Proceedings of the 8th International Conference on Economic Management and Green Development
DOI: 10.54254/2754-1169/103/20242354

A Framework for Decentralized Identity and Credential Management Leveraging Blockchain Technology

Yuhao Liu[1,a,*]

[1]College of Software, Xinjiang University, Urumqi, China
a. guarangpi55@ stu.xju.edu.cn
**corresponding author*

Abstract: As internet technology rapidly advances and the infrastructure continues to evolve, web applications have become an integral part of our daily lives, with electronic identities (e-identities) playing a crucial role. Today, e-identities are vital for verifying personal identity in numerous online activities, and as our dependence on them grows, so does the concern for their security. This increasing importance of e-identity security raises various challenges, highlighting the need for robust solutions. In response to these challenges, this article introduces a novel solution: a blockchain-based decentralized identity and credential management framework. This approach not only enhances security but also offers greater control and flexibility for users. The article delves into the intricacies of the framework, detailing its design and operational mechanism. Additionally, it explores a range of practical applications, demonstrating the framework's versatility and effectiveness through diverse case studies. This comprehensive examination underscores the framework's potential to revolutionize e-identity management and addresses the pressing need for secure and efficient online identity verification in our increasingly digital world.

Keywords: Decentralized Identifiers, Verifiable Credentials, Blockchain, Security and privacy concerns, management framework.

1. Introduction

As technological advancements continue, electronic identities and credentials have become increasingly vital in various facets of our lives, including work, personal endeavors, academia, and other professional activities. We see these identities in various forms such as email, social media, and electronic payment accounts. The growing reliance on electronic identities has led to the storage of vast amounts of personal data, making the security of these identities a paramount concern. Several incidents have underscored this vulnerability; for instance, in 2018, Facebook's major data breach with Cambridge Analytica compromised approximately 87 million user profiles, resulting in significant privacy violations [1]. Similarly, electronic credentials, serving as key instruments for verification, have simplified interactions and enhanced efficiency and convenience in our daily lives. However, like electronic identities, they too are susceptible to security breaches including tampering, leakage, and forgery. A core issue in these challenges is the centralized nature of current electronic identity and credential management systems. Such centralization not only heightens the risk of personal data being targeted but also limits users' control over their information. Conventional authentication methods like usernames and passwords, aside from being vulnerable to attacks, lack cross-platform interoperability, thereby compromising both security and user experience [2]. Blockchain technology introduces a decentralized approach to identity management, obviating the need for centralized institutions. This article builds upon this concept and proposes enhancements [3]. Our contributions are threefold: first, we present a multi-layer architecture for decentralized identity and credential management that integrates blockchain technology. Second, we detail the management process for Decentralized Identifiers (DID) and the mechanisms for secure storage and authentication of Verifiable Credentials (VC). Lastly, we validate the efficacy of our solution in improving identity verification and credential management through a comprehensive case study analysis.

2. Relevant Theories

Blockchain: Blockchain, a concept introduced by Satoshi Nakamoto in 2008, represents a decentralized and distributed ledger system, realized through distributed storage, peer-to-peer transmission, and cryptographic techniques [4]. Within a blockchain framework, data integrity is maintained through encryption and consensus mechanisms, ensuring both tamper resistance and transparency.

Decentralized Storage: Decentralized storage leverages blockchain technology for the distributed online storage of data. This method distributes data across multiple locations to enhance security and accessibility.

Smart Contract: A smart contract is an automated program operating on the blockchain. Its blockchain reliance guarantees tamper resistance and transparency, enabling it to execute operations autonomously.

Decentralized Identity: Decentralized identity utilizes distributed technology to provide a globally unique identifier, empowering users to control their identities independently of central authorities.

Verifiable Credentials: These are digital credentials associated with a decentralized identity, enabling reliable verification of the holder's credentials.

Key Pair: A key pair comprises a public key, used for data encryption, and a private key, for data decryption. This pairing is fundamental to implementing security measures like encryption, decryption, and digital signatures.

Re-encryption: Re-encryption is a technique that alters the encryption of data from one private key to another without decrypting the data itself. This enables data owners to grant third-party access without compromising their private keys [5, 6].

Zero-Knowledge Proof: This is a cryptographic protocol that allows one party (the prover) to prove the truth of a statement to another party (the verifier) without disclosing any additional information [7,8].

3. System Analysis and Application Research

3.1. Architecture

The system architecture of this solution is illustrated in Figure 1, which consists of four components: the blockchain layer, the data layer, the functional layer, and the user layer.

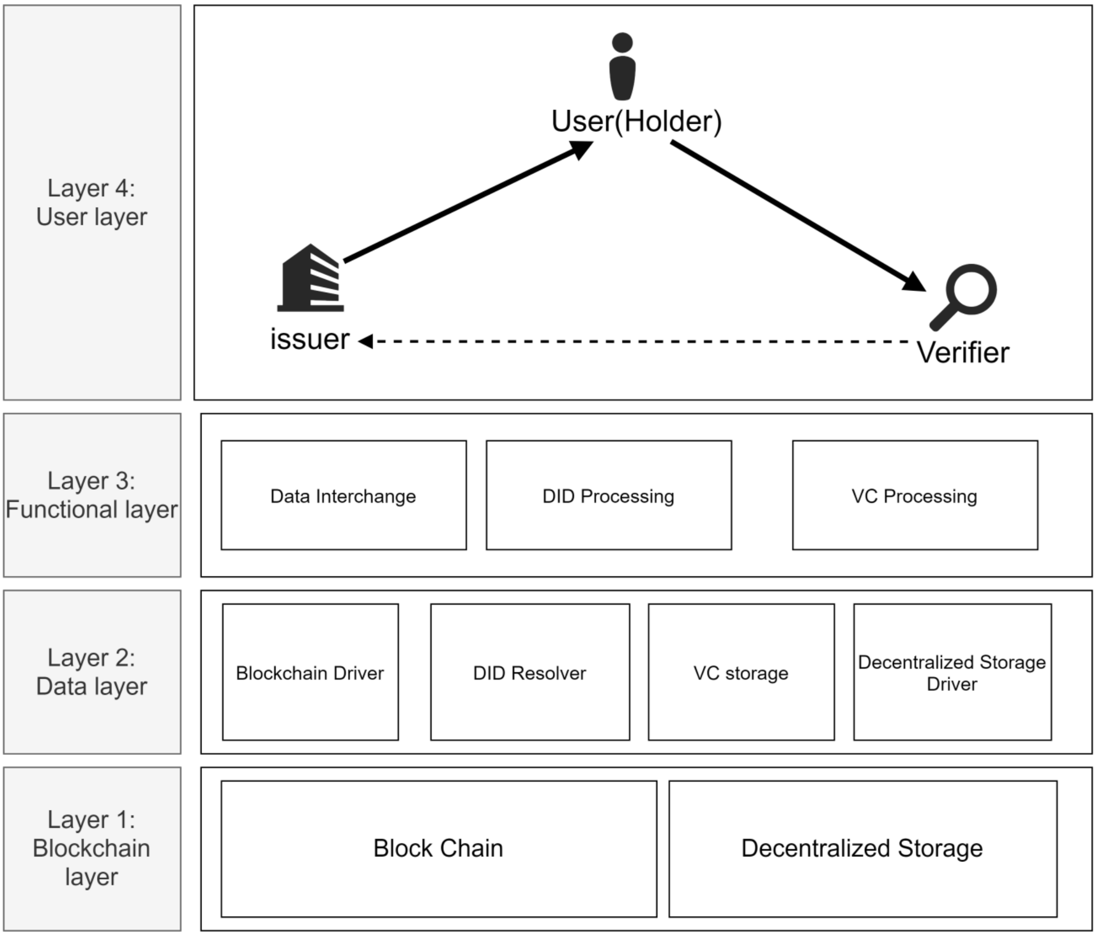


Figure 1: System architecture (Picture credit: Original).

3.1.1. Layer 1: Block chain layer

The first layer forms the foundation of the entire framework, consisting of two components: blockchain and decentralized storage. In this layer, the blockchain stores the ID number of the decentralized identifiers (DID) and their corresponding DID document index values. These index values point to the location of the DID documents within the decentralized storage [9].

For the blockchain component, platforms that support smart contracts, such as Ethereum and EOS, are utilized.

IPFS and Swarm are among the distributed storage solutions employed for the decentralized storage component.

3.1.2. Layer 2: Data layer

The second layer encompasses blockchain drivers, DID resolvers, VC (Verifiable Credentials) storage, and decentralized storage drivers. This layer handles interactions with the blockchain, including reading and writing data, resolving decentralized identities (DIDs), and storing verifiable credentials (VCs).

3.1.3. Layer 3: Functional layer

The third layer is the functionality layer, providing functionality for data exchange, DID processing, and VC processing. Operations such as data interaction between clients and the data layer, DID registration, issuance, verification, and updates, as well as VC issuance and verification, are implemented at this layer.

3.1.4. Layer 4: User layer

The fourth layer directly interfaces with users and includes issuers, users (holders), and verifiers. Users interact with this layer to manage DID registration, VC issuance, and verification. Importantly, the identities of issuers, users, and verifiers are not mutually exclusive; an individual can be an issuer while also being a user and verifier simultaneously.

3.2. DID management

This section will introduce how DID is linked to real identities during registration, ensuring users cannot generate DID duplicates, the composition of DID in Layer 1, provide examples of DID documents. And the update and revocation of DID [10].

3.2.1. DID registration

The user registration process is illustrated in Figure 2:

The system collects the user's biometric feature B.

The DID issuing authority compares the existing data D with the user's biometric information B. If a match is found, it indicates that the user has already registered, and the registration is rejected.

The DID issuing authority verifies the authenticity of the user's biometric information through online video verification or offline authentication points to ensure that it is not forged.

The user's biometric feature B is encrypted and stored within the issuing authority's system for subsequent verification of potential duplicate registrations.

Generate a public-private key pair using the Elliptic Curve Digital Signature Algorithm (ECDSA) and return it to the user, with the system not storing their private key data.

Generate a DID document and store it on a decentralized storage system. Store the ID of the DID in a smart contract on the blockchain and anchor it to the DID document. Return the DID number and smart contract information to the user.

3.2.2. The composition of DID in Layer 1

Figure 3 shows that the DID files are stored in decentralized storage, and the smart contracts on the blockchain store the DID's identifier and the index value of its corresponding DID document in the decentralized storage. When searching for a DID document, the smart contract is queried to retrieve its DID document index value, which is then sent to the DID resolver. The DID resolver resolves the DID document stored in the decentralized storage through the decentralized contract and outputs it.

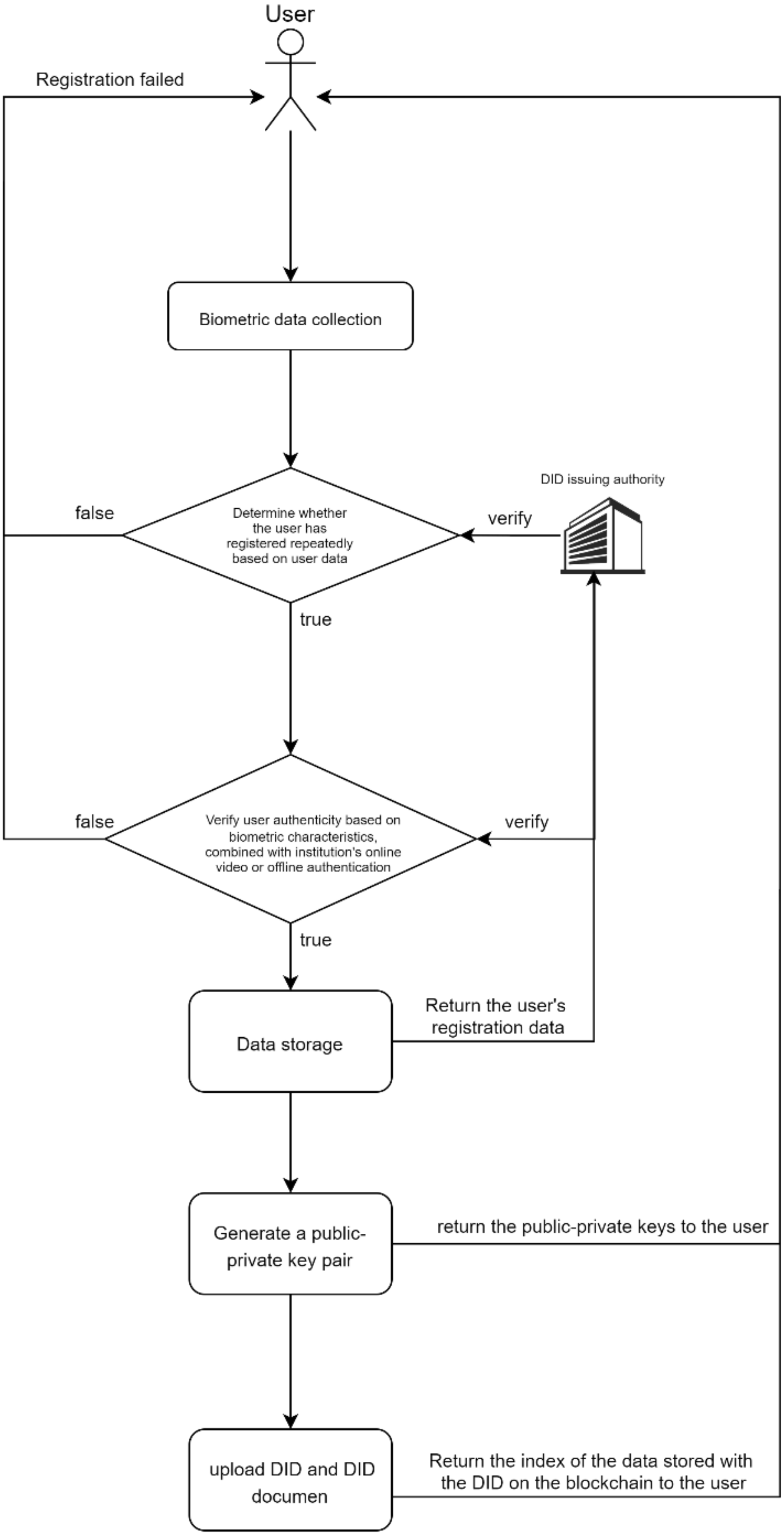

User
Registration failed
Biometric data collection
DID issuing authority
false
verify
Determine whether the user has registered repeatedly based on user data
true
false
verify
Verify user authenticity based on biometric characteristics, combined with institution's online video or offline authentication
true
Data storage
Return the user's registration data
Generate a public-private key pair
return the public-private keys to the user
upload DID and DID documen
Return the index of the data stored with the DID on the blockchain to the user

Figure 2: User registration process diagram (Picture credit: Original).

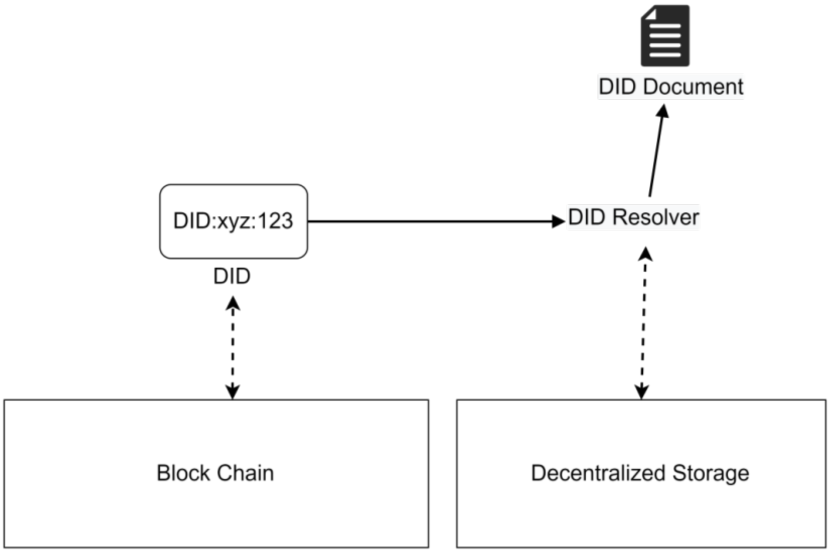


Figure 3: composition of DID (Picture credit: Original).

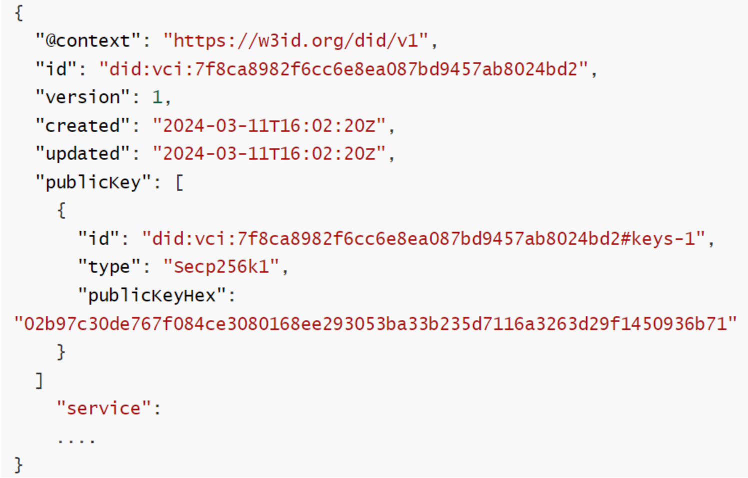
3.2.3. Examples of DID documents

The DID document is written according to the standards provided by W3C, and Table 1 provides explanations for its fields.

Table 1: DID document fields.

filed	meaning
id	The unique identifier of the DID document, representing the specific DID being described.
version	The version number of the document, useful for tracking iterations or updates.
created	Indicates the date and time when the DID document was created.
updated	Indicates the date and time when the DID document was last updated.
Public Key	Contains a list of public keys associated with the DID, providing details like ID, type, and value for each key.
authentication	Lists the public keys or methods used for verifying the identity of the DID subject, typically referencing entries in public Key.
Assertion Method	Specifies methods that can be used to make assertions or signatures, indicating which keys can sign data or messages.
service	Provides information about services offered by the DID subject, including the service's ID, type, and endpoint (URL).
proof	Contains proof of the DID document's integrity and authenticity, including the signature type, creator, and signature value.

DIDs are written in a standardized format to enable DID resolvers to parse DID documents from different DID issuers. Figure 4 illustrates the structure format of a DID.

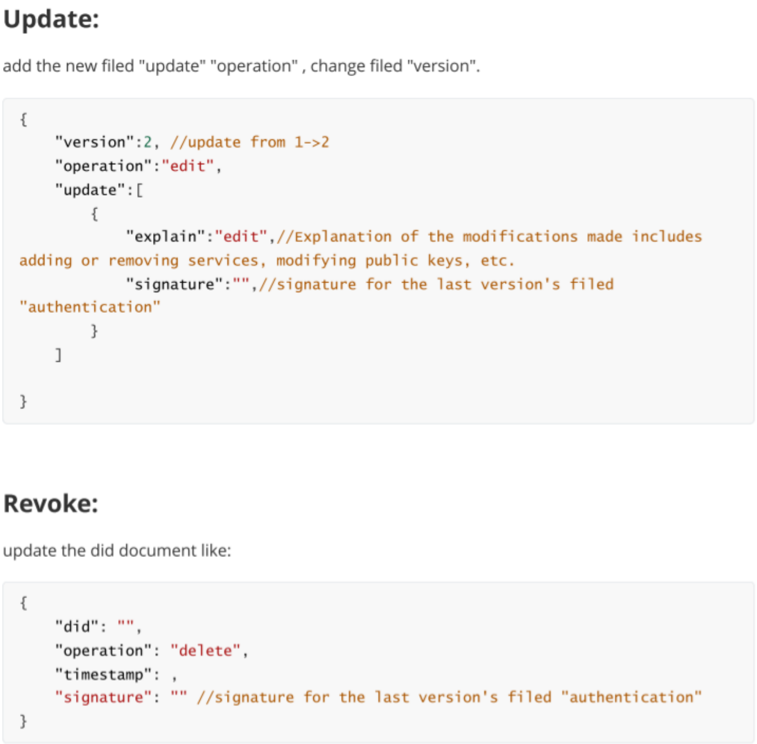


```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:vci:7f8ca8982f6cc6e8ea087bd9457ab8024bd2",
  "version": 1,
  "created": "2024-03-11T16:02:20Z",
  "updated": "2024-03-11T16:02:20Z",
  "publicKey": [
    {
      "id": "did:vci:7f8ca8982f6cc6e8ea087bd9457ab8024bd2#keys-1",
      "type": "Secp256k1",
      "publicKeyHex":
"02b97c30de767f084ce3080168ee293053ba33b235d7116a3263d29f1450936b71"
    }
  ]
    "service":
    ....
}
```

Figure 4: DID structure format (Picture credit: Original).

3.2.4. DID update and revoke

To update or revoke a DID, users need to publish a new DID document and update the index value of the DID document on the blockchain to the latest published value. The new DID document should include a signature of the user's old version attributes using their private key Sk to prove the legitimacy of the new DID document. Figure 5 illustrates the format of the updated and revoked DID documents.



Update:

add the new filed "update" "operation" , change filed "version".

```
{
    "version":2, //update from 1->2
    "operation":"edit",
    "update":[
        {
            "explain":"edit",//Explanation of the modifications made includes
adding or removing services, modifying public keys, etc.
            "signature":"",//signature for the last version's filed
"authentication"
        }
    ]

}
```

Revoke:

update the did document like:

```
{
    "did": "",
    "operation": "delete",
    "timestamp": ,
    "signature": "" //signature for the last version's filed "authentication"
}
```

Figure 5: DID update and revoke (Picture credit: Original).

3.3. VC management

3.3.1. VC Application and Issuance

To establish the authority of the VC issuers, this paper proposes a centralized issuer registration center solution. Issuers are required to register and submit relevant information to the center as shown in table 2, which then conducts an audit process. The registration center is responsible for reviewing and verifying the issuers to ensure their eligibility and trustworthiness in issuing credentials. Once an issuer successfully passes the audit, their information is recorded in the registration center's database and displayed in the client.

Table 2: Issuers information.

Filed	meaning
Website	Official website related to the public Issuer, convenient for entities or natural persons to verify if other information on the Registry is true
Endpoint	Issuer Endpoint, provides a service endpoint for specific information
Description	description by the discoverer
Service Type	Type of service provided by the discoverer
Request Data	The style of request data when users apply for this service (Personal Data)

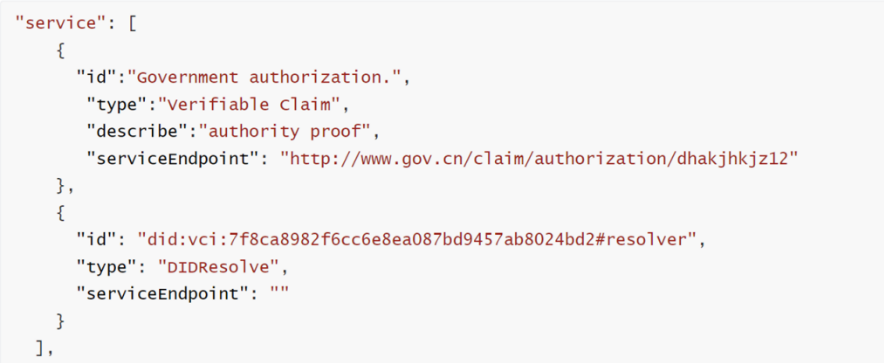
When users utilize the client, they can discover registered issuers by accessing the issuer registration center. The registration center provides a list of issuers and relevant information, allowing users to view the qualifications of issuers and the types of credentials they issue. Additionally, the issuer who has completed the registration can also publish a detailed list of their verifiable claims made during the issuance process, as well as the corresponding information required for applying for these verifiable claims. This way, users can understand the application requirements and processes for different verifiable claims from different issuers.

Furthermore, even if the issuer is not registered in the registration center, they can still issue credentials. The registration center serves the purpose of demonstrating the authority of issuers and providing users with quick access to information about the application process for relevant credentials

3.3.2. Issuer authority proof

While the issuer registration center plays a crucial role in establishing the authority of issuers, it is important to address the scenario where an issuer has not registered with the center. In such cases, the issuer can employ alternative mechanisms to ensure its own authority.

Method one is Registered issuers provide guarantees for unregistered issuers. For instance, The government (already registered) issues a VC to a designated vaccine company (unregistered), specifying authorization information. The vaccine company utilizes this VC as an authentication service and places the endpoint of this VC in the 'service' field of the DID document as shown in figure 6.



```
"service": [
    {
      "id":"Government authorization.",
       "type":"Verifiable Claim",
       "describe":"authority proof",
       "serviceEndpoint": "http://www.gov.cn/claim/authorization/dhakjhkjz12"
    },
    {
      "id": "did:vci:7f8ca8982f6cc6e8ea087bd9457ab8024bd2#resolver",
      "type": "DIDResolve",
      "serviceEndpoint": ""
    }
  ],
```

Figure 6: Authority proof service (Picture credit: Original).

Method tow is embedding the endpoint of certificates proving authority into the "service" field of the DID document. Unregistered issuers can incorporate digital certificates issued by accredited and trusted authentication authorities into their DID documents to ensure their own authority. The authentication authorities must be entities that have undergone authoritative certification and possess credibility.

Verification involved part. The entire verification process involves three participants: the certificate holder, the issuing party, and the verifying party.

Holder: The owner and user of a DID identity. Users create and manage their own DID through a DID client agent.

Issuer: The participant who authenticates the data. This can refer to organizations or individuals who issue or sign verifiable claims for users.

Verifier: The participant who uses the data. Verifiers are typically service providers that users authorize to access their identity and verifiable claims. They verify the identity of the user, the issuer, and the content of the verifiable claims.

3.3.3. Verification flow

Issuance of Credentials: The issuing entity generates a Verifiable Credential (VC) encapsulating specific claims, such as identity data and qualifications. This VC is authenticated via a digital signature using the issuer's private key, affirming the credential's origin and integrity. Subsequently, the issuer dispatches the signed VC to the credential holder.

Management and Storage of Credentials: Upon receipt, the credential holder securely stores the VC, commonly utilizing a digital wallet or another robust storage method. The holder has the capability to manage an array of VCs and selectively disclose them to verifiers as necessary.

Verification Process: In scenarios where a verifier requires confirmation of the holder's specific claim, the holder presents the pertinent VC. The verifier employs the public key from the issuer's Decentralized Identifier (DID) document to authenticate the digital signature on the VC. This crucial step validates that the VC is genuinely issued by the named issuer and remains unaltered.

Additionally, the verifier examines various attributes of the VC, such as its expiry date and revocation status, to assess its ongoing validity. Upon satisfying all verification criteria, the verifier can confidently regard the assertions within the VC as accurate and legitimate.

3.4. VC storage

VC, as a carrier of verifiable information, may contain sensitive data such as personal information and banking account details. The disclosure or malicious tampering of VC can result in severe losses and risks for users. The credibility and reliability of VC directly impact the security and feasibility of the entire identity verification system.

3.4.1. VC storage characteristics

To ensure its credibility and reliability, VC storage must satisfy the following characteristics:

User data is encrypted and stored.

No private keys are stored.

Authentication is required for accessing user data.

Third-party access to user data is allowed only with user authorization

Adding or revoking authorization for third-party access to claims.

To demonstrate the authenticity of VC to third parties without disclosure.

3.4.2. VC storage flow

VC Storage is a database that can be deployed locally or in a cloud-based database. When storing VCs in VC Storage, the VCs are first encrypted using the following encryption formula:

$$\mathrm{C} = \mathrm{Encrypt}_{K_{\mathrm{pub}}}(D) \tag{1}$$

D is the data declared by the third party for access, it is encrypted using the user's public key Kpub to obtain the encrypted data C.

Access to user data is controlled through a robust authentication mechanism, allowing data access only with user permission. When a third-party service provider requests access to user data, this process utilizes Proxy Re-Encryption (PRE) technology to enable secure data access without revealing the user's private key. The Proxy Re-Encryption process is represented by the following formula:

$$K_{\mathrm{re}} = \mathrm{ReKeyGen}\left(K_{\mathrm{priv}}, K_{\mathrm{pub\text{-}T}}\right) \tag{2}$$

Kre is a re-encryption key generated by the user's private key Kpriv and the third party's public key Kpub-T. Then, the user authorizes the third party to access specific data (claims), which are stored using VC. The data is transformed from a form that can only be decrypted by the user to a form that can also be decrypted by the third party using the following formula:

$$C_T = \mathrm{ReEncrypt}(C, K_{\mathrm{re}}) \tag{3}$$

CT is the data that has been re-encrypted and can be decrypted by a third party. Re-encryption enables the data to remain encrypted while allowing authorized third parties to decrypt it and access the data.

After the encryption process is complete, VC storage sends an endpoint to the third party, allowing them to access the stored resources. The third party uses their private key Kpriv-T to access and decrypt the VC containing their claims, and then verifies its authenticity.

$$D_T = \mathrm{Decrypt}_{K_{\mathrm{priv\text{-}T}}}(C_T) \tag{4}$$

If a user does not want to disclose the information in their VC, they can choose to use zero-knowledge proofs to enable third-party verification of their VC without revealing its contents.

3.5. Application research

This section will present two practical case studies to explore the advantages of the proposed solution.

3.5.1. Brand authorization

In commercial activities, brands or manufacturers often require agents to sell their goods on their behalf. To ensure the legitimacy of agents, brands issue authorization certificates to them. It is crucial to securely establish and verify the authorization of agents.

For brands or manufacturers, ensuring the authorization of agents needs to be secure and reliable.

For agents, they need to securely store and quickly prove their legitimate authorization to customers or partners.

Basic Process: The brand or manufacturer issues verifiable agent authorization certificates to agents using the DID program.

Agents can use the DID program to showcase their authorized status in commercial activities and verify its authenticity to customers or partners.

In this case, the proposed solution facilitates fast establishment of authorization between brands and agents, enabling agents to securely present their authorization certificates. It also provides a convenient way for customers to verify the authorization of agents. This solves the problems of easy forgery and difficult verification associated with traditional authorization documents.

3.5.2. One-stop service

In modern life, many transaction applications often require proof from two or more different institutions, making the process cumbersome. For example, when transferring property ownership, one may need to provide property ownership proof from the property management department and property tax payment proof from the tax department. Using the DID program can greatly simplify this process. Here's an example scenario for obtaining degree verification for a master's job application:

Obtaining the first verifiable certificate: The user applies for a bachelor's degree certificate from University A using the DID program. University A issues a digitally signed verifiable certificate (Avc) to the user.

Obtaining the second verifiable degree certificate: The user applies for a master's degree certificate from Graduate School B using the DID program. Graduate School B issues a verifiable certificate (Bvc) to the user.

Presenting verifiable credentials to the company: The user presents the verifiable credentials Avc and Bvc obtained from the university and graduate school to the prospective employer. The company can verify the authenticity of these credentials through the information recorded on the DID, thus completing the educational verification process for the job applicant.

In this case, Using the DID program in transactions that require cross-institution verification can simplify the process, improve the efficiency of issuance and verification, enhance security and transparency, and make storage and management easier compared to paper certificates.

4. Conclusion

This paper begins by evaluating the shortcomings inherent in current digital identity and credential management systems. We propose decentralized identity and verifiable credentials as innovative solutions to overcome these deficiencies. Section 2 introduces pertinent concepts and terminologies. Subsequently, in Section 3, we elucidate the intricacies of our proposed multi-layer framework, clarifying the interrelations among its components. This includes an in-depth discussion on the processes of registering, updating, and revoking Decentralized Identifiers (DIDs), alongside an examination of their manifestations on blockchain and within decentralized storage systems. Additionally, we delve into the issuance and validation of Verifiable Credentials (VCs), highlighting the role of issuing authorities and the encryption methods employed for the secure storage and access of VCs. The efficacy and applicability of our approach are substantiated through detailed analyses of specific use cases.

In today's rapidly evolving digital landscape, there is an escalating need for more robust and efficient systems for managing identities and credentials. Future research endeavors must aim to further refine and develop comprehensive management models, algorithms, and practical applications to cater to this burgeoning demand.

References

[1] Sporny, M., Longley, D., Chadwick, D., & Steele, O. (2020). Verifiable Credentials Data Model v2.0. Retrieved from https://www.w3.org/TR/vc-data-model-2.0/.

[2] Sporny, M., Longley, D., Sabadello, M., Reed, D., Steele, O., & Allen, C. (2019). Decentralized Identifiers (DIDs) v1.0. Retrieved from https://www.w3.org/TR/did-core/.

[3] Brunner, C., Gallersdörfer, U., Knirsch, F., Engel, D., & Matthes, F. (2020). DID and VC: Untangling decentralized identifiers and verifiable credentials for the web of trust. In 2020 the 3rd International Conference on Blockchain Technology and Applications (ICBTA). Xi'an, China. https://doi.org/10. 1145/ 3446983.3446992.

[4] Samir, E., Wu, H., Azab, M., Xin, C., & Zhang, Q. (2022). DT-SSIM: A decentralized trustworthy self-sovereign identity management framework. IEEE Internet of Things Journal, 9(11), 7972–7988. https://doi. org/10.1109/JIOT.2021.3112537.

[5] Tan Pengliu, Xu Teng, Yang Sijia, Tao Zhihui (2024). Review of research on blockchain privacy protection technologies. Advance online publication. https://doi.org/10.19734/j.issn.1001-3695.2023.12.0603.

[6] Liu, Y., He, D., Obaidat, M. S., Kumar, N., Khan, M. K., & Raymond Choo, K.-K. (2020). Blockchain-based identity management systems: A review. Journal of Network and Computer Applications, 166, 102731. https://doi.org/10.1016/j.jnca.2020.102731.

[7] Zhu, X., Xu, H., Zhao, Z., & others. (2021). An Environmental Intrusion Detection Technology Based on WiFi. Wireless Personal Communications, 119(2), 1425-1436.

[8] Cormode, G., Dall'Agnol, M., Gur, T., & Hickey, C. (2023). Streaming Zero-Knowledge Proofs. ArXiv, abs/2301.02161. https://doi.org/10.48550/arXiv.2301.02161.

[9] Alaya, B., Laouamer, L., & Msilini, N. (2020). Homomorphic encryption systems statement: Trends and challenges. Comput. Sci. Rev., 36, 100235. https://doi.org/10.1016/j.cosrev.2020.100235.

[10] Yang, M., & Xia, L. (2019). Secure Data Access Method based on electronic identity for Mobile Internet. IOP Conference Series: Materials Science and Engineering, 569. https://doi.org/10.1088/1757-899X/ 569/ 5/052088.