mathbf{x}}$. Therefore,

$$\boldsymbol{\beta}^* \in \begin{cases} S + \mathbf{x}_k + \hat{\mathbf{x}}_k + \hat{\mathbf{x}} & \text{if } m_i = 0 \\ S^\perp + \mathbf{z}_k + \hat{\mathbf{x}}_k + \hat{\mathbf{x}} & \text{if } m_i = 1 \end{cases}$$

The adversary calculates

$$O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_k+\hat{\mathbf{x}}} = M[O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_k}, \hat{\mathbf{x}}] \text{ and } O_{S^\perp+\mathbf{z}_k+\hat{\mathbf{x}}_k+\hat{\mathbf{x}}} = M[O_{S^\perp+\mathbf{z}_k+\hat{\mathbf{x}}_k}, \hat{\mathbf{x}}]$$

and sets $\mathsf{rvk}^* = (O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_k+\hat{\mathbf{x}}}, O_{S^\perp+\mathbf{z}_k+\hat{\mathbf{x}}_k+\hat{\mathbf{x}}})$. Let us denote $\hat{\mathbf{x}}^* = \hat{\mathbf{x}}_k + \hat{\mathbf{x}}$.
Then only with knowledge of $\hat{\mathbf{x}}$, determining $\hat{\mathbf{x}}^*$ is nothing but a random guess since $\hat{\mathbf{x}}_k$ is uniformly random λ-bit string chosen by the challenger during responding to the i-th signing query. So, the adversary $\mathcal{A}$ gets no extra benefit from the $\mathsf{View}_{\mathcal{A}}$ to determine $\hat{\mathbf{x}}^*$. Therefore,

$\mathcal{A}$ forge the valid tuple $(m^*, \boldsymbol{\sigma}^* = (\boldsymbol{\beta}^*, \mathsf{rvk}^*, \pi^*), \mathsf{ID}^*)$ with non-negligible probability.

$\implies \mathcal{A}$ can calculate $\pi^* \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}^*), \mathsf{ID}^*||\mathsf{rvk}^*)$ with non-negligible probability without knowledge of $\hat{\mathbf{x}}^*$.

$\implies$ NIZK is not computationally sound which is a contradiction.

Case 3: The adversary prepares valid π^* and $\boldsymbol{\beta}^*$. Note that if the adversary prepares valid π^*, then it implies that $\mathcal{A}$ knows the witness $(\mathsf{id}, \hat{\mathbf{x}}^*)$. Then the adversary can prepare rvk^* using $\mathsf{vk}_{\mathsf{id}}$ present in the public list $\mathcal{L}_{\mathsf{vk}}$ and $\hat{\mathbf{x}}^*$.

Then the adversary produces $\boldsymbol{\beta}^*$, rvk^* and π^* with non-negligible probability which contradicts both of the above two cases.

Now, consider the adversary made a query of the form $(\mathsf{id}_k, m^*, \mathsf{ID}^*)$ with $\mathsf{id}_k \in \mathsf{ID}^*$ previously to the signing oracle which gave a response of the form $\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m^*, \mathsf{ID}^*)$ where

$$\boldsymbol{\beta} \in \begin{cases} S + \mathbf{x}_k + \hat{\mathbf{x}}_k & \text{if } m^* = 0 \\ S^{\perp} + \mathbf{z}_k + \hat{\mathbf{x}}_k & \text{if } m^* = 1 \end{cases}$$

$$\mathsf{rvk} = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_k} & \text{if } m^* = 0 \\ O_{S^{\perp}+\mathbf{z}_k+\hat{\mathbf{x}}_k} & \text{if } m^* = 1 \end{cases}$$

$$\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_k), \mathsf{ID}^* || \hat{\mathsf{vk}})$$

Let us assume

$$\boldsymbol{\beta} = \begin{cases} \mathbf{s} + \mathbf{x}_k + \hat{\mathbf{x}}_k & \text{if } m^* = 0 \\ \hat{\mathbf{s}} + \mathbf{z}_k + \hat{\mathbf{x}}_k & \text{if } m^* = 1 \end{cases}$$

Then in order to produce a new signature $\boldsymbol{\sigma}^* = (\boldsymbol{\beta}^*, \mathsf{rvk}^*, \pi^*) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m^*, \mathsf{ID}^*)$ such that $\boldsymbol{\sigma}^* \neq \boldsymbol{\sigma}$ with non-negligible probability, we have the following two cases.

Case A: $\mathsf{rvk}^* = \mathsf{rvk}$ and $\pi^* = \pi$. It implies

$$\boldsymbol{\beta}^* \in \begin{cases} S + \mathbf{x}_k + \hat{\mathbf{x}}_k & \text{if } m^* = 0 \\ S^{\perp} + \mathbf{z}_k + \hat{\mathbf{x}}_k & \text{if } m^* = 1 \end{cases}$$

But, since $\boldsymbol{\sigma}^* \neq \boldsymbol{\sigma}$, it will imply $\boldsymbol{\beta}^* \neq \boldsymbol{\beta}$. Hence,

$$\boldsymbol{\beta}^* = \begin{cases} \mathbf{s}^* + \mathbf{x}_k + \hat{\mathbf{x}}_k & \text{if } m^* = 0 \\ \hat{\mathbf{s}}^* + \mathbf{z}_k + \hat{\mathbf{x}}_k & \text{if } m^* = 1 \end{cases}$$

such that $\mathbf{s}^* \neq \mathbf{s}$ if $m^* = 0$ and $\hat{\mathbf{s}}^* \neq \hat{\mathbf{s}}$ if $m^* = 1$.

So, the adversary produces an element from the subspace S (or the orthogonal subspace $S^{\perp}$) with non-negligible probability which is a contradiction to the fact in **Claim 1** that one can produce an element from S (or $S^{\perp}$) only with probability $2^{-\lambda/2}$ as S is the master secret key.

Case B: $\mathsf{rvk}^* \neq \mathsf{rvk}$. Suppose

$$\mathsf{rvk}^* = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_k+\hat{\mathbf{x}}} & \text{if } m^* = 0 \\ O_{S^{\perp}+\mathbf{z}_k+\hat{\mathbf{x}}_k+\hat{\mathbf{x}}} & \text{if } m^* = 1 \end{cases}$$

and $\hat{\mathbf{x}}^* = \hat{\mathbf{x}}_k + \hat{\mathbf{x}}$. Then $\pi^* \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}^*), \mathsf{ID}^* || \mathsf{rvk}^*)$. So, the adversary produces a non-interactive ZK-proof without knowing the witness with non-negligible probability which contradicts the computational soundness of NIZK. The adversary $\mathcal{A}$ does not know the witness component $\hat{\mathbf{x}}^* = \hat{\mathbf{x}}_k + \hat{\mathbf{x}}$ since $\hat{\mathbf{x}}_k$ was randomly chosen by the challenger $\mathcal{C}$ and kept secret from $\mathcal{A}$.

Therefore, the adversary $\mathcal{A}$ can forge a valid tuple $(m^*, \boldsymbol{\sigma}^* = (\boldsymbol{\beta}^*, \mathsf{rvk}^*, \pi^*), \mathsf{ID}^*)$ such that $(\mathsf{id}_j) \notin \mathcal{L}_C \ \forall \mathsf{id}_j \in \mathsf{ID}^*$, $(m^*, \boldsymbol{\sigma}^*, \mathsf{ID}^*) \notin \mathcal{L}_S$ and $1 \leftarrow \mathsf{qtIRS.Verify}(\mathsf{params}, m^*, \sigma^*, \mathsf{ID}^*)$ only with negligible probability.

Theorem 5. *If there is obfuscation of membership checking circuits as defined in Definition 7 in Section 2 and computationally sound noninteractive ZK proof system* NIZK *as defined in Section 2.3, then our quantum tokenized identity-based ring signature scheme* qtIRS *presented in Section 3.1 achieves strong anonymity according to Def. 10 in Section 2.*

Proof. The challenger and the adversary proceed as follows.

Setup phase: The challenger $\mathcal{C}$ runs the algorithm

$$(\mathsf{params} = \mathsf{crs}, \mathsf{msk} = S) \leftarrow \mathsf{qtIRS.Setup}(\lambda)$$

where λ is the security parameter, $\mathsf{params} = \mathsf{crs} \leftarrow \mathsf{NIZK.Setup}(\lambda)$ is the public parameter set containing λ and the master secret key $S \subseteq \{0,1\}^\lambda$ is a uniformly random $\frac{\lambda}{2}$-dimensional subspace. The challenger sends the params to the adversary $\mathcal{A}$ and keeps the msk secret to itself. Further, $\mathcal{C}$ creates a list $\mathcal{L}_{\mathsf{vk}}$ of user verification keys, a list $\mathcal{L}_C$ of corrupted user identities and a list $\mathcal{L}_S$ of signing queries made by the adversary which are initially empty.

Key generation: The challenger $\mathcal{C}$ runs the key generation algorithm

$$(\mathsf{sk}_{\mathsf{id}_i} = |S\rangle^{\mathbf{x}_i,\mathbf{z}_i}, \mathsf{vk}_{\mathsf{id}_i} = (O_{S+\mathbf{x}_i}, O_{S^\perp+\mathbf{z}_i})) \leftarrow \mathsf{qtIRS.Gen}(\mathsf{params} = \mathsf{crs}, \mathsf{msk} = S, \mathsf{id}_i)$$

that takes as input the parameter set $\mathsf{params} = \mathsf{crs}$, the master secret key $\mathsf{msk} = S$ and the identity of a user $\mathsf{id}_i \in \{0,1\}^* \ \forall i \in [p(\lambda)]$ when $p(\lambda)$ is the number of users in the scheme and generates the secret signing token $\mathsf{sk}_{\mathsf{id}_i} = |S\rangle^{\mathbf{x}_i,\mathbf{z}_i}$ and the corresponding public key $\mathsf{vk}_{\mathsf{id}_i} = (O_{S+\mathbf{x}_i}, O_{S^\perp+\mathbf{z}_i})$ consisting of two obfuscated membership checking circuits for the user with identity id_i where $\mathbf{x}_i, \mathbf{z}_i \leftarrow h(\mathbf{r}_i, \mathsf{id}_i)$ are λ-bit binary strings for uniformly random $\mathbf{r}_i \in \{0,1\}^{2\lambda} \ \forall i \in [p(\lambda)]$. The challenger adds the user identity and the corresponding verification key pair $(\mathsf{id}_i, \mathsf{vk}_{\mathsf{id}_i})$ to the list $\mathcal{L}_{\mathsf{vk}} \ \forall i \in [p(\lambda)]$. The challenger $\mathcal{C}$ sends the list $\mathcal{L}_{\mathsf{vk}}$ to the adversary $\mathcal{A}$ and keeps the secret keys corresponding to specific user identities $\{(\mathsf{id}_1, \mathsf{sk}_{\mathsf{id}_1}), (\mathsf{id}_2, \mathsf{sk}_{\mathsf{id}_2}), \ldots, (\mathsf{id}_{p(\lambda)}, \mathsf{sk}_{\mathsf{id}_{p(\lambda)}})\}$ and random strings $\{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_{p(\lambda)})\}$ secret to itself.

Corruption query: The adversary $\mathcal{A}$ makes $l(\lambda)$ corruption queries (id_j) with $j \in [p(\lambda)]$ to $\mathcal{C}$ where $l(\cdot)$ is a polynomial. The challenger $\mathcal{C}$ returns the secret key $\mathsf{sk}_{\mathsf{id}_j} = |S\rangle^{\mathbf{x}_j,\mathbf{z}_j}$ of the j-th user to $\mathcal{A}$ and adds id_j to the list $\mathcal{L}_C$.

The **Claim 1** of Theorem 4 ensures that given the quantum signing token $\mathsf{sk}_{\mathsf{id}_j}$, the adversary can neither copy it nor learn any information about it.

Signing query: The adversary $\mathcal{A}$ makes $r(\lambda)$ many signing queries $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to $\mathcal{C}$ where $p(\cdot)$ is a polynomial, $m_j \in \{0,1\}$ is the message, $\mathsf{ID}_j \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ is a set of identities chosen by $\mathcal{A}$ and $\mathsf{id}_k \in \mathsf{ID}_j$ is the identity of a user chosen by $\mathcal{A}$. The challenger $\mathcal{C}$ responds with $\boldsymbol{\sigma}_j = (\boldsymbol{\beta}_j, \mathsf{rvk}_j, \pi_j)$ such that $\boldsymbol{\sigma}_j = (\boldsymbol{\beta}_j, \mathsf{rvk}_j, \pi_j) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_j, \mathsf{ID}_j)$ where

$$\boldsymbol{\beta}_j = \begin{cases} \mathbf{s}_j + \mathbf{x}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 0 \\ \hat{\mathbf{s}}_j + \mathbf{z}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 1 \end{cases}$$

$$\mathsf{rvk}_j = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 0 \\ O_{S^\perp+\mathbf{z}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 1 \end{cases}$$

$$\pi_j \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_j), \mathsf{ID}_j || \mathsf{rvk}_j)$$

for $\hat{\mathbf{x}}_j \xleftarrow{R} \{0,1\}^\lambda$, $\mathbf{s}_j \xleftarrow{R} S$ and $\hat{\mathbf{s}}_j \xleftarrow{R} S^\perp$. Then the challenger adds $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to the list $\mathcal{L}_S$.

Note that the adversary $\mathcal{A}$ is allowed to make the same signing query $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ polynomially many times to $\mathcal{C}$. Suppose the adversary makes the query $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to the challenger $q(\lambda)$ times. The responses by the challenger would be $\boldsymbol{\sigma}_{ji} = (\boldsymbol{\beta}_{ji}, \mathsf{rvk}_{ji}, \pi_{ji})$ where

$$\boldsymbol{\beta}_{ji} = \begin{cases} \mathbf{s}_{ji} + \mathbf{x}_k + \hat{\mathbf{x}}_{ji} \text{ if } m_j = 0 \\ \hat{\mathbf{s}}_{ji} + \mathbf{z}_k + \hat{\mathbf{x}}_{ji} \text{ if } m_j = 1 \end{cases}$$

$$\mathsf{rvk}_{ji} = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_{ji}} & \text{if } m_j = 0 \\ O_{S^\perp+\mathbf{z}_k+\hat{\mathbf{x}}_{ji}} & \text{if } m_j = 1 \end{cases}$$

$$\pi_{ji} \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_{ji}), \mathsf{ID}_j || \mathsf{rvk}_{ji})$$

for $\hat{\mathbf{x}}_{ji} \xleftarrow{R} \{0,1\}^\lambda$, $\mathbf{s}_{ji} \xleftarrow{R} S$ and $\hat{\mathbf{s}}_{ji} \xleftarrow{R} S^\perp \ \forall i \in [q(\lambda)]$. Hence, each time the adversary gets a different response and the XORs with the random strings $\hat{\mathbf{x}}_{ji}$ make it impossible to manipulate the results to obtain elements of S or the strings $\mathbf{x}_k, \mathbf{z}_k$.

View: After $l(\lambda)$ corruption queries and $r(\lambda)$ signing queires such that $1 \leq l(\lambda), r(\lambda) \leq p(\lambda)$, the adversary has the access to the information given in Fig. 5.

Anonymity game:

Table 5: $\mathsf{View}^{\mathsf{sAnon}}_{\mathcal{A}}$

$$\boxed{\begin{array}{c}\mathsf{crs}, \quad \mathcal{L}_S = \{(m_{j_i}, \boldsymbol{\sigma}_{ji}, \mathsf{ID}_{j_i}) \big|\ i \in r(\lambda)\}, \quad \mathcal{L}_{\mathsf{vk}} = \{(\mathsf{id}_j, \mathsf{vk}_{\mathsf{id}_j}) \big|\ j \in [p(\lambda)]\}, \\ \mathbf{s}_{i_1} + \mathbf{x}_{i_1}, \ldots, \mathbf{s}_{i_t} + \mathbf{x}_{i_t}, \hat{\mathbf{s}}_{i_{t+1}} + \mathbf{z}_{i_{t+1}}, \ldots, \hat{\mathbf{s}}_{i_{l(\lambda)}} + \mathbf{z}_{i_{l(\lambda)}} \quad \forall \mathsf{id}_{i_k} \in \mathcal{L}_C \\ \text{where } \{\mathbf{s}_{i_1}, \ldots, \mathbf{s}_{i_t}\} \subseteq S \ \ \& \ \ \{\mathbf{s}_{i_{t+1}}, \ldots, \mathbf{s}_{i_{l(\lambda)}}\} \subseteq S^{\perp}\end{array}}$$

$(\mathsf{params} = \mathsf{crs}, \mathsf{msk} = S) \leftarrow \mathsf{qtIRS.Setup}(1^\lambda)$;

$\left(\mathsf{sk}_{\mathsf{id}_j} = |S\rangle^{\mathbf{x}_j, \mathbf{z}_j}, \quad \mathsf{vk}_{\mathsf{id}_j} = (O_{S+\mathbf{x}_j}, O_{S^\perp + \mathbf{z}_j})\right) \leftarrow \mathsf{qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_j), \quad \mathbf{x}_j, \mathbf{z}_j \leftarrow h(\mathbf{r}_j, \mathsf{id}_j),\ \mathbf{r}_j \xleftarrow{R} \{0,1\}^\lambda,\ \forall j \in [p(\lambda)]$;

$\boldsymbol{\sigma}_j = (\boldsymbol{\beta}_j, \mathsf{rvk}_j, \pi_j) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_j, \mathsf{ID}_j),\ m_j \in \{0,1\}, \mathsf{id}_k \in \mathsf{ID}_j \subseteq \{\mathsf{id}_i\}_{i \in p(\lambda)}$;

$\boldsymbol{\beta}_j = \begin{cases} \mathbf{s}_j + \mathbf{x}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 0 \\ \hat{\mathbf{s}}_j + \mathbf{z}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 1 \end{cases}$, $\mathsf{rvk}_j = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 0 \\ O_{S^\perp+\mathbf{z}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 1 \end{cases}$, $\pi_j \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_j), \mathsf{ID}_j || \mathsf{rvk}_j)$

1. The adversary $\mathcal{A}$ chooses a ring of users $\mathsf{ID}^* \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ and a message $m^* \in \{0,1\}$ and sends the query $(\mathsf{id}_{i_0}, \mathsf{id}_{i_1}, m^*, \mathsf{ID}^*)$ to the challenger $\mathcal{C}$ such that $\mathsf{id}_{i_0}, \mathsf{id}_{i_1} \in \mathsf{ID}^*$.
2. Then the challenger chooses a uniformly random bit $b \xleftarrow{R} \{0,1\}$ and responds to the adversary $\mathcal{A}$ with $\boldsymbol{\sigma}^* = (\boldsymbol{\beta}^*, \mathsf{rvk}^*, \pi^*)$ where $(\boldsymbol{\beta}^*, \mathsf{rvk}^*, \pi^*) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_{i_b}}, m^*, \mathsf{ID}^*)$.
3. Then the adversary returns b' as a guess for b.
4. If $\mathsf{id}_{i_0}, \mathsf{id}_{i_1} \notin \mathcal{L}_C$, $(\mathsf{id}_{i_0}, m^*, \mathsf{ID}^*), (\mathsf{id}_{i_1}, m^*, \mathsf{ID}^*) \notin \mathcal{L}_S$ and $b' = b$ then $\mathcal{C}$ returns 1 as output of $\mathsf{Exp}^{\mathsf{Anon}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda)$; otherwise, it returns 0.

We will show that the adversary wins the anonymity game $\mathsf{Exp}^{\mathsf{Anon}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda)$ only with probability negligibly close to $\frac{1}{2}$ i.e., the advantage of $\mathcal{A}$ in the game is

$$\mathsf{Adv}^{\mathsf{Anon}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda) = \left|\Pr[\mathsf{Exp}^{\mathsf{Anon}}_{\mathcal{A},\mathsf{qtIRS}}(\lambda) = 1] - \frac{1}{2}\right| \leq \mu(\lambda)$$

where $\mu(\lambda)$ is a negligible function with $0 \leq \mu(\cdot) \leq 1$. In other words, to the adversary both the users with identity id_{i_0} and id_{i_1} in the ring ID^* are equiprobable signers given $\boldsymbol{\sigma}^*$.

The quantum signing tokens of the user with identities id_{i_0} and id_{i_1} are $\mathsf{sk}_{\mathsf{id}_{i_0}} = |S\rangle^{\mathbf{x}_{i_0}, \mathbf{z}_{i_0}}$ and $\mathsf{sk}_{\mathsf{id}_{i_1}} = |S\rangle^{\mathbf{x}_{i_1}, \mathbf{z}_{i_1}}$. Since $\mathsf{id}_{i_0}, \mathsf{id}_{i_1} \notin \mathcal{L}_C$.

So, $\mathsf{sk}_{\mathsf{id}_0}$, $\mathsf{sk}_{\mathsf{id}_1}$ and their respective Hadamard transforms cannot be measured by $\mathcal{A}$. Thus, it holds that

$$\mathbf{s}_{i_0} + \mathbf{x}_{i_0}, \hat{\mathbf{s}}_{i_0} + \mathbf{z}_{i_0}, \mathbf{s}_{i_1} + \mathbf{x}_{i_1}, \hat{\mathbf{s}}_{i_1} + \mathbf{z}_{i_1} \notin \mathsf{View}_{\mathcal{A}}$$

for any $\mathbf{s}_{i_0}, \mathbf{s}_{i_1} \xleftarrow{R} S$ and $\hat{\mathbf{s}}_{i_0}, \hat{\mathbf{s}}_{i_1} \xleftarrow{R} S^\perp$. Moreover,

$$\begin{aligned} & (\mathsf{id}_{i_0}, m^*, \mathsf{ID}^*), (\mathsf{id}_{i_1}, m^*, \mathsf{ID}^*) \notin \mathcal{L}_S \\ \implies & \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_{i_0}}, m^*, \mathsf{ID}^*), \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_{i_1}}, m^*, \mathsf{ID}^*) \notin \mathsf{View}_{\mathcal{A}} \end{aligned}$$

The adversary $\mathcal{A}$ receives the response $(\boldsymbol{\beta}^*, \mathsf{rvk}^*, \pi^*) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_{i_b}}, m^*, \mathsf{ID}^*)$. Suppose for this signature the random λ-bit string involved is $\hat{\mathbf{x}}^* \xleftarrow{R} \{0,1\}^\lambda$ chosen by the challenger $\mathcal{C}$. Then,

$$\boldsymbol{\beta}^* = \boldsymbol{\alpha}^* + \hat{\mathbf{x}}^* \in \begin{cases} S + \mathbf{x}_{i_b} + \hat{\mathbf{x}}^* & \text{if } m^* = 0 \\ S^\perp + \mathbf{z}_{i_b} + \hat{\mathbf{x}}^* & \text{if } m^* = 1 \end{cases}$$

$$\mathsf{rvk}^* = \begin{cases} O_{S+\mathbf{x}_{i_b}+\hat{\mathbf{x}}^*} & \text{if } m^* = 0 \\ O_{S^\perp+\mathbf{z}_{i_b}+\hat{\mathbf{x}}^*} & \text{if } m^* = 1 \end{cases}$$

$$\pi^* \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_{i_b}, \hat{\mathbf{x}}^*), \mathsf{ID}^* || \mathsf{rvk}^*)$$

We show that for $\mathsf{id}_{i_{\bar{b}}}$ (where $\bar{b} = 1 - b$), the same $\boldsymbol{\beta}^*$ and rvk^* can be produced by choosing different randomness. Consider the randomness

$$\tilde{\mathbf{x}}^* = \begin{cases} \mathbf{x}_{i_{\bar{b}}} + \mathbf{x}_{i_b} + \hat{\mathbf{x}}^* \text{ if } m^* = 0 \\ \mathbf{z}_{i_{\bar{b}}} + \mathbf{z}_{i_b} + \hat{\mathbf{x}}^* \text{ if } m^* = 1 \end{cases}$$

to sign on the same message m^* with respect to the same ring ID^* using the signing token $\mathsf{sk}_{\mathsf{id}_{i_{\bar{b}}}}$ of the user with identity $\mathsf{id}_{i_{\bar{b}}}$. Let $\bar{\boldsymbol{\sigma}}^* = (\bar{\boldsymbol{\beta}}^*, \bar{\mathsf{rvk}}^*, \bar{\pi}^*) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_{i_{\bar{b}}}}, m^*, \mathsf{ID}^*)$ where

$$\bar{\boldsymbol{\beta}}^* \in \begin{cases} S + \mathbf{x}_{i_{\bar{b}}} + \tilde{\mathbf{x}}^* & \text{if } m^* = 0 \\ S^\perp + \mathbf{z}_{i_{\bar{b}}} + \tilde{\mathbf{x}}^* & \text{if } m^* = 1 \end{cases} \implies \bar{\boldsymbol{\beta}}^* \in \begin{cases} S + \mathbf{x}_{i_{\bar{b}}} + (\mathbf{x}_{i_{\bar{b}}} + \mathbf{x}_{i_b} + \hat{\mathbf{x}}^*) & \text{if } m^* = 0 \\ S^\perp + \mathbf{z}_{i_{\bar{b}}} + (\mathbf{z}_{i_{\bar{b}}} + \mathbf{z}_{i_b} + \hat{\mathbf{x}}^*) & \text{if } m^* = 1 \end{cases}$$
$$\implies \bar{\boldsymbol{\beta}}^* \in \begin{cases} S + \mathbf{x}_{i_b} + \hat{\mathbf{x}}^* & \text{if } m^* = 0 \\ S^\perp + \mathbf{z}_{i_b} + \hat{\mathbf{x}}^* & \text{if } m^* = 1 \end{cases}$$

$$\begin{aligned} \bar{\mathsf{rvk}}^* &= \begin{cases} O_{S+\mathbf{x}_{i_{\bar{b}}}+\tilde{\mathbf{x}}^*} & \text{if } m^* = 0 \\ O_{S^\perp+\mathbf{z}_{i_{\bar{b}}}+\tilde{\mathbf{x}}^*} & \text{if } m^* = 1 \end{cases} \\ &= \begin{cases} O_{S+\mathbf{x}_{i_{\bar{b}}}+(\mathbf{x}_{i_{\bar{b}}}+\mathbf{x}_{i_b}+\hat{\mathbf{x}}^*)} & \text{if } m^* = 0 \\ O_{S^\perp+\mathbf{z}_{i_{\bar{b}}}+(\mathbf{z}_{i_{\bar{b}}}+\mathbf{z}_{i_b}+\hat{\mathbf{x}}^*)} & \text{if } m^* = 1 \end{cases} \\ &= \begin{cases} O_{S+\mathbf{x}_{i_b}+\hat{\mathbf{x}}^*} & \text{if } m^* = 0 \\ O_{S^\perp+\mathbf{z}_{i_b}+\hat{\mathbf{x}}^*} & \text{if } m^* = 1 \end{cases} = \mathsf{rvk}^* \end{aligned}$$

and $\bar{\pi}^* \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_{i_{\bar{b}}}, \tilde{\mathbf{x}}^*), \mathsf{ID}^* || \mathsf{rvk}^*)$.

Therefore, both the strings $\boldsymbol{\beta}^*$ and $\bar{\boldsymbol{\beta}}^*$ are sampled uniformly from the same coset $S + \mathbf{x}_{i_b} + \hat{\mathbf{x}}^*$ or $S^\perp + \mathbf{z}_{i_b} + \hat{\mathbf{x}}^*$ according as $m^* = 0$ or 1. So, the binary string $\boldsymbol{\beta}^*$ can be produced by both the users with identity id_{i_0} and id_{i_1}. Also, $\mathsf{rvk}^* = \bar{\mathsf{rvk}}^*$. Now, the only difference between the signatures $\boldsymbol{\sigma}^*$ and $\bar{\boldsymbol{\sigma}}^*$ is that the non-interactive ZK-proofs π^* and $\bar{\pi}^*$ are different due to different witnesses.

$$\pi^* \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_{i_b}, \hat{\mathbf{x}}^*), \mathsf{ID}^* || \mathsf{rvk}^*) \text{ and } \bar{\pi}^* \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_{i_{\bar{b}}}, \tilde{\mathbf{x}}^*), \mathsf{ID}^* || \mathsf{rvk}^*)$$

Therefore, π^* and $\bar{\pi}^*$ both are non-interactive ZK-proofs over the same statement $\mathsf{ID}^* || \mathsf{rvk}^*$ with respect to different witnesses.

Therefore, if $\mathcal{A}$ wins the game with non-negligible advantage, then it returns $b' = b$ with probability non-negligibly better than $\frac{1}{2}$. It implies that $\mathcal{A}$ can identify the signer id_{i_b} with probability non-negligibly better than $\frac{1}{2}$. It implies that the adversary $\mathcal{A}$ can return id_{i_b} a part of the witness of the non-interactive ZK-proof by seeing the proof π^* since we have shown that the other two elements $\boldsymbol{\beta}^*$ and rvk^* of the signature reveal no information about id_{i_b}. It contradicts that NIZK is a zero-knowledge proof system.

Hence, the adversary can win the game only with a negligible advantage.

4 D-qtIRS: Signature delegation in qtIRS

4.1 Description of the protocol

In this Section, we propose a quantum tokenized Identity-based ring signature with signature delegation D-qtIRS= (Setup, Gen, Sign, DGen, DSign, Verify) where Setup, Gen, Sign and Verify are same as in qtIRS. The DGen is a QPT algorithm run by the lessor to generate and send the delegated signing token and necessary additional information to the lessee. The DSign is a QPT algorithm run by the lessee to generate a signature. We describe the D-qtIRS protocol in Figure 6.

4.2 Security analysis

Theorem 6. *Our quantum-tokenized IRS scheme* qtIRS *presented in Fig. 6 attains strong existential unforgeability against adaptive chosen-message-and-identity attacks with signature delegation as per Def. 12 in Section 2 assuming the existence of obfuscation of membership checking circuits defined in Def. 7 in Section 2 and computationally sound non-interactive ZK proof system* NIZK *defined in Section 2.3.*

Fig. 6: Delegation of signature in quantum-tokenized IRS scheme D-qtIRS

$(\mathsf{params}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$: This algorithm is same as qtIRS.Setup.

$(\mathsf{sk}_{\mathsf{id}}, \mathsf{vk}_{\mathsf{id}}) \leftarrow \mathsf{Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id})$: This algorithm is same as qtIRS.Gen.

$\sigma \leftarrow \mathsf{Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, m, \mathsf{ID})$: This algorithm is same as qtIRS.Sign.

$(\mathsf{rsk}, \gamma = ((\mathsf{rvk}_0, \pi_0), (\mathsf{rvk}_1, \pi_1))) \leftarrow \mathsf{DGen}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, \mathsf{ID} = (\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_n))$: This QPT algorithm is run by the lessor who delegates the signature to the lessee.

i. It extracts the security parameter λ and the common reference string crs from the
i. If $\mathsf{id}_i \notin \mathsf{ID}$, it returns $\perp$ and terminates otherwise it follows the below steps.
ii. It Samples uniformly $\hat{\mathbf{x}}_0, \hat{\mathbf{x}}_1 \leftarrow \{0,1\}^\lambda$.
iii. It computes $|S\rangle^{\mathbf{x}_i+\hat{\mathbf{x}}_0, \mathbf{z}_i+\hat{\mathbf{x}}_1} \leftarrow \mathsf{QOTP}\big(|S\rangle^{\mathbf{x}_i,\mathbf{z}_i}, (\hat{\mathbf{x}}_0, \hat{\mathbf{x}}_1)\big)$ using $\mathsf{sk}_{\mathsf{id}_i} = |S\rangle^{\mathbf{x}_i,\mathbf{z}_i}$, $\hat{\mathbf{x}}_0$ and $\hat{\mathbf{x}}_1$. It sets $\mathsf{rsk} = |S\rangle^{\mathbf{x}_i+\hat{\mathbf{x}}_0, \mathbf{z}_i+\hat{\mathbf{x}}_1}$.
iv. It computes $O_{S+\mathbf{x}_i+\hat{\mathbf{x}}_0} = M[O_{S+\mathbf{x}_i}, \hat{\mathbf{x}}_0]$ and $O_{S^\perp+\mathbf{z}_i+\hat{\mathbf{x}}_1} = M[O_{S^\perp+\mathbf{x}_i}, \hat{\mathbf{x}}_1]$ and sets $\mathsf{rvk}_0 = O_{S+\mathbf{x}_i+\hat{\mathbf{x}}_0}$ and $\mathsf{rvk}_1 = O_{S^\perp+\mathbf{z}_i+\hat{\mathbf{x}}_1}$.
v. It sets $\mathbf{y}_0 = \mathsf{ID}||\mathsf{rvk}_0$ and $\mathbf{y}_1 = \mathsf{ID}||\mathsf{rvk}_1$.
vi. It computes $\pi_0 \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_i, \hat{\mathbf{x}}_0), \mathbf{y}_0)$ and $\pi_1 \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_i, \hat{\mathbf{x}}_1), \mathbf{y}_1)$.
vii. It sets the additional information $\gamma = ((\mathsf{rvk}_0, \pi_0), (\mathsf{rvk}_1, \pi_1))$.
viii. It sends the modified quantum token rsk through a secure quantum channel and the additional necessary classical information $\gamma = ((\mathsf{rvk}_0, \pi_0), (\mathsf{rvk}_1, \pi_1))$ through a secure classical channel to the lessee.

$\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi) \leftarrow \mathsf{DSign}(m, \mathsf{rsk}, \gamma = ((\mathsf{rvk}_0, \pi_0), (\mathsf{rvk}_1, \pi_1)))$: The lessee receives the delegated signing token and necessary additional classical information $(\mathsf{rsk}, \gamma = ((\mathsf{rvk}_0, \pi_0), (\mathsf{rvk}_1, \pi_1)))$ and runs the QPT algorithm DSign using a message m, quantum signing token $\mathsf{rsk} = |S\rangle^{\mathbf{x}_i+\hat{\mathbf{x}}_0, \mathbf{z}_i+\hat{\mathbf{x}}_1}$, two randomized verification keys rvk_0, rvk_1 and the ZK-proofs π_0 and π_1 as follows

i. It computes

$$\boldsymbol{\beta} = \begin{cases} \mathsf{Measure}\big(|S\rangle^{\mathbf{x}_i+\hat{\mathbf{x}}_0, \mathbf{z}_i+\hat{\mathbf{x}}_1}\big) & \text{if } m = 0 \\ \mathsf{Measure}\Big(H^{\otimes\lambda}\big(|S\rangle^{\mathbf{x}_i+\hat{\mathbf{x}}_0, \mathbf{z}_i+\hat{\mathbf{x}}_1}\big)\Big) & \text{if } m = 1 \end{cases}$$
$$= \begin{cases} \mathsf{Measure}(|S\rangle^{\mathbf{x}_i+\hat{\mathbf{x}}_0, \mathbf{z}_i+\hat{\mathbf{x}}_1}) & \text{if } m = 0 \\ \mathsf{Measure}\big(|S^\perp\rangle^{\mathbf{z}_i+\hat{\mathbf{x}}_1, \mathbf{x}_i+\hat{\mathbf{x}}_0}\big) & \text{if } m = 1 \end{cases}$$

ii. It sets

$$\mathsf{rvk} = \begin{cases} \mathsf{rvk}_0 \text{ if } m = 0 \\ \mathsf{rvk}_1 \text{ if } m = 1 \end{cases}$$
$$\pi = \begin{cases} \pi_0 \text{ if } m = 0 \\ \pi_1 \text{ if } m = 1 \end{cases}$$

It outputs the signature, the randomized verification key and the zk-proof $(\boldsymbol{\beta}, \mathsf{rvk}, \pi)$ for message m with respect to the ring of users ID.

$b \leftarrow \mathsf{Verify}(\mathsf{params}, m, \boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi), \mathsf{ID})$: This algorithm is same as qtIRS.Verify.

Proof. The challenger $\mathcal{C}$ and the adversary $\mathcal{A}$ proceed as follows.

Setup phase: The challenger $\mathcal{C}$ runs the algorithm

$$(\mathsf{params} = \mathsf{crs}, \mathsf{msk} = S) \leftarrow \mathsf{D\text{-}qtIRS.Setup}(\lambda)$$

where λ is the security parameter, $\mathsf{crs} \leftarrow \mathsf{NIZK.Setup}(\lambda)$ is the public parameter set containing λ and the master secret key $S \subseteq \{0,1\}^\lambda$ is a uniformly random $\frac{\lambda}{2}$-dimensional subspace. The challenger sends the params to the

adversary $\mathcal{A}$ and keeps the msk secret to itself. Further, $\mathcal{C}$ creates a list $\mathcal{L}_{\mathsf{vk}}$ of user verification keys, a list $\mathcal{L}_C$ of corrupted user identities and a list $\mathcal{L}_S$ of signing queries made by the adversary which are initially empty.

Key generation: The challenger $\mathcal{C}$ runs the key generation algorithm

$$(\mathsf{sk}_{\mathsf{id}_i} = |S\rangle^{\mathbf{x}_i,\mathbf{z}_i}, \mathsf{vk}_{\mathsf{id}_i} = (O_{S+\mathbf{x}_i}, O_{S^\perp+\mathbf{z}_i})) \leftarrow \mathsf{D\text{-}qtIRS.Gen}(\mathsf{params} = \mathsf{crs}, \mathsf{msk} = S, \mathsf{id}_i)$$

that takes as input the parameter set $\mathsf{params} = \mathsf{crs}$, the master secret key $\mathsf{msk} = S$ and the identity of a user $\mathsf{id}_i \in \{0,1\}^*$ $\forall i \in [p(\lambda)]$ when $p(\lambda)$ is the number of users in the scheme and generates the secret signing token $\mathsf{sk}_{\mathsf{id}_i} = |S\rangle^{\mathbf{x}_i,\mathbf{z}_i}$ and the corresponding public key $\mathsf{vk}_{\mathsf{id}_i} = (O_{S+\mathbf{x}_i}, O_{S^\perp+\mathbf{z}_i})$ consisting of two obfuscated membership checking circuits for the user with identity id_i where $\forall i \in [p(\lambda)]$, $\mathbf{x}_i, \mathbf{z}_i \leftarrow h(\mathbf{r}_i, \mathsf{id}_i)$ for randomly chosen 2λ-bit binary string $\mathbf{r}_i$. The challenger adds the user identity and the corresponding verification key pair $(\mathsf{id}_i, \mathsf{vk}_{\mathsf{id}_i})$ to the list $\mathcal{L}_{\mathsf{vk}}$ $\forall i \in [p(\lambda)]$. The challenger $\mathcal{C}$ sends the list $\mathcal{L}_{\mathsf{vk}}$ to the adversary $\mathcal{A}$ and keeps the secret keys corresponding to specific user identities $\{(\mathsf{id}_1, \mathsf{sk}_{\mathsf{id}_1}), (\mathsf{id}_2, \mathsf{sk}_{\mathsf{id}_2}), \ldots, (\mathsf{id}_{p(\lambda)}, \mathsf{sk}_{\mathsf{id}_{p(\lambda)}})\}$ and random strings $\{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_{p(\lambda)}\}$ secret to itself.

Corruption query: The adversary $\mathcal{A}$ makes $l(\lambda)$ adaptive corruption queries to $\mathcal{C}$. The corruption query is of the form (id_j) with $j \in [p(\lambda)]$ where $l(\cdot)$ is a polynomial. The challenger $\mathcal{C}$ returns the secret key $\mathsf{sk}_{\mathsf{id}_j} = |S\rangle^{\mathbf{x}_j,\mathbf{z}_j}$ of the j-th user to $\mathcal{A}$ and adds id_j to the list $\mathcal{L}_C$.

The **Claim 1** of Theorem 4 ensures that given the quantum signing token $\mathsf{sk}_{\mathsf{id}_j}$, the adversary can neither copy it nor learn any information about it.

Signing query: The adversary $\mathcal{A}$ makes $r(\lambda)$ many signing queries $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to $\mathcal{C}$ where $p(\cdot)$ is a polynomial, $m_j \in \{0,1\}$ is a message, $\mathsf{ID}_j \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ is a set of identities chosen by $\mathcal{A}$ and $\mathsf{id}_k \in \mathsf{ID}_j$ is the identity of a user chosen by $\mathcal{A}$. The challenger $\mathcal{C}$ responds with $\boldsymbol{\sigma}_j = (\boldsymbol{\beta}_j, \mathsf{rvk}_j, \pi_j)$ such that $\boldsymbol{\sigma}_j = (\boldsymbol{\beta}_j, \mathsf{rvk}_j, \pi_j) \leftarrow \mathsf{D\text{-}qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_j, \mathsf{ID}_j)$ where

$$\boldsymbol{\beta}_j = \begin{cases} \mathbf{s}_j + \mathbf{x}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 0 \\ \hat{\mathbf{s}}_j + \mathbf{z}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 1 \end{cases}$$

$$\mathsf{rvk}_j = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_j} \text{ if } m_j = 0 \\ O_{S^\perp+\mathbf{z}_k+\hat{\mathbf{x}}_j} \text{ if } m_j = 1 \end{cases}$$

$$\pi_j \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_j), \mathsf{ID}_j || \mathsf{rvk}_j)$$

for $\hat{\mathbf{x}}_j \xleftarrow{R} \{0,1\}^\lambda$, $\mathbf{s}_j \xleftarrow{R} S$ and $\hat{\mathbf{s}}_j \xleftarrow{R} S^\perp$. Then the challenger adds $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to the list $\mathcal{L}_S$.

Note that the adversary $\mathcal{A}$ is allowed to make the same signing query $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ polynomially many times to $\mathcal{C}$. Suppose the adversary makes the query $(\mathsf{id}_k, m_j, \mathsf{ID}_j)$ to the challenger $q(\lambda)$ times. The responses by the challenger would be $\boldsymbol{\sigma}_{ji} = (\boldsymbol{\beta}_{ji}, \mathsf{rvk}_{ji}, \pi_{ji})$ where

$$\boldsymbol{\beta}_{ji} = \begin{cases} \mathbf{s}_{ji} + \mathbf{x}_k + \hat{\mathbf{x}}_{ji} \text{ if } m_j = 0 \\ \hat{\mathbf{s}}_{ji} + \mathbf{z}_k + \hat{\mathbf{x}}_{ji} \text{ if } m_j = 1 \end{cases}$$

$$\mathsf{rvk}_{ji} = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_{ji}} \text{ if } m_j = 0 \\ O_{S^\perp+\mathbf{z}_k+\hat{\mathbf{x}}_{ji}} \text{ if } m_j = 1 \end{cases}$$

$$\pi_{ji} \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_{ji}), \mathsf{ID}_j || \mathsf{rvk}_{ji})$$

for $\hat{\mathbf{x}}_{ji} \xleftarrow{R} \{0,1\}^\lambda$, $\mathbf{s}_{ji} \xleftarrow{R} S$ and $\hat{\mathbf{s}}_{ji} \xleftarrow{R} S^\perp$ $\forall i \in [q(\lambda)]$. Hence, each time the adversary gets different responses and the XORs with the random strings $\hat{\mathbf{x}}_{ji}$ makes it difficult to manipulate the results to obtain elements of S or the strings $\mathbf{x}_k, \mathbf{z}_k$.

Delegation query: The adversary sends $m(\lambda)$ signing token delegation request of the form (id_j) with $j \in [p(\lambda)]$. The challenger chooses a ring of users ID such that $\mathsf{id}_j \in \mathsf{ID} \subseteq \{\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_{p(\lambda)}\}$ and responds with

$$(\mathsf{rsk}_j, \gamma_j = ((\mathsf{rvk}_{j_0}, \pi_{j_0}), (\mathsf{rvk}_{j_1}, \pi_{j_1})) \leftarrow \mathsf{D\text{-}qtIRS.DGen}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_j}, \mathsf{ID})$$

where

$$\mathsf{rsk}_j = |S\rangle^{\mathbf{x}_j+\hat{\mathbf{x}}_{j_0},\mathbf{z}_j+\hat{\mathbf{x}}_{j_1}}, \mathsf{rvk}_{j_0} = O_{S+\mathbf{x}_j+\hat{\mathbf{x}}_{j_0}}, \mathsf{rvk}_{j_1} = O_{S^\perp+\mathbf{z}_j+\hat{\mathbf{x}}_{j_1}},$$

$$\pi_{j_0} \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_j, \hat{\mathbf{x}}_{j_0}), \mathsf{ID}||\mathsf{rvk}_{j_0}),\ \pi_{j_1} \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_j, \hat{\mathbf{x}}_{j_1}), \mathsf{ID}||\mathsf{rvk}_{j_1}).$$

The challenger $\mathcal{C}$ and adds (id_j) to the list $\mathcal{L}_{\mathsf{del}}$.

View: After $l(\lambda)$ corruption queries and $r(\lambda)$ signing queries and $m(\lambda)$ signature delegation queries such that $1 \le l(\lambda), r(\lambda), m(\lambda) \le p(\lambda)$.

For $\mathcal{L}_C$, let the adversary $\mathcal{A}$ measures the first t signing tokens $\mathsf{sk}_{\mathsf{id}_j}$ for $j \in \{i_1, \ldots, i_t\} \subseteq [l(\lambda)]$ directly yielding $\mathcal{A}$ to have the knowledge $\{\mathbf{s}_{i_1} + \mathbf{x}_{i_1}, \ldots, \mathbf{s}_{i_t} + \mathbf{x}_{i_t}\}$ where $\{\mathbf{s}_{i_1}, \ldots, \mathbf{s}_{i_t}\} \subseteq S$ and remaining $(l(\lambda) - t)$ many tokens $\mathsf{sk}_{\mathsf{id}_j}$ for $j \in \{i_{(t+1)}, \ldots, i_{l(\lambda)}\} \subseteq [l(\lambda)]$, after Hadamard transform resulting $\mathcal{A}$ to have the knowledge of $\{\hat{\mathbf{s}}_{i_{t+1}} + \mathbf{z}_{i_{t+1}}, \ldots, \hat{\mathbf{s}}_{i_{l(\lambda)}} + \mathbf{z}_{i_{l(\lambda)}}\}$ where $\{\mathbf{s}_{i_{t+1}}, \ldots, \mathbf{s}_{i_{l(\lambda)}}\} \subseteq S^{\perp}$. Finally, $\mathcal{A}$ has the information view as presented in Fig. 6.

Table 6: $\mathsf{View}_{\mathcal{A}}^{\mathsf{sEU\text{-}ACMID}}$

$$\boxed{\begin{array}{c} \mathsf{crs}, \quad \mathcal{L}_{\mathsf{vk}} = \{(\mathsf{id}_j, \mathsf{vk}_{\mathsf{id}_j}) |\ \ j \in [p(\lambda)]\}, \quad \mathcal{L}_S = \{(m_{j_i}, \boldsymbol{\sigma}_{ji}, \mathsf{ID}_{j_i}) |\ \ i \in r(\lambda)\}, \\ \mathbf{s}_{i_1} + \mathbf{x}_{i_1}, \ldots, \mathbf{s}_{i_t} + \mathbf{x}_{i_t}, \hat{\mathbf{s}}_{i_{t+1}} + \mathbf{z}_{i_{t+1}}, \ldots, \hat{\mathbf{s}}_{i_{l(\lambda)}} + \mathbf{z}_{i_{l(\lambda)}} \quad \forall \mathsf{id}_{i_k} \in \mathcal{L}_C, \\ \text{where } \{\mathbf{s}_{i_1}, \ldots, \mathbf{s}_{i_t}\} \subseteq S \ \ \& \ \ \{\mathbf{s}_{i_{t+1}}, \ldots, \mathbf{s}_{i_{l(\lambda)}}\} \subseteq S^{\perp}, \quad (\mathsf{rsk}_j, \gamma_j) \quad \forall \mathsf{id}_j \in \mathcal{L}_{\mathsf{del}} \end{array}}$$

$(\mathsf{params} = \mathsf{crs}, \mathsf{msk} = S) \leftarrow \mathsf{qtIRS.Setup}(1^{\lambda}); \quad \left(\mathsf{sk}_{\mathsf{id}_j} = |S\rangle^{\mathbf{x}_j, \mathbf{z}_j},\ \mathsf{vk}_{\mathsf{id}_j} = (O_{S+\mathbf{x}_j}, O_{S^{\perp}+\mathbf{z}_j})\right) \leftarrow \mathsf{qtIRS.Gen}(\mathsf{params}, \mathsf{msk}, \mathsf{id}_j),$

$\mathbf{x}_j, \mathbf{z}_j \leftarrow h(\mathbf{r}_j, \mathsf{id}_j),\ \mathbf{r}_j \xleftarrow{R} \{0,1\}^{\lambda} \quad \forall j \in [p(\lambda)];$
$\boldsymbol{\sigma}_j = (\boldsymbol{\beta}_j, \mathsf{rvk}_j, \pi_j) \leftarrow \mathsf{qtIRS.Sign}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_k}, m_j, \mathsf{ID}_j), \quad m_j \in \{0,1\}, \quad \mathsf{id}_k \in \mathsf{ID}_j \subseteq \{\mathsf{id}_i\}_{i \in p(\lambda)},$

$$\boldsymbol{\beta}_j = \begin{cases} \mathbf{s}_j + \mathbf{x}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 0 \\ \hat{\mathbf{s}}_j + \mathbf{z}_k + \hat{\mathbf{x}}_j \text{ if } m_j = 1 \end{cases}, \quad \mathsf{rvk}_j = \begin{cases} O_{S+\mathbf{x}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 0 \\ O_{S^{\perp}+\mathbf{z}_k+\hat{\mathbf{x}}_j} & \text{if } m_j = 1 \end{cases}, \quad \pi_j \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_k, \hat{\mathbf{x}}_j), \mathsf{ID}_j||\mathsf{rvk}_j),$$

$(\mathsf{rsk}_j, \gamma_j = ((\mathsf{rvk}_{j_0}, \pi_{j_0}), (\mathsf{rvk}_{j_1}, \pi_{j_1})) \leftarrow \mathsf{D\text{-}qtIRS.DGen}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_j}, \mathsf{ID}), \quad \mathsf{rsk}_j = |S\rangle^{\mathbf{x}_j+\hat{\mathbf{x}}_{j_0}, \mathbf{z}_j+\hat{\mathbf{x}}_{j_1}},$
$\mathsf{rvk}_{j_0} = O_{S+\mathbf{x}_j+\hat{\mathbf{x}}_{j_0}}, \quad \mathsf{rvk}_{j_1} = O_{S^{\perp}+\mathbf{z}_j+\hat{\mathbf{x}}_{j_1}}, \quad \pi_{j_0} \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_j, \hat{\mathbf{x}}_{j_0}), \mathsf{ID}||\mathsf{rvk}_{j_0}), \quad \pi_{j_1} \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_j, \hat{\mathbf{x}}_{j_1}), \mathsf{ID}||\mathsf{rvk}_{j_1})$

Forge: Suppose $\mathcal{L}_{\mathsf{del}} \cap \mathsf{ID} = \{\mathsf{id}_j\}$ and the adversary have $(\mathsf{rsk}_j, \gamma_j)$ where $\mathsf{rsk} = |S\rangle^{\mathbf{x}_j+\mathbf{x}_0, \mathbf{z}_j+\mathbf{x}_1}$. Let us assume that the adversary wins the game with probability $\mu(\lambda)$ where $\mu(\cdot)$ is a non-negligible function. Then the following two cases may happen.

Case 1: The adversary outputs two valid tuples $(m_1^*, \boldsymbol{\sigma}_1^*, \mathsf{ID})$ and $(m_2^*, \boldsymbol{\sigma}_2^*, \mathsf{ID})$ with probability $\mu(\lambda)$ which pass the verification successfully such that $(m_1^*, \boldsymbol{\sigma}_1^*, \mathsf{ID}), (m_2^*, \boldsymbol{\sigma}_2^*, \mathsf{ID}) \notin \mathcal{L}_S$, $(m_1^*, \boldsymbol{\sigma}_1^*) \neq (m_2^*, \boldsymbol{\sigma}_2^*)$, $(\mathsf{id}) \notin \mathcal{L}_C\ \forall \mathsf{id} \in \mathsf{ID}$.

Suppose he adversary outputs two valid tuples $(m_1^*, \boldsymbol{\sigma}_1^*, \mathsf{ID})$ and $(m_2^*, \boldsymbol{\sigma}_2^*, \mathsf{ID})$ which pass the verification successfully such that $(m_1^*, \boldsymbol{\sigma}_1^*, \mathsf{ID}), (m_2^*, \boldsymbol{\sigma}_2^*, \mathsf{ID}) \notin \mathcal{L}_S$ and $(\mathsf{id}) \notin \mathcal{L}_C\ \forall \mathsf{id} \in \mathsf{ID}$. Let $\boldsymbol{\sigma}_1^* = (\boldsymbol{\beta}_1^*, \mathsf{rvk}_1^*, \pi_1^*)$ and $\boldsymbol{\sigma}_2^* = (\boldsymbol{\beta}_2^*, \mathsf{rvk}_2^*, \pi_2^*)$. Then we will show that $(m_1^*, \boldsymbol{\sigma}_1^*) = (m_2^*, \boldsymbol{\sigma}_2^*)$. Note that the adversary has the delegated token as extra information than the EU-ACMI game in def. 9 that gives $\mathsf{rsk}_j = |S\rangle^{\mathbf{x}_j+\mathbf{x}_0, \mathbf{z}_j+\mathbf{x}_1}$, $\gamma_j = ((\mathsf{rvk}_{j_0}, \pi_{j_0}), (\mathsf{rvk}_{j_1}, \pi_{j_1}))$.

Claim 1: $m_1^* = m_2^*$.
Proof of claim 1: If possible let $m_1^* \neq m_2^*$, without loss of generality let $m_1^* = 0$, $m_2^* = 1$. Then, $\boldsymbol{\beta}_1^* \in S + \mathbf{x}_j + \mathbf{x}_0$ and $\boldsymbol{\beta}_2^* \in S^{\perp} + \mathbf{z}_j + \mathbf{x}_1$.

However, the adversary $\mathcal{A}$ can measure either $|S\rangle^{\mathbf{x}_j+\mathbf{x}_0, \mathbf{z}_j+\mathbf{x}_1}$ or $H^{\otimes\lambda}\left(|S\rangle^{\mathbf{x}_j+\mathbf{x}_0, \mathbf{z}_j+\mathbf{x}_1}\right)$ since $\mathcal{A}$ cannot clone a quantum state. Thus, it can obtain either an element of $S + \mathbf{x}_j + \mathbf{x}_0$ or $S^{\perp} + \mathbf{z}_j + \mathbf{x}_1$ but not both together.

Hence, $m_1^* = m_2^*$.

Claim 2: $\boldsymbol{\sigma}_1^* = \boldsymbol{\sigma}_2^*$ i.e., $(\boldsymbol{\beta}_1^*, \mathsf{rvk}_1^*, \pi_1^*) = (\boldsymbol{\beta}_2^*, \mathsf{rvk}_2^*, \pi_2^*)$.
Proof of claim 2:Without loss of generality, let $m_1^* = m_2^*$. If possible let $(\boldsymbol{\beta}_1^*, \mathsf{rvk}_1^*, \pi_1^*) \neq (\boldsymbol{\beta}_2^*, \mathsf{rvk}_2^*, \pi_2^*)$.

Now, $\boldsymbol{\beta}_1^*, \boldsymbol{\beta}_2^* \in S + \mathbf{x}_j + \mathbf{x}_0$. Let $\boldsymbol{\beta}_1^* = \mathbf{s}_1 + \mathbf{x}_j + \mathbf{x}_0$, $\boldsymbol{\beta}_2^* = \mathbf{s}_2 + \mathbf{x}_j + \mathbf{x}_0$. If $\mathbf{s}_1 = \mathbf{s}_2$, then $\mathsf{rvk}_1^* = \mathsf{rvk}_2^* = \mathsf{rvk}_0 = O_{S+\mathbf{x}_j+\mathbf{x}_0}$ and $\pi_1^* = \pi_2^* = \pi_0 = \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id_j}, \mathbf{x}_0), \mathsf{ID}||\mathsf{rvk}_0)$. This implies, $(\boldsymbol{\beta}_1^*, \mathsf{rvk}_1^*, \pi_1^*) = (\boldsymbol{\beta}_2^*, \mathsf{rvk}_2^*, \pi_2^*)$.

So, in order to get $(\boldsymbol{\beta}_1^*, \mathsf{rvk}_1^*, \pi_1^*) \neq (\boldsymbol{\beta}_2^*, \mathsf{rvk}_2^*, \pi_2^*)$, we must have $\mathbf{s}_1 \neq \mathbf{s}_2$ i.e., $\boldsymbol{\beta}_1^* \neq \boldsymbol{\beta}_2^*$. Then the adversary $\mathcal{A}$ gets two distinct elements of $\boldsymbol{\beta}_1^*, \boldsymbol{\beta}_2^* \in S + \mathbf{x}_j + \mathbf{x}_0$ given a signle copy of $\mathsf{rsk} = |S\rangle^{\mathbf{x}_j+\mathbf{x}_0, \mathbf{z}_j+\mathbf{x}_1}$ which contradicts the basic features of quantum information.

Therefore, $\boldsymbol{\sigma}_1^* \neq \boldsymbol{\sigma}_2^*$.

Case 2: The adversary outputs one valid tuple $(m, \boldsymbol{\sigma}, \mathsf{ID}^*)$ with non-negligible probability $\mu(\lambda)$ which passes the verification successfully such that $(m, \boldsymbol{\sigma}, \mathsf{ID}^*) \notin \mathcal{L}_S$, $(\mathsf{id}) \notin \mathcal{L}_C\ \forall \mathsf{id} \in \mathsf{ID}^*$ and $\mathsf{ID} \neq \mathsf{ID}^*$. Then we will show a contradiction to the existence of computationally sound non-interactive ZK proof system NIZK. Since the statement

$(\mathsf{ID}^*||\mathsf{rvk})$ is different from $(\mathsf{ID}||\mathsf{rvk})$, the corresponding proofs must be different. Most importantly, the witnesses for the ZK proofs present in γ are not given to the adversary $\mathcal{A}$. Let $\boldsymbol{\sigma} = (\boldsymbol{\beta}, \mathsf{rvk}, \pi)$ where

$$\boldsymbol{\beta} = \begin{cases} \mathbf{s} + \mathbf{x}_j + \hat{\mathbf{x}} \text{ if } m = 0 \\ \hat{\mathbf{s}} + \mathbf{z}_j + \hat{\mathbf{x}} \text{ if } m = 1 \end{cases}$$

$$\mathsf{rvk} = \begin{cases} O_{S+\mathbf{x}_j+\hat{\mathbf{x}}} \quad \text{if } m = 0 \\ O_{S^\perp+\mathbf{z}_j+\hat{\mathbf{x}}} \text{ if } m = 1 \end{cases}$$

$$\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{crs}, (\mathsf{id}_j, \hat{\mathbf{x}}), \mathsf{ID}^*||\mathsf{rvk})$$

Then, the adversary can generate a non-interactive ZK proof π on statement $(\mathsf{ID}^*||\mathsf{rvk})$ without the knowledge of the randomness $\hat{\mathbf{x}}$ involved in the witness with non-negligible probability $\mu(\lambda)$ which contradicts the computational soundness of the non-interactive ZK proof system NIZK. Hence, if $\mathcal{A}$ can generate a valid signature on a message m with respect to a ring ID^* different from the one specified by the challenger during $\mathsf{D\text{-}qtIRS.Gen}$, then it contradicts the computational soundness of the NIZK protocol.

Hence, our assumption was wrong. An adversary $\mathcal{A}$ can win only with negligible probability.

4.3 Leased key revocation

Suppose a lessor with signing token $\mathsf{sk}_{\mathsf{id}_i}$ wants to lease one signing authority to a lessee during her one-week vacation. Then she runs the QPT algorithm DGen given in Fig. 6 generates

$$(\mathsf{rsk}, \gamma = ((\mathsf{rvk}_0, \pi_0), (\mathsf{rvk}_1, \pi_1))) \leftarrow \mathsf{DGen}(\mathsf{params}, \mathsf{sk}_{\mathsf{id}_i}, \mathsf{ID} = (\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_n))$$

She sends $(\mathsf{rsk}, \gamma = ((\mathsf{rvk}_0, \pi_0), (\mathsf{rvk}_1, \pi_1)))$ to the lessee using secure communication channel. Using that delegated token and additional classical information the lessee can sign one message of his choice. Suppose the lessor's one-week vacation is over and the lessee claims that he has not used the lessor's signing authority. The lessor now wants to revoke her signing authority given to the lessee. Then the lessor randomly chooses a message $m \xleftarrow{R} \{0,1\}$ and asks the lessee to sign that message by running the QPT algorithm DSign given in Fig. 6. The lessee signs that message and sends the signature to the lessor as a proof. The lessor verifies the signature and gets convinced that her signing authority given to the lessee is revoked. The strong existential unforgeability of $\mathsf{D\text{-}qtIRS}$ against adaptive chosen message and identity attack with signature delegation ensures that the lessee can no longer generate a new signature after the successful verification of the signature on the dummy message.

If the lessor provides n signing tokens to the lessee among which only t tokens are used by him in the given time, then the lessor can revoke the remaining $n - t$ tokens simiarly.

5 Conclusion

Throughout this paper, we have proposed the first IRS qtIRS in the quantum world and its powerful variant $\mathsf{D\text{-}qtIRS}$ and analyzed their stronger security in terms of unforgeability and anonymity in the standard model from the obfuscation of membership checking circuits and NIZK. We have justified the existence of the obfuscation of membership-checking circuits. The $\mathsf{D\text{-}qtIRS}$ is the first IRS that allows signature delegation and revocation. In the near future, we would like to enhance its efficiency in terms of signature size.

References

1. Wiesner, S. (1983). *Conjugate coding.* ACM Sigact News, 15(1), 78-88.
2. Nielsen, M. A. and Chuang, I. L. (2010). *Quantum computation and quantum information.* Cambridge University Press.
3. Shamir, A. (1985). *Identity-based cryptosystems and signature schemes.* In Advances in Cryptology: Proceedings of CRYPTO 84 4 (pp. 47-53). Springer Berlin Heidelberg.
4. Rivest, R. L., Shamir, A. and Tauman, Y. (2001). *How to leak a secret.* In Advances in Cryptology—ASIACRYPT 2001: 7th International Conference on the Theory and Application of Cryptology and Information Security Gold Coast, Australia, December 9–13, 2001 Proceedings 7 (pp. 552-565). Springer Berlin Heidelberg.
5. Bender, A., Katz, J. and Morselli, R. (2006). *Ring signatures: Stronger definitions, and constructions without random oracles.* In Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3 (pp. 60-79). Springer Berlin Heidelberg.

6. Malavolta, G. and Schröder, D. (2017). *Efficient ring signatures in the standard model.* In Advances in Cryptology–ASIACRYPT 2017: 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II 23 (pp. 128-157). Springer International Publishing.
7. Zhang, F. and Kim, K. (2002). *ID-based blind signature and ring signature from pairings.* In Advances in Cryptology—ASIACRYPT 2002: 8th International Conference on the Theory and Application of Cryptology and Information Security Queenstown, New Zealand, December 1–5, 2002 Proceedings 8 (pp. 533-547). Springer Berlin Heidelberg.
8. Chow, S. S., Yiu, S. M. and Hui, L. C. (2005). *Efficient identity-based ring signature.* In Applied Cryptography and Network Security: Third International Conference, ACNS 2005, New York, NY, USA, June 7-10, 2005. Proceedings 3 (pp. 499-512). Springer Berlin Heidelberg.
9. Au, M. H., Liu, J. K., Yuen, T. H. and Wong, D. S. (2006). *ID-based ring signature scheme secure in the standard model.* In Advances in Information and Computer Security: First International Workshop on Security, IWSEC 2006, Kyoto, Japan, October 23-24, 2006. Proceedings 1 (pp. 1-16). Springer Berlin Heidelberg.
10. Herranz, J. (2007). *Identity-based ring signatures from RSA.* Theoretical Computer Science, 389(1-2), 100-117.
11. Au, M. H., Liu, J. K., Susilo, W. and Zhou, J. (2013). *Realizing fully secure unrestricted ID-based ring signature in the standard model based on HIBE.* IEEE Transactions on Information Forensics and Security, 8(12), 1909-1922.
12. Deng, L., Jiang, Y. and Ning, B. (2019). *Identity-based linkable ring signature scheme.* IEEE Access, 7, 153969-153976.
13. Zhandry, M. (2019). *Quantum lightning never strikes the same state twice.* In 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Eurocrypt 2019 (pp. 408-438). Springer Verlag.
14. Le, H. Q., Vo, B., Duong, D. H., Susilo, W., Le, N. T., Fukushima, K. and Kiyomoto, S. (2021). *Identity-based linkable ring signatures from lattices.* IEEE Access, 9, 84739-84755.
15. Hu, M., Liu, Z., Ren, X. and Zhou, Y. (2024). *Linkable ring signature scheme with stronger security guarantees.* Information Sciences, 680, 121164.
16. Ben-David, S. and Sattath, O. (2023). *Quantum tokens for digital signatures.* Quantum, 7, 901.
17. Tang, Y., Xia, F., Ye, Q., Wang, M., Mu, R. and Zhang, X. (2021). *Identity-Based Linkable Ring Signature on NTRU Lattice.* Security and Communication Networks, 2021(1), 9992414.
18. Shmueli, O. (2022, June). *Public-key quantum money with a classical bank.* In Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing (pp. 790-803).
19. Shmueli, O. (2022, August). *Semi-quantum tokenized signatures.* In Annual International Cryptology Conference (pp. 296-319). Cham: Springer Nature Switzerland.
20. Hofheinz, D., Malone-Lee, J. and Stam, M. (2007). *Obfuscation for cryptographic purposes.* In Theory of Cryptography: 4th Theory of Cryptography Conference, TCC 2007, Amsterdam, The Netherlands, February 21-24, 2007. Proceedings 4 (pp. 214-232). Springer Berlin Heidelberg.
21. Aaronson, S. and Christiano, P. (2012, May). *Quantum money from hidden subspaces.* In Proceedings of the forty-fourth annual ACM symposium on Theory of computing (pp. 41-60).
22. Groth, J. (2016). *On the size of pairing-based non-interactive arguments.* In Advances in Cryptology–EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II 35 (pp. 305-326). Springer Berlin Heidelberg.
23. Coladangelo, A., Liu, J., Liu, Q. and Zhandry, M. (2021). *Hidden cosets and applications to unclonable cryptography.* In Advances in Cryptology–CRYPTO 2021: 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16–20, 2021, Proceedings, Part I 41 (pp. 556-584). Springer International Publishing.
24. Gay, R. and Pass, R. (2021, June). *Indistinguishability obfuscation from circular security.* In Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing (pp. 736-749).
25. Brakerski, Z., Döttling, N., Garg, S. and Malavolta, G. (2023). *Candidate iO from homomorphic encryption schemes.* Journal of Cryptology, 36(3), 27.
26. Gao, W., Fu, T., Qin, B., Dong, X., Zhao, Z. and Liu, M. *Logarithmic Identity-Based Ring Signature Over Lattices and Linkable Variant.* Available at SSRN 4898352.
27. Gao, W., Yao, H., Qin, B., Dong, X., Zhao, Z. and Zeng, J. (2024). *Post-Quantum Secure ID-Based (Threshold) Linkable Dual-Ring Signature and Its Application in Blockchain Transactions.* Cryptography, 8(4), 48.
28. Wang, J. (2008). *Identity-based ring signature from lattice basis delegation.* Tsinghua University, Beijing.
29. Buser, M., Liu, J. K., Steinfeld, R. and Sakzad, A. (2022, June). *Post-quantum ID-based ring signatures from symmetric-key primitives.* In International Conference on Applied Cryptography and Network Security (pp. 892-912). Cham: Springer International Publishing.