Bulletin of Networking, Computing, Systems, and Software – www.bncss.org, ISSN 2186-5140
Volume 15, Number 1, pages 6–11, January 2026

Short Revocable Group Signatures with Compact Revocation Lists from SPS-EQ (preliminary version)

Yuka Yonezawa*, Toru Nakanishi*, Teruaki Kitasuka*, Zhuotao Lian*
*Graduate School of Advanced Science and Engineering, Hiroshima University, Higashi-Hiroshima 739-8527, Japan
Email: {m242085, t-nakanishi, kitasuka, lian-zhuotao}@hiroshima-u.ac.jp

***Abstract*—Group signature schemes offer privacy-enhancing user authentication by enabling a user to anonymously prove membership in a group. However, revoking the membership of anonymous users is challenging, and thus various revocable group signature schemes have been proposed. Existing schemes that rely on zero-knowledge proofs often suffer from long signature sizes and large revocation lists, resulting in poor efficiency and scalability. In this paper, we propose a revocable group signature scheme where both the revocation list size and the signature length are reduced by using randomizable SPS-EQ and set commitments without requiring zero-knowledge proofs. Compared to the previous scheme, signature size in our proposed scheme is reduced to 58%. Our implementation on a PC confirms the efficiency and practicality of the proposed scheme.**

I. Introduction

In the widely used ID-based user authentication, personal information such as name, address, telephone number, and furthermore service usage history are linked with a user ID. Then the information is stored in the server, but there is a possibility that it is leaked, which can be a privacy problem for users. As a solution to this problem, an authentication scheme called a group signature [1] has been proposed. The group signature scheme is a digital signature scheme in which a user who belongs to a group can anonymously prove the membership. In this scheme, a group manager (GM) manages the membership to a group and an opener has the authority to identify the signer from a signature in cases of dispute. In the group signature scheme, user IDs are not passed to the server, and thus a privacy-preserving user authentication is realized. Group signature schemes can be extended to anonymous credential (AC) systems [2], where an issuer issues a certificate on attributes to a user, and the user can select and prove a part of attributes anonymously. A group signature scheme requires a revocation mechanism to invalidate signatures generated by a signer removed from the group. Since it is not easy to check the revocation due to the anonymity of signatures, lots of revocable group signature (R-GS) schemes have been proposed.

In [3], Libert, Peters and Yung first propose a scalable R-GS scheme (LPY scheme), where the signature length and the signature generation and verification times are $\mathcal{O}(1)$, and the public key size and membership certificate size are sublinear for a maximum number of group members N and a maximum number of revoked users r. However, since it is based the standard model, the signature consists of about 100 elements and the signature length is large, due to the cost of the zero-knowledge proof used. Therefore, in [4], Ohara et al. proposed an improved version of the CS (Complete Subtree)-based LPY scheme in a random oracle model. In this scheme, the number of elements in signatures is reduced to 18 elements and thus short signatures are achieved. Following the methodology of [4] introduces a revocable group signature scheme that is secure under simple assumptions. On the other hand, the revocation list size in [4] is $\mathcal{O}(r \log \frac{N}{r})$, which is a problem when the number of revoked users is large. This is because the latest revocation list must be obtained and referenced by the signer every single time during the authentication process.

An existing R-GS scheme [5] addresses the issue of revocation list size by using a vector commitment to compress every K revocation entries generated through the SD (Subset Difference) method. This technique effectively reduces the size of the revocation list to $\mathcal{O}(\frac{r}{K})$. However, as a trade-off, the signing time is increased to $\mathcal{O}(K)$. Furthermore, as mentioned in [3], since the Groth-Sahai proof in the standard model is used and is based on the SD method, which is more complex than the CS method, the signature size in this scheme is about three times that of [3].

In [6], Sugimoto and Nakanishi proposed an efficient R-GS scheme with compact revocation list, based on Ohara et al.'s scheme. In this scheme, the size of the revocation list is reduced by compressing nodes in a tree structure based on the CS method with vector commitment. In the case of compressing every K nodes in the revocation list, the size of the revocation list is at most $\mathcal{O}(\frac{r \log \frac{N}{r}}{K})$. On the other hand, the trade-off of shortening the revocation list is that the signature length increases due to the additional zero-knowledge proofs for the verifications of vector commitment and AHO signatures.

In this paper, we propose a shorter R-GS scheme with a compact revocation list, based on SPS-EQ signatures [7] and set commitments [7]. The previous scheme [6] employs AHO signatures, vector commitments [7], and zero-knowledge proofs, all of which contribute to complex signing computations and long signatures. In contrast, SPS-EQ signatures and set commitments can be randomized without the need for zero-knowledge proofs, but they cannot be directly integrated into

the R-GS scheme [6]. By partially utilizing zero-knowledge proofs, we construct a shorter and secure R-GS scheme with a compact revocation list. We further demonstrate the effectiveness of the proposed scheme by implementing it on a PC. Compared to the previous scheme [6], our proposed scheme offers the following advantages:

- **Reduced Signature Length:** Our approach reduces the signature length to about 60% compared to the scheme [6] (from 3,392 Bytes to 2,049 Bytes in our experiments).
- **Compact Revocation Lists:** The use of set commitments allows for more reduction of the revocation list size, achieving about half the size of [6] under similar conditions.
- **Constant-Size Membership Certificates:** Our scheme utilizes $O(1)$-size membership certificates, a notable improvement over the $O(\log N)$-size certificates required in [6].

II. Preliminaries

A. Bilinear map

In this paper, we use bilinear groups with a bilinear map. Let $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_T$ be cyclic groups of the same prime order p. Let P and $\hat{P}$ be generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. Then, the bilinear map e: $\mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ satisfies the following bilinearity and non-degeneracy:

Bilinearity: for any $P \in \mathbb{G}_1, Q \in \mathbb{G}_2, a, b \in \mathbb{Z}_p$,

$$e(aP, bQ) = e(P, Q)^{ab}$$

Non-degeneracy: $e(P, \hat{P}) \neq 1_{\mathbb{G}_T}$, where $1_{\mathbb{G}_T}$ is the identity element of $\mathbb{G}_T$.

The above bilinear map can be realized by pairing on elliptic curves.

B. Signatures of Knowledge

In this paper, we use signatures based on proofs of knowledge (SPK) derived from zero-knowledge proofs of knowledge via Fiat-Shamir heuristic. A zero-knowledge proof of knowledge is an interactive protocol between a prover P and a verifier V to prove knowledge of secret information satisfying a certain relation without revealing the secret information. We adopt SPKs to prove the knowledge of secret values $x_1, \cdots, x_k \in \mathbb{Z}_p$ on a message m for a equation $y = x_1 P_1 + \cdots + x_k P_k$.

C. SPS-EQ

We utilize SPS-EQ (structure-preserving signatures on equivalent classes) [7]. In a structure-preserving signature scheme, all components of messages, public keys, and the signatures are group elements of bilinear groups. Thus, cryptographic outputs such as a set commitment can be signed. In the SPS-EQ, one can sign multiple messages which are viewed as equivalence classes of group element vectors. For a group $\mathbb{G}$ with a prime order p, a message vector $\vec{M} = (m_i)_{i\in[\ell]} \in \mathbb{G}^\ell$ is related to $\vec{M'} = (m'_i)_{i\in[\ell]} \in \mathbb{G}^\ell$ through the equivalence relation $\mathcal{R} = \{(\vec{M}, \vec{M'}) \,|\, \exists s \in \mathbb{Z}_p : \vec{M'} = s\vec{M}\}$. Here, $\forall i \in [\ell], \exists \theta_i \in \mathbb{Z}_p : m_i = \theta_i m_1 \Leftrightarrow m'_i = \theta_i m'_1$, which implies that for message vectors within the same equivalence class, the discrete logarithmic relationships between their elements are preserved. Moreover, for random scalar $\mu \in \mathbb{Z}_p$, $\mu\vec{M}$ becomes randomized under the DDH assumption, making it indistinguishable from the original $\vec{M}$. By randomizing the message and signature within the same equivalence class, it is possible to generate anonymized signatures without relying on zero-knowledge proofs.

D. Set Commitment

Set commitment [7] is a scheme for committing to a set of elements. This scheme enables the verification of whether a given subset of elements is included the committed set. As the security requirements, the definitions of binding and hiding are defined as for standard commitment schemes, and subset soundness requires it to be infeasible to perform opening to subsets that are not contained in the committed set.

For compact representation, as in [7], we define the polynomial $f_S(x) := \prod_{s_i \in S}(x - s_i) = \sum_{i=0}^{N} f_i \cdot x^i$ for a set $S = \{s_1, ..., s_N\} \in \mathbb{Z}_p^N$, where f_i is the coefficient of x^i in this polynomial. For a group generator P, since $f_S(a)P = \sum_{i=0}^{N}(f_i \cdot a^i)P$, one can efficiently compute $f_S(a)P$ when given $(a^i P)_{i=0}^N$ but not a itself.

III. Model and Security of Revocable Group Signatures

In this paper, we adopt the LPY model [3] as well as to the previous R-GS scheme [4].

A. Model

A revocable group signature (R-GS) scheme consists of the following six polynomial-time algorithms.

- **Setup**: It takes as inputs security parameter $\kappa \in \mathbb{N}$, the number of group members $N \in \mathbb{N}$, and outputs the group public key gpk, the group manager's (GM's) private key gsk, the opener's secret key osk, and the public information $St = (St_{users}, St_{trans})$. St is initialized to $St_{users} = \emptyset, St_{trans} = \epsilon$ (empty string).
- **Join**: This is an interactive protocol between the GM and a joining user who will be a signer. Let κ and gpk be the inputs of the user i, and κ, gpk, St and gsk be the inputs of the GM. When **Join** is executed, the user receives a membership certificate cred and the user's private key usk. In addition, for the protocol history $transcript_i$, St is updated with $St_{users} := St_{users} \cup \{i\}, St_{trans} := St_{trans} || \langle i, transcript_i \rangle$.
- **Revoke**: It takes as inputs gpk, gsk, a time epoch $t \in \mathbb{N}$, and a set of revoked users $R_t \subset St_{users}$ at time t, and outputs a revocation list RL_t at time t.
- **Sign**: It takes as inputs gpk, t, RL_t, cred, usk, and a bit-string message M for a signer i. If $i \in R_t$, it outputs $\perp$, and otherwise it outputs a group signature σ.
- **Verify**: It takes as inputs σ, t, RL_t, M and gpk, and outputs 1 if σ is valid, 0 otherwise.
- **Open**: It takes as inputs M, t, RL_t, σ, osk, gpk and St. It tracks the signer i of σ and outputs $i \in St_{users}$, or $\perp$.

B. Security

As in [4], we define the three security requirements of an R-GS scheme, which are informally as follows.

- **Anonymity**: No adversary without osk can identify the signer from a signature, and can distinguish whether signers of two group signatures are the same or not.
- **Non-frameability**: No adversary who can corrupt the GM and the opener can produce a valid group signature which is traced to an honest user.
- **Misidentification resistance**: No adversary without gsk can produce a valid group signature which is traced to outside of the set of non-revoked and corrupted users.

IV. Previous Scheme

In the previous R-GS scheme [6], the size of the revocation list is reduced using vector commitments while keeping the efficient revocation using the CS (Complete Subtree), as follows.

First, GM generates a binary tree with height h and number of leaves 2^h. Then, it assigns IDs to the nodes in breadth-first order with the root as 0, and assigns each user to a leaf. In the case of Fig. 1, nodes 7 to 14 correspond to users. When adding a new user, GM issues certificates $(A_{u_0}, \cdots A_{u_\ell})$ to the user for node ID $(u_0, u_1, \cdots, u_\ell)$ on the path from the root to the leaf assigned to the user. In Fig. 1, the certificates (A_0, A_1, A_4, A_9) are issued to the user at node 9.

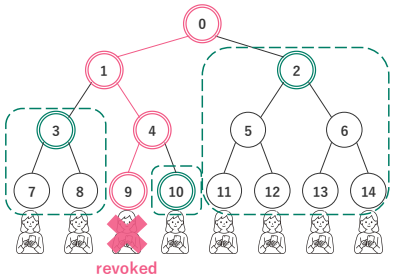

Fig. 1. Generation of revocation list

When a revocation occurs, GM searches a set of *cover nodes* using the CS method. Each cover node is a root of a subtree whose subtrees consist only of leaf nodes of non-revoked users. Then, GM sets a split parameter K and splits the set of cover nodes by K. For each partitioned set, compress it by a vector commitment and compute the commitment C. For each C, generate an SPS signature (AHO signature) B_C of each C, and set them to a revocation list. In the case of Fig. 1, when user 9 is revoked, the cover node IDs are $\{2, 3, 10\}$.

When signing, the signer proves that a node u_i in the path from the root to the signer's leaf is a cover node in the revocation list, which means non-revocation, as follows. Select a vector commitment C that contains the node u_i. Then, prove that u_i in the certificate A_{u_i} is included in C, by SPKs of the certificates A_{u_i} and B_C and the vector commitment.

In the previous scheme [6], the revocation list is compressed with vector commitments. The size of the revocation list is reduced from $\mathcal{O}(r \log \frac{N}{r})$ in [3], [4] to $\mathcal{O}(\frac{r \log \frac{N}{r}}{K})$, where r is the number of revoked users and N is the number of all users. On the other hand, the vector commitment and signature verifications are added, and thus the signature length is increased for their zero-knowledge proofs.

V. Proposed R-GS Scheme

A. Construction Idea

The previous R-GS scheme [6] with compact RL (revocation list) is based on Ohara et al.'s R-GS scheme [4]. In the scheme with the compact RL, using vector commitments, the cover nodes in the current RL are accumulated. However, the verification of opening the vector commitment needs a public parameter g_i of indexing the i-th node in the commitment. To conceal the index i for anonymity of the signer, the verification of an SPS signature on g_i is proved in zero-knowledge, which increases the signature length (and signing/verification costs). We adopt a set commitment instead of the vector commitment to reduce the overhead, where such an index is not needed and the verification is simpler. In the previous R-GS scheme, an AHO signature is used as the SPS signature and the zero-knowledge proof (SPK) of the verification of AHO signature is needed in the group signature. In this paper, we adopt an SPS-EQ signature as the SPS signature in the combination with the set commitment. Since the SPS-EQ signature and set commitment can be randomized and the zero-knowledge proofs are not needed, we can reduce the signature size. To integrate the SPS signature to the R-GS scheme, our R-GS scheme is based on the anonymous credential (AC) system [7] constructed using the SPS-EQ signatures. The AC system is similar to group signature schemes, as mentioned in Introduction. In the AC system, an issuer issues an attribute certificate (SPS signature) certifying user's attributes (in the form of set commitment) to a user, and the user can anonymously select and prove a part of attributes to a verifier.

Difference from SPS-EQ-based AC. In the SPS-EQ-based AC system [7], the certificate consists of an SPS-EQ signature on message vector (C_i, rC_i, P), where C_i is a set commitment on the attributes, and r is a random. The random ρ on the set commitment is regarded as the user's secret key usk. On the other hand, in our R-GS scheme, the signed message vector is modified to (C_i, upk, P) with $\mathsf{upk} = \mathsf{usk}P$. For **Open** algorithm, as in Ohara et al. [4], the linear encryption is adopted, where the upk is encrypted to identify the joining user. In our scheme, the knowledge of the same usk is proved by SPKs so that it is ensured that the upk is certified in an SPS-EQ signature and encrypted by the linear encryption.

A revocation certificate to certify the cover nodes at the revocation epoch is also an SPS-EQ signature on a set commitment to the cover nodes. As in the previous R-GS scheme [6], all the cover nodes are partitioned into sets of nodes, and each set is committed and signed with the current time. The certificate is published in the RL, and thus it cannot be randomized by a user's secret. But, for the randomization of the SPS-EQ signature and the set commitment under the DDH assumption, the set commitment has to be randomized.

To resolve this discrepancy, we utilize a non-randomized set commitment with $\rho = 1$, and in the group signature, we adopt the conventional approach of committing and the zero-knowledge proof (SPK). On the other hand, only the SPS-EQ signature is randomized on the perfect adaption of signatures of SPS-EQ, as shown in [7]. Furthermore, for showing the non-revoked user, the signer has to prove that a node in the set commitment of the membership certificate is the same as that in the revocation certificate. In our scheme, a Pedersen commitment to the node is calculated, and the verifications of opening in the both set commitments are proved by SPKs on the Pedersen commitments. Although the SPKs are added, the simple SPS-EQ verification and randomization of signature can reduce the signature length, compared to the previous R-GS scheme using AHO signatures.

Since we accumulate the nodes on the path to the signer in the membership certificate (in the previous R-GS scheme, they are not accumulated), we achieve $O(1)$-size certificate, for $O(\log N)$-size certificate in the previous R-GS scheme.

B. Construction

The proposed scheme consists of the following algorithms:

- **Setup** $(1^\kappa, N)$

1) Generate bilinear group parameters $\mathsf{BG} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, P, \hat{P})$ with security parameter κ. Randomly choose $a \xleftarrow{R} \mathbb{Z}_p$ and compute the public parameters for set commitment $\mathsf{pp_{SC}} = ((a^i P, a^i \hat{P})_{i\in[N]})$ with the total number of users N. Choose a hash function $H : \mathbb{Z}_p \xrightarrow{R} \{0,1\}^*$.
2) Randomly choose $F_1, F_2, F_3 \xleftarrow{R} \mathbb{G}_1 \backslash \{1\}, \xi_1, \xi_2, \xi_3 \xleftarrow{R} \mathbb{Z}_p^*$. Compute $G_1 = \xi_1 F_1 + \xi_3 F_3, G_2 = \xi_2 F_2 + \xi_3 F_2$. Set the public key for the linear encryption $\mathsf{pk}_{Enc} = (F_1, F_2, F_3, G_1, G_2)$ and the secret key $\mathsf{sk}_{Enc} = (\xi_1, \xi_2, \xi_3)$.
3) For $k = 3$, randomly choose $(x_{\mathsf{cred},i})_{i\in[k]} \xleftarrow{R} \mathbb{Z}_p^{*k}$ as the secret key for the SPS-EQ signature of the membership certificate, and set $\mathsf{sk_{cred}} = (x_{\mathsf{cred},i})_{i\in[k]}$. Compute the public key $\mathsf{pk_{cred}} = (\hat{X}_{\mathsf{cred},i})_{i\in[k]} = (x_{\mathsf{cred},i}\hat{P})_{i\in[k]}$.
4) For $k = 2$, randomly choose $(x_{RL,i})_{i\in[k]} \xleftarrow{R} \mathbb{Z}_p^{*k}$ as the secret key for the SPS-EQ signature of the revocation list $\mathsf{sk}_{RL} = (x_{RL,i})_{i\in[k]}$. Compute the public key $\mathsf{pk}_{RL} = (\hat{X}_{RL,i})_{i\in[k]} = (x_{RL,i}\hat{P})_{i\in[k]}$.
5) Randomly choose $Q \in \mathbb{G}_1$ and $\hat{Q} \in \mathbb{G}_2$ for commitments.
6) Output the group public key $\mathsf{gpk} = (\mathsf{BG}, \mathsf{pp}_{SC}, \mathsf{pk}_{Enc}, \mathsf{pk_{cred}}, \mathsf{pk}_{RL}, Q, \hat{Q})$, the GM's secret key $\mathsf{gsk} = (a, \mathsf{sk_{cred}}, \mathsf{sk}_{RL})$, and the opener's secret key $\mathsf{osk} = (\mathsf{sk}_{Enc})$.

- **Join**

1) The user randomly chooses a secret key $\mathsf{usk} \xleftarrow{R} \mathbb{Z}_p^*$ and computes the public key $\mathsf{upk} \leftarrow \mathsf{usk} \cdot P$. The user computes a standard digital signature Sig_i for the message upk and sends it to GM.
2) The user sends an SPK to GM showing knowledge of α such that $\mathsf{upk} = \alpha P$. If the verification fails, the process terminates.
3) GM assigns the user to a leaf u_ℓ in the binary tree. Let $\langle v \rangle := (u_0, u_1, ..., u_\ell)$ be the nodes on the path from the root node to that leaf node.
4) Let the polynomial be $f_{\langle v \rangle}(x) := \prod_{u_i \in \langle v \rangle}(x - u_i) = \sum_{i=0}^{\ell} f_i \cdot x^i$ (f_i is the coefficient of the i-th term of the polynomial). The user randomly chooses $\rho \xleftarrow{R} \mathbb{Z}_p^*$ and computes the set commitment C_i as follows.
$$\begin{aligned} C_i &= \rho \cdot f_{\langle v \rangle}(a) \cdot P \\ &= \rho \cdot (f_0 a^0 P + f_1 a^1 P + ... + f_\ell a^\ell P) \in \mathbb{G}_1^* \end{aligned}$$
The user sends C_i and $R = \rho P$ to GM, along with an SPK proving knowledge of ρ such that $R = \rho P$
5) GM verifies the correctness of C_i by checking if $e(C_i, \hat{P}) = e(R, f_{\langle v \rangle} \cdot \hat{P})$. If this equation or the SPK verification fails, the process terminates. GM randomly chooses $y \xleftarrow{R} \mathbb{Z}_p^*$ and computes an SPS-EQ signature $\sigma_i = (Z, Y, \hat{Y})$ for the message (C_i, upk, P) using the secret key $\mathsf{sk_{cred}}$, where $Z = y(x_{\mathsf{cred},1} C_i + x_{\mathsf{cred},2}\mathsf{upk} + x_{\mathsf{cred},3} P)$, $Y = \frac{1}{y}P$, and $\hat{Y} = \frac{1}{y}\hat{P}$. GM sends σ_i to the user.
6) The user checks if σ_i satisfies the SPS-EQ signature verification equations: $e(C_i, \hat{X}_{\mathsf{cred},1}) \cdot e(\mathsf{upk}, \hat{X}_{\mathsf{cred},2}) \cdot e(P, \hat{X}_{\mathsf{cred},3}) = e(Z, \hat{Y})$ and $e(Y, \hat{P}) = e(P, \hat{Y})$. If they hold, the user obtains the membership certificate $\mathsf{cred} \leftarrow (C_i, \sigma_i, \rho, \mathsf{usk})$.
7) Finally, GM adds i and $transcript_i = (\mathsf{upk}, \sigma_i, Sig_i)$ to St_{trans}.

- **Revoke** $(t, R_t, \mathsf{gsk}, \mathsf{gpk})$

1) For the set of revoked users R_t at the current time t, apply the CS method to obtain the set of cover nodes $(u'_1, u'_2, ..., u'_{num})$. Here $num \leq |R_t| \log \frac{N}{|R_t|}$, for N is the total number of users.
2) Let the vector be $\vec{u} = (u'_1, u'_2, ..., u'_{num})$. Define a split parameter K and divide the elements of $\vec{u}$ every K elements, as follows, where $\Omega = \lceil num/K \rceil$:
$$\begin{aligned} \vec{u}_1 &= (u'_1, u'_2, ..., u'_K) \\ \vec{u}_2 &= (u'_{K+1}, u'_{K+2}, ..., u'_{2K}) \\ &\vdots \\ \vec{u}_\Omega &= (u'_{(\Omega-1)K+1}, u'_{(\Omega-1)K+2}, ..., u'_{num}) \end{aligned}$$
3) GM computes the set commitment for the elements of each $\vec{u_k}$ for $1 \leq k \leq \Omega$ with $\rho = 1$ as:
$$C_{\vec{u_k}} = f_{\vec{u_k}}(a)P = \sum_{u'_j \in \vec{u_k}} (x - u'_j)P \in \mathbb{G}_1^*$$
4) For all $1 \leq k \leq \Omega$, GM randomly chooses $y \xleftarrow{R} \mathbb{Z}_p^*$ and uses the secret key sk_{RL} to sign $C_{\vec{u_k}}$ from step 3 along with the time $T = tP$. This is, compute $Z = y(x_{RL,1} C_{\vec{u_k}} + x_{RL,2} T), Y = \frac{1}{y}P, \hat{Y} = \frac{1}{y}\hat{P}$ to generate the SPS-EQ signature $\sigma_{\vec{u_k}} \leftarrow (Z, Y, \hat{Y})$ for the message $(C_{\vec{u_k}}, T)$ using the secret key sk_{RL}.

Outputs the following revocation list RL_t:

$$RL_t = (t, \vec{u}, \{\sigma_{\vec{u_k}}, C_{\vec{u_k}}\}_{k=1}^{\Omega})$$

- **Sign**$(\mathsf{gpk}, t, RL_t, \mathsf{usk}, \mathsf{cred}, M)$
The following SPKs are generated as an SPK for a conjunction of statements, ensuring that the same secret knowledge maintains its identity across them.

1) For the anonymous verification, randomize the SPS-EQ signature of the membership certificate cred. Randomly choose $\mu_{\mathsf{cred}}, \psi_{\mathsf{cred}} \xleftarrow{R} \mathbb{Z}_p^*$, and compute $\sigma_i' = ((Z_i', Y_i', \hat{Y}_i') = (\psi_{\mathsf{cred}}\mu_{\mathsf{cred}} Z, \frac{1}{\psi_{\mathsf{cred}}} Y, \frac{1}{\psi_{\mathsf{cred}}}\hat{Y}))$. The message of σ_i' is also randomized by μ_{cred} as $(C'_{i,1}, C'_{i,2}, C'_{i,3}) = \mu_{\mathsf{cred}}(C_i, \mathsf{upk}, P)$, and let $\mathsf{cred}' = ((C'_{i,1}, C'_{i,2}, C'_{i,3}), \sigma_i')$.
2) To prove that the leaf node of the signer is included in the subtree rooted at node u_i within C_i, the following W_i is computed and randomized to W_i' using μ_{cred}:

$$W_i = \rho \cdot f_{\langle v \rangle \setminus u_i}(a) P, \quad W_i' = \mu_{\mathsf{cred}} \cdot W_i$$

Furthermore, to hide u_i, compute a commitment to u_i as follows. Randomly choose $\nu \xleftarrow{R} \mathbb{Z}_p^*$, and compute the commitment $C_{u_i} = u_i \hat{P} + \nu \hat{Q}$. Compute an SPK showing knowledge of u_i and ν, and let this SPK be Π^{u_i}. Using the commitment C_{u_i}, $f_{u_i}(a)\hat{P}$ can be as:

$$f_{u_i}(a)\hat{P} = (a - u_i)\hat{P} = a\hat{P} - (C_{u_i} - \nu\hat{Q})$$

Here, let $\psi_i = a\hat{P} - (C_{u_i} - \nu\hat{Q})$. The verification of $u_i \in \langle v \rangle$ in the set commitment $C'_{i,1}$ is $e(W_i', \psi_i) = e(C'_{i,1}, \hat{P})$. This equation can be transformed into $e(W_i', a\hat{P} - C_{u_i}) e(C'_{i,1}, \hat{P})^{-1} = e(W_i', Q)^{-\nu}$. The SPK showing the knowledge of ν s.t. this verification equation is computed, and let this SPK be $\Pi^{W_i'}$.
3) Compute an SPK showing knowledge of $(\mathsf{usk}, \mathsf{cred})$ such that $C'_{i,2} = \mathsf{usk} \cdot C'_{i,3}, C'_{i,3} = \mu_{\mathsf{cred}} \cdot P$. Let this SPK be $\Pi^{\mathsf{cred}'}$.
4) Prove that $C_{\vec{u_k}}$ contains the cover node u_j' for the signer's leaf node. Using the RL_t, find the set commitment $C_{\vec{u_k}}$ that contains the cover node u_j'. Randomly choose $\rho_{C_{\vec{u_k}}} \xleftarrow{R} \mathbb{Z}_p^*$ and compute the commitment of $C_{C_{\vec{u_k}}} = C_{\vec{u_k}} + \rho_{C_{\vec{u_k}}} Q$. Randomize the SPS-EQ signature $\sigma_{\vec{u_k}}$. Choose $\psi_{\vec{u_k}} \xleftarrow{R} \mathbb{Z}_p^*$ randomly, and using $\sigma_{\vec{u_k}}$, compute $\sigma'_{\vec{u_k}} = ((Z'_{\vec{u_k}}, Y'_{\vec{u_k}}, \hat{Y}'_{\vec{u_k}}) = (\psi_{\vec{u_k}} Z, \frac{1}{\psi_{\vec{u_k}}} Y, \frac{1}{\psi_{\vec{u_k}}}\hat{Y}))$. The first verification equation of the SPS-EQ signature is $e(C_{C_{\vec{u_k}}} - \rho_{C_{\vec{u_k}}} Q, \hat{X}_{RL,1}) \cdot e(tP, \hat{X}_{RL,2}) = e(Z'_{\vec{u_k}}, \hat{Y}'_{\vec{u_k}})$. Transforming this equation, we get $e(C_{C_{\vec{u_k}}}, \hat{X}_{RL,1}) \cdot e(tP, \hat{X}_{RL,2}) \cdot e(Z'_{\vec{u_k}}, \hat{Y}'_{\vec{u_k}})^{-1} = e(Q, \hat{X}_{RL,1})^{\rho_{C_{\vec{u_k}}}}$. Compute an SPK $\Pi^{\sigma_{\vec{u_k}}}$ showing knowledge of $\rho_{C_{\vec{u_k}}}$.
5) To prove that u_j' is contained in $C_{\vec{u_k}}$, compute the following witness W_j, randomly choose $p_{W_j} \xleftarrow{R} \mathbb{Z}_p$, and compute the commitment to W_j as $C_{W_j} = W_j + \rho_{W_j} Q$.

$$W_j = f_{\vec{u_k} \setminus u_j'}(a) P$$

To show $u_j' = u_i$, we reuse $C_{u_i} = u_i \hat{P} + \nu \hat{Q}$. Then when $u_j' = u_i$, we obtain

$$f_{u_j'}(a)\hat{P} = (a - u_j')\hat{P} = a\hat{P} - (C_{u_i} - \nu\hat{Q})$$

Thus, from $e(W_j, f'_{u_j}(a)\hat{P}) = e(C_{\vec{u_k}}, \hat{P})$, we obtain $e(C_{W_j} - \rho_{W_j} Q, a\hat{P} - C_{u_i} + \nu\hat{Q}) = e(C_{C_{\vec{u_k}}} - \rho_{C_{\vec{u_k}}} Q, \hat{P})$. Generate an SPK Π^{W_j} showing knowledge of $(\nu, \rho_{W_j}, \rho_{W_j} \cdot \nu, \rho_{C_{\vec{u_k}}})$. Also, to prove the correctness of $\rho_{W_j} \cdot \nu$, let $\xi = \rho_{W_j} \cdot \nu$, and randomly choose $\rho_\xi, \rho_\nu \xleftarrow{R} \mathbb{Z}_p^*$. Compute the commitment $C_\xi = \xi P + \rho_\xi Q$, and $C_\nu = \nu P + \rho_\nu Q$. Generate an SPK $\Pi^{\rho_{W_j} \cdot \nu}$ showing $(\xi, \rho_\xi, \nu, \rho_\nu, \rho_{W_j}, \rho')$ that satisfies:

$$C_\xi = \xi P + \rho_\xi Q, \quad C_\nu = \nu P + \rho_\nu Q, \quad C_\xi = \rho_{W_j} C_\nu + \rho' Q$$

Here, let $\rho' = \rho_\xi - \rho_{W_j}\rho_\nu$. The equation implies $C_\xi = \rho_{W_j} \cdot \nu \cdot P + (\rho_{W_j} \cdot \rho_\nu + \rho') \cdot Q$, i.e, $\xi = \rho_{W_j} \cdot \nu$.
6) Randomly choose $\gamma, \zeta \xleftarrow{R} \mathbb{Z}_p^*$. Compute the ciphertext of $\mathsf{upk} = \mathsf{usk} \cdot P$, as $\psi_1 = \gamma F_1, \psi_2 = \zeta F_2, \psi_3 = (\gamma + \zeta) F_3, \psi_4 = \gamma G_1 + \zeta G_2 + \mathsf{usk} \cdot P$. Generate an SPK Π^{Enc} showing the knowledge of $(\gamma, \zeta, \mathsf{usk})$.
7) Output the signature $\sigma = (\mathsf{cred}', W_i', \sigma'_{\vec{u_k}}, C_{C_{\vec{u_k}}}, C_{W_j}, C_{u_i}, C_\xi, C_\nu, (\psi_1, \psi_2, \psi_3, \psi_4), \Pi^{W_i'}, \Pi^{u_i}, \Pi^{\mathsf{cred}'}, \Pi^{\sigma_{\vec{u_k}}}, \Pi^{W_j}, \Pi^{\rho_{W_j} \cdot \nu}, \Pi^{Enc})$.

- **Verify**$(\sigma, M, \mathsf{gpk})$ The verifier checks the two verification equations of the SPS-EQ signature σ_i' and the second verification equation of $\sigma'_{\vec{u_k}}$, and each SPK included in the signature σ. If all are valid, it outputs valid; otherwise, it outputs invalid.
- **Open**$(\mathsf{osk}, \sigma, M)$ Calculate $\mathsf{upk}' = (\psi_4 - (\xi_1\psi_1 + \xi_2\psi_2 + \xi_3\psi_3))$. If there exists an entry $\langle i, transcript_i \rangle = (\mathsf{upk}, \sigma_i, Sig_i)$ in St_{trans} such that $\mathsf{upk}' = \mathsf{upk}$, then output i; otherwise, it outputs $\perp$.

VI. Experiment Results

To demonstrate the effectiveness of the proposed scheme in terms of the authentication time, we implemented the proposed scheme on a PC (WSL2 (Ubuntu 22.04.5LTS)) with an Intel Core i7-13700 CPU (2.1GHz) and 16GB of memory by C language with the GMP library and the pairing library ELiPS [8] (BLS curve-461bit), and measured the computation times for the signing, verification and witnesses W_i, W_j. We also compared the signature length and revocation list size of the proposed scheme with those of the previous schemes [4] and [6]. For 128-bit security, each $\mathbb{G}_1$, $\mathbb{G}_2$ element and $\mathbb{Z}_p$ element is represented with 512 bits in the library [8]. Also, the height of the CS method tree is fixed at 20.

A. Signature Length

Table II shows the comparison of signature lengths among the previous R-GS schemes [4], [6] and the proposed scheme. The $i|\mathbb{G}|+j|\mathbb{Z}_p|$ means i elements of $\mathbb{G}_1$ or $\mathbb{G}_2$ and j elements of $\mathbb{Z}_p$. Since the proposed scheme adopts SPS-EQ and the set commitment, the number of SPK elements in **Sign** is smaller than that in the previous scheme [6] with compact revocation

TABLE I
REVOCATION LIST SIZE COMPARISON

	[4]	$K = 100$		$K = 500$		$K = 1,000$	
		[6]	Ours	[6]	Ours	[6]	Ours
$N = 10,000$ $R = 1,000$	192KB	5.1KB	2.6KB	1.0KB	0.5KB	0.5KB	0.3KB
$N = 100,000$ $R = 10,000$	1,920KB	51.2KB	25.6KB	10.2KB	5.1KB	5.1KB	2.6KB
$N = 1,000,000$ $R = 100,000$	19,200KB	512KB	256KB	102KB	51.2KB	51.2KB	25.6KB

lists, and thus the signature length is also smaller. On the other hand, the signature length is larger than that of the previous scheme [4] without compact revocation lists.

TABLE II
SIGNATURE LENGTH COMPARISON

	# of elements	Signature Length
[4]	$5\|\mathbb{G}\| + 13\|\mathbb{Z}_p\|$	1,152 Byte
[6]	$30\|\mathbb{G}\| + 23\|\mathbb{Z}_p\|$	3,392 Byte
Ours	$19\|\mathbb{G}\| + 13\|\mathbb{Z}_p\|$	2,049 Byte

B. Processing Time

Fig. 2 shows the computation times of **Sign**, **Verify**, **Sign+Verify**, and the witnesses W_i, W_j in **Sign** of the proposed scheme, for 10,000 total users and 1,000 revoked users, when the split parameter K for cover nodes is varied. While the processing time of **Verify** is constant, the processing time of **Sign** increases as K increases. This is because the computation of witness W_j of **Sign** depends on K. Although the processing time of **Sign** increases, it remains around 200ms even when $K = 1,000$, demonstrating that the processing time is sufficiently practical.

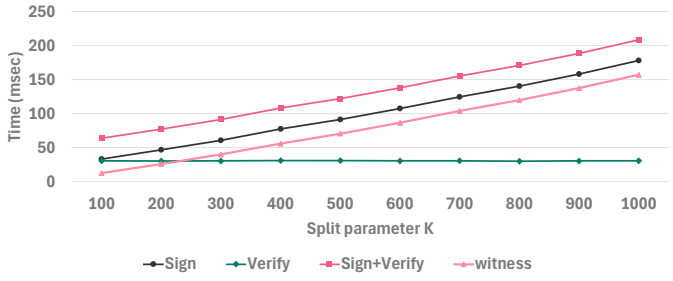

Fig. 2. Processing times of the proposed scheme with K

C. Revocation List Size

Table I compares the sizes of the revocation list (RL) in [4], [6] and our scheme. The split parameter is set to $K = 100, 500, 1000$, while the total number of users N and the number of revoked users R are varied such that the revocation ratio is fixed at $R/N = 0.1$. In the proposed scheme, the RL size is reduced from [4], and the reduction is larger as K is increased, similarly to [6]. When $K = 1,000$, the RL size is about 25KB for $N = 1,000,000$, which is half the size of the RL in [6].

VII. CONCLUSION

In this paper, we have proposed an R-GS scheme using SPS-EQ signatures and set commitments, where the size of the RL and the signature length are reduced. From the previous R-GS scheme with compact RL, the signature size is reduced to 58%. Our implementation results show that even when the revocation list was compressed to 1/1000, **Sign** and **Verify** times were around 200ms, indicating sufficient practicality.

Future work directions include extending our revocation mechanism to revocable anonymous credential systems handling multiple attributes, and supporting more expressive capabilities such as attribute-based revocation and proving relationships between attributes. Concurrently, maintaining and enhancing the efficiency achieved in this work (i.e., short signature length and compact revocation list size), particularly through optimization of the SPK components, is crucial for practicality. Exploring these directions aims to realize practical, privacy-preserving next-generation credential systems.

REFERENCES

[1] D. Chaum and E. van Heyst, “Group signatures,” in *Advances in Cryptology—EUROCRYPT ’91*, vol. 547, Lecture Notes in Computer Science, pp. 257–265, Springer-Verlag, 1991.
[2] J. Camenisch and A. Lysyanskaya, “Dynamic accumulators and application to efficient revocation of anonymous credentials,” in *Advances in Cryptology—CRYPTO 2002*, vol. 2442, Lecture Notes in Computer Science, pp. 61–76, Springer-Verlag, 2002.
[3] B. Libert, T. Peters, and M. Yung, “Scalable group signatures with revocation,” in *Advances in Cryptology—EUROCRYPT 2012*, vol.7237, Lecture Notes in Computer Science, pp.609–627, Springer-Verlag, 2012.
[4] K. Ohara, K. Emura, G. Hanaoka, A. Ishida, K. Ohta, and Y. Sakai, “Shortening the Libert-Peters-Yung revocable group signature scheme by using the random oracle methodology,” *IEICE Trans. Fundamentals*, vol.E102-A, no.9, pp.1101–1117, 2019.
[5] S. Sadiah and T. Nakanishi, “Revocable group signatures with compact revocation list using vector commitments,” *IEICE Trans. Fundamentals*, vol.E100-A, no.8, pp.1672–1682, 2017.
[6] K. Sugimoto and T. Nakanishi, “Reducing revocation lists in CS-based revocable group signature scheme using vector commitment,” in *CANDAR 2021*, pp.175–181, 2021.
[7] G. Fuchsbauer, C. Hanser, and D. Slamanig, “Structure-preserving signatures on equivalence classes and their application to anonymous credentials,” *Journal of Cryptology*, vol.32, pp.498–546, 2019.
[8] Y. Takahashi, Y. Nanjo, T. Kusaka, Y. Nogami, T. Kanenari, and T. Tatara, “An implementation and evaluation of pairing library ELiPS for BLS curve with several techniques,” in *ITC-CSCC 2019*, 2019.