TAC: A unified trust anchor framework based on consortium blockchain

Feng Zhang1, Wenbao Jiang1* and Boxuan Shi1

1 School of Information Management, Beijing Information Science Technology University, Beijing 100192, China

*Corresponding author’s e-mail: jiangwenbao@tsinghua.org.cn

Abstract. In the current network, there are still many security issues, such as identity fraud and address spoofing. In traditional Public Key Infrastructure (PKI), most trust models adopt a centralized trust model with a single-point trust anchor. The existing PKI system lacks a unified authentication framework and is low efficiency. Aiming at the problems of how to store public keys efficiently and how to make the network secure and trustworthy, this paper proposes a new trust anchor framework based on consortium blockchain, named Trust Anchor Chain (TAC). This study uses the form of consortiu m to manage TAC, proposes a hierarchical naming ID, named TACID, to uniquely identify the user in TAC, and proposes a new data structure, named Struct Radix Trie, to store public key information in TAC to improve efficiency. Section 4, TAC has superiority through the comparison between local experiments and remote experiments. In TAC, users can access each other in the TAC, get the public key, and ensure the correctness of the identity. All users work together to maintain TAC to achieve equal network interconnection, security and credibility.

1. Introduction

Under the public key cryptosystem, the public key digital signature technology relies on the Certificate Authority (CA)[1] certificate issued by the Public Key Infrastructure (PKI)[2] to bind the entity identity and public key to ensure the authenticity of the entity public key. Binding the user's public key to the user's identity in the form of a public key certificate forms a mature solution to the network security problem. However, PKI introduces a trusted third-party CA, which brings the cost of certificate management, storage, computing.

1) The issuing, distribution, acquisition, verification and revocation of certificates are more complicated;

2) The online certificate directory is required to provide users with certificate downloading service and status query service at any time, which increases maintenance overhead;

3) If the user communicates with more objects, the user musts store and manage these certificates locally, which increases the usage overhead of the client;

4) The solution to large-scale key management is generally to adopt a method of adding physically CAs, there is also a problem of cross-certification and trust management.

5) Users must create multiple digital identities among multiple service providers to access their services separately.

From a user experience perspective, the process of creating multiple digital identities is inconvenient and cumbersome because the user must repeat the same registration process and remember multiple
passwords for different services. The provider's central server is the main target of hackers, user's private key is vulnerable to hackers.

This study proposes a unified trust anchor framework to provide the user around the world with a unified digital identity. This framework is decentralized, manageable, traceable and auditable to prevent multiple attacks and provide users with high security. In the next sections, this paper discusses consortium blockchain, the advantages and disadvantages of existing trust models and blockchain-based PKI.

### 1.1. Consortium Blockchain

Bitcoin[3] began to appear in 2008, which was introduced by Nakamoto, then blockchain technology attracted great attention. Blockchain technology uses block chain data structure to validate and store data, uses distributed node consensus algorithms to generate and update data, uses cryptography to ensure data transfer and access security, utilizes automated script code to implement the contract. Blockchain technology[4] mainly includes the data layer, network layer, consensus layer and contract layer. Consortium blockchain is based on blockchain. The member of consortium blockchain is a certain group and needs to meet the entry requirements of consortium blockchain. In consortium blockchain, it requires all participants need permission. Through the analysis, consortium blockchain has characteristics such as traceable, decentralization and data consistency. These characteristics provide a basis to implement TAC.

### 1.2. Blockchain-based PKI

1) Certcoin[5]: Certcoin was proposed by Fromknecht to implement alternative, public and decentralized certification schemes. Certcoin uses a consistency guarantee based on blockchain-based cryptocurrencies such as Bitcoin and Namecoin to create a decentralized PKI to ensure identity retention. However, certcoin does not have authentication. That is to say, no matter who first applies for identity ownership, it has it. Second, if both online and offline keys are compromised, they don't have enough solutions. As a result, identity owners cannot safely reclaim their identity, which can result in an identity being unusable.

2) Authcoin[6]: Authcoin proposed by Benjamin implements a decentralized PKI scheme. To mitigate cybersquatting and Sybil attacks, Authcoin shared some basic ideas with Certcoin, except that it uses a flexible challenge-response scheme for authentication when publishing public keys. In detail, Authcoin emphasizes the actual binding of the user when registering the public key by adding a complex challenge-response step that allows it to adapt to Sybil attacks. However, the performance cost increases with its interactive communication steps, and the solution does not consider the credibility of the person performing the operations during the verification and authentication process.

3) IKP[7]: Instant Karma PKI (IKP) was proposed by Matsumoto to implement an improved PKI solution. IKP uses a blockchain-based mechanism that automatically responds to CA's misbehavior and provides incentives for those who can help identify bad behavior. IKP focuses on intelligently monitoring the misbehaving CA through Ethereum through intelligent contracts. However, IKP faces the same problems as the above, and malicious users may use fake identities for fraud.

4) Cecoin[8]: Bo Qin's proposed cecoin implements a distributed PKI solution based on Bitcoin technology, providing multiple certificates and identity distribution functions. However, the efficiency of Bitcoin itself and the waste of resources are more serious. In addition, the use of mine work as a CA node, then the emergence of the mine pool means a large CA node. Again, it can cause a single point of failure.

5) Certledger[9]: M.Y. Kubilay proposes a public log and a blockchain-based PKI model that is more transparent and validates its operations. In CertLedger, all TLS clients can verify the final state of the log, making split-world attacks impossible. However, there are still CA nodes in the certledger.

In order to solve these problems, this study proposes to implement a unified, decentralized, manageable, traceable, auditable new trust anchor framework. In this paper, we propose a unified trust anchor framework named TAC, which is based on consortium blockchain. This study proposes to set the
global unique identifier, named $TAC_{\text{id}}$, to identify the user identity, the $TAC_{\text{id}}$ is generated by $TAC$ according to the user’s identity information, which needs the user to provide certain proofs, for example, a work certificate, ID Card or passport. The more detailed identity information, the more reliable it will be in $TAC$. $TAC$ generates and distributes the different $TAC_{\text{id}}$ based on different identity information. This study does not set up a third-party trust anchor, but different users will be divided and managed, according to the country, region, and organization, to meet the requirements of worldwide users. Users can also access each other in the $TAC$, get the public key, and establish a reliable link.

2. Trust Anchor Chain
In this section, This paper describes the design connotation of $TAC$. Our goal is the $TAC_{\text{user}}$ is data node which saved in $TAC$. The overall structure of the $TAC$ framework is shown in Figure 1. This paper proposes to divide organizations in $TAC$ according to the country region as a different starting node of the different $TAC_{\text{id}}$. This means that if Bob is applying for registration in China, then his $TAC_{\text{id}}$ will start with CN.

![Consortium Blockchain of Trust Anchor](image)

Figure 1 The overall structure of the $TAC$ framework

For example, Bob want to register an account in $TAC$.
1. The system generates an $TAC_{\text{id}}$ based on the personal information provided by Bob, including address, name, etc..

2. Bob can choose to use an encryption software to generate the public key locally and upload it to $TAC$, or the system can generate it offline. We recommend local generation.

3. Bob registered activities generates a registration transaction, submitted to $TAC$, after being certified, this transaction will be stored in $TAC$.

4. If Bob wants to get someone else’s public key, he can query it on $TAC$, or get it by the other party, Bob to verify if it is trust worthy. For a best viewing experience the used font must be Times New Roman, on a Macintosh use the font named times, except on special occasions, such as program code.

So, we make the basic description about $TAC_{\text{user}}$. 
We assume that a $TAC_{user}$ is the tuple $(x, y, z)$ where $x \in ENT$, $y \in ENT \cup ID$ and $z \in DATA$. The set $ENT \cup ID$ is assumed to be empty. This allow $TAC_{user}$ exit in the other PKI that connect public keys to identities or authorisations.

Simply, we have $PK \subseteq ENT \times (ENT \cup ID) \times DATA$. We use the following functions to obtain the component values from a pubkey $K \in PK$ given by $K = (x, y, z)$:

$$O(C) = x, S(C) = y, D(C) = z.$$  

Generally, these functions may return the organization of the $TAC_{user}$, $O(C)$, the subject of the $TAC_{user}$, $S(C)$, the data of the $TAC_{user}$, $D(C)$. Let $seqPK$ be the set of all sequences of pubkeys from $PK$. If $PK = (a_1, b_1, c_1), (a_2, b_2, c_2)$, then we have

$$seqPK = \{<>, (a_1, b_1, c_1), (a_2, b_2, c_2), (a_1, b_1, c_2), (a_2, b_2, c_1)\}.$$  

Let $Num(A)$ indicate the number of elements in the set $A$. So we construct the set of all pubkey paths, $PK\_PATH$, from the seqPK.

$$PK\_PATH = \{(1, pubkey) \mid pubkey \in PK\} \cup \{s \in seqPK \mid Num(s) \geq 2 \land S(s(i)) = 0(s(i + 1)), 1 \leq i \leq Num(s) - 1\}.$$  

We have the following rules:
1) $\forall pubkey \in PK \cdot O(pubkey) \in ORG$
2) $\forall p \in PK\_PATH \cdot O(p(1)) \in ORG$
3) $\forall pubkey \in PK \cdot Num(s(pubkey)) = 1$

Rule 1 states that all pubkey are managed by the set of ORG. Rule 2 states that all pubkey path begin with a element of the ORG. Rule 3 states that each pubkey there is a unique $TAC_{ID}$.

### 3. Struct Radix Trie

In this section, this paper discusses the technologies used in TAC. Here this section states in sequence: Struct Radix Trie, User structure, Transaction structure. This study uses consortium blockchain to achieve decentralization and traceability and use Struct Radix Trie to achieve a unified trust anchor and manageable goals. In section 6, This paper analyses the advantages of Struct Radix Trie compared with hierarchical naming and flat naming on storing and managing the pubkey. In TAC, this study uses hierarchical naming to generate $TAC_{ID}$ as the globally unique identifier of the $TAC_{user}$. For example,

| Name     | Description                                      |
|----------|--------------------------------------------------|
| organization | the member of the consortium organization, manage a part of $TAC$ |
| identifier    | the subject identity or authorisation for the public key |
| pubkey    | the public key value for this identifier          |
| data      | additional data fields                           |
| UALuser   | untrusted address lists                          |

| Name     | Description                                      |
|----------|--------------------------------------------------|
| ENT      | the set of $TAC$ entities(users, organizations, etc.) |
| ID       | the set of all $TAC_{user}$ identifiers           |
| DATA     | the set of all additional $TAC_{user}$ data       |
| PK       | the set of all $TAC_{pubkey}$                     |
| ORG      | the set of all organizations                     |

In addition, in order to better describe the unified trust anchor framework. This study begins the development with some high level definitions for fundamental sets (or types), as shown in Table 2.
Alice is a student at Tsinghua University now. She applied to TAC for registration and this study can assign CN.bj.thu.Alice as her TACID. So, in order to store the TACID, this study proposes Struct Radix Trie based on Radix Tree[10], in order to improve efficiency and facilitate management. The data structure is shown in Figure 2.

In Figure 2, this study uses CN. as a root organization node in the Struct Radix Trie, different organizations under the CN. node will be stored hierarchically according to their respective regions. Similarly, the public key information under the other organization nodes in TAC also be stored in this way.

This study maintains a Struct Radix Trie outside the TAC to facilitate users to query public key information. The operations such as registration, update and revocation about TAC_pubkey need to be performed in the TAC and recorded in the blockchain. It is convenient for security audits and traceability when encountering security issues. Different organizations manage the public key information of their own regions, clarify the main body of responsibility, strengthen the flexibility and autonomy of organizations.

4. Evaluation
In this section, This paper evaluates TAC. This TAC experiment is based on the Ethereum platform and uses the golang for development. The local system is Drawin, 2.6 GHz Intel Core i7 and 16G memory. This study sets up three computers to form a LAN, uses Aliyun server to simulate public network and uses zookeeper to build a centralized trust root cluster, and trust anchor cluster, called centralized trust anchor model(CTA). The operating system in Aliyun is Centos6.5.

4.1. Routing experiment
In this test, 100,000 pieces of data are inserted respectively (10 first-level nodes, each of which has 10 second-level nodes, each of which has 100 third-level nodes, and each of the three-level nodes has 100 nodes.), to 1 million pieces of data (10 first-level nodes, each of which has 10 second-level nodes, each of which has 100 third-level nodes, each of which has 1000 nodes). As shown in the Figure 3, the
horizontal and vertical are the number of nodes adding the public key, and the vertical axis is the total time required.

![Figure 3](image_url)

*Figure 3 The time of adding pubkey*

From the Figure 3, we can find that the curve of local TAC after expanding 1000 times and the curve of local CTA are basically the same, and the three curves are basically linear, indicating that the increase of time is stable compared with the increase of the node. After our tested, CTA’s spending time has increased dramatically is because of the network latency.

4.2. Additional experiment

Based on 10 million data nodes, this study also tested some important function modules. The results are shown in Table 3.

| Operation           | Operation Times | Average Time   |
|---------------------|-----------------|----------------|
| ECC GeneratePubkey  | 1000000         | 19042 ns/op    |
| Create Block        | 30000           | 55335 ns/op    |
| Verify ECC_Pubkey   | 20000           | 94187 ns/op    |

This study uses Elliptical Curve Cryptography (ECC) in the TAC to generate the public key and verify the legitimacy of the TAC_user’s signature. From the test data, shown in Table 3, the efficiency is in line with our expectations.

5. Conclusion

In this study, we design a unified trust anchor framework based on consortium chain, named TAC. TAC is mainly used to manage and store public key information and transactions about public key changes. TAC is an unmodifiable, decentralized mechanism that guarantees data consistency. TAC can support large-scala data and prevent multiple attacks. Users can obtain public key information through TAC to ensure the correctness and authenticity of the identity. When there is a node deliberately makes a mistake, TAC will add its address in UAL. But, TAC also needs improvement. We will also develop browser plug-ins for users to make it easier to access TAC, faster queries, and more secure communication.
Acknowledgments
This work was supported by the National Natural Science Foundation of China through project 61540020.

References
[1] Adams, C., & Lloyd, S. (1999). Understanding public-key infrastructure: concepts, standards, and deployment considerations. Sams Publishing.
[2] Housley, R., & Polk, W. T. (2008). Internet X. 509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. RFC, 5280, 1-151.
[3] Nakamoto, S., & Bitcoin, A. (2008). A peer-to-peer electronic cash system.
[4] F. Zhang, B. Shi, W. Jiang. (2018). Review of key technology and its application of blockchain, Chinese Journal of Network and Information Security.
[5] Fromknecht, C., Velicanu, D., & Yakoubov, S. (2014). Certcoin: A namecoin based decentralized authentication system 6.857 class project. Unpublished class project.
[6] Leiding, B., Cap, C. H., Mundt, T., & Rashidibajgan, S. (2016). Authcoin: validation and authentication in decentralized networks. arXiv preprint arXiv:1609.04955.
[7] Matsumoto, S., & Reischuk, R. M. (2016). IKP: Turning a PKI Around with Blockchains. IACR Cryptology ePrint Archive, 2016, 1018.
[8] Qin, B., Huang, J., Wang, Q., Luo, X., Liang, B., & Shi, W. (2017). Cecoin: A decentralized PKI mitigating MitM attacks. Future Generation Computer Systems.
[9] Kubilay, M. Y., Kiraz, M. S., & Mantar, H. A. (2019). Certledger: A new pki model with certificate transparency based on blockchain. Computers & Security, 85, 333-352.
[10] Zhang, W., Tang, H., Byna, S., & Chen, Y. (2018, November). DART: distributed adaptive radix tree for efficient affix-based keyword search on HPC systems. In Proceedings of the 27th International Conference on Parallel Architectures and Compilation Techniques (pp. 1-12).