Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

STAR-AC: Traceable Anonymous Credentials With Selective Disclosure and Attribute-Level Revocation for the Marine Internet of Things

Biao Hu
Hunan University of Science and Technology
Liang Bai
Hunan University of Science and Technology
Jin Wang
Hunan University of Science and Technology
Yuzhen Liu
Hunan University of Science and Technology
Xiaoliang Wang
Hunan University of Science and Technology
Kuanching Li
aliric@hnust.edu.cn

Hunan University of Science and Technology

Research Article

Keywords:

Posted Date: November 24th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11611675/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

STAR-AC: Traceable Anonymous Credentials With Selective Disclosure and Attribute-Level Revocation for the Marine Internet of Things

Biao Hu[1,2], Liang Bai[1], Jin Wang[1], Yuzhen Liu[1], Xiaoliang Wang[1,2*], Kuanching Li[1*]

[1]School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan, 411201, China.
[2]Hunan Provincial University Key Laboratory of Knowledge Processing and Networked Manufacturing, Xiangtan, 411201, China.

*Corresponding author(s). E-mail(s): fengwxl@hnust.edu.cn; aliric@hnust.edu.cn;
Contributing authors: biaohu@mail.hnust.edu.cn; liangbai77@hnust.edu.cn; jinwang@hnust.edu.cn; liuyuzhen@hnust.edu.cn;

Abstract

The Marine Internet of Things (MIoT) requires secure and efficient attribute authentication with identity privacy and fine-grained revocation. However, existing credential schemes often often incur high authentication overhead and have limitations in fine-grained disclosure, revocation, and traceability. To address these challenges, this paper proposes STAR-AC: traceable anonymous credentials with selective disclosure and attribute-level revocation for MIoT. STAR-AC integrates threshold credential issuance, efficient selective disclosure, attribute-level revocation, conditional traceability, and ledger-based state management. It enables marine vehicles to disclose only required attributes while preserving privacy. Efficient credential showing and verification algorithms reduce computational and communication overhead, with costs mainly depending on the number of disclosed attributes. Attribute-level revocation preserves other valid attributes, while offline witness and incremental state updates reduce authentication and revocation overhead. Moreover, one-time pseudonyms preserve vehicle identity privacy while enabling efficient identity tracing. Formal security analysis proves attribute authenticity, anonymity, unlinkability, revocation security,

state integrity, and traceability. Experimental results demonstrate that STAR-AC reduces credential showing communication and computational overhead compared with an existing state-of-the-art scheme. The results also show efficient identity tracing, low-cost incremental revocation-state maintenance, and high-throughput ledger-state retrieval and updates under concurrent requests.

Keywords: Marine Internet of Things, anonymous attribute credentials, selective disclosure, attribute-level revocation, traceability.

1 Introduction

The Marine Internet of Things (MIoT) is becoming a critical infrastructure for ocean observation, maritime transportation, offshore energy, environmental monitoring, and marine resource management[1, 2]. In a typical MIoT environment, heterogeneous entities such as unmanned surface vehicles, autonomous underwater vehicles, buoys, ships, coastal stations, satellites, and maritime service platforms cooperate to collect, forward, and process sensing data[3, 4]. These entities often operate across different administrative domains and communicate through unstable links. Compared with terrestrial IoT systems, MIoT networks will face more challenging deployment conditions, including intermittent connectivity, dynamic topology, and weak physical protection[5, 6]. These characteristics make secure and efficient authentication a fundamental requirement for reliable marine services.

Authentication in MIoT should not only verify whether a marine vehicle is legitimate, but also determine whether it satisfies specific attributes or service requirements[7]. For example, a verifier may need to check whether a marine vehicle has a valid mission role, sensing capability, navigation permission, regional authorization, or data access privilege. Directly revealing all identity and all attributes of the vehicle, however, may expose sensitive information such as mission type, operational region, organizational affiliation, or device capability[8, 9]. Therefore, anonymous attribute-based authentication with selective disclosure is necessary. That is to say a marine vehicle should disclose only the required attributes while hiding its real identity and unrelated attributes.

Existing privacy-preserving authentication studies can be broadly classified into two categories. The first focuses on anonymous credentials and selective disclosure, enabling users to prove attribute validity while hiding their identities and undisclosed attributes[10–12]. Liu *et al.* [10] proposed a threshold-based vehicle-to-infrastructure authentication scheme supporting attribute-based disclosure in vehicular networks. However, as the number of all attributes increases, its communication and verification costs also grow rapidly. Sun *et al.* [12] introduced a fine-grained delegation and revocation of authorization attributes, allowing users to securely delegate partial attributes and revoke them when necessary. However, its delegation and revocation rely on additional verification structures, causing computation and communication costs to increase with the number of delegated attributes. Wu *et al.* [13] introduced a

consistency-proof mechanism to support continuous authentication in zero-trust networks while enabling selective disclosure with minimized attribute exposure. However, it provides limited support for complex attribute structures during authentication.

In a word, they often require proofs for hidden attributes, increasing showing costs with the total number of attributes even when few are disclosed. In addition to this efficiency limitation, weak credential binding may allow attributes from different credentials to be combined, resulting in unauthorized attribute showing.

The second category focuses on revocation, and blockchain-based state management[14–16]. Blockchain-assisted authentication, dynamic pseudonyms, and group signatures improve state transparency, traceability, and cross-domain trust[17, 18]. Wijethilaka *et al.* [19] proposed a blockchain-based decentralized authentication and authorization scheme for 5G network slicing. However, in large-scale and high-frequency authentication scenarios, the blockchain interaction and on-chain storage overhead of this scheme become difficult to tolerate. Jiang et al. [20] proposed a device-assisted threshold password-based single sign-on scheme for cross-domain authentication, enabling single login access across multi-domains. Wang *et al.* [21] employed group signatures and dynamic pseudonyms to achieve anonymous authentication and message verifiability in vehicle-to-roadside unit scenarios. However, it lacks fine-grained attribute-based control and management. In short, these schemes support only user-based or credential-based revocation, which is insufficient for MIoT scenarios. Thus, MIoT authentication should support selective disclosure, attribute-level revocation, enforceable traceability, and efficient ledger-state maintenance.

To address these challenges, this paper proposes STAR-AC: traceable anonymous credentials with selective disclosure and attribute-level revocation for the MIoT. STAR-AC integrates threshold credential issuance, efficient selective disclosure, attribute-level revocation, identity traceability, and ledger-based state management. It enables marine vehicles to disclose only required attributes while preserving identity privacy and preventing unauthorized attribute disclosure. By combining randomized signatures with a showing proof, STAR-AC achieves efficient credential showing and verification, with costs mainly depending on the number of disclosed attributes. Attribute level revocation and incremental state updates reduce revocation and management overhead without affecting other valid attributes. Moreover, one-time pseudonyms preserve anonymity against external adversaries while supporting efficient identity tracing. These features enable secure and efficient attribute authentication and revocation management in large-scale MIoT environments. The main contributions of this paper are as follows:

- **A traceable anonymous credentials with selective disclosure and attribute-level revocation for MIoT.** This paper proposes STAR-AC, a unified framework that integrates threshold credential issuance, selective disclosure, attribute-level revocation, conditional traceability, and ledger-based state management. STAR-AC enables marine vehicles to selectively disclose attributes while preserving identity privacy and supporting attribute-level revocation.
- **An efficient anonymous attribute credential protocol.** STAR-AC supports selective disclosure of attributes while hiding vehicle identities and unrelated

attributes. It binds the disclosed attributes to the same credential and prevents unauthorized attribute showing. By combining randomized signatures with a showing proof, STAR-AC develops efficient credential showing and verification algorithms that reduce communication and computational overhead, with costs mainly depending on the number of disclosed attributes.

- **An integrated credential management mechanism combining revocation, tracing, and state management.** STAR-AC supports attribute level revocation without affecting other valid attributes. It reduces authentication overhead through offline witness maintenance and improves revocation-state management through incremental updates. It also enables identity tracing while preserving anonymity against external adversaries.
- **Formal security analysis and experimental evaluation.** Formal security proofs establish attribute authenticity, showing anonymity and unlinkability, attribute-level revocation security, revocation-state integrity, and authorized traceability. Experimental results demonstrate lower communication and computational overhead for credential showing and verification compared with an existing state-of-the-art scheme. The results also show efficient identity tracing, incremental revocation-state maintenance, and concurrent ledger-state processing.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 presents the system model, formal definition, and security model and assumptions. Section 4 gives the high-level workflow and details STAR-AC. Section 5 provides a security proof and analysis. Section 6 reports the performance analysis and evaluation. Finally, concluding remarks and future directions are presented in Section 7.

2 Related Work

2.1 Anonymous Credentials With Selective Disclosure

Anonymous credentials allow uses to prove possession of certified attributes without revealing unnecessary identity information. Existing works mainly focus on credential aggregation[10, 22], randomized showing[23], selective disclosure[24], and lightweight verification. Zhao *et al.* [23] proposed a randomized aggregate-signature-based anonymous credential scheme that improves the efficiency of credential generation and verification. Yang *et al.* [25] further considered decentralized identity and self-sovereign credential aggregation, enabling users to aggregate credentials issued by multiple issuers and present authentication tokens. Chen *et al.* [26] applied designated-verifier anonymous credentials and supports selective disclosure and traceability. However, it does not further support efficient attribute-level revocation or on-chain maintenance of large-scale revocation states. Yang *et al.* [24] studied privacy-preserving cross-organizational identification and uses selective disclosure to reduce unnecessary exposure of user information. Zhang *et al.* [27] proposed a satellite-earth staged authentication protocol that combines anonymous credentials with a dual-chain architecture. However, its communication overhead increases rapidly with the total number of attributes.

In addition, several studies investigate threshold attribute-based credentials and redactable signatures[12, 13, 28]. Shi *et al.* [29] proposed a threshold attribute-based credential scheme with redactable signatures that allows a user to obtain attribute-based credentials from threshold issuers and selectively disclose part of the certified information. Li *et al.* [30] considered dynamic threshold anonymous credentials and batch showing, aiming to reduce authentication cost when multiple credentials need to be shown together. However, this scheme lacks support for attribute-level revocation and traceability. Zhang *et al.* [31] studied multi-attribute anonymous credentials, selective disclosure, identity tracing, and attribute revocation, and is one of the closest works to the proposed construction. However, their scheme incurs high overhead when the revocation scale is large.

Although existing schemes support privacy-preserving attribute authentication, they often incur showing overhead related to the total number of attributes, provide only coarse-grained revocation, or require costly tracing procedures. In contrast, STAR-AC uses attribute-based credentials and a hidden context, so only disclosed attributes appear in the showing token while cross-credential attribute mixing is prevented. Thus, online communication only depends on disclosed attributes rather than the total certified attributes.

2.2 Revocation and Blockchain-Based State Management

Revocation mechanisms are critical for practical anonymous authentication. Existing works have explored blockchain-assisted identity management[19, 32], dynamic pseudonyms, group signatures[21], verifiable credentials[33, 34], trusted execution environments, and conditional privacy mechanisms. Wijesundara *et al.* [33] used decentralized identifiers, verifiable credentials, IPFS, and smart contracts to authenticate firmware in smart-home IoT systems. Kwon *et al.* [34] combined blockchain, verifiable credentials, and elliptic-curve cryptography to support mutual authentication. Shi *et al.* [32] used blockchain and lightweight cryptographic mechanisms to authenticate medical IoT devices. But, it does not support anonymous attribute credentials and selective disclosure.

Zhang *et al.* [35] applied decentralized identity, verifiable credentials, threshold signatures, and proxy signatures to cross-domain authentication. Liu *et al.* [36] relied on trusted execution environments to support lightweight authentication, pseudonym management, tracing, and revocation. However, the non-revocation proof in this scheme incurs high communication overhead. Xie *et al.* [37] used blockchain to manage UAV credentials and authentication state in Internet of Vehicles scenarios. However, this scheme lacks support for a low-overhead traceable showing mechanism. These works demonstrate the utility of blockchain and pseudonym mechanisms for authentication, but lack unified support for selective disclosure and attribute-based revocation in anonymous attribute-based credentials.

Compared with existing studies, STAR-AC unifies anonymous attribute-based credentials, fine-grained revocation, traceability, and ledger-based state maintenance. Each attribute is assigned a hidden revocation tag, enabling attribute-based revocation while preserving selective disclosure. Non-revocation witnesses are maintained offline,

and only lightweight proofs are verified during showing. Pseudonym indices support efficient tracing, while ledger states and incremental updates reduce revocation costs.

Overall, prior works usually focus either on anonymous credential showing or on blockchain-assisted authentication and revocation management. These schemes cannot achieve low communication overhead while simultaneously supporting attribute-based revocation, state-based non-revocation proofs, traceability, and incremental ledger updates. STAR-AC addresses this gap through attribute-based credentials, a hidden credential, pseudonym indices, and accumulator-based revocation management.

3 Problem Formulation

3.1 System Model

As illustrated in Fig. 1, STAR-AC consists of five entities: the maritime authority (MA), marine vehicles (MVs), issuers, verifiers, and a public ledger. Their roles are described as follows.

3.1.1 MA

The MA is a trusted management entity. It initializes the system, generates and distributes issuer key shares, and registers MVs. It verifies credential issuance requests, maintains registration and tracing records. It also manages attribute revocation, signs authorized revocation states, and supports identity tracing.

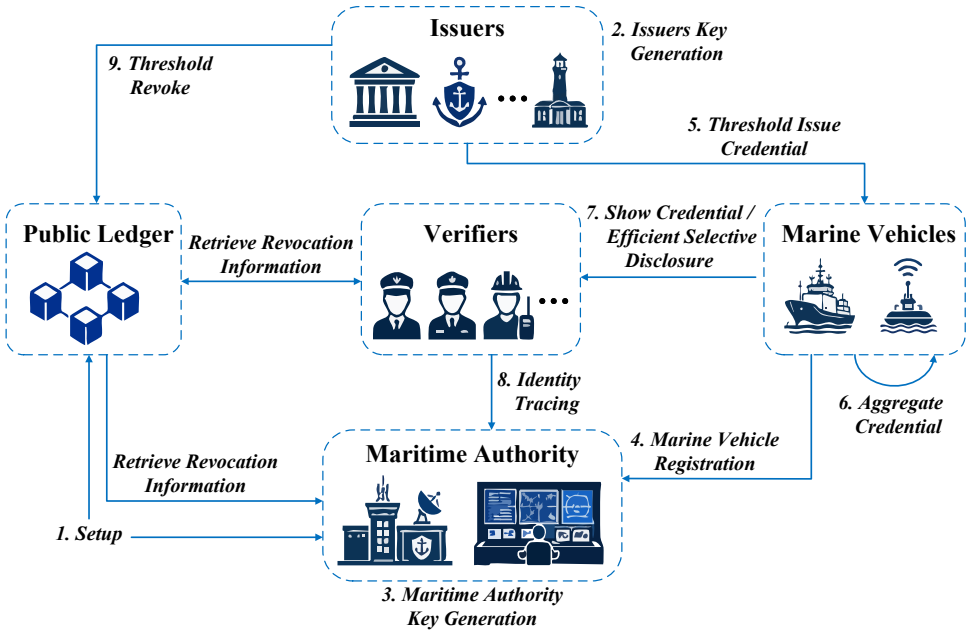


Fig. 1 System Model.

3.1.2 MV

MVs register with the MA and obtain attribute credentials from threshold issuers. They update non-revocation witnesses for valid attributes according to the latest revocation state. During authentication, each MV uses a one-time pseudonym to selectively disclose required attributes and prove credential validity and attribute non-revocation without revealing its real identity.

3.1.3 Issuer

The system contains n_I issuers with a threshold t_I. Each issuer holds secret shares securely distributed by the MA. At least t_I issuers are required to issue valid attribute credentials and authorize attribute revocation. Issuers follow the protocol and sign only MA-approved requests. Fewer than t_I compromised shares do not reveal the threshold signing keys under the assumed security model.

3.1.4 Verifier

Verifiers authenticate MVs based on their disclosed attributes. They verify the showing proofs and current revocation state without obtaining the MVs' real identities. Each verifier records used pseudonym indices to prevent repeated credential presentations.

3.1.5 Public Ledger

The public ledger stores revocation states and update records. It verifies threshold revocation signatures, accumulator updates, and MA signatures before accepting new states. MVs and verifiers retrieve the latest revocation state from the ledger.

3.2 Formal Definition

STAR-AC consists of the following algorithms.

- **Setup** $(1^\lambda, n_I, t_I, q, L) \to pp$. The MA generates the bilinear groups, hash functions, accumulator structured reference string (SRS), pseudonym index domain, and public parameters.
- **MAKeyGen** $(pp) \to (ask, apk)$. The MA generates its signing key pair and publishes the initial revocation state.
- **IssuerKeyGen** $(pp) \to (pk_{\text{ext}}, rpk, \{isk_i, rsk_i\}_{i=1}^{n_I})$. The MA generates threshold credential signing keys and revocation authorization keys for issuers.
- **MVRegister** $(id, pp, apk) \to (usk, pseucred)$. The MA registers the MV, generates registration secrets, and certifies privately assigned one-time pseudonyms. It stores registration and tracing records.
- **CredIssue**$(id, A) \to \{\hat{\sigma}_{i,j}\}_{i\in T, j\in[1,q]}$. The MA verifies and authorizes the attribute request. Issuers generate blinded signature shares and sign the credential digest. Shares are released after threshold authorization.
- **CredAgg** $(\{\widehat{\sigma}_{i,j}\}_{i\in I, j\in[1,q]}) \to cred$. The MV verifies, unblinds, and aggregates signature shares from at least t_I issuers. It also verifies threshold issuance authorization.

- **WitnessUpdate** $(cred, st_e, R_e, \Delta R) \rightarrow WitList$. The MV verifies the revocation state and updates non-membership witnesses for unrevoked attributes. Incremental updates are used when available.
- **CredShow** $(cred, pseucred, J_{\text{req}}, ctx, st_e, R_e) \rightarrow tok$. The MV selects an unused pseudonym and discloses the requested attributes. It randomizes their signatures and non-membership witnesses and generates a showing proof bound to the session and revocation state.
- **CredVerify** $(tok, J_{\text{req}}, ctx, st_e, R_e) \rightarrow \{0, 1\}$. The verifier checks the pseudonym, revocation state, disclosed attributes, randomized signatures, and showing proof. It accepts only if all checks pass.
- **Revoke** $(id, cid, J_{\text{rev}}, st_e, R_e) \rightarrow st_{e+1}$. The MA verifies the revocation request and updates the accumulator with threshold authorization. The ledger verifies and publishes the new state.
- **Trace** $(tok, TraceTable, apk) \rightarrow id/\perp$. The MA verifies the pseudonym of an accepted token and retrieves its registered identity from the private tracing records. It returns $\perp$ if verification or lookup fails.

3.3 Security Model and Assumptions

STAR-AC is analyzed in the random oracle model. The MA and public ledger are trusted. The MA securely generates and distributes issuer key shares. Malicious distributed key generation is not considered. Issuers sign only MA-authorized requests. The adversary only obtains fewer than t_I issuer shares. The probabilistic polynomial-time (PPT) adversary $\mathcal{A}$ may control MVs, verifiers, and public channels. It may obtain credentials, issuance transcripts for non-challenge vehicles, public revocation states, and showing tokens. It may also submit authentication requests. The MA signing key, private tracing records, and honest challenge MV secrets remain uncompromised. The security analysis uses the following assumptions.

1. **Signature security [38]:** The threshold attribute signature scheme is unforgeable under chosen-message attacks. The adversary may obtain public keys, authorized issuance transcripts, blinded signature shares, and fewer than t_I signing shares. A forgery is a valid attribute signature on credential without MA-authorized issuance. MA signatures and threshold revocation authorization signatures are also unforgeable.
2. **Accumulator security [39]:** The bilinear inverse-exponent problem is hard in the Type-III bilinear groups. The accumulator SRS is $SRS_L = \{g^{\tau^\ell}, \tilde{g}^{\tau^\ell}\}_{\ell=0}^{L}$, where $\tau \in \mathbb{Z}_p^*$ is secret. Given SRS_L, no PPT adversary can output (ρ, Ω), where $\rho \in \mathbb{Z}_p^*$ and $\tau + \rho \neq 0$, satisfying $\Omega = e(g, \tilde{g})^{1/(\tau+\rho)}$ except with negligible probability.
3. **Privacy [40]:** DDH is hard in G_1. For a hidden random $\mu \in \mathbb{Z}_p^*$ and polynomially many independent random bases $R_\ell \in G_1$, the distributions $\{(R_\ell, R_\ell^\mu)\}_\ell$and$\{(R_\ell, Z_\ell)\}_\ell$ are computationally indistinguishable, where Z_ℓ are independent uniform group elements. This multi-base property follows from DDH through hybrid replacements.

The anonymity experiment considers two honest MVs with identical disclosed attributes, authorization results, session context, and revocation state. The adversary

may observe polynomially many tokens and request adaptive showings, but cannot corrupt or trace challenge MVs. Holder secrets, private pseudonym assignments, identity-linked registration and issuance records, and tracing records remain secret.

Each showing uses a fresh pseudonym and independent randomizers. Indices are sampled uniformly without replacement from disjoint private sets, with identical conditional public selection distributions. Public issuance records and metadata reveal neither challenge identities nor common pseudonym ownership. No public cross-group encoding reveals a holder secret or the discrete logarithm of h_1 relative to g. Identifying network metadata and physical-layer side channels are not considered.

All hash functions are independent random oracles. The reduction algorithms maintain consistent random-oracle responses, including those involving protected holder secrets. Pseudonym bases are embedded at their first oracle definition. Adaptive target selection and earlier base queries incur polynomial guessing losses. Unlinkability compares the showing views of two showings from one honest MV and from two honest MVs under matching public requests and histories.

Security Goals: Attribute authenticity prevents unauthorized attribute showing and cross-credential mixing. Attribute revocation security rejects revoked attributes. Showing anonymity and unlinkability protect honest MVs against external adversaries under the stated privacy conditions. Revocation state integrity prevents unauthorized updates. Authorized traceability allows the MA to identify accepted tokens through private tracing records.

4 Construction

This section presents the construction of STAR-AC. The main notations are summarized in Table 1.

4.1 High-Level Workflow

The high-level workflow consists of STAR-AC following phases:

1. **System Initialization:** The MA generates the public parameters, bilinear groups, and accumulator parameters. It initializes the revocation set and accumulator.
2. **MA Key Generation:** The MA generates its key pair and publishes the authenticated initial revocation state.
3. **Issuer Key Generation:** The MA generates threshold credential signing and revocation authorization keys, publishes verification keys, and securely distributes secret shares to issuers.
4. **MV Registration:** The MA verifies MV registration, generates registration secrets, certifies one-time pseudonyms, and stores registration and tracing records.
5. **Credential Issuance:** The MA authorizes attribute requests and generates issuance records. Issuers verify the authorization and generate blinded signature shares, which are released after threshold approval.
6. **Credential Aggregation:** The MV removes the blinding factors, verifies the signature shares, and aggregates them into attribute credentials.

Table 1 Main Notations

Notation	Description
λ, p	Security parameter and group order
G_1, G_2, G_T, e	Bilinear groups and pairing
$g, h_1, \tilde{g}, h_2$	Group generators
$H, H_1, H_A, H_{\text{ctx}}, H_\pi$	Hash functions
n_I, t_I, q, L	Issuer count, threshold, attribute count, and revocation limit
pp, SRS	Public parameters and reference string
ask, apk	MA key pair
pk_{ext}, rpk	Credential and revocation verification keys
id, μ	MV identity and holder secret
A, α_j, ρ_j	Attributes, encoded values, and revocation tags
cid, χ, Σ_j	Credential ID, context, and signature
$cred, I$	Credential and disclosed indices
$\kappa, P_u, Cert_\kappa$	Pseudonym index, value, and certificate
R_e, acc_e, st_e	Revoked tags, accumulator, and state
$A_{e,j}, B_{e,j}, \pi_{\text{show}}, tok$	Non-membership witness, proof, and token

7. **Witness Update:** The MV obtains the current revocation state and updates the non-membership witnesses for its valid attributes.
8. **Credential Showing and Verification:** The MV randomizes attribute signatures and non-membership witnesses, and generates a showing proof using a one-time pseudonym. The verifier checks the token and current revocation state.
9. **Attribute Revocation and Ledger Update:** The MA updates the accumulator with threshold revocation authorization. The ledger verifies and publishes the new revocation state.
10. **Tracing:** The MA uses the certified pseudonym and its private tracing records to identify the registered MV.

4.2 Construction

Setup$(1^\lambda, n_I, t_I, q, L) \rightarrow pp$. The MA generates Type-III bilinear groups (G_1, G_2, G_T, p, e) of prime order p, where $e : G_1 \times G_2 \rightarrow G_T$ is a non-degenerate bilinear pairing. The generators are $g, h_1 \in G_1$ and $\tilde{g}, h_2 \in G_2$. The blinding generators h_1, h_2 are independently generated. Their discrete logarithms relative to $g, \tilde{g}$ remain unknown to the adversary. The MA defines hash functions, $\mathrm{H} : \{0,1\}^* \rightarrow \mathbb{Z}_p, \mathrm{H}_A : \{0,1\}^* \rightarrow \mathbb{Z}_p^*, \mathrm{H}_{ctx}, \mathrm{H}_\pi : \{0,1\}^* \rightarrow \mathbb{Z}_p$, and $\mathrm{H}_1 : \{0,1\}^* \rightarrow G_1 \setminus \{1_{G_1}\}$. For an identity id and attribute (j, m_j), the MA defines $h_{id} = \mathrm{H}_1(id)$, and $\alpha_j = \mathrm{H}_A(j||m_j)$. The attribute revocation tag is $\rho_j = \mathrm{H}_A(\mu||j||m_j)$, where μ is the registered holder secret. Each attribute type uses a fixed index j across credentials. For the revoked tag set R_e, the MA defines $f_e(X) = \prod_{\rho \in R_e}(X + \rho) = \sum_{\ell=0}^{|R_e|} f_\ell X^\ell$. The MA selects $\tau \xleftarrow{\$} \mathbb{Z}_p^*$ and computes $SRS = \{g^{\tau^\ell}, \tilde{g}^{\tau^\ell}\}_{\ell=0}^{L}$, where L is the maximum number of revoked tags. It securely deletes τ after SRS generation. The accumulator is $acc_e = \prod_{\ell=0}^{|R_e|}(\tilde{g}^{\tau^\ell})^{f_\ell}$. Each

state satisfies $|R_e| \leq L$. The MA initializes $R_0 = \emptyset, acc_0 = \tilde{g}, dig_0 = \mathrm{H}(\mathrm{sort}(R_0))$ and defines $st_e = (e, acc_e, dig_e, \sigma_{st,e})$, where $dig_e = \mathrm{H}(\mathrm{sort}(R_e))$ and $\mathrm{sort}(R_e)$ is the sorting function for the elements of R_e. The MA defines a global pseudonym index domain $K \subseteq \{0,1\}^\lambda$. For each registered MV, it randomly assigns a private set $K_{id} \subseteq K$. The assigned sets are disjoint. Each MV maintains $Used \subseteq K_{id}$, and each verifier maintains $UsedList$. The parameters satisfy $1 \leq t_I \leq n_I$, where n_I is the number of issuers, t_I is the threshold, and q is the number of certified attributes. The MA publishes $pp = (G_1, G_2, G_T, p, e, g, h_1, \tilde{g}, h_2, \mathrm{H}, \mathrm{H}_1, \mathrm{H}_A, \mathrm{H}_{ctx}, \mathrm{H}_\pi, K, n_I, t_I, q, L, SRS)$.

MAKeyGen$(pp) \rightarrow (ask, apk)$. The MA selects $ask \xleftarrow{\$} \mathbb{Z}_p^*$ and computes $apk_1 = g^{ask}, apk_2 = \tilde{g}^{ask}$. It sets $apk = (apk_1, apk_2)$ and computes $\sigma_{st,0} = \mathrm{H}_1(0||acc_0||dig_0)^{ask}$. The MA publishes $st_0 = (0, acc_0, dig_0, \sigma_{st,0})$ through the public ledger and retains ask.

IssuerKeyGen$(pp) \rightarrow (pk_{ext}, rpk, \{isk_i, rsk_i\}_{i=1}^{n_I})$. The MA independently selects $x, y_\alpha, y_\chi, y_\mu, y_\rho, rsk \in \mathbb{Z}_p^*$. For each $\zeta \in \{x, y_\alpha, y_\chi, y_\mu, y_\rho, rsk\}$, it generates $f_\zeta(X) = \zeta + \sum_{\ell=1}^{t_I-1} a_{\zeta,\ell} X^\ell$, where the coefficients are independently selected from $\mathbb{Z}_p$. For $t_I > 1$, the leading coefficient is nonzero. For each issuer I_i, $i \in [1, n_I]$, the MA computes $\zeta_i = f_\zeta(i) \pmod p$ and assigns $isk_i = (x_i, y_{\alpha,i}, y_{\chi,i}, y_{\mu,i}, y_{\rho,i}), rsk_i$. It computes $X_i = \tilde{g}^{x_i}, \widetilde{Y}_{\nu,i} = \tilde{g}^{y_{\nu,i}}, Y_{\nu,i} = g^{y_{\nu,i}}$ for $\nu \in \{\alpha, \chi, \mu, \rho\}$, and $rpk_i = \tilde{g}^{rsk_i}$. The MA computes $X = \tilde{g}^x, \widetilde{Y}_\nu = \tilde{g}^{y_\nu}, Y_\nu = g^{y_\nu}, rpk = \tilde{g}^{rsk}$. It sets $pk = (X, \widetilde{Y}_\alpha, \widetilde{Y}_\chi, \widetilde{Y}_\mu, \widetilde{Y}_\rho)$, $pk_{ext} = (pk, Y_\alpha, Y_\chi, Y_\mu, Y_\rho)$. For each threshold set $T \subseteq [1, n_I]$, $|T| = t_I$, it computes $\lambda_i = \prod_{\ell \in T, \ell \neq i} \ell(\ell - i)^{-1} \pmod p$. The MA securely distributes (isk_i, rsk_i) to issuer I_i. It publishes and authenticates pk_{ext}, rpk, and all issuer verification keys.

Key Relations. For every $\nu \in \{\alpha, \chi, \mu, \rho\}$, the keys satisfy

$$e(Y_{\nu,i}, \tilde{g}) = e(g, \widetilde{Y}_{\nu,i}), e(Y_\nu, \tilde{g}) = e(g, \widetilde{Y}_\nu), X = \prod_{i \in T} X_i^{\lambda_i},$$
$$\widetilde{Y}_\nu = \prod_{i \in T} \widetilde{Y}_{\nu,i}^{\lambda_i}, Y_\nu = \prod_{i \in T} Y_{\nu,i}^{\lambda_i}, rpk = \prod_{i \in T} rpk_i^{\lambda_i} \quad (1)$$

MVRegister$(id, pp, apk) \rightarrow (usk, pseucred)$. The MV submits id to the MA through a secure channel. The MA selects $usk, \mu \xleftarrow{\$} \mathbb{Z}_p^*$ and computes $h_{id} = \mathrm{H}_1(id), upk = h_{id}^{usk}, B_\mu = h_{id}^\mu, Sig = \mathrm{H}_1(id||upk||B_\mu)^{ask}$. The MA randomly assigns a private set K_{id} of unused indices from K, sampled uniformly without replacement. For every $\kappa \in K_{id}$, it computes $Q_\kappa = \mathrm{H}_1(\mathrm{H}(apk)||\kappa)$, $P_{u,\kappa} = Q_\kappa^\mu, Cert_\kappa = \mathrm{H}_1(P_{u,\kappa}||\kappa)^{ask}$. The MA stores $TraceTable[\mathrm{H}(P_{u,\kappa}||\kappa)] = id$ and generates $pseucred = (\mu, B_\mu, Sig, K_{id}, \{Cert_\kappa\}_{\kappa \in K_{id}})$. It sends $(usk, pseucred)$ to the MV through a secure channel. The MV computes $upk = h_{id}^{usk}$ and verifies $B_\mu = h_{id}^\mu$, $e(Sig, \tilde{g}) = e(\mathrm{H}_1(id||upk||B_\mu), apk_2)$, $e(Cert_\kappa, \tilde{g}) = e(\mathrm{H}_1(Q_\kappa^\mu||\kappa), apk_2)$ for every $\kappa \in K_{id}$. The MA stores $RegList[id] = (id, upk, \mu, B_\mu, Sig, K_{id})$. The MV securely stores usk and $pseucred$.

CredIssue$(id, A) \rightarrow \{\hat{\sigma}_{i,j}\}_{i \in T, j \in [1,q]}$. The MV holds $A = \{(j, m_j)\}_{j=1}^q$. It selects $\nu \leftarrow \mathbb{Z}_p^*$ and sends (id, A, ν) to the MA through an authenticated secure channel. The MA verifies the requesting MV's identity and retrieves $RegList[id]$. It checks

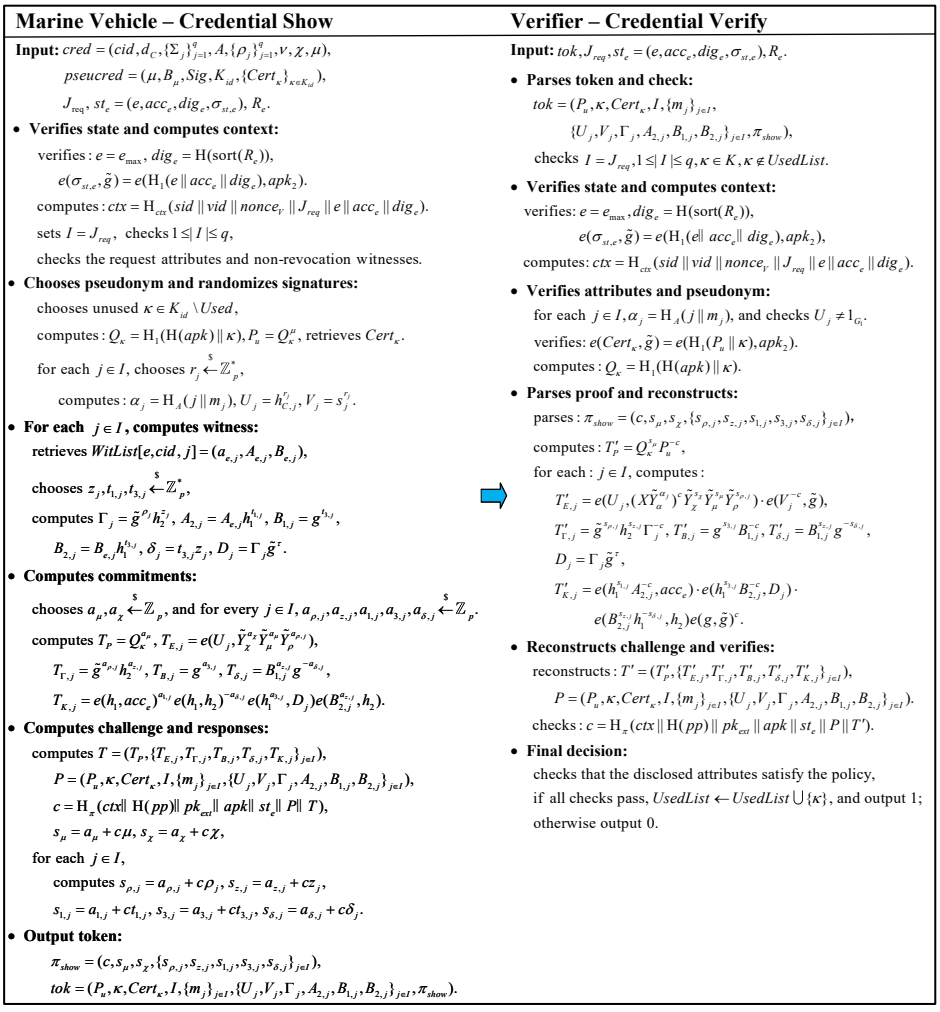


Fig. 2 Credential Showing and Verification.

the registration, requested attributes, and issuance policy. For each $j \in [1, q]$, it computes $\alpha_j = \mathrm{H}_A(j||m_j), \rho_j = \mathrm{H}_A(\mu||j||m_j)$. The MA selects $r_C \xleftarrow{\$} \mathbb{Z}_p^*$ and computes $d_0 = \mathrm{H}(id||upk||B_\mu||\mathrm{H}(A)||\nu||\ r_C||\{(j, \alpha_j, \rho_j)\}_{j=1}^q)$. For every j, it computes $h_{C,j} = \mathrm{H}_1(d_0||j)$ and selects $o_j \xleftarrow{\$} \mathbb{Z}_p$ to compute $C_{\mu,j} = h_{C,j}^{\mu} g^{o_j}$. The MA computes $d_C = \mathrm{H}(d_0||\{(h_{C,j}, C_{\mu,j})\}_{j=1}^q)$, $\chi = \mathrm{H}_A(d_C||\nu), cid = \mathrm{H}(id||d_C)$, $\zeta_C = \mathrm{H}_1(d_C)^{ask}$. It checks the uniqueness of the issuance record and stores $PendingList[id, cid] = (id, cid, upk, B_\mu,\ A, \nu, r_C, d_0, d_C, \chi, \zeta_C, \{(j, \alpha_j, \rho_j, h_{C,j}, C_{\mu,j})\}_{j=1}^q)$. The MA sends $(cid, d_C, \chi, r_C, \zeta_C, \{\rho_j, h_{C,j},\ C_{\mu,j}, o_j\}_{j=1}^q)$ to the MV through a secure channel. The MV recomputes $d_0, h_{C,j}, d_C, \chi, cid$ and verifies $\rho_j = \mathrm{H}_A(\mu||j||m_j)$, $C_{\mu,j} = h_{C,j}^{\mu} g^{o_j}$, $e(\zeta_C, \tilde{g}) = e(\mathrm{H}_1(d_C), apk_2)$. After successful verification, it sends $req_C = (id, upk, B_\mu, A, \nu, r_C, cid, d_C, \zeta_C,\ \{\rho_j, C_{\mu,j}\}_{j=1}^q)$ to each selected issuer I_i. Issuer I_i

recomputes $\alpha_j, d_0, h_{C,j}, d_C, \chi, cid$ and verifies $e(\zeta_C, \tilde{g}) = e(\mathrm{H}_1(d_C), apk_2)$. It checks the request against the MA-authorized record. For every $j \in [1,q]$, it computes $a_{i,j} = x_i + y_{\alpha,i}\alpha_j + y_{\chi,i}\chi + y_{\rho,i}\rho_j \pmod p$, $\hat{\sigma}_{i,j} = h_{C,j}^{a_{i,j}} C_{\mu,j}^{y_{\mu,i}}$. The issuer also computes $\sigma_{C,i} = \mathrm{H}_1(d_C)^{rsk_i}$ and sends $(id, cid, d_C, \chi, \sigma_{C,i})$ to the MA. The MA verifies $e(\sigma_{C,i}, \tilde{g}) = e(\mathrm{H}_1(d_C), rpk_i)$. After collecting valid shares from a threshold set T, $|T| = t_I$, it computes $\sigma_C = \prod_{i\in T} \sigma_{C,i}^{\lambda_i}$ and verifies $e(\sigma_C, \tilde{g}) = e(\mathrm{H}_1(d_C), rpk)$. The MA stores $ReqList[id, cid] = (id, cid, upk, B_\mu, A, \nu, r_C, d_0,\ d_C, \chi, \sigma_C, \{(j, \alpha_j, \rho_j, h_{C,j}, C_{\mu,j})\}_{j=1}^q)$, returns (cid, d_C, σ_C) to the MV, and confirms the issuance to the participating issuers. Each issuer then sends $\{\hat{\sigma}_{i,j}\}_{j=1}^q$ to the MV.

CredAgg$(\{\hat{\sigma}_{i,j}\}_{i\in T, j\in[1,q]}) \to cred$. The MV obtains valid shares from a threshold set T, where $|T| = t_I$. For every $i \in T$ and $j \in [1,q]$, it computes $\sigma_{i,j} = \hat{\sigma}_{i,j} Y_{\mu,i}^{-o_j}$ and verifies $e(\sigma_{i,j}, \tilde{g}) = e(h_{C,j}, X_i \widetilde{Y}_{\alpha,i}^{\alpha_j} \widetilde{Y}_{\chi,i}^{\chi} \widetilde{Y}_{\mu,i}^{\mu} \widetilde{Y}_{\rho,i}^{\rho_j})$. Invalid shares are discarded. The MV computes $s_j = \prod_{i\in T} \sigma_{i,j}^{\lambda_i}$ and sets $\Sigma_j = (h_{C,j}, s_j)$. It verifies $e(s_j, \tilde{g}) = e(h_{C,j}, X \widetilde{Y}_{\alpha}^{\alpha_j} \widetilde{Y}_{\chi}^{\chi} \widetilde{Y}_{\mu}^{\mu} \widetilde{Y}_{\rho}^{\rho_j})$. The MV obtains (cid, d_C, σ_C) from the MA and verifies $e(\sigma_C, \tilde{g}) = e(\mathrm{H}_1(d_C), rpk)$. It checks that cid, d_C match the issuance request and stores $cred = (cid, d_C, \{\Sigma_j\}_{j=1}^q, A, \{\rho_j\}_{j=1}^q, \nu, \chi, \mu)$.

Signature Relations. The shares satisfy

$$\begin{aligned} \hat{\sigma}_{i,j} &= h_{C,j}^{a_{i,j} + \mu y_{\mu,i}} g^{o_j y_{\mu,i}}, \\ \sigma_{i,j} &= h_{C,j}^{x_i + y_{\alpha,i}\alpha_j + y_{\chi,i}\chi + y_{\mu,i}\mu + y_{\rho,i}\rho_j}. \end{aligned} \tag{2}$$

The aggregated signature satisfies

$$s_j = h_{C,j}^{x + y_\alpha \alpha_j + y_\chi \chi + y_\mu \mu + y_\rho \rho_j}. \tag{3}$$

CredShow$(cred, pseucred, J_{req}, ctx, st_e, R_e) \to tok$. The verifier sends $(sid, vid, J_{req}, nonce_V)$ to MV. Both parties obtain $st_e = (e, acc_e, dig_e, \sigma_{st,e})$ and verify $e = e_{\max}, dig_e = \mathrm{H}(\mathrm{sort}(R_e))$, $e(\sigma_{st,e}, \tilde{g}) = e(\mathrm{H}_1(e||acc_e||dig_e), apk_2)$. They compute $ctx = \mathrm{H}_{ctx}(sid||vid||nonce_V||J_{req}||e||acc_e||dig_e)$. The MV sets $I = J_{req}$, where $1 \le |I| \le q$, and checks the requested attributes and their non-revocation witnesses.

The MV randomly selects an unused $\kappa \in K_{id} \setminus Used$ and computes $Q_\kappa = \mathrm{H}_1(\mathrm{H}(apk)||\kappa), P_u = Q_\kappa^\mu$. It retrieves $Cert_\kappa$. For every $j \in I$, it computes $\alpha_j = \mathrm{H}_A(j||m_j)$, selects $r_j \xleftarrow{\$} \mathbb{Z}_p^*$ and computes $U_j = h_{C,j}^{r_j}, V_j = s_j^{r_j}$. The MV retrieves $WitList[e, cid, j] = (a_{e,j}, A_{e,j}, B_{e,j})$. It selects $z_j, t_{1,j}, t_{3,j} \xleftarrow{\$} \mathbb{Z}_p^*$ and computes $\Gamma_j = \tilde{g}^{\rho_j} h_2^{z_j}, A_{2,j} = A_{e,j} h_1^{t_{1,j}}, B_{1,j} = g^{t_{3,j}}, B_{2,j} = B_{e,j} h_1^{t_{3,j}}, \delta_j = t_{3,j} z_j \pmod p, D_j = \Gamma_j \tilde{g}^\tau$. The MV selects $a_\mu, a_\chi \xleftarrow{\$} \mathbb{Z}_p$ and, for every $j \in I$, $a_{\rho,j}, a_{z,j}, a_{1,j}, a_{3,j}, a_{\delta,j} \xleftarrow{\$} \mathbb{Z}_p$. It computes $T_P = Q_\kappa^{a_\mu}$, $T_{E,j} = e(U_j, \widetilde{Y}_\chi^{a_\chi} \widetilde{Y}_\mu^{a_\mu} \widetilde{Y}_\rho^{a_{\rho,j}})$, $T_{\Gamma,j} = \tilde{g}^{a_{\rho,j}} h_2^{a_{z,j}}$, $T_{B,j} = g^{a_{3,j}}, T_{\delta,j} = B_{1,j}^{a_{z,j}} g^{-a_{\delta,j}}$. The MV computes or retrieves $e(h_1, acc_e)$ and $e(h_1, h_2)$. It computes $T_{K,j} = e(h_1, acc_e)^{a_{1,j}} e(h_1, h_2)^{-a_{\delta,j}} \cdot e(h_1^{a_{3,j}}, D_j) e(B_{2,j}^{a_{z,j}}, h_2)$.

The MV forms $T = (T_P, \{T_{E,j}, T_{\Gamma,j}, T_{B,j}, T_{\delta,j}, T_{K,j}\}_{j\in I})$ and $P = (P_u, \kappa, Cert_\kappa, I, \{m_j\}_{j\in I}, \{U_j, V_j, \Gamma_j, A_{2,j}, B_{1,j}, \quad B_{2,j}\}_{j\in I})$. It computes $c =$

$\mathrm{H}_\pi(ctx||\mathrm{H}(pp)||pk_{ext}||apk||\ st_e||P||T)$. The MV computes $s_\mu = a_\mu + c\mu \pmod p$, $s_\chi = a_\chi + c\chi \pmod p$ and, for every $j \in I$, $s_{\rho,j} = a_{\rho,j} + c\rho_j \pmod p$, $s_{z,j} = a_{z,j} + cz_j \pmod p$, $s_{1,j} = a_{1,j} + ct_{1,j} \pmod p$, $s_{3,j} = a_{3,j} + ct_{3,j} \pmod p$, $s_{\delta,j} = a_{\delta,j} + c\delta_j \pmod p$. The MV generates $\pi_{show} = (c, s_\mu, s_\chi, \{s_{\rho,j}, s_{z,j}, s_{1,j}, s_{3,j}, s_{\delta,j}\}_{j\in I})$ and $tok = (P_u, \kappa, Cert_\kappa, I, \{m_j\}_{j\in I}, \{U_j, V_j, \Gamma_j, A_{2,j}, B_{1,j}, B_{2,j}\}_{j\in I}, \pi_{show})$. It sends tok to the verifier and marks κ as used.

Showing Proof Relations. The proof π_{show} demonstrates knowledge of

$$\mathcal{W} = (\mu, \chi, \{\rho_j, z_j, t_{1,j}, t_{3,j}, \delta_j\}_{j\in I}), \tag{4}$$

satisfying

$$P_u = Q_\kappa^\mu, \tag{5}$$

and for every $j \in I$,

$$\begin{aligned} E_j &= M_{\chi,j}^\chi M_{\mu,j}^\mu M_{\rho,j}^{\rho_j}, \Gamma_j = \tilde{g}^{\rho_j} h_2^{z_j}, B_{1,j} = g^{t_{3,j}}, B_{1,j}^{z_j} = g^{\delta_j}, \\ K_j &= e(h_1, acc_e)^{t_{1,j}} e(h_1, D_j)^{t_{3,j}} \cdot e(B_{2,j}, h_2)^{z_j} e(h_1, h_2)^{-\delta_j}, \end{aligned} \tag{6}$$

where

$$\begin{aligned} E_j &= \frac{e(V_j, \tilde{g})}{e(U_j, X\widetilde{Y}_\alpha^{\alpha_j})}, M_{\nu,j} = e(U_j, \widetilde{Y}_\nu), \nu \in \{\chi, \mu, \rho\} \\ K_j &= \frac{e(A_{2,j}, acc_e)e(B_{2,j}, D_j)}{e(g, \tilde{g})}. \end{aligned} \tag{7}$$

These relations are included in the showing proof without separately evaluating E_j, $M_{\nu,j}$, or K_j during proof generation.

CredVerify$(tok, J_{req}, ctx, st_e, R_e) \rightarrow \{0, 1\}$. The verifier parses $tok = (P_u, \kappa, Cert_\kappa, I, \{m_j\}_{j\in I}, \{U_j, V_j, \Gamma_j, A_{2,j}, \ B_{1,j}, B_{2,j}\}_{j\in I}, \pi_{show})$. It checks $I = J_{req}, 1 \le |I| \le q, \kappa \in K, \kappa \notin UsedList$.

The verifier obtains $st_e = (e, acc_e, dig_e, \sigma_{st,e})$ and verifies $e = e_{\max}, dig_e = \mathrm{H}(\mathrm{sort}(R_e))$, and $e(\sigma_{st,e}, \tilde{g}) = e(\mathrm{H}_1(e||acc_e||dig_e), apk_2)$. It computes $ctx = \mathrm{H}_{ctx}(sid||vid||nonce_V||J_{req}||e||acc_e||dig_e)$. For every $j \in I$, it computes $\alpha_j = \mathrm{H}_A(j||m_j)$ and checks $U_j \neq 1_{G_1}$. It also verifies $e(Cert_\kappa, \tilde{g}) = e(\mathrm{H}_1(P_u||\kappa), apk_2)$. The verifier computes $Q_\kappa = \mathrm{H}_1(\mathrm{H}(apk)||\kappa)$, and parses $\pi_{show} = (c, s_\mu, s_\chi, \{s_{\rho,j}, s_{z,j}, s_{1,j}, s_{3,j}, s_{\delta,j}\}_{j\in I})$. It computes $T'_P = Q_\kappa^{s_\mu} P_u^{-c}$, and for every $j \in I$, $T'_{E,j} = e(U_j, (X\widetilde{Y}_\alpha^{\alpha_j})^c \widetilde{Y}_\chi^{s_\chi} \widetilde{Y}_\mu^{s_\mu} \widetilde{Y}_\rho^{s_{\rho,j}}) \cdot e(V_j^{-c}, \tilde{g})$, $T'_{\Gamma,j} = \tilde{g}^{s_{\rho,j}} h_2^{s_{z,j}} \Gamma_j^{-c}$, $T'_{B,j} = g^{s_{3,j}} B_{1,j}^{-c}$, $T'_{\delta,j} = B_{1,j}^{s_{z,j}} g^{-s_{\delta,j}}$. The verifier computes $D_j = \Gamma_j \tilde{g}^\tau$ and uses the precomputed value $e(g, \tilde{g})$ to compute $T'_{K,j} = e(h_1^{s_{1,j}} A_{2,j}^{-c}, acc_e) \cdot e(h_1^{s_{3,j}} B_{2,j}^{-c}, D_j) \cdot e(B_{2,j}^{s_{z,j}} h_1^{-s_{\delta,j}}, h_2) e(g, \tilde{g})^c$. It forms $T' = (T'_P, \{T'_{E,j}, T'_{\Gamma,j}, T'_{B,j}, T'_{\delta,j}, T'_{K,j}\}_{j\in I})$. The verifier reconstructs P and checks $c = \mathrm{H}_\pi(ctx||\mathrm{H}(pp)||pk_{ext}||apk||st_e||P||T')$.

It also checks that the disclosed attributes satisfy the requested authorization policy. If any check fails, the verifier outputs 0. Otherwise, it adds κ to $UsedList$ and outputs 1.

WitnessUpdate$(cred, st_e, R_e, \Delta R) \rightarrow WitList$. The MV obtains $st_e = (e, acc_e, dig_e, \sigma_{st,e})$ and verifies $e = e_{\max}, dig_e = \mathrm{H}(\mathrm{sort}(R_e))$, $e(\sigma_{st,e}, \tilde{g}) = e(\mathrm{H}_1(e||acc_e||dig_e), apk_2)$. Invalid states are rejected. For every $j \in [1, q]$, the MV verifies $\rho_j = \mathrm{H}_A(\mu||j||m_j)$ and checks $\rho_j \notin R_e$. Revoked attributes are removed from the active witness list.

Incremental Update. If the previous valid witness and authenticated $\Delta R = R_e \setminus R_{e-1}$ are available, the MV computes $u(X) = \prod_{\rho \in \Delta R}(X + \rho)$. For each unrevoked attribute, it checks $u(-\rho_j) \neq 0$ and retrieves $WitList[e-1, cid, j] = (a_{e-1,j}, A_{e-1,j}, B_{e-1,j})$. The MV computes $a_{e,j} = a_{e-1,j} u(-\rho_j)^{-1}$ and $f_e(X) = f_{e-1}(X)u(X)$. It computes $b_{e,j}(X) = \frac{1 - a_{e,j} f_e(X)}{X + \rho_j}$. For $b_{e,j}(X) = \sum_\ell b_{j,\ell} X^\ell$, the MV computes $A_{e,j} = g^{a_{e,j}}, B_{e,j} = \prod_\ell (g^{\tau^\ell})^{b_{j,\ell}}$. It stores $WitList[e, cid, j] = (a_{e,j}, A_{e,j}, B_{e,j})$.

Full Recalculation. If the previous witness or authenticated incremental data is unavailable, the MV computes $f_e(X) = \prod_{\rho \in R_e}(X + \rho)$ and, for every unrevoked attribute, $a_{e,j} = f_e(-\rho_j)^{-1}$, $b_{e,j}(X) = \frac{1 - a_{e,j} f_e(X)}{X + \rho_j}$. It computes $A_{e,j}$ and $B_{e,j}$ as above and stores the updated witness.

Non-Membership Relations. Both methods satisfy

$$\begin{aligned} &a_{e,j} f_e(X) + b_{e,j}(X)(X + \rho_j) = 1, \\ &e(A_{e,j}, acc_e) e(B_{e,j}, \tilde{g}^{\tau + \rho_j}) = e(g, \tilde{g}). \end{aligned} \tag{8}$$

Revoke$(id, cid, J_{rev}, st_e, R_e) \rightarrow st_{e+1}$. The MA receives (id, cid, J_{rev}), where $J_{rev} \subseteq [1, q]$. It retrieves $RegList[id]$ and $ReqList[id, cid]$ and verifies the request authorization, registered credential, and target attributes. The MA verifies $e(\sigma_C, \tilde{g}) = e(\mathrm{H}_1(d_C), rpk)$. For every $j \in J_{rev}$, it retrieves the certified ρ_j and computes $R_J = \{\rho_j : j \in J_{rev}, \rho_j \notin R_e\}$. If $R_J = \emptyset$, the MA returns st_e. Otherwise, it computes $R_{e+1} = R_e \cup R_J$ and checks $|R_{e+1}| \leq L$. The MA computes $f_e(X) = \prod_{\rho \in R_e}(X + \rho) = \sum_{\ell=0}^{m} f_\ell X^\ell, m = |R_e|$, $u(X) = \prod_{\rho \in R_J}(X + \rho) = \sum_{\ell=0}^{s} u_\ell X^\ell, s = |R_J|$, $f_{e+1}(X) = f_e(X)u(X) = \sum_{\ell=0}^{m+s} f'_\ell X^\ell$, where $f'_\ell = \sum_{a+b=\ell} f_a u_b$. Missing coefficients are zero. The MA computes $\Delta_\ell = f'_\ell - f_\ell$ and $acc_{e+1} = acc_e \prod_{\ell=0}^{m+s} (\tilde{g}^{\tau^\ell})^{\Delta_\ell}$, $dig_{e+1} = \mathrm{H}(\mathrm{sort}(R_{e+1}))$. It selects $r_R \xleftarrow{\$} \mathbb{Z}_p^*$ and computes $reqCom = \mathrm{H}(id||cid||J_{rev}||r_R)$. The MA privately sends the complete revocation request to the approving issuers and computes $d_R = \mathrm{H}(\mathrm{H}(pp)||apk||rpk||reqCom||e||acc_e||dig_e||(e+1)||acc_{e+1}||dig_{e+1}||\mathrm{H}(\mathrm{sort}(R_J))||u(X))$. Each approving issuer verifies the request authorization, certified tags, and state transition. Issuer I_i computes $\sigma_{rev,i} = \mathrm{H}_1(d_R)^{rsk_i}$. The MA verifies the shares, selects a valid threshold set T, and computes $\sigma_{rev} = \prod_{i \in T} \sigma_{rev,i}^{\lambda_i}$. It verifies $e(\sigma_{rev}, \tilde{g}) = e(\mathrm{H}_1(d_R), rpk)$ and computes $\sigma_{st,e+1} = \mathrm{H}_1((e+1)||acc_{e+1}||dig_{e+1})^{ask}$. The MA submits the authorized update to the public ledger. The ledger checks that e is the current epoch and verifies $R_{e+1} = R_e \cup R_J, |R_{e+1}| \leq L$, $f_{e+1}(X) = f_e(X)u(X)$, $acc_{e+1} = \prod_{\ell=0}^{m+s} (\tilde{g}^{\tau^\ell})^{f'_\ell}$, $dig_{e+1} = \mathrm{H}(\mathrm{sort}(R_{e+1}))$. It recomputes d_R and verifies $e(\sigma_{rev}, \tilde{g}) = e(\mathrm{H}_1(d_R), rpk)$, $e(\sigma_{st,e+1}, \tilde{g}) = e(\mathrm{H}_1((e+1)||acc_{e+1}||dig_{e+1}), apk_2)$. If any check fails, the update is rejected. Otherwise, the

ledger stores $RevEvent_{e+1} = (reqCom, d_R, \sigma_{rev}, \mathrm{H}(\mathrm{sort}(R_J)), R_J, u(X))$ and publishes $st_{e+1} = (e+1, acc_{e+1}, dig_{e+1}, \sigma_{st,e+1})$. The new state becomes effective for subsequent credential showing.

Trace$(tok, TraceTable, apk) \rightarrow id/\perp$. Given a valid accepted token, the MA extracts $(P_u, \kappa, Cert_\kappa)$ and verifies $e(Cert_\kappa, \tilde{g}) = e(\mathrm{H}_1(P_u||\kappa), apk_2)$. It searches $TraceTable[\mathrm{H}(P_u||\kappa)]$ and returns the registered identity id. If the certificate is invalid or no record exists, it outputs $\perp$.

5 Security Proof and Analysis

Based on the aforementioned security model and assumptions, we analyze the correctness and security of STAR-AC. The analysis covers attribute authenticity, showing anonymity and unlinkability, attribute-level revocation, revocation state integrity, and authorized traceability.

5.1 Correctness

Theorem 1 (Correctness) *For honestly generated credentials and valid non-revocation witnesses, CredVerify accepts the showing token if the requested attributes satisfy the authorization policy and the pseudonym index has not been used at the verifier.*

Proof For each attribute j, the blinded signature share generated by issuer I_i is $\hat{\sigma}_{i,j} = h_{C,j}^{a_{i,j}} C_{\mu,j}^{y_{\mu,i}} = h_{C,j}^{a_{i,j}+\mu y_{\mu,i}} Y_{\mu,i}^{o_j}$. After removing the blinding factor, the MV obtains $\sigma_{i,j} = \hat{\sigma}_{i,j} \cdot Y_{\mu,i}^{-o_j} = h_{C,j}^{x_i + y_{\alpha,i}\alpha_j + y_{\chi,i}\chi + y_{\mu,i}\mu + y_{\rho,i}\rho_j}$. Lagrange aggregation gives $s_j = \prod_{i\in T} \sigma_{i,j}^{\lambda_i} = h_{C,j}^{x + y_\alpha \alpha_j + y_\chi \chi + y_\mu \mu + y_\rho \rho_j}$. The aggregated signature therefore satisfies $e(s_j, \tilde{g}) = e(h_{C,j}, X\tilde{Y}_\alpha^{\alpha_j}\tilde{Y}_\chi^{\chi}\tilde{Y}_\mu^{\mu}\tilde{Y}_\rho^{\rho_j})$. The randomized signature $(U_j, V_j) = (h_{C,j}^{r_j}, s_j^{r_j})$ satisfies the same verification relation: $e(V_j, \tilde{g}) = e(U_j, X\tilde{Y}_\alpha^{\alpha_j}\tilde{Y}_\chi^{\chi}\tilde{Y}_\mu^{\mu}\tilde{Y}_\rho^{\rho_j})$. Since $h_{C,j} \neq 1_{G_1}$ and $r_j \in \mathbb{Z}_p^*$, the randomized signature satisfies $U_j \neq 1_{G_1}$. Hence, it passes the non-identity check in *CredVerify*.

For an unrevoked attribute, $\rho_j \notin R_e$. Thus, $f_e(-\rho_j) \neq 0$. The witness coefficients are $a_{e,j} = f_e(-\rho_j)^{-1}, b_{e,j}(X) = \frac{1 - a_{e,j} f_e(X)}{X + \rho_j}$. They satisfy $a_{e,j} f_e(X) + b_{e,j}(X)(X + \rho_j) = 1$. Consequently, $e(A_{e,j}, acc_e) e(B_{e,j}, \tilde{g}^\tau \tilde{g}^{\rho_j}) = e(g, \tilde{g})$. For the randomized witness, the MV computes $\Gamma_j = \tilde{g}^{\rho_j} h_2^{z_j}, D_j = \Gamma_j \tilde{g}^\tau$, $A_{2,j} = A_{e,j} h_1^{t_{1,j}}, B_{1,j} = g^{t_{3,j}}, B_{2,j} = B_{e,j} h_1^{t_{3,j}}$. With $\delta_j = t_{3,j} z_j$, bilinearity gives $K_j = e(h_1, acc_e)^{t_{1,j}} e(h_1, D_j)^{t_{3,j}} \cdot e(B_{2,j}, h_2)^{z_j} e(h_1, h_2)^{-\delta_j}$. The prover's responses satisfy all relations in the showing proof.

Substituting them into the verifier's reconstruction equations yields $\mathcal{T}' = \mathcal{T}$. Hence, $c = \mathrm{H}_\pi(ctx||\mathrm{H}(pp)||pk_{\mathrm{ext}}||apk||st||\ \mathcal{P}||\mathcal{T}')$. The MA certified pseudonym and authenticated revocation state also pass their verification equations. Therefore, *CredVerify* accepts the valid token. □

5.2 Soundness and Zero-Knowledge of the Showing Proof

Lemma 1 (Knowledge Soundness) *The joint showing proof is knowledge-sound for the relations in Eq. (5) and Eq. (6) in the classical random oracle model. A PPT prover that generates*

an accepting proof for a fresh statement with non-negligible probability admits an expected polynomial-time extractor for the complete witness.

Proof The statement fixes the public parameters, verification keys, session context, authenticated state, and token fields. All group elements belong to their designated prime-order groups, and witness values belong to $\mathbb{Z}_p$. Consider two accepting transcripts with the same statement and commitments but distinct challenges c, c'. For $\Delta = c - c' \neq 0$, each witness component is extracted as $w = (s_w - s'_w)\Delta^{-1} \pmod p$. This yields $\mathcal{W} = (\mu, \chi, \{\rho_j, z_j, t_{1,j}, t_{3,j}, \delta_j\}_{j \in I})$. Dividing the verification equations and raising the results to Δ^{-1} gives $P_u = Q_\kappa^\mu$ and, for every $j \in I$, $E_j = M_{\chi,j}^{\chi} M_{\mu,j}^{\mu} M_{\rho,j}^{\rho_j}$, $\Gamma_j = \tilde{g}^{\rho_j} h_2^{z_j}$, $B_{1,j} = g^{t_{3,j}}$, $B_{1,j}^{z_j} = g^{\delta_j}$, and $K_j = e(h_1, acc_e)^{t_{1,j}} e(h_1, D_j)^{t_{3,j}} e(B_{2,j}, h_2)^{z_j} e(h_1, h_2)^{-\delta_j}$. Since g has order p, $\delta_j = t_{3,j} z_j \pmod p$. Shared responses yield the same μ, χ for all disclosed attributes. Thus, the extracted witness satisfies all showing proof relations.

The Fiat-Shamir hash binds the full statement and commitment vector. Write ε_q for the probability of a fresh accepting proof whose challenge input was queried. With at most $Q_\pi \geq 1$ queries, the general forking lemma gives $\Pr[\text{Fork}] \geq \varepsilon_q \left(\frac{\varepsilon_q}{Q_\pi} - \frac{1}{p} \right)$. Acceptance without the corresponding query has probability at most $1/p$. Non-negligible proving success therefore permits expected polynomial-time extraction. Previously supplied honest statements already have valid witnesses. □

Lemma 2 (Zero-Knowledge) *The joint showing proof π_{show} is computationally zero-knowledge in the random oracle model.*

Proof A simulator selects c, s_μ, s_χ, and all attribute responses uniformly from $\mathbb{Z}_p$. It reconstructs $\mathcal{T}' = (T'_P, \{T'_{E,j}, T'_{\Gamma,j}, T'_{B,j}, T'_{\delta,j}, T'_{K,j}\}_{j \in I})$ using the verification equations. It then programs $\mathrm{H}_\pi(ctx \| \mathrm{H}(pp) \| pk_{\text{ext}} \| apk \| st \| \mathcal{P} \| \mathcal{T}') = c$. Other random oracle queries are answered consistently. Since $Q_\kappa \neq 1_{G_1}$ and $T'_P = Q_\kappa^{s_\mu} P_u^{-c}$, the reconstructed pseudonym commitment is uniformly distributed in G_1. For at most q_H previous queries to H_π, the probability that the programmed input was queried earlier is at most q_H/p. The simulator aborts in this event. The simulated challenges and responses have the same distribution as those in honest proofs. Therefore, the simulated proof is computationally indistinguishable from an honest proof, except with negligible probability.

For N_π adaptively selected valid statements and at most Q_π other oracle queries, the total simulation error is at most $N_\pi(Q_\pi + N_\pi)/p$. The simulator also generates programmed accepting transcripts for well-formed statements in the intermediate games. It recomputes all commitments from the current statement. Equality with the honest proof distribution is asserted only for valid statements. □

Lemma 3 (Randomized Non-Membership Witness Hiding) *Randomized non-membership witnesses for valid unrevoked tags are computationally indistinguishable from their ideal distributions in the simulated showing view, provided the permitted auxiliary information is consistently simulatable and contains no cross-group encoding of the discrete logarithm of h_1 relative to g.*

Proof For an unrevoked tag ρ with witness (A, B), the randomized elements are $\Gamma = \tilde{g}^{\rho} h_2^{z}, A_2 = A h_1^{t_1}, B_1 = g^{t_3}, B_2 = B h_1^{t_3}$ where z, t_1, t_3 are independently sampled from $\mathbb{Z}_p^*$. The showing proof is first replaced with the simulator of Lemma 2.

Since h_1, h_2 are generators, Γ and A_2 are independent and each is within statistical distance $1/p$ of uniform. For (B_1, B_2), a DDH reduction receives (g, g^a, g^t, T) and sets $h_1 = g^a, B_1 = g^t, B_2 = BT$. The reduction generates the accumulator parameters, original witnesses, and remaining public values. It generates non-target elements according to their current hybrid distributions and simulates the showing proof using Lemma 2. If $T = g^{at}$, the target pair follows the real distribution. If T is uniform, B_2 is independent of B_1. All permitted auxiliary information is simulated consistently and contains no cross-group encoding of a. Therefore, a distinguisher between the two distributions yields a DDH distinguisher. Including statistical sampling differences, one witness replacement satisfies $\Delta_{\text{wit}} \leq \varepsilon_{\text{DDH}} + \frac{4}{p}$.

For two valid witnesses, the triangle inequality gives $\varepsilon_{\text{wit}}^{(1)} \leq 2\varepsilon_{\text{DDH}} + \frac{8}{p}$. For N_W witnesses across adaptive showing sessions, successive replacements give $\varepsilon_{\text{wit}} \leq N_W(\varepsilon_{\text{DDH}} + \frac{4}{p})$. All showing proofs remain simulated during these replacements. Each target DDH instance is embedded independently, while other public elements retain their current hybrid distributions.

Hence, the randomized witnesses hide their original values under the stated conditions. □

5.3 Security of Anonymous Attribute Showing

Theorem 2 (Attribute Authenticity) *An accepted showing authenticates its disclosed attributes under a common holder secret and an MA-authorized issuance context, except with negligible probability.*

Proof Lemma 1 extracts common values μ, χ and a tag ρ_j for each disclosed attribute. The extracted relation gives $e(V_j, \tilde{g}) = e(U_j, X \tilde{Y}_{\alpha}^{\alpha_j} \tilde{Y}_{\chi}^{\chi} \tilde{Y}_{\mu}^{\mu} \tilde{Y}_{\rho}^{\rho_j})$, where $\alpha_j = \mathrm{H}_A(j \| m_j)$. Since $U_j \neq 1_{G_1}$, (U_j, V_j) is a valid signature on $(\alpha_j, \chi, \mu, \rho_j)$. The reduction records all MA-authorized message tuples. An extracted tuple absent from these records gives a signature forgery. Otherwise, its authorized record satisfies $\alpha_j = \mathrm{H}_A(j' \| m_j')$, $\rho_j = \mathrm{H}_A(\mu \| j' \| m_j')$. Except for a random-oracle collision, $(j', m_j') = (j, m_j)$. Hence, $\rho_j = \mathrm{H}_A(\mu \| j \| m_j)$. All disclosed signatures share the extracted μ, χ. Distinct authorized issuance contexts have distinct $\chi = \mathrm{H}_A(d_C \| \nu)$, except for collisions. The attributes therefore belong to one authorized issuance context. The relation $P_u = Q_{\kappa}^{\mu}$ also binds them to the certified pseudonym. An unauthorized showing thus yields a signature forgery or a hash collision, subject to the polynomial extraction loss in Lemma 1. Its success probability is negligible. □

Theorem 3 (Showing Anonymity and Unlinkability) *STAR-AC provides showing anonymity and unlinkability against PPT external adversaries in the random oracle model.*

Proof Anonymity compares showings from two honest MVs with matching disclosed attributes, authorization results, and public session information. Unlinkability compares two showing slots owned by one honest MV with the same slots owned by two honest MVs. Each slot has matching public requests and authenticated states across experiments. Both experiments contain the same registered candidates and permitted auxiliary records. Only the

designated owners differ. Challenge MVs remain uncorrupted and untraced. Their private assignments and identifying records remain hidden. Protected showings use fresh indices and independent randomizers.

Game G_0 is the first real experiment. *Game* G_1 replaces all protected proofs with the simulator of Lemma 2. For at most N_S proofs and Q_π other challenge-oracle queries, $\varepsilon_{\mathrm{ZK}} \leq \frac{N_S(Q_\pi+N_S)}{p}$. Proofs remain simulated throughout the intermediate games.

Game G_2 replaces randomized non-membership elements. Each Γ_j and $A_{2,j}$ becomes an independent uniform element of its designated group, with statistical distance at most $1/p$ per replacement. For each remaining pair, a DDH reduction receives (g, g^a, g^t, T) and sets $h_1 = g^a, \quad B_{1,j} = g^t, \quad B_{2,j} = B_{e,j}T$. The reduction generates holder secrets, signing keys, accumulator parameters, and original witnesses. It computes issuance records and non-target elements under their current hybrid distributions. If $T = g^{at}$, the target pair is real. If T is uniform, $B_{2,j}$ is uniform and independent of $B_{1,j}$. Simulated proofs require neither a nor t. The SRS and signing keys are independent of a, and the permitted view contains no cross-group encoding of a. Thus, each replacement reduces to DDH.

Game G_3 replaces pseudonym and randomized signature pairs individually. A reduction receives (g, A, R, T), where $A = g^a$, and embeds a as the protected holder secret. The reduction generates MA and issuer keys. Non-target H_1 outputs are sampled as g^β, with β retained privately. For registration and issuance bases $h_{id} = g^{\beta_0}$ and $h_{C,j} = g^{\beta_j}$, it computes $B_\mu = A^{\beta_0}, \quad C_{\mu,j} = A^{\beta_j} g^{o_j}$. Issuance digests, authorizations, and blinded signature shares are then computed without knowing a. Protected revocation tags are cached by holder and attribute pair. For a query $\mathrm{H}_A(\hat{\mu}\|j\|m)$, the reduction tests $g^{\hat{\mu}} = A$. Matching queries use the protected cache; other queries use the ordinary oracle table. The cache also preserves answers issued before issuance.

For a non-target base $Q_\kappa = g^\gamma$, the reduction computes $P_u = A^\gamma$. For a non-target randomized signature, it samples $v \in \mathbb{Z}_p^*$ and computes $U_j = g^v, \quad V_j = U_j^{\omega_j} A^{v y_\mu}$, where $\omega_j = x + y_\alpha \alpha_j + y_\chi \chi + y_\rho \rho_j$. These values preserve the showing distribution under the shared holder secret and issuance context. Earlier replacements retain their ideal distributions. For a target pseudonym, the reduction sets $Q_\kappa = R, \quad P_u = T$, and generates the matching certificate $Cert_\kappa = \mathrm{H}_1(P_u\|\kappa)^{ask}$ and private tracing record. The base is embedded at its first oracle definition. Guessing this event handles earlier queries. Fresh indices prevent conflicts with earlier showings. For a target randomized signature, a separate reduction sets $U_j = R, \quad V_j = R^{\omega_j} T^{y_\mu}$. If $T = R^a$, the pair is real. If T is uniform, V_j is uniform and independent of U_j, since $y_\mu \neq 0$. Each reduction changes one pair, preserves all non-target distributions, and recomputes matching certificates and derived values. These simulations cover permitted auxiliary records and adaptive showing queries.

In *Game* G_3, protected pseudonyms, signature second components, and randomized non-membership elements are independent of the designated owners. Certificates depend only on public pseudonyms and indices. Simulated proofs depend only on public statements and oracle tables.

Game G_4 changes the designated owners. Public requests, states, and permitted auxiliary records remain unchanged. Private index assignments have identical conditional public selection distributions. Therefore, $\Pr[G_4 = 1] = \Pr[G_3 = 1]$. This holds for both anonymity and unlinkability.

Games G_5, G_6, G_7 reverse the hybrids for the second ownership choice, restoring signature and pseudonym pairs, randomized witnesses, and honest proofs. Thus, G_7 is the second real experiment. Write N_U for the number of registered candidates, $Q_1^* \geq 1$ for the number of first H_1 input definitions, and $k_{\max}$ for the maximum disclosed attributes per showing. Each

direction uses at most $N_S(2k_{\max}+1)$ DDH replacements. Adaptive target selection and early base queries incur a guessing loss of at most $N_U Q_1^*$.

Hence, for either experiment,

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{priv}} \le 2\varepsilon_{\mathrm{ZK}} + 2N_U Q_1^* N_S(2k_{\max}+1)\varepsilon_{\mathrm{DDH}} + \varepsilon_{\mathrm{stat}}, \tag{9}$$

where $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{priv}} = 2|\Pr[b'=b] - 1/2|$, and $\varepsilon_{\mathrm{DDH}}$ bounds the constructed DDH advantages. Sampling differences and collisions satisfy $\varepsilon_{\mathrm{stat}} = O(\frac{Q_H^2 + N_S(k_{\max}+1)}{p})$, where Q_H bounds all random-oracle calls. All counts are polynomial, so both advantages are negligible. □

5.4 Security of Attribute-Level Revocation

Lemma 4 (Non-Membership Soundness) *A PPT adversary cannot produce a valid non-membership witness for a revoked tag, except with negligible probability.*

Proof Suppose an adversary $\mathcal{A}$ generates a valid non-membership witness $(A_{e,j}^*, B_{e,j}^*)$ for $\rho_j \in R_e$, satisfying $e(A_{e,j}^*, acc_e)e(B_{e,j}^*, \tilde{g}^{\tau+\rho_j}) = e(g,\tilde{g})$. Since $\rho_j \in R_e$, $f_e(X) = (X+\rho_j)F_j(X)$, where $F_j(X) = \prod_{\rho\in R_e\setminus\{\rho_j\}}(X+\rho)$. If $\tau+\rho_j = 0$, both pairing terms on the left equal 1_{G_T}. This contradicts $e(g,\tilde{g}) \neq 1_{G_T}$. Otherwise, substituting $acc_e = \tilde{g}^{f_e(\tau)}$ gives $[e(A_{e,j}^*, \tilde{g}^{F_j(\tau)})e(B_{e,j}^*, \tilde{g})]^{\tau+\rho_j} = e(g,\tilde{g})$. Thus, $\Omega = e(A_{e,j}^*, \tilde{g}^{F_j(\tau)})e(B_{e,j}^*, \tilde{g}) = e(g,\tilde{g})^{1/(\tau+\rho_j)}$. Since $\deg F_j \le L-1$, the public SRS can compute $\tilde{g}^{F_j(\tau)}$. Hence, Ω solves the assumed bilinear inverse-exponent problem. This contradicts the accumulator security assumption. Therefore, the adversary's success probability is negligible. □

Lemma 5 (Witness Update Correctness) *Both full recomputation and incremental updating generate valid non-membership witnesses for unrevoked attributes.*

Proof Full recomputation sets $a_{e,j} = f_e(-\rho_j)^{-1}, b_{e,j}(X) = \frac{1-a_{e,j}f_e(X)}{X+\rho_j}$. Thus, $a_{e,j}f_e(X) + b_{e,j}(X)(X+\rho_j) = 1$. For incremental updating, $f_e(X) = f_{e-1}(X)u(X)$, where $u(X) = \prod_{\rho\in\Delta R}(X+\rho)$. If $\rho_j \notin R_e$, then $u(-\rho_j) \neq 0$. Hence, $a_{e,j} = a_{e-1,j}u(-\rho_j)^{-1} = f_{e-1}(-\rho_j)^{-1}u(-\rho_j)^{-1} = f_e(-\rho_j)^{-1}$. The updated coefficient $b_{e,j}(X)$ is computed from the same polynomial identity. Both methods therefore satisfy $e(A_{e,j}, acc_e)e(B_{e,j}, \tilde{g}^{\tau+\rho_j}) = e(g,\tilde{g})$. Thus, incremental updating preserves witness correctness. □

Theorem 4 (Attribute Revocation Security) *A PPT adversary cannot produce an accepted showing for an attribute revoked in the authenticated state used by the verifier, except with negligible probability.*

Proof Consider an accepted token disclosing an attribute (j, m_j) revoked for its certified holder in state ste_e. Lemma 1 extracts $\mu, \rho_j, z_j, t_{1,j}, t_{3,j}, \delta_j$. By Theorem 2, $\rho_j = \mathrm{H}_A(\mu\|j\|m_j) \in R_e$, except with negligible probability. The extracted relations satisfy $\Gamma_j = \tilde{g}^{\rho_j}h_2^{z_j}$, $\delta_j = t_{3,j}z_j$. Define $A_{e,j}^* = A_{2,j}h_1^{-t_{1,j}}$, $B_{e,j}^* = B_{2,j}h_1^{-t_{3,j}}$. Substitution into the extracted K_j relation gives $e(A_{e,j}^*, acc_e)e(B_{e,j}^*, D_j) = e(g,\tilde{g})e(B_{e,j}^*, h_2)^{z_j}$. Since

$D_j = \tilde{g}^{\tau+\rho_j} h_2^{z_j}$, cancellation yields $e(A^*_{e,j}, acc_e)e(B^*_{e,j}, \tilde{g}^{\tau+\rho_j}) = e(g, \tilde{g})$. Thus, $(A^*_{e,j}, B^*_{e,j})$ is a valid non-membership witness for $\rho_j \in R_e$, contradicting Lemma 4. Using another tag requires an unauthorized signature message or a hash collision by Theorem 2. Successful revoked-attribute showing therefore has negligible probability. □

Theorem 5 (Attribute-Level Revocation Correctness) *Revoking an attribute (j, m_j) prevents its showing across credentials issued under the same registered holder secret μ, without invalidating other unrevoked attributes.*

Proof The revocation tag is $\rho_j = \mathrm{H}_A(\mu \| j \| m_j)$. The same attribute uses a fixed identifier j across credentials. Thus, credentials issued under the same μ assign the same tag to (j, m_j). Adding ρ_j to R_e revokes this attribute across these credentials. For another attribute (k, m_k) with $\rho_k \notin R_e$, $a_{e,k} = f_e(-\rho_k)^{-1}$ remains well-defined. By Lemma 5, a valid non-membership witness can be generated for this attribute. Therefore, revocation affects the targeted attribute without invalidating other unrevoked attributes. □

5.5 Revocation Authorization and Traceability

Theorem 6 (Revocation State Integrity) *An adversary cannot cause the public ledger to accept an unauthorized revocation state, except with negligible probability.*

Proof Each revocation transition is bound to the authorization digest $d_R = \mathrm{H}(\mathrm{H}(pp)||apk||rpk||reqCom||e||acc_e||dig_e||(e+1)||acc_{e+1}||dig_{e+1}||\mathrm{H}(\mathrm{sort}(R_J))||u(X))$. The MA obtains authorization shares from a threshold set T and aggregates them into $\sigma_{\mathrm{rev}} = \prod_{i \in T} \sigma_{\mathrm{rev},i}^{\lambda_i}$. The resulting signature satisfies $e(\sigma_{\mathrm{rev}}, \tilde{g}) = e(\mathrm{H}_1(d_R), rpk)$. Each participating issuer authorizes only a valid revocation request and state transition. The adversary cannot recover the threshold authorization key during the security challenge. An unauthorized transition with a new authorization digest therefore requires a forgery of the threshold authorization signature, except with negligible hash-collision probability. The ledger also checks $R_{e+1} = R_e \cup R_J$, $f_{e+1}(X) = f_e(X)u(X)$, $acc_{e+1} = \tilde{g}^{f_{e+1}(\tau)}$, and $dig_{e+1} = \mathrm{H}(\mathrm{sort}(R_{e+1}))$. The updated state is signed by the MA: $\sigma_{st,e+1} = \mathrm{H}_1((e+1)||acc_{e+1}||dig_{e+1})^{ask}$. The ledger verifies this signature before publication. Therefore, the adversary cannot modify an authorized revocation state or introduce an unauthorized transition without breaking the signature security or the assumed integrity of the ledger. □

Theorem 7 (Authorized Traceability) *Every accepted showing token is traceable to its registered pseudonym holder by the trusted MA, except with negligible probability.*

Proof An accepted token contains a certified pair satisfying $e(Cert_\kappa, \tilde{g}) = e(\mathrm{H}_1(P_u \| \kappa), apk_2)$. The MA certifies only registered pseudonym pairs and retains their tracing records. An accepted pair absent from these records gives a certificate forgery. Otherwise, the private assignment identifies its registered holder *id*. Except for a hash collision, $TraceTable[\mathrm{H}(P_u \| \kappa)] = id$. The registered secret satisfies $P_u = Q_\kappa^{\mu_{id}}$. For a newly generated statement, Lemma 1 extracts μ^* satisfying $P_u = Q_\kappa^{\mu^*}$. Since $Q_\kappa \neq 1_{G_1}$ has order p, $\mu^* = \mu_{id}$

Table 2 Comparison of Security Properties

Scheme	Threshold Issuance	Threshold Revocation	Anonymity	Unlinkability	Unforgeability	Traceability	Selective Disclosure	Fine-grained Revocation
[30]	√	×	√	√	√	NA	√	×
[12]	×	×	√	√	√	NA	√	credential-level
[41]	×	×	√	√	√	√	×	×
[19]	×	×	NA	√	√	√	×	×
[31]	√	√	√	√	√	√	√	credential/user level
[35]	×	×	√	√	√	√	partial VC attribute	VC-level
[25]	×	×	√	√	√	×	√	×
STAR-AC	√	√	√	√	√	√	√	attribute level

(mod p). The same secret appears in all disclosed signature relations. Theorem 2 binds these signatures to authorized issuance records for that secret. Copied honest tokens retain the same binding. The MA therefore returns the registered pseudonym holder. A missing or inconsistent result requires a certificate forgery, a collision, or a violation of the established proof or signature properties. Each event has negligible probability. □

Overall Security

Theorem 8 (Security of STAR-AC) *Under the assumptions and security models in Section 3.3, STAR-AC provides showing correctness, attribute authenticity, showing anonymity and unlinkability against external adversaries, attribute-level revocation security and correctness, revocation state integrity, and authorized traceability to registered pseudonym holders, except with negligible probability. Revocation security applies to the authenticated state used by the verifier.*

Proof Theorem 1 establishes showing correctness. Lemmas 1 and 2 establish knowledge soundness and zero-knowledge of the joint showing proof. Theorem 2 establishes attribute authenticity. Under the stated privacy conditions, Theorem 3 establishes showing anonymity and unlinkability through joint-view simulation and DDH hybrids. Lemmas 4 and 5 establish non-membership soundness and witness update correctness. Theorems 4 and 5 establish attribute-level revocation security and correctness. Theorems 6 and 7 establish revocation state integrity and authorized traceability to registered pseudonym holders. Each failure event has negligible probability under its stated assumptions. Their union remains negligible over polynomially many protocol executions. Thus, STAR-AC satisfies the stated properties within the defined security models. □

6 Performance Analysis and Evaluation

6.1 Theoretical Analysis and Comparison

In this subsection, we comprehensively analyze and compare STAR-AC in terms of functional features, security properties, and theoretical overhead.

6.1.1 Functional Features and Security Properties

Table 2 compares the security properties of STAR-AC with those of existing schemes. Most schemes support anonymity, unlinkability, and unforgeability. However, their

Table 3 Comparison of Communication and Computational Overheads

Scheme	Communication Complexity	Computational Complexity	
	Credential Showing	Show Credential	Verify Credential
[30]	$(5q+4)\|G_1\| + 4(q+1)\|G_2\| + \|G_T\| + (2q+5)\|\mathbb{Z}_p\|$	$(3q+k+8)t_{e1} + (5q+6)t_{e2} + 3t_p$	$2(q+1)t_{e1} + 4t_{e2} + 4(q+2)t_p$
STAR-AC	$(5k+2)\|G_1\| + k\|G_2\| + (6k+4)\|\mathbb{Z}_p\| + \|I\|$	$(10k+2)t_{e1} + (5k+2)t_{e2} + (3k+2)t_p + 2kt_{eT}$	$(11k+2)t_{e1} + (5k+3)t_{e2} + (5k+4)t_p + t_{eT}$

support for threshold issuance, threshold revocation, traceability, selective disclosure, and fine-grained revocation varies.

Scheme [12] supports threshold issuance and selective disclosure but does not support threshold revocation or fine-grained revocation. Scheme [41] provides credential-level revocation, while [31] supports threshold issuance and revocation at the credential or user level. Scheme [35] supports partial VC attribute disclosure and VC-level revocation. In contrast, STAR-AC supports both threshold issuance and threshold revocation while enabling attribute-level revocation. It also provides anonymity, unlinkability, unforgeability, traceability, and selective disclosure. Certified one-time pseudonyms support authorized identity tracing. Attribute-specific revocation tags allow individual attributes to be revoked without invalidating other valid attributes.

6.1.2 Theoretical Overhead

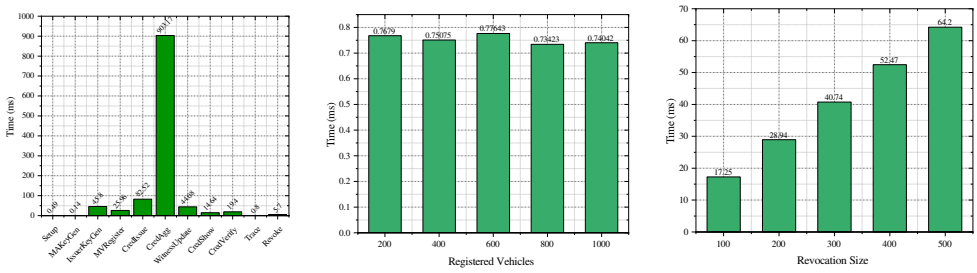
Table 3 compares the communication and computational overheads of [30] and STAR-AC during credential showing and verification. Here, $|G_1|$, $|G_2|$, $|G_T|$, and $|\mathbb{Z}_p|$ denote the sizes of elements in the corresponding groups and scalar field. The terms t_{e1}, t_{e2}, t_{eT}, and t_p denote the costs of exponentiation in G_1, G_2, and G_T, and bilinear pairing, respectively. Moreover, q denotes the total number of attributes in a credential, while k denotes the number of disclosed attributes. The term $|I|$ denotes the encoded size of the disclosed attribute index set.

As shown in Table 3, the communication overhead of [30] increases linearly with q. In contrast, STAR-AC requires $(5k+2)|G_1| + k|G_2| + (6k+4)|\mathbb{Z}_p| + |I|$ for credential showing. Its communication cost mainly depends on k, since only the randomized signatures, revocation tags, and non-revocation proofs of disclosed attributes are transmitted.

For computation, the showing and verification costs of [30] increase with q, while those of STAR-AC increase with k. STAR-AC uses shared proof responses and optimized pairing computations to reduce repeated operations. Its computational overhead includes the generation and verification of attribute-level non-revocation proofs. Therefore, when k is much smaller than q, STAR-AC can achieve lower online communication and computational costs while supporting selective disclosure, authorized tracing, and attribute-level revocation.

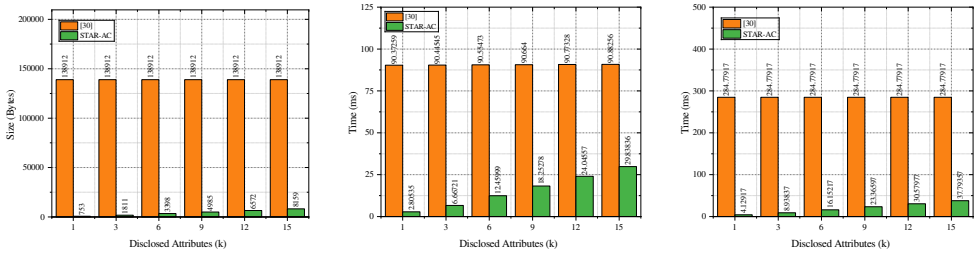
6.2 Experimental Analysis

To evaluate the performance of STAR-AC, we implemented the scheme in C++ using the BLS12-381 curve. The experiments were conducted on a desktop computer with an



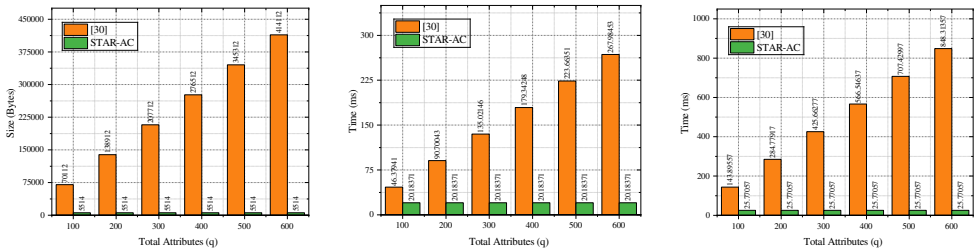
(a) Algorithm Runtime. (b) Trace Overhead. (c) Revocation Overhead.

Fig. 3 Computational overhead evaluation of STAR-AC.



(a) Communication Overhead. (b) Showing Overhead. (c) Verification Overhead.

Fig. 4 Performance comparison under a fixed total number of attributes ($q = 200$).



(a) Communication Overhead. (b) Showing Overhead. (c) Verification Overhead.

Fig. 5 Performance comparison under a fixed number of disclosed attributes ($k = 10$).

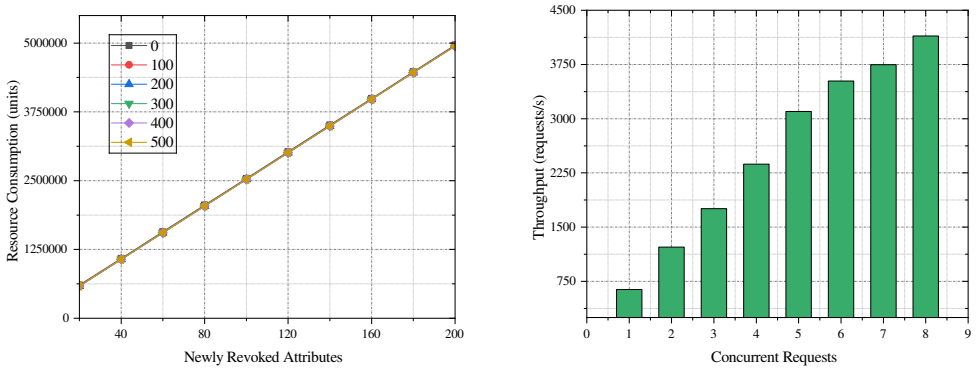
Intel Core i5-13600KF processor, 32 GB of RAM, and Ubuntu 20.04. The hash functions were implemented using SHA-256 and hash-to-curve operations. All credential related timings were averaged over 100 independent executions.

6.2.1 Algorithm Performance

Fig. 3 presents the computational performance of STAR-AC. As shown in Fig. 3(a), *CredAgg* requires the highest processing time of 903.17 ms, followed by CredIssue at 82.52 ms. In contrast, *CredShow* and *CredVerify* require only 14.64 ms and 19.40 ms,

respectively. The remaining operations also incur relatively low overhead. These results show that most computational costs are associated with credential preparation, while online showing and verification remain efficient.

Fig. 3(b) evaluates tracing performance under different registration scales. As the number of registered MVs increases from 200 to 1000, the tracing time remains between 0.73 ms and 0.78 ms. This demonstrates efficient identity tracing with low sensitivity to registration scale. Fig. 3(c) evaluates revocation performance. As the revocation size increases from 100 to 500, the processing time increases from 17.25 ms to 64.20 ms. The moderate growth indicates that STAR-AC can efficiently handle revocation operations involving multiple attributes.



(a) State update cost.

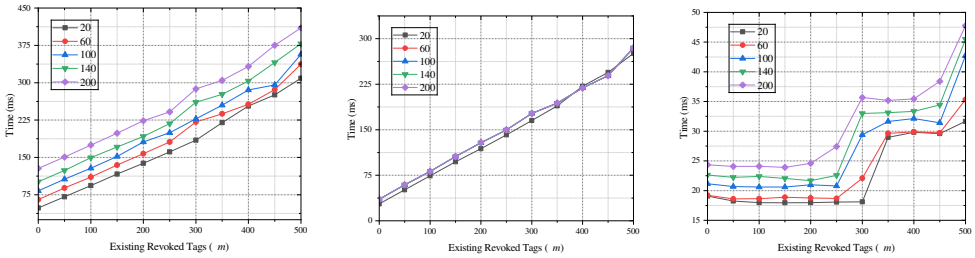
(b) State retrieval latency.

Fig. 6 Revocation state management performance.

6.2.2 Credential Showing and Verification Performance

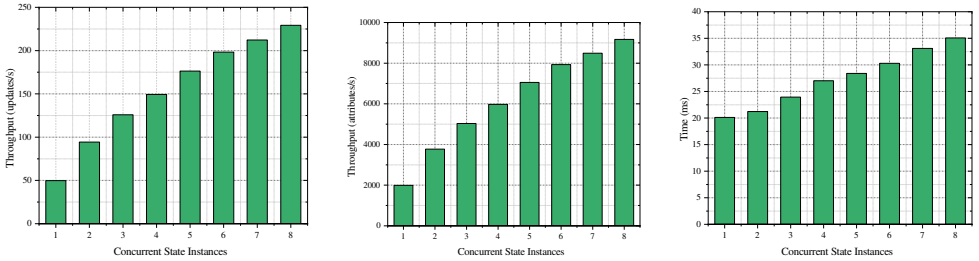
Fig. 4 compares STAR-AC with [30] under $q = 200$. As shown in Fig. 4(a), the communication overhead of STAR-AC increases from 753 to 8159 bytes as k increases from 1 to 15, remaining much lower than the 138912 bytes required by [30]. Fig. 4(b) shows that the showing time of STAR-AC increases from 2.81 to 29.84 ms, compared with approximately 90 ms for [30]. As shown in Fig. 4(c), its verification time remains below 40 ms, compared with approximately 284.78 ms for [30]. These results demonstrate the efficiency of STAR-AC in selective attribute disclosure, with significantly lower communication and computational overhead.

Fig. 5 compares the performance under $k = 10$. As shown in Fig. 5(a), STAR-AC maintains a constant communication overhead of 5514 bytes as q increases from 100 to 600, while [30] increases from 70112 to 414112 bytes. Fig. 5(b) and Fig. 5(c) show that the showing and verification times of STAR-AC remain stable at approximately 20.18 ms and 25.77 ms, respectively. At $q = 600$, [30] requires 267.98 ms and 848.31 ms, respectively. These results demonstrate the low overhead and favorable scalability of STAR-AC for selective attribute disclosure.



(a) End-to-end update latency. (b) Revocation processing time. (c) State update confirmation time.

Fig. 7 Revocation update performance.



(a) Update throughput. (b) Revocation throughput. (c) Update confirmation latency.

Fig. 8 Concurrent update performance.

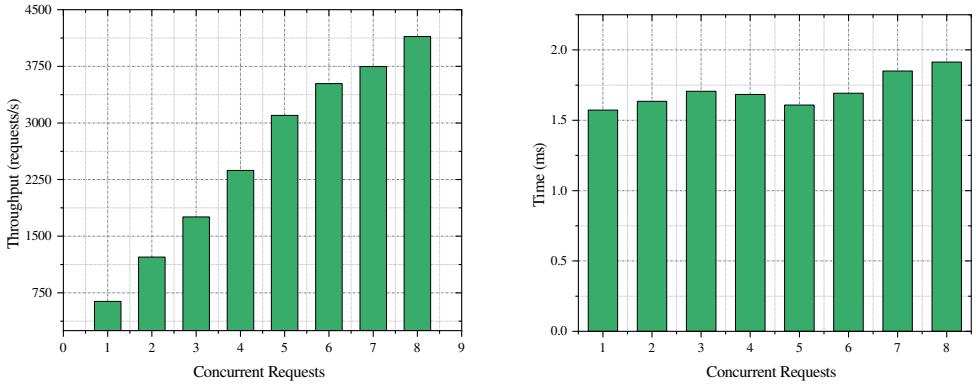
6.2.3 Revocation State Management Performance

Fig. 7 evaluates revocation update performance under different historical revocation sizes. As shown in Fig. 7(a), the end-to-end latency increases from 48.61 to 309.11 ms as m increases from 0 to 500 under $s = 20$, and from 127.84 to 409.99 ms under $s = 200$. Fig. 7(b) shows that local processing accounts for most of the update time, while Fig. 7(c) shows that the confirmation time remains below 48 ms. These results demonstrate efficient state update confirmation across different revocation sizes.

Fig. 6 evaluates revocation-state management performance. As shown in Fig. 6(a), resource consumption increases from approximately 0.60 million to 4.96 million units as s increases from 20 to 200 under $m = 0$, while remaining nearly unchanged across historical revocation sizes for a fixed s. Fig. 6(b) shows that state retrieval latency remains between 1.35 and 1.70 ms. These results demonstrate predictable update costs and efficient state retrieval.

6.2.4 Concurrent Ledger Performance

Fig. 8 evaluates concurrent updates to independent revocation states. As shown in Fig. 8(a) and Fig. 8(b), update throughput increases from 49.83 to 229.21 updates/s, while revocation throughput increases from 1993.10 to 9168.45 attributes/s. Fig. 8(c) shows that confirmation latency increases moderately from 20.11 to 35.06 ms. These results demonstrate efficient parallel processing of independent revocation updates.



(a) Retrieval throughput.

(b) Retrieval latency.

Fig. 9 Concurrent retrieval performance.

Fig. 9 evaluates concurrent state retrieval performance. As shown in Fig. 9(a), retrieval throughput increases from 636.65 to 4144.29 requests/s as concurrency increases from 1 to 8. Fig. 9(b) shows that retrieval latency remains low, ranging from 1.57 to 1.91 ms. These results demonstrate efficient concurrent state retrieval.

Overall, STAR-AC achieves low credential showing and verification overhead, efficient attribute revocation, and stable state retrieval. Its favorable communication and computational performance, together with scalable ledger throughput, supports its practical application in large-scale MIoT environments.

7 Concluding Remarks and Future Work

This paper proposed STAR-AC: traceable anonymous credentials with selective disclosure and attribute-level revocation for MIoT. STAR-AC integrates threshold credential issuance, efficient selective disclosure, attribute-level revocation, and authorized identity tracing. It protects vehicle identities and undisclosed attributes while preventing unauthorized attribute disclosure. Offline witness maintenance and incremental state updates reduce authentication and revocation overhead. Security analysis examines attribute authenticity, anonymity, unlinkability, revocation security, state integrity, and traceability under the stated assumptions. Experimental results demonstrate lower credential showing and verification costs than the compared scheme. The results also show efficient identity tracing, revocation-state maintenance, and concurrent ledger processing.

Future work will focus on further reducing witness update costs and improving authentication efficiency under intermittent maritime communication. Large-scale MIoT deployments will also be investigated to evaluate practical performance under dynamic network conditions.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant 62372168), National Key R&D Program of China (No. 2023YFC2811102), the Open Project of the Key Laboratory of Artificial Intelligence and Information Processing, Education Department of Guangxi Zhuang Autonomous Region (Hechi University) (Grant 2024GXZDSY010), Hunan Province Science and Technology Innovation Plan Project (No. 2023ZJ1020), and the Hunan Provincial Postgraduate Scientific Research Innovation Projects (Grants LXBZZ2024276 and CX20251547).

Data Availability Statement

This study does not rely on external datasets. All experimental data were randomly generated for performance evaluation.

Declaration of generative AI and AI-assisted technologies

During the preparation of this manuscript, the generative AI tool ChatGPT was used solely for language editing, grammar improvement, and stylistic revisions of the text. The tool was not used to generate scientific content, analyze data, or draw conclusions. The authors reviewed, verified, and take full responsibility for the finalized text and all scientific claims presented in this publication.

References

[1] Son, U., Huh, J.-H.: A survey of cyber security for maritime autonomous surface ships: Opportunities, challenges, and future directions. IEEE Transactions on Intelligent Transportation Systems **26**(6), 7343–7361 (2025)

[2] Chaudhary, M., Goyal, N., Benslimane, A., Awasthi, L.K., Alwadain, A., Singh, A.: Underwater wireless sensor networks: Enabling technologies for node deployment and data collection challenges. IEEE Internet of Things Journal **10**(4), 3500–3524 (2023)

[3] Hu, B., Wang, X., Wang, J., Liu, Y., Li, K., Tsang, K.F.: Hybrid-blockchain-based cross-domain authentication for internet of marine things. Computer Standards & Interfaces, 104140 (2026)

[4] Jahanbakht, M., Xiang, W., Hanzo, L., Azghadi, M.R.: Internet of underwater things and big marine data analytics—a comprehensive survey. IEEE Communications Surveys & Tutorials **23**(2), 904–956 (2021)

[5] Lin, B., Zhang, M., Jiang, Y., Qi, S., Yang, M., Wen, Z., Na, Z.: Harsh weather-oriented edge intelligence empowered maritime communication-computing converged network resource allocation. IEEE Internet of Things Journal **13**(8),

15334–15347 (2026)

[6] Zhang, S., Hu, B., Liang, W., Li, K.-C., Gupta, B.B.: A caching-based dual k-anonymous location privacy-preserving scheme for edge computing. IEEE Internet of Things Journal **10**(11), 9768–9781 (2023)

[7] Vangala, A., Agrawal, S., Das, A.K., Pal, S., Kumar, N., Lorenz, P., Park, Y.: Big data-enabled authentication framework for offshore maritime communication using drones. IEEE Transactions on Vehicular Technology **73**(7), 10196–10210 (2024)

[8] Ye, J., Cao, X., Xie, S.: Designing anonymous signature-based identity authentication scheme for ocean multilevel transmission. IEEE Internet of Things Journal **11**(13), 24310–24322 (2024)

[9] Zhang, S., Hu, B., Liang, W., Li, K.-C., Pathan, A.-S.K.: A trajectory privacy-preserving scheme based on transition matrix and caching for iiot. IEEE Internet of Things Journal **11**(4), 5745–5756 (2024)

[10] Liu, Z., Yao, N., Bai, S., Mai, T.: Priv2i: Privacy-preserving v2i authentication protocol with fine-grained access control. Ad Hoc Networks **183**, 104122 (2026)

[11] Wang, X., Xu, L., Zhou, L., Liu, Y., Xiong, N., Li, K.-C.: Large language model-driven probabilistic trajectory prediction in the internet of things using spatio-temporal encoding and normalizing flows. Digital Communications and Networks (2025)

[12] Sun, M., Lai, J., Wu, W., Yang, Y., Chu, C.-K., Deng, R.H.: How to securely delegate and revoke partial authorization credentials. IEEE Transactions on Dependable and Secure Computing **22**(2), 979–996 (2024)

[13] Wu, T., Li, G., Wang, J., Xiao, B., Song, Y.: Ppca: Privacy-preserving continuous authentication scheme with consistency proof for zero-trust architecture networks. IEEE Internet of Things Journal **12**(11), 17596–17609 (2025)

[14] Cao, C., Xu, C., Jiang, C., Zhang, Z., Dong, X., Chen, K.: Lpbt-sso: Password-based threshold single-sign-on authentication from lwe. IEEE Transactions on Dependable and Secure Computing **22**(1), 491–504 (2025)

[15] Wang, X., Zeng, P., Liu, G., Li, K.-C., Liu, Y., Hu, B., Palmieri, F.: A privacy-preserving certificate-less aggregate signature scheme with detectable invalid signatures for vanets. Journal of Information Security and Applications **89**, 104001 (2025)

[16] Liu, G., Li, H., Le, J., Wang, N., Liu, Y., Xiang, T.: Lrcpa: Lattice-based robust and conditional privacy-preserving authentication for vanets. IEEE Transactions on Vehicular Technology **74**(3), 4698–4712 (2025)

[17] Chen, Y., Zhang, J., Wei, X., Wang, Y., Cui, J.: Cross-domain authentication scheme for vehicles based on given virtual identities. IEEE Internet of Things Journal **11**(9), 15869–15879 (2024)

[18] He, G., He, T., Chen, R., Xu, B., Zhu, H., Zhang, L., Guo, N.: Multi-device collaborative authentication for internet of things. IEEE Internet of Things Journal **12**(14), 27753–27768 (2025)

[19] Wijethilaka, S., Yadav, A.K., Braeken, A., Liyanage, M.: Blockchain-based secure authentication and authorization framework for robust 5g network slicing. IEEE transactions on network and service management **21**(4), 3988–4005 (2024)

[20] Jiang, C., Xu, C., Yang, G., Zhang, Z., Chen, J.: Device-enhanced password-based threshold single-sign-on authentication. IEEE Transactions on Information Forensics and Security **20**, 2006–2021 (2025)

[21] Wang, W., Han, Z., Zhu, Y., Gadekallu, T.R., Wang, W., Su, C.: Enhanced v2r authentication for vanets using group signatures and dynamic pseudonyms. IEEE Transactions on Intelligent Transportation Systems **26**(9), 14144–14153 (2025)

[22] Zhang, S., Yan, Z., Liang, W., Li, K.-C., Dobre, C.: Baka: Biometric authentication and key agreement scheme based on fuzzy extractor for wireless body area networks. IEEE Internet of Things Journal **11**(3), 5118–5128 (2024)

[23] Zhao, Y., Cai, J., Ying, Z., Li, W., Liu, X., Zhang, L.: Optimized authentication algorithm for privacy-preserving anonymous credentials using randomized aggregate signatures. Journal of King Saud University Computer and Information Sciences **38**(2), 42 (2026)

[24] Yang, Z., Qu, X., Chen, Z., Sun, G.: Coichain: Blockchain scheme for privacy data authentication in cross-organizational identification. Computer Communications **233**, 108054 (2025)

[25] Yang, Y., Ching, W.K., Huang, M., Innet, S., Yang, G., Pang, H., Deng, R.H.: Disc: Decentralized identity system with self-sovereign credential aggregation. IEEE Transactions on Information Forensics and Security **20**, 12871–12886 (2025)

[26] Chen, B., Guan, Q., Li, J., Wang, J., He, J., He, D., Xiang, T.: Dva-aclf: Designated verifier asymmetric anonymous credential scheme for low-altitude flight. IEEE Transactions on Network Science and Engineering **13**, 5752–5767 (2026)

[27] Zhang, Y., Zhou, X., Kong, J., Guo, C., Chen, Y., Qin, X.: An anonymous credential-based efficient staged dual-chain authentication protocol for satellite–earth links. Journal of King Saud University Computer and Information Sciences **37**(3), 30 (2025)

[28] Zhang, S., Yan, Z., Liang, W., Li, K.-C., Di Martino, B.: Bcae: A blockchain-based cross domain authentication scheme for edge computing. IEEE Internet of Things Journal **11**(13), 24035–24048 (2024)

[29] Shi, R., Feng, H., Yang, Y., Yuan, F., Li, Y., Pang, H.H., Deng, R.H.: Threshold attribute-based credentials with redactable signature. IEEE Transactions on Services Computing **16**(5), 3751–3765 (2023)

[30] Li, C., Ning, J., Xu, S., Lin, C., Li, J., Shen, J.: Dtacb: Dynamic threshold anonymous credentials with batch-showing. IEEE Transactions on Information Forensics and Security **19**, 7744–7758 (2024)

[31] Zhang, Q., Ren, B., Zhong, H., Wang, F., Zeng, M., Cui, J.: Distributed multi-attribute anonymous certificate management scheme with fine-grained revocation for unmanned aerial vehicles. IEEE Transactions on Dependable and Secure Computing **23**(3), 6859–6873 (2026)

[32] Qiu, S., Li, J., Di, X., Li, X., Wu, Y., Ibrahim, M.: Lightweight mutual authentication scheme based on blockchain for internet of medical things. IEEE Internet of Things Journal **12**(7), 8848–8861 (2024)

[33] Wijesundara, W., Lee, J.-S., Aloupogianni, E., Tith, D., Suzuki, H., Obi, T.: Didauth-iotfw: Decentralized firmware authentication for smart home iot devices using verifiable credentials. Internet of Things **34**, 101788 (2025)

[34] Kwon, D., Das, A.K., Park, Y.: A blockchain-based mutual authentication scheme with data sovereignty for iot-based healthcare digital twin environments. IEEE Transactions on Consumer Electronics **71**(4), 9401–9414 (2025)

[35] Zhang, Z., Ren, W., Zhang, X., Sun, Y., Zhu, T., Choo, K.-K.R.: A blockchain and signature based scheme for cross-domain authentication with decentralized identity. Computer Standards & Interfaces **94**, 103994 (2025)

[36] Liu, X., Wang, M., Jing, H., Zhang, R., Guo, Z.: A lightweight authentication scheme for vanets based on secgear. IEEE Transactions on Vehicular Technology **74**(7), 10789–10801 (2025)

[37] Xie, M., Chang, Z., Li, H., Min, G.: Basuv: A blockchain-enabled uav authentication scheme for internet of vehicles. IEEE Transactions on Information Forensics and Security **19**, 9055–9069 (2024)

[38] Pointcheval, D., Sanders, O.: Short randomizable signatures. In: Cryptographers' Track at the RSA Conference, pp. 111–126 (2016). Springer

[39] Papamanthou, C., Tamassia, R., Triandopoulos, N.: Authenticated hash tables based on cryptographic accumulators. Algorithmica **74**(2), 664–712 (2016)

[40] Boneh, D.: The decision diffie-hellman problem. In: International Algorithmic Number Theory Symposium, pp. 48–63 (1998). Springer

[41] Feng, X., Cui, K., Wang, L., Liu, Z., Ma, J.: Pbag: A privacy-preserving blockchain-based authentication protocol with global-updated commitment in iovs. IEEE Transactions on Intelligent Transportation Systems **25**(10), 13524–13545 (2024)