Non-transferable Blockchain-based Identity Authentication

Yuxia Fu
Institute of Information Engineering

Jun Shao
Zhejiang Gongshang University

Qingjia Huang
School of Cyber Security, University of Chinese Academy of Sciences

Qihang Zhou
Institute of Information Engineering

Huamin Feng
Beijing Electronic Science and Technology Institute

Xiaoqi Jia (✉ jiaxiaoqi@iie.ac.cn)
Institute of Information Engineering

Ruiyi Wang
Institute of Information Engineering

Wenzhi Feng
School of Cyber Security, University of Chinese Academy of Sciences

Research Article

Keywords: Identity authentication, Privacy preservation, Single Sign-On, Ring signature, Diffie-Hellman key exchange

Posted Date: January 23rd, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2482221/v1

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Peer-to-Peer Networking and Applications on May 8th, 2023. See the published version at https://doi.org/10.1007/s12083-023-01481-1.

Non-transferable Blockchain-based Identity Authentication

Yuxia Fu[1,2], Jun Shao[3], Qingjia Huang[1,2], Qihang Zhou[1,2], Huamin Feng[4], Xiaoqi Jia[1,2*], Ruiyi Wang[1,2] and Wenzhi Feng[1,2]

[1*]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, 100080, China.

[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, 100093, China.

[3]Zhejiang Gongshang University, Zhejiang, 310018, China.

[4]Beijing Electronic Science & Technology Institute, Beijing, 100070, China.

*Corresponding author(s). E-mail(s): jiaxiaoqi@iie.ac.cn;
Contributing authors: fuyuxia@iie.ac.cn; chn.junshao@gmail.com; huangqingjia@iie.ac.cn; zhouqihang@iie.ac.cn; oliver_feng@yeah.net; wangruiyi@iie.ac.cn; fengwenzhi@iie.ac.cn;

Abstract

Due to the identification functionality, identity authentication is the first and primary security step in many information systems. There exist many works dedicated to giving secure identity authentication. However, most of the existing schemes suffer from at least one of the following problems: heavy account management, single point of failure, and privacy leakage. To tackle these challenges, we propose two blockchain-based identity authentication schemes in this paper. One is based on the famous Diffie-Hellman key exchange protocol and is efficient but with user-verifier interaction. The other utilizes the ring signature, which is non-interactive with a small computational cost. Besides the traditional security properties, such as unforgeability and identity anonymity, our proposed schemes can hold non-transferability, i.e., the verifier cannot prove the user's identity authentication to any third party. At last, the extensive experimental results demonstrate that our proposals are practical and efficient.

Keywords: Identity authentication, Privacy preservation, Single Sign-On, Ring signature, Diffie-Hellman key exchange

1 Introduction

Due to the identification functionality, identity authentication is the first and primary security step in many information systems. According to a report from Grand View Research [1], the global identity verification market was valued at USD 8.48 billion in 2021 and is expected to expand at a compound annual growth rate (CAGR) of 16.7% from 2022 to 2030. Many academic and industry works [2–9] have been dedicated to securing identity authentication. The existing schemes can be roughly classified into four categories: the isolated user identity (SILO) model, the federated model, the user-centric model, and the self-sovereign identity (SSI) model.

In the SILO model [5, 12, 13], the user cannot use the identity from one service provider to authenticate himself/herself to other service providers. With the proliferation of applications

Table 1 Comparison between our proposal and existing identity authentication solutions

Properties	Shuaib et al. [10]	Mahmood et al. [11]	Chang et al. [8]	Jia et al. [12]	DH-based scheme	RS-based scheme
No online CA	✓	✗	✗	✓	✓	✓
Decentralized storage	✗	✗	✗	✗	✓	✓
Unforgeability	✓	✓	✓	✓	✓	✓
Identity anonymity	✓	✓	✗	✓	✓	✓
Non-transferability	✓	✗	✗	✗	✓	✓
Untraceability	✗	✓	✗	✓	✗	✓

and online services, users would be overwhelmed with multiple accounts, which also could increase the risk of identity leakage. To solve the problem, the federated model [11, 14, 15] was proposed. However, cross-service-provider authentication can only happen between trusted entities, and it needs to address the problem of a single point of failure. The user-centric identity model [8, 16–20] can realize the cross-service-provider authentication between any two service providers. Currently, the Single Sign-On (SSO) scheme, one user-centric identity authentication scheme, has been widely used in many systems, such as Amazon OpenID, Facebook, and Google [7]. Nevertheless, the user-centric identity model still has the single point of failure problem and may lead to some privacy issues. For example, identity providers can effortlessly get what services users access, which raises many privacy concerns. Some SSI models [10, 21–30] have recently been proposed based on blockchain technology. It aims to give users better control over their identity data, emphasizing data minimization and decentralized identity storage. However, it still has privacy problems. In particular, the service provider can prove the user's identity authentication to others. Once the adversary can collect enough information, he/she can infer the user's activity trace.

As mentioned above, almost all of the existing identity authentication schemes suffer from one of the following problems as least, including heavy identity management, a single point of failure, and privacy leakage. To alleviate these problems, we propose two new identity authentication schemes utilizing blockchain [31], Diffie-Hellman key exchange [32], and ring signature [33, 34]. In particular, we store (anonymous) credentials on the blockchain to simplify identity management and simultaneously remove the single point of failure. Furthermore, we employ the Diffie-Hellman key exchange and ring signature to disallow the verifier to prove the identity authentication to others. Table 1 gives the comparison between our proposal and some representative identity authentication schemes. In a nutshell, the main contributions of this work are as follows.

- We summarize the properties a competent identity authentication scheme should satisfy, including no online CA, decentralized storage, unforgeability, identity anonymity, non-transferability, and untraceability. See the details in Section 2.3.
- We propose two identity authentication schemes by integrating the blockchain technique with the Diffie-Hellman (DH) key exchange technique or the ring signature (RS) technique. Both proposed schemes can satisfy the first five properties, and the RS-based scheme can also satisfy the untraceability property.
- The detailed security analysis demonstrates that the DH-based and RS-based schemes satisfy the claimed properties. The extensive experimental results show that our proposals perform well in terms of feasibility and effectiveness.

The rest of this paper is organized as follows. In Section 2, we present the system and security models and identify our design goals. Then, we give some preliminary information, including blockchain, Diffie-Hellman key exchange, and ring signature in Section 3. Section 4 presents the details of our proposed schemes and security analysis, followed by performance evaluation in Section 5. Section 6 reviews the related works. In the end, Section 7 gives the conclusions of our paper.

2 System Model and Design Goal

In this section, we give our system model and security model, and identify our design goals.

2.1 System Model

As shown in Fig. 1, we in this paper consider a typical blockchain-based identity authentication scenario, which primarily consists of four types of entities: a certificate authority (CA), users, verifiers, and a blockchain system.

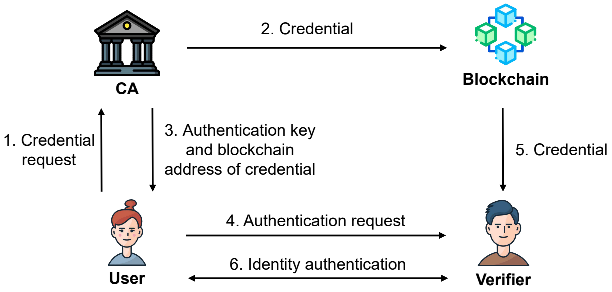
CA
2. Credential
Blockchain
1. Credential request
3. Authentication key and blockchain address of credential
5. Credential
4. Authentication request
6. Identity authentication
User
Verifier

Fig. 1 System model

- **CA**: The certificate authority (CA) [35] is the entity that signs and issues anonymous credentials, which contain attributes the corresponding user has but not revealing the user's identity. One simple attribute example is whether the user works for an organization. Furthermore, the CA records the anonymous credentials on the blockchain for validation, update, and revocation.
- **User**: The user is the owner of some anonymous credential on the blockchain. He/She can use the anonymous credential to prove his/her attribute to the verifier without revealing his/her identity.
- **Verifier**: The verifier could be a person, a service provider, or a system. He/She verifies whether the user has specific attributes according to the anonymous credential on the blockchain.
- **Blockchain**: The main functionality of the blockchain is to store anonymous credentials. In this paper, we mainly use the tamper resistance and time series of the blockchain.

2.2 Security Model

In our system model, due to the nature of the identity authentication scenario, we make the following assumptions:

- We assume that the users could be malicious. Particularly, the user would like to prove to the verifier that he/she does hold the attribute he/she does not have.
- We assume that the verifier could be malicious too. In particular, the verifier is interested in obtaining the user's identity. Furthermore, the verifier would like to sell the information of user's identity authentication to others.
- Based on the nature of CA, we simply assume that the CA is fully honest.
- As other blockchain-based systems [36, 37], we assume that the blockchain is semi-trusted. Specifically, the blockchain system will execute the protocol faithfully, but the nodes in the blockchain system would be interested in the user's identity and activity trace.
- We assume that one single credential does not reveal the identity of the corresponding user.

2.3 Design Goals

Based on the system model and security model, our design goal in this paper is to develop identity authentication satisfying the following properties.

- **No online CA**. To minimize the communication overhead and mitigate the single point of failure, the identity authentication scheme should allow the user and verifier to authenticate without relying on any online CA.
- **Decentralized storage**. To defend against the single point of failure, it would be better that the identity authentication scheme employ decentralized storage.
- **Unforgeability**. The basic security requirement for an identity authentication scheme is that anyone cannot forge any attribute he/she does not hold.
- **Identity anonymity**. Another basic security requirement for an identity authentication scheme is that anyone cannot get the identity information from the credentials or the interactions with the user.
- **Non-transferability**. To protect the user's privacy, we need non-transferability, i.e., the verifier cannot prove the identity authentication

to others. In this case, the (malicious) verifier cannot sell the information related to the identity authentication if the buyer does not trust the verifier.

- **Untraceability**. In some cases, non-transferability is insufficient if the information buyer trusts the verifier. Hence, we also require untraceability, i.e., the verifier can verify whether the user holds the attribute while not knowing the used credential.

3 Preliminaries

Before delving into the design of our proposed identity authentication schemes, we would like to review some basic knowledge related to the blockchain [31], Diffie-Hellman key exchange [32, 38], and ring signature [33].

3.1 Blockchain

Blockchain [31] is a technique that maintains a public ledger in a group of distributed entities that may not have a trust relationship. Due to the underlying consensus algorithm and cryptographic primitives, tamper resistance is the primary property of blockchain. In particular, the data cannot be modified once recorded on the blockchain. Furthermore, most existing blockchain systems can only support the data appending operation. In this case, the data is time-series. If there exist two records on the blockchain related to the same content, only the latter one is considered valid. In this paper, we mainly use the tamper resistance and time series of blockchain.

3.2 Diffie-Hellman Key Exchange

The Diffie-Hellman (DH) key exchange protocol [32, 38] is a method for two parties, Alice and Bob, to establish a shared secret over a public network. It proceeds as follows. At first, Alice and Bob agree on a finite cyclic group G generated by an element g with a big prime order q. Then, Alice (resp. Bob) chooses a random value a (resp. b) from Z_q^* and sends g^a (resp. g^b) to Bob (resp. Alice). At last, Alice (resp. Bob) can get the shared secret g^{ab} by using $(g^b)^a$ (resp. $(g^a)^b$).

3.3 Ring signature

Ring signature (RS) [39] is a special kind of digital signature that allows the creation of signatures on behalf of an ad hoc group of signers without revealing the signer. In this paper, we employ the scheme due to Abe et al. [40] as the underlying ring signature scheme in our proposal. The details are given as follows.

- `Setup`: On input a security parameter λ, the system setup algorithm `Setup` outputs the system parameter $\texttt{param} = (G, g, q, H)$, where G is a finite cyclic group generated by an element g with a big prime order q, and $H : \{0,1\} \rightarrow Z_q^*$ is a cryptographic-secure hash function.
- `KeyGen`: On input the system parameter $\texttt{param} = (G, g, q, H)$, the key generation algorithm `KeyGen` outputs a public/private key pair (y_i, x_i), where x_i is chosen from Z_q^* randomly and $y_i = g^{x_i}$.
- `Sign`: On input a group of n verifying keys $\texttt{PK} = \{y_{i_0}, \cdots, y_{i_{n-1}}\}$, a message m, and a signing key x_{i_k} $(0 \leq k \leq n-1)$, the signing algorithm `Sign` outputs a ring signature $\sigma = (t_0, s_0, ..., s_{n-1})$, where $t_{k+1} = H(\texttt{PK}\|m\|g^r)$, $t_j = H(\texttt{PK}\|m\|g^{s_{j-1}} y_{i_{j-1}}^{t_{j-1}})$ for $(j = k+2, \cdots, n-1, 0, \cdots, k)$, $s_k = r - x_{i_k} t_k \bmod q$, and $r, s_0, \cdots, s_{k-1}, s_{k+1}, \cdots, s_{n-1}$ are chosen randomly from Z_q^*.
- `Vrf`: On input a ring signature $\sigma = (t_0, s_0, ..., s_{n-1})$ on behalf of the verifying keys $\texttt{PK} = \{y_{i_0}, \cdots, y_{i_{n-1}}\}$ on message m, the verification algorithm `Vrf` outputs 1 if the following equation holds; otherwise, it outputs 0. $t_0 \overset{?}{=} H(\texttt{PK}\|m\|g^{s_{n-1}} y_{i_{n-1}}^{t_{n-1}})$, where $t_j = H(\texttt{PK}\|m\|g^{s_{j-1}} y_{i_{j-1}}^{t_{j-1}})$ for $j = 1, \cdots, n-1$.

4 Our Proposed Identity Authentication Scheme

In this section, we start with an interactive DH-based scheme that satisfies the first five properties mentioned in Section 2.3. After that, we offer an RS-based scheme satisfying all the properties listed in Section 2.3. We also give the analysis of these two schemes in this section.

4.1 Our DH-based Identity Authentication

We have four phases in the DH-based Identity Authentication scheme, namely System Setup (Setup) phase, Credential Issue (Issue) phase, Identity Authentication (Authen) phase, and Credential Update (Update) phase. Setup phase will generate the system parameter, and the CA will publish the anonymous credential on the blockchain for the corresponding user in Issue phase. With the anonymous credential, the verifier can authenticate the user in Authen phase. At last, the CA can update the credential in Update phase. For simplicity, we assume that entities communicate with each other via secure channels, which can be realized by TLS protocol [41].

4.1.1 The description

System Setup Phase

In Setup phase, the CA generates the system parameter $\mathtt{param} = (G, g, q, H)$, where G is a finite cyclic group generated by an element g with a big prime order q as that in the Diffie-Hellman key exchange protocol, and H is a cryptographic-secure hash function $H : \{0,1\}^* \rightarrow \{0,1\}^\ell$. Furthermore, the CA chooses a secret key k and generates a signing/verifying key pair $(sk_{\mathtt{ca}}, vk_{\mathtt{ca}})$ for a signature scheme.

Credential Issue Phase

In Issue phase, the CA firstly verifies the identity information $\mathtt{id}_i$ of the user U_i. Then, the CA checks whether U_i holds the attribute(s) he/she claims. If not, the CA aborts this phase. Otherwise, the CA continues to do the following steps. Particularly, the CA generates a transaction (credential)

$$\mathtt{tx}_{i_t} = y_{\mathtt{u},i_t} \| \mathtt{attr}_{i_t} \| S_{i_t} \| \sigma_{i_t},$$

where $y_{\mathtt{u},i_t} = g^{x_{\mathtt{u},i_t}}$, $x_{\mathtt{u},i_t}$ is a random element from Z_q^*, $\mathtt{attr}_{i_t}$ is the attribute(s) U_i claims, $S_{i_t} = H(\mathtt{id}_i \| y_{\mathtt{u},i_t} \| k)$, and σ_{i_t} is the CA's signature on $y_{\mathtt{u},i_t} \| \mathtt{attr}_{i_t} \| S_{i_t}$. After that, as shown in Fig. 2 the CA records $\mathtt{tx}_{i_t}$ on the underlying blockchain and sends the authentication key $x_{\mathtt{u},i_t}$ and the blockchain address $\mathtt{addr}_{i_t}$ of $\mathtt{tx}_{i_t}$ to U_i. It is worth mentioning that $\mathtt{attr}_{i_t}$ will not reveal $\mathtt{id}_i$, and each user can have multiple transactions (credentials).

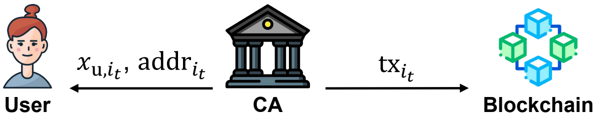

Fig. 2 Credential issue in the DH-based scheme

Identity Authentication Phase

When U_i needs to pass the identity authentication by the verifier V_j, they invoke Authen phase as follows.

- U_i sends the blockchain address $\mathtt{addr}_{i_t}$ of $\mathtt{tx}_{i_t}$ to the verifier V_j as the request for the identity authentication.
- Upon receiving the request from U_i, V_j retrieves $\mathtt{tx}_{i_t}$ from the underlying blockchain and checks its validity, including whether $\mathtt{tx}_{i_t}$ is not revoked[1], and whether σ_{i_t} is a valid CA's signature on $y_{\mathtt{u},i_t} \| \mathtt{attr}_{i_t} \| S_{i_t}$. If one of them does not pass, V_j refuses U_i's request. Otherwise, as shown in Fig. 3, V_j chooses a random u_{j_t} from Z_q^* and sends $v_{j_t} = g^{u_{j_t}}$ to U_i.
- On receiving v_{j_t} from V_j, U_i computes $R_t = v_{j_t}^{x_{\mathtt{u},i_t}}$ and sends it to V_j.
- After receiving R_t from U_i, V_j checks whether $R_t \overset{?}{=} y_{\mathtt{u},i_t}^{u_{j_t}}$ holds. If so, U_i passes the identity authentication by V_j; otherwise, V_j refuses U_i's request.

Note that R_t can be considered the shared key in the Diffie-Hellman key exchange protocol with the input $(y_{\mathtt{u},i_t}, v_{j_t})$.

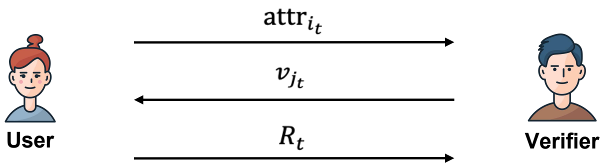

Fig. 3 Identity authentication of DH-based scheme

Credential Update Phase

When the user's attribute changes or the authentication key x is lost or revealed, the CA or the

[1] We will give more details in Update phase

user should update the corresponding transaction. We have the two following cases in Update phase.

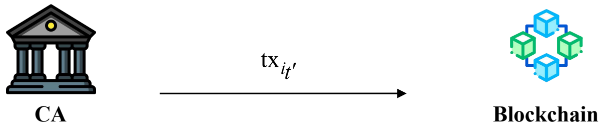

Fig. 4 Credential revocation in the DH-based scheme

- In the first case, the CA updates the credential (transaction) proactively. If the transaction $\texttt{tx}_{i_t}$ needs to be revoked, as shown in Fig. 4, the CA generates a transaction $\texttt{tx}_{i_{t'}} = \texttt{revoke} \| \texttt{addr}_{i_t} \| \sigma_{i_{t'}}$, where $\texttt{revoke}$ is the label of revocation, $\texttt{addr}_{i_t}$ is the blockchain address of $\texttt{tx}_{i_t}$, $\sigma_{i_{t'}}$ is the CA's signature on $\texttt{revoke} \| \texttt{addr}_{i_t}$. Once the transaction $\texttt{tx}_{i_{t'}}$ is recorded on the blockchain, $\texttt{tx}_{i_t}$ will be revoked. If $\texttt{tx}_{i_t}$ needs to be updated but not revoked, as shown in Fig. 5 the CA will also generate another transaction $\texttt{tx}'_{i_t} = y'_{u,i_t} \| \texttt{attr}'_{i_t} \| S'_{i_t} \| \sigma'_{i_t}$, where $y'_{u,i_t} = g^{x'_{u,i_t}}$, x'_{u,i_t} is a random element from Z_q^*, $\texttt{attr}'_{i_t}$ is the new attribute(s), $S'_{i_t} = H(\texttt{id}_i \| y'_{u,i_t} \| k)$, and σ'_{i_t} is the CA's signature on $y'_{u,i_t} \| \texttt{attr}'_{i_t} \| S'_{i_t}$. At last, the CA records $\texttt{tx}_{i_{t'}}$ and $\texttt{tx}'_{i_t}$ on the blockchain and sends the new authentication key x'_{u,i_t} to the corresponding user via a secure channel along with the blockchain address $\texttt{addr}'_{i_t}$ of $\texttt{tx}'_{i_t}$.
- In the second case, the CA updates the credential (transaction) on the user's request. In particular, the CA verifies the identity information $\texttt{id}_i$ and the new claimed $\texttt{attr}'_{i_t}$ (if the request is not a revocation one). After that, the user sends the blockchain address $\texttt{addr}_{i_t}$ of the transaction (credential) $\texttt{tx}_{i_t}$. The CA checks whether $S_{i_t} \overset{?}{=} H(\texttt{id}_i \| y_{\texttt{u},i_t} \| k)$ holds. If not, the CA aborts this phase. Otherwise, the CA generates transactions and sends necessary data to the user, as in the first case.

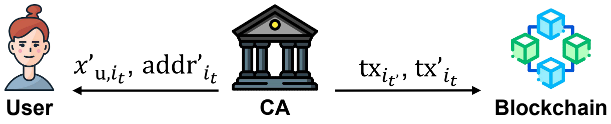
User
x'_{u,i_t}, addr'_{i_t}
CA
$\mathrm{tx}_{i_{t'}}$, tx'_{i_t}
Blockchain

Fig. 5 Credential update in the DH-based scheme

In order to protect the relationship information between $\texttt{tx}_{i_{t'}}$ and $\texttt{tx}'_{i_t}$, the CA should send $\texttt{tx}'_{i_t}$ to the blockchain with other transactions (credentials).

4.1.2 Analysis of the DH-based Scheme

In the following, we will analyze the properties of the above DH-based identity authentication scheme one by one.

No online CA

According to the description of the DH-based identity authentication scheme, the user and the verifier can execute Authen phase only with the help of the blockchain but without any interaction with the CA. Therefore, the DH-based identity authentication scheme holds the property of having no online CA.

Decentralized storage

According to the description of the DH-based identity authentication scheme, all the credentials are recorded on the blockchain, which is a decentralized system. Therefore, the DH-based identity authentication scheme holds the property of decentralized storage.

Unforgeability

We have three parts for the analysis of the unforgeability property.

- First, anyone except the CA cannot generate valid signatures in the transactions, according to the unforgeability of the underlying signature scheme.
- Second, without knowing $x_{\texttt{u},i_t}$ or u_{j_t}, no one can generate R_t with the input $y_{\texttt{u},i_t}$ and v_{j_t}, based on the CDH assumption[2], which follows the security of the Diffie-Hellman key exchange protocol.
- Third, the user cannot abuse the update functionality. In particular, the CA checks whether the user holds $\texttt{id}$ and the claimed new $\texttt{attr}'$ before updating $\texttt{tx} = y \| \texttt{attr} \| S \| \sigma$. In other words, the user cannot update $\texttt{tx}$, even if he/she holds $\texttt{attr}$ but is not corresponding to the identity $\texttt{id}$.

[2] Given (g, g^a, g^b), it is intractable to compute g^{ab}.

Therefore, the DH-based identity authentication scheme holds the unforgeability property.

Identity anonymity

We have two parts for the analysis of the property of identity anonymity.

- First, no one can get the identity information from single credential (transaction). Recall the content of the credential (transaction) $\mathtt{tx} = y\|\mathtt{attr}\|S\|\sigma$. It is easy to see that y and σ won't reveal any information about the identity. Furthermore, we assume that attr does not reveal the identity either. Regarding $S = H(\mathtt{id}\|y\|k)$, anyone who does not know k cannot obtain any information about id due to the security of the underlying hash function.
- Second, multiple attr's related to the same user may reveal the identity information. However, this attack works only if multiple tx's can be linked. It is easy to see that y, attr and σ cannot be used to link different tx. Regarding $S = H(\mathtt{id}\|y\|k)$, it is useless either due to the security of the underlying hash function. Furthermore, no one can link attr and its new version $\mathtt{attr}'$, since $\mathtt{attr}'$ is distributed with other transactions.

Therefore, the DH-based identity authentication scheme holds the property of identity anonymity.

Non-transferability

To show the non-transferability of the proposed DH-based scheme, we need to show that the verifier can also generate the authentication data. According to the description of the DH-based scheme, the authentication data R_t can be generated by the user or the verifier alone. Therefore, the DH-based identity authentication scheme holds the property of identity anonymity.

Untraceability

Though the verifier cannot prove to others that the user has executed the identity authentication with him/her, he/she knows which transaction (credential) the user used. As a result, several verifiers interacting with the same user can conspire to obtain the user's activity trace. Therefore, the DH-based identity authentication scheme doesn't hold the property of untraceability.

4.2 Our RS-based Identity Authentication

In this part, we will propose RS-based identity authentication scheme that is non-interactive and untraceability. Like the DH-based scheme, we also have phases Setup, Issue, Authen, and Update in the RS-based scheme.

4.2.1 The description

System Setup Phase

In Setup phase, the CA generates the system parameter $\mathtt{param} = (G, g, q, H)$ by running RS.Setup. Furthermore, every verifier also runs RS.KeyGen to get the signing/verifying key pair $(x_{\mathtt{v},i}, y_{\mathtt{v},i})$.

Credential Issue Phase

It is almost the same as that in the DH-based scheme, except the generation of $(x_{\mathtt{u},i_t}, y_{\mathtt{u},i_t})$. In particular, $(x_{\mathtt{u},i_t}, y_{\mathtt{u},i_t})$ is the signing/verifying key pair generated from RS.KeyGen.

Identity Authentication Phase

When U_i wants to use the transaction (credential) $\mathtt{tx}_{i_t} = y_{\mathtt{u},i_t}\|\mathtt{attr}_{i_t}\|S_{i_t}\|\sigma_{i_t}$ to prove to the verifier V_j that he/she holds the specific attribute(s) $\mathtt{attr}^* \subseteq \mathtt{attr}_{i_t}$, U_i does the following steps.

- U_i randomly chooses $n-2$ unrevoked credentials (transactions) $\{\mathtt{tx}_0, \cdots, \mathtt{tx}_{n-3}\}$ on the blockchain, where all the corresponding attributes satisfy $\mathtt{attr}_m \supseteq \mathtt{attr}^*$ for $(0 \leq m \leq n-3)$.
- U_i sorts $(y_{\mathtt{u},0}, \cdots, y_{\mathtt{u},n-3}, y_{\mathtt{u},i_t}, y_{\mathtt{v},j})$ as the ascending order to get $\mathtt{PK} = \{y_{i_0}, \cdots, y_{i_{n-1}}\}$, where $y_{\mathtt{u},m}$ is the corresponding value in $\mathtt{attr}_m$ for $(0 \leq m \leq n-3)$, and $y_{\mathtt{v},j}$ is V_j's verifying key.
- U_i runs RS.Sign with the input $(\mathtt{PK}, \mathtt{request}, x_{\mathtt{u},i_t})$, where $x_{\mathtt{u},i_t}$ is the authentication key corresponding to $\mathtt{tx}_{i_t}$. After that, U_i can get the ring signature $\sigma_{\mathtt{u},t}$.
- As shown in Fig. 6, U_i sends $\sigma_{\mathtt{u},t}$ to V_j along with a set of blockchain address $\{\mathtt{addr}_0, \cdots, \mathtt{addr}_{n-3}, \mathtt{addr}_{i_t}\}$ and the authentication request request, where $\{\mathtt{addr}_0, \cdots, \mathtt{addr}_{n-3}, \mathtt{addr}_{i_t}\}$ are corresponding to $\{\mathtt{tx}_0, \cdots, \mathtt{tx}_{n-3}, \mathtt{tx}_{i_t}\}$.

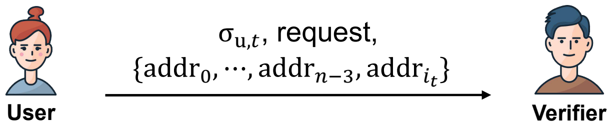

Fig. 6 Identity authentication of RS-based scheme

Upon receiving the data from U_i, V_j does the following steps.

- V_j retrieves $\{\mathtt{tx}_0, \cdots, \mathtt{tx}_{n-3}, \mathtt{tx}_{i_t}\}$ from the blockchain according to $\{\mathtt{addr}_0, \cdots, \mathtt{addr}_{n-3}, \mathtt{addr}_{i_t}\}$. If one of them is revoked, V_j aborts this phase; otherwise, he/she continues to do the following steps.
- V_j sorts $(y_{\mathrm{u},0}, \cdots, y_{\mathrm{u},n-3}, y_{\mathrm{u},i_t}, y_{\mathrm{v},j})$ as the ascending order to get $\mathtt{PK}' = \{y_{i_0}, \cdots, y_{i_{n-1}}\}$.
- V_j runs `RS.Vrf` with the input $(\mathtt{PK}', \mathtt{request}, \sigma_{\mathrm{u},t})$. If the output is 1, then the authentication succeeds; otherwise, the authentication fails.

Credential Update Phase

It is almost the same as that in the DH-based scheme, except the generation of $(x'_{\mathrm{u},i_t}, y'_{\mathrm{u},i_t})$. In particular, $(x'_{\mathrm{u},i_t}, y'_{\mathrm{u},i_t})$ is the signing/verifying key pair generated from `RS.KeyGen`.

4.2.2 Analysis of the RS-based Scheme

In the following, we will analyze the properties of the above RS-based identity authentication scheme one by one.

No online CA & Decentralized storage & Identity anonymity

The RS-based scheme is almost the same as the DH-based scheme, except for `Authen` phase. Hence, we can easily obtain the properties of no online CA, decentralized storage, and identity anonymity for the RS-based scheme.

Unforgeability

We have two parts for the analysis of unforgeability property.

- First, similar to the DH-based scheme, anyone except the CA cannot generate valid signatures in the transactions, and the user cannot abuse the update functionality.
- Second, without knowing the signing key corresponding to one of verifying keys in `PK`, on one can generate a valid ring signature on `request` on behalf of `PK`, due to the unforgeability of the underlying ring signature.

Therefore, the RS-based identity authentication scheme holds the property of unforgeability.

Non-transferability

Similar to the DH-based signature, we need to show that the verifier can also generate the authentication data. As we can see from the description of the RS-based scheme, the authentication data is $\sigma_{\mathrm{u},t}$. Meanwhile, according to the anonymity of the underlying ring signature, $\sigma_{\mathrm{u},t}$ can be generated by the user or the verifier alone from the view of others. Therefore, the RS-based identity authentication scheme holds the property of identity anonymity.

Untraceability

According to the anonymity of the underlying ring signature, no one can tell whether the same signer generates two ring signatures on behalf of the same group of verifying keys. In other words, the verifier cannot link any two identity authentications. Therefore, the RS-based identity authentication scheme holds the property of untraceability.

5 Performance Evaluation

In this section, we would like to evaluate our proposals' performance.

We evaluate the presented proposals in a 64 bit Window 10 operating system computer, which has a 16.0 GB RAM, Intel(R) Core(TM) i5-7300HQ CPU with 2.5 GHz frequency. And the other configurations are as shown in Table 2. Then, we present numerical consequences and discussions.

Table 2 Authentication system operating environment

Name	**Specification**
Cryptographic library	Bouncy Castle Crypto APIs (https://www.bouncycastle.org/)
Blockchain	TestNet
Elliptic curve	secp256k1

The initial evaluation of our schemes involves the phases of credential issue, identity authentication, and credential update. To eliminate the influence of the experimental environment episodes, we set up 100 authentication events for both schemes and recorded the corresponding average time. As illustrated in Fig. 7, the consumption times of the credential issue and credential update phases in the DH-based scheme and the RS-based scheme are from 2.58 ms to 2.7 ms, respectively, behaving well. During the identity authentication phase, the DH-based scheme costs 4.02 ms; meanwhile, the cost of the RS-based scheme increases as the number of public keys grows. The time consumed for identity authentication increases from 7.32 ms to 31.24 ms when the number of participating public keys increases from 1 to 10. It is well known that as the number of participating public keys increases, the scheme provides more robust security.

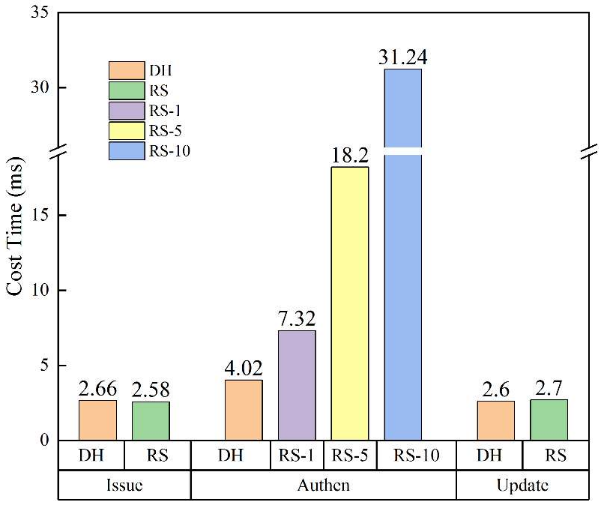

Fig. 7 Time consumed at each phase of identity authentication

Extensive experiments demonstrate the effectiveness of our algorithms, i.e., the DH-based algorithm is more prompt and is more affected by bandwidth because of its interactive authentication. Meanwhile, the RS-based algorithm can achieve higher safety with less communication.

6 Related Works

In this section, we mainly review identity authentication solutions, which can be roughly divided into four types, namely, the isolated user identity (SILO) model [5, 12, 13], federated [7, 11, 14, 15], user-centric [8, 16–20], and self-sovereign identity (SSI) [10, 21–30]. And the details are as follows:

Josang *et al.* [13] proposed an SILO model to manage the identity, in which, when a user wishes to access a service from different service providers (SPs), he/she must visit and authenticate with each SP separately. However, with the proliferation of applications and online services, users are overwhelmed with multiple accounts [5, 12] (along with their identifiers and corresponding credentials), which raises the need for user identity management solutions to reduce management complexity. Furthermore, each SP stores many user identities and has varying security protection capabilities, increasing the risk of information leakage.

Mahmood *et al.* [11] proposed a federated model to alleviate the above challenges. This model creates the centralized identity paradigm, where identity data is shared between trusted entities, such as government services and educational institutions. Therefore, it uses a centralized identity authentication model to enable users to access multiple services with a single set of credentials. Using a federated model helps reduce identity management complexity and relieves the risk of information leakage from varying security levels of services. However, the federated model can only be applied between trusted entities and suffers from a single point of failure.

Chang *et al.* [8] proposed the user-centric identity model to overcome the issue that the federated model can only be applied between trusted entities. In this model, the SPs and the IdPs may not always have a preexisting trust relationship, known as the Single Sign-On (SSO) system. Examples of this model are the use of third-party accounts, such as Amazon OpenID, Facebook, and Google, to access online services. This broadens the application of scenarios; moreover, the model can reduce identity management complexity. Nevertheless, the user-centric identity model is not as successful as expected. It not only suffers from a single point of failure but also leads to privacy issues. For example, the IdPs [8, 16–18] can effortlessly get what services have been accessed by users, which raises many privacy concerns. Moreover, the Cambridge Analytica scandal of misusing people's personal information from Facebook to influence voters in the US Elections [42] has raised serious concerns about private data collection and analysis by prominent online centralized SPs. This has led to the search for more secure and privacy-preserving schemes for digital identity management.

Yang *et al.* [21] proposed a blockchain-based SSI model to protect the privacy of the identity user. This model aims to provide users with better control over their identity data, with a strong emphasis on data minimization and decentralized identity storage. Especially blockchain holds great potential to facilitate a decentralized identity storage environment among untrusted entities. Hence, the model solves a single point of failure problem. However, the existing blockchain-based SSIs [22–30]are still in their infancy and have privacy problems, such as the fact that verifiers possess authentication interaction information and may transfer it to another person or a third party. It is fair to say that almost all of the existing identity authentication schemes can solve the above issues perfectly and efficiently. Different from these studies, we consider privacy-preserving, efficient, and distributed identity of authentication schemes while guaranteeing the requirements in subsection 2.3.

7 Conclusion

In this paper, we have proposed two blockchain-based identity authentication schemes utilizing Diffie-Hellman key exchange and ring signature, respectively. The detailed analysis and extensive experimental results demonstrate that our proposals are secure, efficient, and effective. Compared to the existing identity authentication schemes, our proposals can prevent the verifier (service provider) from proving the identity authentication to others.

Funding This work was supported by the following:

- Program of Key Laboratory of Network and Assessment Technology, the Chinese Academy of Sciences;
- Program of Beijing Key Laboratory of Network Security and Protection Technology;
- National Key Research and Development Program of China [2019YFB1005201];
- National Key Research and Development Program of China [2021YFB2910109];
- Strategic Priority Research Program of Chinese Academy of Sciences [XDC02010900];
- National Natural Science Foundation of China [61772078].

Data availability The source code of the proposal can be found at https://github.com/fuyx0305/experiment.

Declarations

Ethics approval This work does not involve any work related to ethics.

Consent to publish All authors consent to publication.

Conflicts of interest All authors declare that they have no potential competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author Contribution Miss. Yuxia Fu participated in all processes of the paper, including the original design and writing of the idea of non-transferable blockchain-based identity authentication and writing each part of the paper. Prof. Jun Shao participated in the original design and guided the writing of the paper. Mr. Qingjia Huang was the leading performer in the experimental part of the paper. Miss. Qihang Zhou organized and analyzed the experimental data. Prof. Huamin Feng and Prof. Xiaoqi Jia raised the problems of the existing identity authentication system. Miss. Ruiyi Wang designed the DH-based identity authentication process, and Mr. Wenzhi Feng designed the RS-based identity authentication process. In the final stage, all authors discussed and revised the content of each part of this paper.

References

[1] GVR Report coverIdentity Verification Market Size, Share and Trends Report Identity Verification Market Size, Share and Trends Analysis Report By Component, By Type, By Deployment Mode, By Organization Size, By Verticals, By Region, And Segment Forecasts, 2022 - 2030. https://www.grandviewresearch.com/industry-analysis/identity-verification-market-report

[2] Satybaldy, A., Nowostawski, M., Ellingsen, J.: Self-sovereign identity systems. In: IFIP International Summer School on Privacy and Identity Management, pp. 447–461 (2020). Springer

[3] Cheng, X., Zhang, Z., Chen, F., Zhao, C., Wang, T., Sun, H., Huang, C.: Secure identity authentication of community medical internet of things. IEEE Access **7**, 115966–115977 (2019)

[4] Norta, A., Matulevičius, R., Leiding, B.: Safeguarding a formalized blockchain-enabled identity-authentication protocol by applying security risk-oriented patterns. Computers & Security **86**, 253–269 (2019)

[5] Cao, Y., Yang, L.: A survey of identity management technology. In: 2010 IEEE International Conference on Information Theory and Information Security, pp. 287–293 (2010). IEEE

[6] Amor, A.B., Abid, M., Meddeb, A.: A privacy-preserving authentication scheme in an edge-fog environment. In: 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), pp. 1225–1231 (2017). IEEE

[7] Kurniawan, R.: Perancangan dan implementasi sistem otentikasi oauth 2.0 dan pkce berbasis extreme programming (xp). Jurnal Pendidikan dan Teknologi Indonesia **2**(2), 81–91 (2022)

[8] Chang, C.-C., Lee, C.-Y.: A secure single sign-on mechanism for distributed computer networks. IEEE Transactions on Industrial Electronics **59**(1), 629–637 (2011)

[9] Wang, J., Wu, L., Choo, K.-K.R., He, D.: Blockchain-based anonymous authentication with key management for smart grid edge computing infrastructure. IEEE Transactions on Industrial Informatics **16**(3), 1984–1992 (2019)

[10] Shuaib, M., Hassan, N.H., Usman, S., Alam, S., Bhatia, S., Agarwal, P., Idrees, S.M.: Land registry framework based on self-sovereign identity (ssi) for environmental sustainability. Sustainability **14**(9), 5400 (2022)

[11] Mahmood, K., Li, X., Chaudhry, S.A., Naqvi, H., Kumari, S., Sangaiah, A.K., Rodrigues, J.J.: Pairing based anonymous and secure key agreement protocol for smart grid edge computing infrastructure. Future Generation Computer Systems **88**, 491–500 (2018)

[12] Jia, X., He, D., Kumar, N., Choo, K.-K.R.: A provably secure and efficient identity-based anonymous authentication scheme for mobile edge computing. IEEE Systems Journal **14**(1), 560–571 (2019)

[13] Josang, A., AlZomai, M., Suriadi, S.: Usability and privacy in identity management architectures. In: ACSW Frontiers 2007: Proceedings of 5th Australasian Symposium on Grid Computing and e-Research, 5th Australasian Information Security Workshop (Privacy Enhancing Technologies), and Australasian Workshop on Health Knowledge Management and Discovery, pp. 143–152 (2007). Australian Computer Society

[14] Zhu, H., Hu, J., Chang, S., Lu, L.: Shakein: secure user authentication of smartphones with single-handed shakes. IEEE transactions on mobile computing **16**(10), 2901–2912 (2017)

[15] Khattak, Z.A., Sulaiman, S., Ab Manan, J.-L.: A study on threat model for federated identities in federated identity management system. In: 2010 International Symposium on Information Technology, vol. 2, pp. 618–623 (2010). IEEE

[16] Ghasemisharif, M., Kanich, C., Polakis, J.: Towards automated auditing for account and session management flaws in single sign-on deployments. In: 2022 IEEE Symposium on Security and Privacy (SP), pp. 1524–1524 (2022). IEEE Computer Society

[17] Karim, A., Adnan, M.A.: An openid based authentication service mechanisms for internet of things. In: 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), pp. 687–692 (2019).

IEEE

[18] Jøsang, A., Pope, S.: User centric identity management. In: AusCERT Asia Pacific Information Technology Security Conference, vol. 22, p. 2005 (2005). Citeseer

[19] El Maliki, T., Seigneur, J.-M.: A survey of user-centric identity management technologies. In: The International Conference on Emerging Security Information, Systems, and Technologies (SECUREWARE 2007), pp. 12–17 (2007). IEEE

[20] Jøsang, A., Pope, S.: User centric identity management. In: AusCERT Asia Pacific Information Technology Security Conference, vol. 22, p. 2005 (2005). Citeseer

[21] Yang, M., Zhu, T., Liang, K., Zhou, W., Deng, R.H.: A blockchain-based location privacy-preserving crowdsensing system. Future Generation Computer Systems **94**, 408–418 (2019)

[22] Ma, Z., Jiang, M., Gao, H., Wang, Z.: Blockchain for digital rights management. Future Generation Computer Systems **89**, 746–764 (2018)

[23] Li, H., Tian, H., Zhang, F., He, J.: Blockchain-based searchable symmetric encryption scheme. Computers &amp; Electrical Engineering **73**, 32–45 (2019)

[24] Ebrahimi, A.: Identity management service using a blockchain providing certifying transactions between devices. Google Patents. US Patent 9,722,790 (2017)

[25] Van Bokkem, D., Hageman, R., Koning, G., Nguyen, L., Zarin, N.: Self-sovereign identity solutions: The necessity of blockchain technology. arXiv preprint arXiv:1904.12816 (2019)

[26] Malik, N., Nanda, P., Arora, A., He, X., Puthal, D.: Blockchain based secured identity authentication and expeditious revocation framework for vehicular networks. In: 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), pp. 674–679 (2018). IEEE

[27] Shuaib, M., Hassan, N.H., Usman, S., Alam, S., Bhatia, S., Mashat, A., Kumar, A., Kumar, M.: Self-sovereign identity solution for blockchain-based land registry system: a comparison. Mobile Information Systems **2022** (2022)

[28] Stokkink, Q., Pouwelse, J.: Deployment of a blockchain-based self-sovereign identity. In: 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 1336–1342 (2018). IEEE

[29] Stokkink, Q., Ishmaev, G., Epema, D., Pouwelse, J.: A truly self-sovereign identity system. In: 2021 IEEE 46th Conference on Local Computer Networks (LCN), pp. 1–8 (2021). IEEE

[30] Kuo, T.-T., Ohno-Machado, L.: Modelchain: Decentralized privacy-preserving healthcare predictive modeling framework on private blockchain networks. arXiv preprint arXiv:1802.01746 (2018)

[31] Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system (2008)

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

[33] Backes, M., Dttling, N., Hanzlik, L., Kluczniak, K., Schneider, J.: Ring signatures: Logarithmic-size, no setup—from standard assumptions. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques (2019)

[34] Ateniese, G., Camenisch, J., Joye, M., Tsudik, G.: A practical and provably secure

coalition-resistant group signature scheme. In: Annual International Cryptology Conference (2000)

[35] wikioedia: Certificate authority (2022). Accessed Oct 21, 2022

[36] Zheng, H., Shao, J., Wei, G.: Attribute-based encryption with outsourced decryption in blockchain. Peer-to-Peer Networking and Applications **13**(5), 1643–1655 (2020)

[37] Guan, Y., Zheng, H., Shao, J., Lu, R., Wei, G.: Fair outsourcing polynomial computation based on the blockchain. IEEE Transactions on Services Computing **15**(5), 2795–2808 (2022)

[38] Abusukhon, A., Anwar, M.N., Mohammad, Z., Alghannam, B.: A hybrid network security algorithm based on diffie hellman and text-to-image encryption algorithm. Journal of Discrete Mathematical Sciences and Cryptography **22**(1), 65–81 (2019)

[39] Rivest, R.L., Shamir, A., Tauman, Y.: How to leak a secret. In: International Conference on the Theory and Application of Cryptology and Information Security, pp. 552–565 (2001). Springer

[40] Abe, M., Ohkubo, M., Suzuki, K.: 1-out-of-n signatures from a variety of keys. In: International Conference on the Theory and Application of Cryptology and Information Security, pp. 415–432 (2002). Springer

[41] Rescorla, E.: The transport layer security (tls) protocol version 1.3. Rfc, Internet Engineering Task Force (IETF) (August 2018). https://www.rfc-editor.org/rfc/rfc8446

[42] Wikipedia: Facebook–Cambridge Analytica Data Scandal. https://en.wikipedia.org/wiki/Facebook%E2%80%93Cambridge_Analytica_data_scandal Accessed Feb 4, 2021