Cecoin: A decentralized PKI mitigating MitM attacks

Article *in* Future Generation Computer Systems · October 2017

DOI: 10.1016/j.future.2017.08.025

CITATIONS
21

READS
163

6 authors, including:

bo bo Qin
South China University of Technology
102 PUBLICATIONS **1,004** CITATIONS
SEE PROFILE

Qin Wang
Swinburne University of Technology
8 PUBLICATIONS **36** CITATIONS
SEE PROFILE

Future Generation Computer Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Cecoin: A decentralized PKI mitigating MitM attacks

Bo Qin [a], Jikun Huang [a], Qin Wang [b], Xizhao Luo [c,*], Bin Liang [a], Wenchang Shi [a]

[a] Key Laboratory of Data Engineering and Knowledge Engineering, Ministry of Education, School of Information, Renmin University of China, Beijing, China
[b] School of Electronic and Information Engineering, Beihang University, Beijing, China
[c] School of Computer Science & Technology, Soochow University, China

HIGHLIGHTS

- We propose a new public key infrastructure framework based on Bitcoin.
- We present a concrete distributed PKI scheme referred to as Cecoin.
- Cecoin is quipped with enhanced security and desirable performance.

ARTICLE INFO

Article history:
Received 1 January 2017
Received in revised form 26 May 2017
Accepted 14 August 2017
Available online xxxx

Keywords:
PKI
Blockchain
Decentralized consensus

ABSTRACT

For numerous applications, it is essential to reliably link a public key with its owner. The current solution is to employ the well-known Public Key Infrastructure (PKI), represented by a trusted certificate authority (CA), to fulfill this assignment by signing the certificate for the public key after validating its owner. However, due to the centralized architecture, it raises the single-point failure problem with unpredictable threats. In this paper, we present a distributed certificate scheme, referred to as Cecoin which is inspired by the well-known Bitcoin by employing its irreversible unforgeability and public verifiability. In Cecoin, the certificates can be treated as currencies and recorded on block chain, which removes the single point failure problem. The miners can verify the validity of certificates following a set of rules to ensure ownership consistency, and allow an identity to bind multiple public-key certificates. For efficient retrieval and verification of certificates, and quick operations, we incorporate the modified Merkle Patricia tree and employ it to implement a distributed Certificate Library. To allow the owner to transfer the possession of identity, we design an online fair exchange protocol without a trusted third party. Security and efficiency analyses show that our Cecoin provides strong security with desirable efficiency.

1. Introduction

For the purpose of security, it is necessary for the user to authenticate the binding relationship between a public key and its owner. Public Key Infrastructure (PKI) [1] is such an important infrastructure that can provide a trusted digital identity, which is called certificate, to guarantee a correct association between them. During the process of communication, PKI allows others to validate and authenticate the certificates to ensure the confidentiality and integrity of the transmitted information. Therefore, the approaches to make an assurance and provide an improvement for the trust underlying the foundation of PKI. It is acknowledged that in a traditional structure, the assurance was delivered in the form of certificate, essentially a signature by a trusted third party via the public key. However, such centralized approaches brings challenges to security, especially the problem of single-point failure. This kind of deficiency represents the risk caused by a hierarchically structured system which is highly depended on a single central authorities. Commonly, the centralized authority in PKI is called Certificate Authority (CA) [2], and it is responsible for assigning and managing certificates for users.

Due to its strict hierarchy, we can deduce how the risk happens. In one case, as a result of inflexible dependence, once CA being hacked, adversary could distribute false certificates with pseudo identity [3,4]. Such misfortune can be exemplified by the incidents as DigiNotar [5], been attacked in 2011, which distributed numerous faked certificates issued by Google, and Comodo [6], which caused chaos through network. In another case, the subordinate agencies to issue certificates might be granted excessive power due to the aim of alleviating the bottleneck restriction of resources. The Trustwave [7], for example, has even arbitrarily granted an unbeknown company issuing unauthorized digital certificates. What

* Corresponding author.
E-mail address: xzluo@suda.edu.cn (X. Luo).

https://doi.org/10.1016/j.future.2017.08.025
0167-739X/

is more, the inconsistency between the real identities and the users is another subtle problem. For the reason of colossal market, there is no certificate company could monopoly all the trust of people. Therefore the existence of multiple to multiple mapping relations in CAs lead to the inconsistency of identities, which might issue the same identity to different entities and give adversaries chances to distribute false certificates. All those problems in PKIs are inevitable under the centralized structure.

Various approaches have been proposed to solve the above problem, mainly in centralize ways and in decentralized ways. The centralized schemes, represented by IDentity-based Public Key Infrastructure (ID-PKI) [8,9] and Certificateless Public Key Infrastructure (CL-PKI) [10], simplify the certificates management process by providing a fully trusted third party. However, these schemes have to work in a closed virtual private network instead of an open network. The other solutions are, predictably, decentralize schemes with the aim to separate the power of CAs. Certificate Transparency (CT) project [11], proposed by Google, employed the distributed servers to monitor peers, and Pretty Good Privacy (PGP) [12], proposed in the early 1994, introduced an distributed trust model by bringing a common intermediator among communicators. In recent years, researchers have devoted themselves to achieve consistency via an innovative technical tool called blockchain, and the corresponding schemes, represented by Certcoin [13,14], Authcoin [15] and IKP [16], have been successfully implemented. However, there still exist unsolvable problems, such as ignoring threats or lacking of diversities, which have left room for further improvement. We will discuss about them detailly in Section 2.

In this paper, we propose a distributively blockchain-based PKI named Cecoin. Firstly, by removing the trusted third party, the task to distribute and manage certificates is accordingly accomplished by miners, who will separate the power from CAs. As they are operated in Bitcoin system, the miners are dominated by incentive mechanism and distributed consensus protocol to ensure the consistency, and the guaranteed consistency of identities can avoid issuing false certificates. Secondly, for the purpose of adaptive availability, Cecoin provides services of multi-certificate and identity assignment. We augment the structure of Merkle Patricia tree to implement a distributed Certificate Library which stores all valid certificates. Besides, people who owns certificates can freely sell his identities and certificates online, and the exchange is fair without requiring a trusted third party. Thirdly, we implemented the prototype of Cecoin based on Bitcoin structure. The security of the scheme is analyzed in detail and our scheme prevents the MitM attack in real network atmosphere. As for efficiency, the time complexity of computing overhead is O(1) and the spatial complexity of storage overhead is O(n). Therefore, the analyses show that our scheme provides strong security with desirable efficiency.

The rest of the paper is organized as follow. Section 2 describes commonly used types of PKIs. In Section 3, we discuss the supporting technologies of Cecoin. Section 4 describes design of Cecoin. In Section 5, we introduce the implementation Cecoin in detail. Section 6 and Section 7 analyze the security and evaluation of Cecoin. Finally, Section 8 draws a conclusion of our scheme.

2. Related work

Studies have been conducted in various aspects to improve the security of PKI. The existing approaches mainly fall into two categories, centralized and decentralized.

2.1. Centralized approaches

Centralized approaches enhance system security by relying on a trusted third party. They are widely deployed in our network, and here are three types of implementation to achieve the goal of PKI:

CA: Certificate authorities (CA) are institutions or organizations through a network which are treated as trusted third parties. The current standard used in large-scale is X.509 [2]. The authorities possess the right of certification meaning that they can sign individuals, organizations or another CA's certificates. The structure of CA system likes a tree, which origins from a single root and then develops to various branches. However, once the initial CA under risk, such as being attacked or being faked, the whole system will be destroyed. The centralized power is unavoidably at the risk in various aspects.

ID-PKI: The system based on the method of IBE names IDentity-based Public Key Cryptography (ID-PKC) and the corresponding structure develops into IDentity-based Public Key Infrastructure (ID-PKI). Identify-Based Encryption (IBE) was firstly proposed by Shamir [8]. The purpose of IBE is to simplify the management process of certification in CA system via treating their identities as public keys. The binding relationships tightly link individuals to their certificates, which has greatly reduced the cost of communication by abandoning the process of exchanging and storing. Until Boneh delivered the implementation of IBE through bilinear pairing [9], the CA system began to draw attention on this scheme. However, IBE is more centralized than traditional CA due to its key generation center which issues individuals' private keys matched to their identities. This risk makes it not applicable towards open network.

CL-PKI: Certificateless Public Key Infrastructure (CL-PKI) employs the idea of certificateless cryptography [10]. It is a variant of IBE which avoids the inherent escrow in identity-based cryptography. CL-PKI relies on the use of a trusted third party (TTP) in possession of a master key, other than the use of certificates to guarantee the certification of public keys. However, in contrast to IBE, CL-PKI does not suffer from the key escrow property and it can be regarded as intermediate between traditional certificated PKI and IBE-PKI. However, it is centralized in structure which results in the problems brought by centralization as described above.

2.2. Decentralized approaches

Decentralized approaches enhance system security by separating powers of CAs through distributed architectures. The main problem of distributed system is its consensus and synchronization. After several years research on Bitcoin system, several schemes have been proposed and based on that technology. Here we state five schemes and the latter three schemes are based on Blockchain:

PGP: Pretty Good Privacy (PGP) [12] has build up a system which makes authentication entirely decentralized. Each user in the system acts as an authority aiming to ensure lots of bindings between other users and their public keys. To make it clear, Alice can trust Bob through their common intermediator Carol, and Carol makes an commitment to Alice with the signature of Bob. When Alice has successfully verified the authenticity of Bob's public key, then Alice decides to trust Bob and they have build up a reliable relation. However, the system cannot avoid malicious users who generate large numbers of false keys and connections. What is more, the missing incentives and the lack of punishments in order to motivate users in opposite make the system stick in narrow range.

CT: Certificate Transparency (CT) project was proposed by Google [11] as a distributed prototype to prevent the latent incidents caused by highly strong centralization. It is the most popular

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.

system to date which is available in both Chrome and Firefox. The project has introduced transparency into the workings of CAs via maintaining public, append-only logs through millions of independent servers world-wide. The distributed logs in various severs record all certificates issued which can be monitored and censored by servers in parallel. However, although CT possesses properties of publicity and auditability, it still relies on the numerous severs controlled by the same company, which means we cannot authenticate whether the majority of severs is trusted or attacked by others. What is more, CT cannot be able to revoke the false certificates issued.

Certcoin: Certcoin was proposed by Fromknecht [13,14] to implement an alternative, public and decentralized authentication scheme. Certcoin employs the consistency guarantees provided by blockchain-based cryptocurrencies, such as Bitcoin and Namecoin, to establish a decentralized PKI which can ensure the identity retention. However, on the one hand, Certcoin cannot prevent squatting identities of legal users as other schemes. And on the other hand, it cannot fulfill the user's demand of using multiple public keys under the same identity.

Authcoin: Authcoin was proposed by Benjamin [15] to realize a decentralized PKI scheme. For the purpose to mitigate the squatting and Sybil attacks, Authcoin has shared some basic ideas with Certcoin except that it employs a flexible challenge–response scheme for validation and authentication when public keys are issued. In detail, Authcoin stresses the actual binding when users register public keys by adding a complex challenge–response step, and this mechanism makes it resilient to Sybil attacks. However, the performance cost is increasing along with its interactive communication steps and the scheme does not consider the credibility of whoever performs the operations in validation and authentication process.

IKP: Instant Karma PKI (IKP) was proposed by Matsumoto [16] to achieve an improved PKI scheme. IKP employs a blockchain-based mechanism which can offer automatic responses to the misbehavior of CA and incentives for those whoever helps to detect the misbehavior activities. IKP focuses on monitoring misbehaving CAs through the automation of Ethereum through the smart contract. However, IKP confronts the problem which is the same as above schemes that the malicious users may spitefully register with fake identities to execution fraud.

3. An overview of supporting technologies

In this section, we will discuss the technologies used in Cecoin. Here we state in sequence: the Blockchain, the Merkle Patricia tree and the Fair Exchange Protocol based on smart contract.

3.1. Blockchain

Blockchain technology has come into people's view due to the emerge of Bitcoin since Satoshi proposed the white paper in an email sent to the members of Cypherpunk [17]. Bitcoin is an innovative cryptocurrency which fundamentally relies on Blockchain technology. After its transient steady development, Bitcoin has surprisingly soared up into the sky. Researchers have proposed various Bitcoin-class cryptocurrencies such as Litecoin [18], Primecoin [19] and Namecoin [20,21], etc. What is more, some other researchers have directly extracted the core technical principle of Bitcoin system, which is blockchain, to build platforms such as Hyperledger [22] and Ethereum [23,24] for further application developments. Notably, the most inspiring applications to date are notarization, crowd-funding, and smart contract.

Blockchain consists a series of technologies including cryptography, peer-to-peer network, and distributed consensus protocol. Therefore it possesses the inherit properties of tamper-proofing, decentralization, and consistency. We will briefly explain them as follow:

- *Cryptography:* Blockchain is a hash-pointed linked list [25] where each block stores the unique hash value of the previous block. The one-wayness of hash ensures the integrity of contents in block. If any details have been modified on purpose, the hash pointer would fail to link the correct path. We could employ the tamper-resistant property to ensure the certificates stored on blockchain integral.
- *Peer-to-peer network:* Blockchain exists in P2P network which represents the fact that each node is equal to their peers. For the pros, the distributed nodes remove the single-point failure. But for the cons, the separated nodes make it difficult to ensure the consistency. Fortunately, more and more distributed consensus protocols have been proposed and fully discussed corresponding to the development of blockchain.
- *Distributed consensus protocol:* Proof-of-Work (PoW) [26] mechanism in Bitcoin lets miners compete for the rewards of verifying transactions via computing a hash puzzle problem. The consistency relies on the computing power owned by all (group) miners and it is highly secure except for whoever has controlled more than half of power (which is almost impossible). Simultaneously, other consensus are being proposed such as Proof-of-Stake (PoS) [27], Delegated Proof-of-stake (DPoS) [28], Practical Byzantine Fault Tolerance (PBFT) [29], etc.

In this paper, we use the blockchain-based structure to achieve our system and the Proof-of-Work mechanism to ensure the consistency of certificates.

3.2. The merkle patricia tree

The Merkle Patricia tree is used in Ethereum [30] to store the state of account denoted as (*address*, *state*). Compared to the common Merkle tree, the Merkle Patricia tree has added two features inspired from Trie, which makes it able to simply add or remove nodes and efficiently look up items. Besides above, it forms a persistent data structure which can provide the proof of memberships. There are three kinds of nodes in the Merkle Patricia tree of Ethereum (quoted from [24]):

- *Leaf:* It consists two items. The first item corresponds to the nibbles in the key not already accounted for, by the accumulation of keys and branches traversed from the root. The hex-prefix encoding method is used and the second parameter to the function is required to be true.
- *Extension:* It consists two items. The first item corresponds to a series of nibbles of size greater than one that are shared by at least two distinct keys past the accumulation of nibbles keys and branches as traversed from the root. The hex-prefix encoding method is used and the second parameter to the function is required to be false.
- *Branch:* It consists seventeen items. The first sixteen items correspond to each of the sixteen possible nibble values for the keys at this point in their traversal. The seventeenth item is used in the case of this being a terminator node and thus a key being ended at this point in its traversal.

In this paper, we will employ the similar data structure to store the certificates in order to make it efficiently operated and looked up. And we need to make some changes when considering the aim to allowing an identity being bound with multiple public-keys in certificates.

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.

3.3. The fair exchange protocol based on smart contract

A fair exchange protocol is that two parties achieve an agreement in exchange with no one can gain extra advantage by unfair means. In most cases [31,32], fair exchange protocols require a trusted third party who acts as an intermediary to ensure the fairness of the exchange. However, in a decentralized system based on blockchain, there is no such a fully trusted third party. Therefore, we need to pick up a fair exchange protocol in decentralized way. Due to the script being automatic executed in blockchain-based cryptocurrencies, it can be further applied as a smart contract. Inspired by this, researchers has proposed several trustless payment schemes [33–35].

4. The system model

In this section, we provide an overview of the key features of Cecoin. We begin by introducing the main design goals, and then describing the system model and necessary components in the Cecoin architecture, and finally narrating the motivating scenarios in real network atmosphere.

4.1. Design goals

As mentioned in previous sections, our purpose is to propose a secure blockchain-based enhanced PKI named Cecoin, which provides a solution to the inevitable problems caused by centralization. To achieve it, Cecoin should satisfy the following goals in detail:

- *Achieve basic function of PKI:* As an enhanced scheme of PKI, Cecoin should have the fundamental functions of PKI. Therefore, Cecoin should provide services of certificate operation containing registering, revoking, renewing and verifying. The scheme can be able to achieve the distribution and management of certificates.
- *Enhance security:* In order to improve the security of PKI, Cecoin should be reliably decentralized to avoid the risk under the centralization. It should provide the guarantee of consistency to prevent the issuance of false certificates and protect the users' communication security. Specially, our scheme should prevent the prevailing man-in-middle attack.
- *Provide service of multi-certificate:* Most Internet service providers provided their services in multiplatform which caused that their applications have to be developed in different languages under different developers, and then the diversity resulted in different cryptographic algorithms which asks for clients of different keys to log in the different platforms. Under the premise of ensuring the consistency of identity, Cecoin should satisfy the requirement that an identity can be used to register multiple certificates for different scenarios.
- *Provide service of identity assignment:* In the real world, when company amalgamation and acquisition happens, the ownership of original identity would be transferred. Therefore, changes in identity should be allowed in Cecoin for practicability, here we definite the word "identity assignment" to describe the changes, notably, the identity transferring needs digital token (here is Cecoin currency) to provide warranties and achieve fairness.

Cecoin not only avoids the registration and distribution of pseudo certificates, but also prevents the risk in communication like MitM attack. In addition, the scheme provides two extra services, including the requirement of multi-certificate in case that one identity maps to multiple public keys in different scenarios, and the requirement of online certificate transferring in case that clients have to transfer ownership of certificates to others.

4.2. System model

There are three types of entities in our scheme, Miners, Certificate owner, and Certificate user. As shown in Fig. 1, all of them are nodes in the P2P network. The nodes recorded on blockchain in Cecoin share the common public data, and the request of certificates management and decentralized Certificate Library are implemented by the modified Merkle Patricia tree. Here, we briefly describe the responsibility of entities shown as follow, and all operations mentioned will be described detailly in later sections.

- *Miner:* The concept of Miner is the same as that of miners in Bitcoin. Miners in Cecoin play the role of traditional **CA**s, and they are responsible for recording the requests from Certificate owner and guaranteeing the correctness and consistency of certificates according to the distributed consensus mechanism, such as Proof-of-Work.
- *Certificate owner:* The concept of Certificate owner is the one who claims to own certificates. Certificate owners in Cecoin act as traditional **Domain**s, and they could make the requests of registering, renewing, revoking and identity transferring by creating special transactions to efficiently manage their certificates and identities.
- *Certificate user:* The concept of Certificate User is whoever legally uses certificates. Certificate users play the role of traditional **Client**s and they could search the certificates in Certificate Library to authenticate the identity of another communicator with the aim to protect their communication.

In order to make the distributed network reliable, there are two interactive entities of shared public data in Cecoin, which are the certificate operations recorded on blockchain and the latest certificate stored in decentralized Certificate Library. Here we state the functional responsibilities to make it clear.

- *On blockchain:* Blockchain (BC) stores the special transactions which represents the certificate operations of certificate and the request from Certificate owner. Once the Miner collects enough special transactions, he should immediately confirm it and then construct blocks to record the valid transactions. Whenever the data fixed on the blockchain, it is irreversible and persistent. What is more, the public operative data on blockchain makes it traceable for remedy after loss.
- *In Certificate Library:* Certificate Library (CL) is a distributed database correlated to the previous blockchain to store the updated certificates. We have made some changes on the Merkle Patricia tree, and then employ it to implement this decentralized Certificate Library. When special transactions are recorded, Miners should update the Certificate Library according to requests from that transaction. As for certificate users, they can validate the applied certificate in this library.

It is to be observed that, firstly, the above two kinds of shared data in Cecoin can both protect the integrality and consistency of certificates, but their usage are sightly different. BC stores a verified certificate operation of request which has been validated by Miners while CL stores all valid certificates in the network. Secondly, the data stored in the two structures interacts with each other. The head of each block in BC stores the root hash values of Merkle Patricia tree which represents the current status of the certificate library, therefore any operations such as addition, deletion, and change of certificates can lead to difference on block. In turn, the transactions stored in the BC would affect the status of CL, that means the certificate operation stored in BC would map to CL in form of the certificates being inserted and deleted in CL's Merkle Patricia tree, which results in changes of hash value.

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.

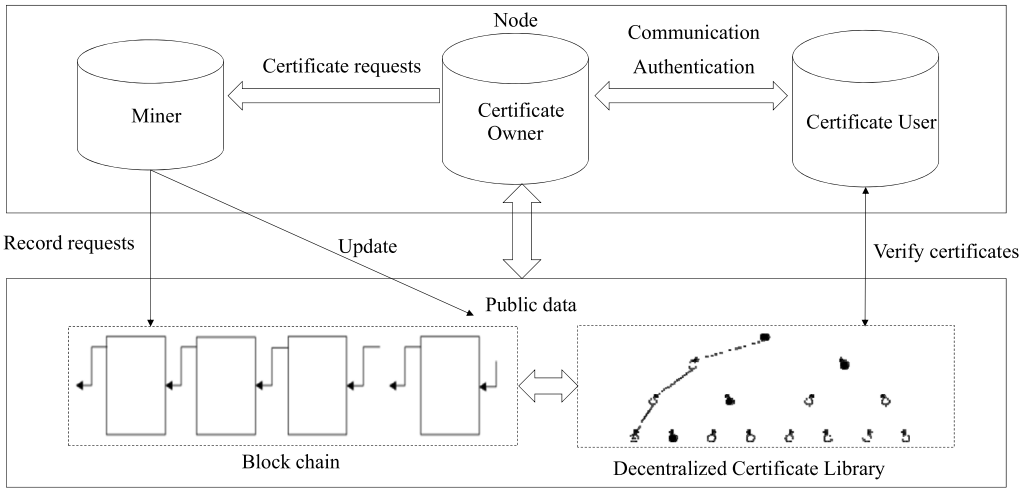


Fig. 1. The system model of Cecoin.

4.3. Application scenarios

The users of Cecoin can be divided into two categories, the one acts as Certificate owner who owns or holds the certificates. The other one acts as Certificate user who uses certificates to protect transmitted information. We will describe the possible application scenarios in Cecoin.

As for the Certificate owner who owns the right, he could submit the requests and applications to the network waiting for validation by miners to manage his own certificates and identities. And there are four practical scenarios:

- *Registry:* In order to get a digital identity, Certificate owner needs to register certificates with identities in Cecoin before communication. They throw their request to the public waiting for miners to valid them into the block. Once succeed, the owner has its legal certificates.
- *Revoking:* When certificates are no longer used, Certificate owners should revoke their certificates to prevent risks, such as certificates being misused by users, or being used to cheat by attackers.
- *Renewing:* Once certificates being expired, Certificate owners could renew them to extend the term of validity under the same private key. Or in case that when the private keys of certificates are lost, owners should renew them with a new private key.
- *Identity Assignment:* When companies amalgamation and acquisition happens, the ownership of original identity should be transferred to its new owner. The association between identities and their public keys will be changed via a transaction in our system.

As for the Certificate user who validates certificates from their owners, they need authenticate identity of the communicator by validate the certificate in Cecoin. As a result, they can able to use the public keys of certificates from other communicators to encrypt transmitted information under the communication.

5. The proposal

In this section we propose a primary prototype inspired by Bitcoin system. Obviously, it can also be deployed on other blockchain-based platforms, such as Ethereum. Considering its stability and robustness, we finally work with similar ideas from Bitcoin.

5.1. System architecture

The system architecture of the prototype is shown as Fig. 2. We employ some existing modules in Bitcoin to achieve decentralization and consistency, including the PoW protocol, P2P network and data structure. To achieve the purposes of our Cecoin, we have modified and added some modules, and our contribution can be briefly summarized as follow.

Firstly, we have changed data structure to add our certificate functions. New member fields are added to the class of block and transaction, which are denoted as *CertMerkleHash* in block header and *cert* in output of transaction. The Merkle Patricia tree is used to achieve a decentralized Certificate Library. It is noted that *CertMerkleHash* in block header is a hash value of the root in the modified Merkle Patricia tree, which could ensure consistency of the decentralized consensus. The implementation details of the modified Merkle Patricia tree will be depicted in the following section.

Secondly, we have modified the original verification mechanism to adapt our system. The modified mechanism has changed the validation of transactions and blocks into verifications of certificate requests from Certificate owner and the consistency of the modified Merkle Patricia tree. The correct associations between users and its public key makes the certificate reliable. We will depict it in detail when we describe the implementation later.

At last, we have implemented the modules of Certificate request, Mining and Authentication in application layer. Certificate request is for Certificate owner to achieve the capability of submitting certificate requests to Miners and the self-management of their own certificates. Mining module is responsible for validating the requests and updating the Certificate Library. Authentication module allows Certificate users to verify the validity of certificates and authenticate identities of communicators, which prevents adversary from impersonated fraud.

5.2. Decentralized certificate library

The structure of the modified Merkle Patricia tree is shown in Fig. 3. As we can see, the height of Branch-Domain and Domain-Cert are both 128, and there are three kinds of nodes in the modified Merkle Patricia tree defined as follow:

- *Branch nodes:* They have seventeen items. The first item is the hash of hashes stored in child nodes. The last sixteen items are child nodes, and it corresponds to each of possible

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.

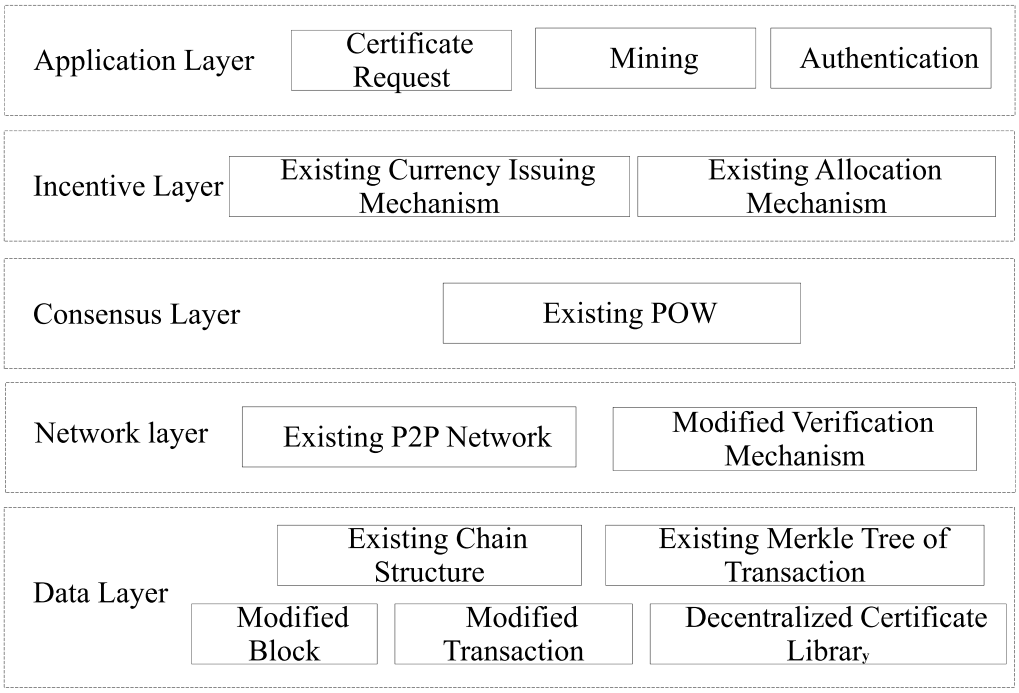


Fig. 2. Implementation architecture of prototype.

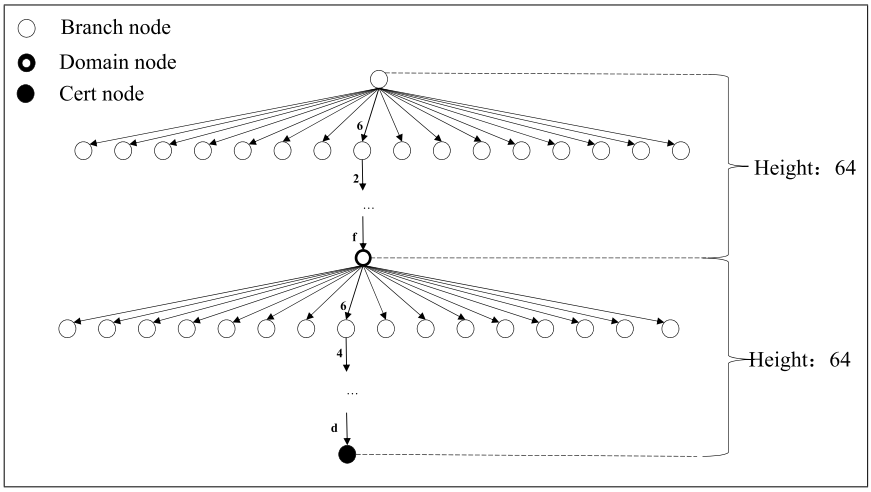


Fig. 3. Structure of the modified Merkle Patricia tree.

nibble values for the keys at this point in their traversal, which is as same as the first sixteen items in Branch of trie in Ethereum.

- *Domain nodes:* They have eighteen items. The first item is the hash of hashes stored in child nodes. The extra item stores the information of address bound with domain. The other sixteen items are the same as Branch nodes. It should be noted that the depth of a Domain node must be 64.
- *Certificate nodes:* Certificates nodes are leaf nodes in the bottom of tree. And they store just two items, certificate and its hash.

For the purpose to improve the efficiency of updating Certificate Library permanently, we have employed Merkle Patricia tree to implement the decentralized Certificate Library. Once Miners execute legitimate certificates requests from Certificate owner, certificates will be correspondingly inserted in or removed out from the Certificate Library, which results in a heavy computation overhead. Merkle Patricia tree from Ethereum is a cryptographically authenticated data structure used to store (*key, value*) pairings. It provides the efficiency of $O(log(n))$ for inserting, retrieving and removing items, which is an excellent selection for our scheme.

The tuple of (*address, state*) in Ethereum is an one-to-one relationship, where the *address* represents account address of user and the *state* is balance owned by its account. However, data in Cecoin is triple, which denotes as (*address, domain, cert*). Notably, both (*address, domain*) and (*domain, cert*) in triples are one-to-many associations. In order to ensure consistency of domain and provide service of multi-certificates, we convert triple (*address, domain, cert*) to a tuple (*key, address, cert*), and *key* represents path of *cert* in the tree. We stipulate that:

$$key = sha256(domain).toHex() + sha256(cert.SerialNumber).toHex()$$

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.

We have provided certificate operations for inserting, removing, and retrieving, which separately denotes as *insert(addr, domain, cert)*, *remove(domain, cert)* and *find(domain, cert)*.

5.3. The modified validation mechanism

We have changed the validation rules of *transactions* and *blocks* to verify certificate requests from Certificate owner and ensure the consistency of decentralized Certificate Library.

5.3.1. Transactions validation

As mentioned in the preceding paragraphs, we have modified the structure of transaction in Bitcoin to make it represent certificate requests. In order to ensure the certificate requests valid, the following validation rules for transactions are added in Cecoin:

- *Fee:* Fee of transactions should be checked. Certificate owner should pay some cecoins to miners for their contributions. We stipulate that each request to register or revoke costs 0.0005 units of Cecoin (CEC), but the request to renew costs 0.001 CEC.
- *ScriptSig:* ScriptSigs of transactions should be checked. The signatures of transactions prevent the case that adversaries steal coins or certificates that do not belong to him.
- *Validity of certificates to be removed:* When transaction represents the process of revoking or renewing, certificates to be removed from decentralized Certificate Library should be checked and make sure they are originally in Certificate Library.
- *Prevention of preemptive registration:* When transaction represents the process of registering a certificate with a new identity, the ownership of identity should be authenticated. The ownership of identity in certificate would be authenticated by sending a challenge to the actual owner.
- *Expiry date of certificates to be inserted:* When transaction represents the process of registering or renewing, certificates to be inserted should be checked, to ensure they are still alive in the period of validity.
- *Consistency of identity:* When transaction represents the process of registering or renewing, the binding pair (*address, domain*) of requests should be checked, to guarantee the consistency of the ownerships of identities.

If all the transaction rules above are satisfied, it could be put into memory pool of transaction. Otherwise, it should be discarded.

5.3.2. Blocks validation

Miner build up blocks to store transactions of certificate requests. For the purpose of decentralization and consistency, miners should mutually verify blocks constructed by each other. And the validation rules for blocks in Cecoin are shown as following:

- *Validate transactions:* To audit misbehavior of the miner who records false certificates in his block, miners should validate all transactions in each block they received (as stated in Section 5.3.1), and make sure that transactions logged in the block are correct and valid.
- *Validate consistency of Certificate Library:* To achieve the consistency of decentralized Certificate Library, miners should check whether *CertMerkleHash* in block header matches to their collected transactions in the block.
- *Check consensus mechanism:* At last, miners should compete to its rewards by validating block header according to Proof of Work mechanism as it is in Bitcoin system.

If all the block rules are valid, miners can accept it. Otherwise, it should be discarded.

```
{
 "hash"      : <hash of the transaction>
  "ver"      : <version>
  "vin_sz"   : <number of the transaction input>
  "vout_sz"  : <number of the transaction output>
  "lock_time" : <lock time>
  "size"     :<size of the transaction>
  "input"    :[
  { "prev_out"  :[
            "hash"  : <hash of the previous transaction referenced>
            "n"     : <order number of the output in the transaction>    ],
      "scriptSig"  : <signature to redeem the output>
  }    {
     "prev_out"   :[
           ……  ]  }  ],
  "output"   :[
    {
       "value"       : <amount of cecoins to be assigned>
       "cert"        : <certificate to be registered>
       "scriptPubKey" : <script to designate recipient>
    }    {
       ……}  ]
}
```

Fig. 4. The special transactions of operations by domains.

5.4. Modular design

In this part, we will describe our modular design, which includes three components: Certificate request, Validation (Mining) and Authentication. Here we give our detail discussion as follow.

5.4.1. Module of certificate request

As described in Section 4, Certificate owner could submit certificate requests to internet waiting for validation with the aim to manage certificates and identities through transactions. We have added a member field *cert* in data structure based on Bitcoin to represent the certificate requests. The structure of transactions in Cecoin is shown in Fig. 4. Here we will discuss it in detail.

Registering: The transaction script is consisted by two types of outputs: coin-output or certificate-output. There is at least one certificate-output together with coin-output. It is noted that the field *value* and *cert* are mutually exclusive, and they cannot be zero/null or non-zero/non-null at the same time. Of course, the transaction should include at least one input pointed by an unspent transaction coin-output to pay for his request.

Revoking: Similar to the registering, a transaction for revoking should include at least one input from an unspent transaction coin-output to pay to the miners. But there are two differences:

- *Output of transaction:* In revoking process, there is none of certificates to be inserted into Certificate Library. Therefore none of outputs in transactions for revoking are certificate-outputs.
- *Input of transaction:* In revoking process, Certificate owner should illustrate the certificate that is removed from Certificate Library in previous transaction. So there is at least one input refers to the previous certificate-output which records the registered certificates.

Renewing: Transactions for renewing consists the process of revoking an expired certificates and registering an new one. A renewing transaction has at least two inputs and one output. For the two inputs, one represents a coin-output to pay to miners, the other should be a certificate-output to illustrate the certificate to be removed.

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.

Identity Assignment: There are two types of transactions to make achieve the identity assignment: one is Buyer-transaction sent by the one who wants to obtain identities, another is assignment-transaction sent by the original owner.

- *Buyer-transaction:* Buyers send his transactions with the purchase cecoins to pay for identities he wants to obtain at first. Note that he would specified the redemption terms in *scriptPubKey* of the coin-output, which means that only when specified identity is assigned to buyer within a given period of time T could the owner redeems the coins. To achieve this, we added an external type of script *op_checkownership* which has two parameters: $<$ *address* $>$ and $<$ *identity* $>$, representing the ownership of a given identity and a Cecoin address. If the address stored in the domain node of $<$ *identity* $>$ matches $<$ *address* $>$, it returns *true*; otherwise, returns false. So the *scriptPubKey* of Buyer-transaction would be: *if* $<$ *s_pubkey* $>$ *op_checksig* $<$ *b_address* $><$ *domain* $>$ *op_checkownerelse* $<$ *b_pubkey* $>$ *op_checksig* $<$ *T* $>$ *op_checklocktime*
- *Assignment-transaction:* Only if owner sends a correct assignment-transaction could the owner redeems coins from the buyer. The assignment-transaction is similar to the renewing process, except for two points: The first is the owner should revoke all certificates bound with the given identities in inputs of assignment-transaction. The second is a new certificate with the identity should be registered in certificate-output, and the certificate-output should specify the Cecoin address of buyer.

5.4.2. Module of validation

The module of validation is for Miner, and it represents the mining process in Bitcoin. Miners in Cecoin record transactions for certificate requests and update the decentralized Certificate Library when they are mining. The process of mining is divided into two parts: *valid transactions collection* and *block construction*.

Valid collection of transactions: The first thing for mining is to collect valid transactions. Only valid transactions can be recorded in blocks and therefore the valid certificate requests can be legally responded.

Block construction: Miners record valid transactions of certificate requests in blocks and update Certificate Library when blocks are constructed. They do following things to build valid Cecoin blocks:

- *Add transactions to block:* When miners construct a block, they add valid transactions from memory pool to the block. The Merkle hash of the transactions would be recorded in block header.
- *Update Certificate Library:* Then miners update the decentralized Certificate Library according to the transactions. Certificates specified in certificate-inputs of transactions would be removed from the modified Merkle Patricia tree, certificates in outputs would be inserted and the expired certificates should be removed. Besides, the hash of the updated Merkle Patricia tree would be recorded in block header.
- *Build block header:* The last thing of block construction is to construct block header with the PoW mechanism to ensure consistency. The block header records the detail information of the block body.

5.4.3. Module of authentication

The module of authentication is for Certificate users, which allow them to authenticate the identity of whoever communicates with them. Certificate users receive and fetch certificates from Certificate owners. And whenever get the new certificates, they will update their Certificate Library according to transactions recorded in the valid blocks. Then, they could authenticate certificates by searching them in their Certificate Library. If there is no such certificates, it turns invalid. Otherwise, the certificates are valid.

6. Security analysis

One of the important purpose of this paper is to enhance security of PKI and to prevent the issuance of pseudo certificates. In this section we will analyze the security performance of Cecoin. Before our formal analysis, we firstly declare the security assumption.

There are three entities in our attack model: **Communicators** consists certificate owner and user. The owner of certificates is well behaved in that he would revoke the false certificates whenever he detects a false certificate with his identity which he has never registered. The user who received certificates is well behaved in that he would disconnect whenever he knows an impersonated certificate. Therefore we assume that both parties in communication are benign and prevent all private keys from leaking. **Miners** are the entities who control the massive computing power. We assume that the computing power in Cecoin is uniform distribution and most of miners in Cecoin are honest. **Adversaries** are the threats of the system with spiteful aims. Here, there are three ways to insert a impersonate certificate into Certificate Library:

- *False certificate requests:* The malicious Certificate owner sends a transaction to register a certificate with identity owned by someone else which already exists in Certificate Library.
- *Preemptive registration:* The malicious Certificate owner sends a transaction for preemptive registration, and attempts to register a certificate with a new identity which is not belong to him.
- *Misbehaved Miner:* The malicious or compromised Miner attempts to insert an impersonated certificate to Certificate Library.

6.1. False certificate requests

In this scenario, the adversary attempts to register a false certificate by binding an identity owned by others. When adversary launches such an attack, he may encounter two situations.

Register the impersonate certificate to adversary's address: If adversary generates a certificate with the association between the victim's identity and his own address and then sends such transactions to public, miners will discard the transaction since the address is inconsistent for the identity with *address* stored in *Domain Node*.

Register the impersonate certificate to victim's address: The reason for failure of registering in previous situation is the inconsistency of addresses. To make it successful, adversary might register the certificate to victim's address to avoid the check of consistency rules. Since there is no violation of the rules, it would work. However, if he did that, he would hand the manipulation right of the certificate over to the victim. As long as victim check certificates owned by him in Cecoin, he would find there is a certificate he had never registered. Victim could revoke it at any time.

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.

Table 1
Compare with related work.

Design goals		CAs	IBE	PGP	Certificate transparency	IKP	Certcoin	Authcoin	Cecoin
Basic function	Registration	✓	✓	✓	✓	✓	✓	✓	✓
	Revoking	✓	✓	✗	✓	✗	✓	✗	✓
	Renewing	✗	✗	✗	✗	✗	✓	✗	✓
	Validation	✓	N/A	✓	✓	✓	✓	✓	✓
Multi-certificate		✓	✗	✓	✓	✓	✗	✓	✓
Identity assignment		✗	✗	✗	✗	✗	✗	✗	✓
Prevention form	False certificate requests	✓	✓	✗	✓	✓	✓	✓	✓
	Preemptive registration	✗	✗	✗	✗	✗	✗	✓	✓
	Misbehaved CAs	✗	✗	✗	✗	✓	✓	✗	✓

6.2. Preemptive registration

Preemptive registration means malicious Certificate owner attempts to register identities or certificates before the actual owners. As describe in Section 5.3.1, when received a transaction for validation, miners would check the ownership of identity in certificate via sending a challenge to the actual owner. If adversary sends a transaction for preemptive registration, the actual owner could refuse to respond the challenge. Then miners would alarm that the certificate is a false one and immediately discard the transaction.

6.3. Misbehaved miner

The last scenario is the misbehaved miner which means a malicious or compromised Miner attempts to insert a false certificate into Certificate Library, and the computing power of the misbehaved Miner is less than 51% in Cecoin. If the misbehaved miner recorded false certificates in his block, other miners in peer will check the block as invalid and then discard it since it contains invalid transaction. For other miners, refusing blocks means that they have a chance to get more rewards as an incentive which push them to monitoring others. Under the assumption of uniformly distributed computing power, the distributed consensus protocol and incentive mechanism will ensure to detect the misbehavior of Miners and make financial penalties.

7. Evaluation

In this section, we will evaluate Cecoin in two parts as follow, one is to assess the completion of goals set before, another is to evaluate the efficacy.

7.1. Assessment on completion of goals

We have evaluated the completion of goals set in previous section and compared with the schemes in related work. And the detailed results are shown in Table 1.

For the positive side, most schemes have achieved the basic functions of registration, revoking and validation. As we can see, IBE does not need to validate the associations of identities and public keys. However, for the negative side, PGP, IKP and Authcoin have not achieved the goal of revoking, and what is more, except for Certcoin and Cecoin, other schemes have not achieved the goal of renewing.

For the actual service, IBE and Certcoin cannot satisfy the requirement of one identity bound with multiple certificates. Except for Cecoin, all schemes cannot support execute identity assignment online without a trusted third party. For the security, as we have analyzed in Section 6, Cecoin could prevent from attacks including but not limit to false certificate requests, preemptive registration and misbehaved CA (or Miner), which has greatly enhanced the security compared to related works.

7.2. Efficiency

In this part, we evaluate the overhead in Cecoin prototype. It consists computing overhead caused by certificate operations and the storage overhead of Certificate Library.

7.2.1. Computation overhead

As we have modified Merkle Patricia tree into an constant height to implement decentralized Certificate Library, the computing overhead for inserting, deleting and searching certificates in the Library will be greatly reduced. The computing overhead for such certificate operations are proportional to the height of the tree. Just as we mentioned in previous section, the height of the modified Merkle Patricia tree is always 128, which is constant, meaning that the computing overhead of Cecoin is correspondingly constant. Therefore the time complexity is $O(1)$ as shown in Fig. 5(a).

7.2.2. Storage overhead

The storage overhead in Cecoin is mainly made up by the Certificate Library. Assume that the number of certificates stored in the distributed certificate library is n and the correspond storage cost of the library is N. In the worst case, not a single one of the storage path in the distributed Certificate Library coincides with others, and the cost reaches to $N = 128*n$. In the best case, all nodes in the distributed storage path in Certificate Library fitly coincides except for leaf nodes, and it reaches to $N = 127 + n$. Therefore, we can conclude that the storage space of N is linear with the number of stored certificate tree n, so the space complexity is $O(n)$ as shown in Fig. 5(b). Notably, for the miners with rich computing and storage resources, they can afford the overhead above.

8. Concluding remarks

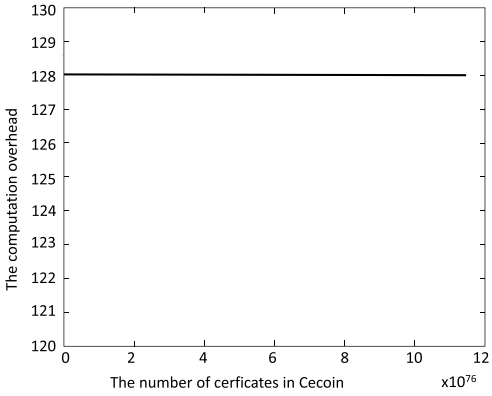
In this paper, we provided a secure enhanced structure of PKI named Cecoin which distributively blockchain-based. The scheme processes the guarantee of consistency to prevent from false certificates. Besides, it provides practical services of multi-certificates and identity assignment. The scheme are achieved in prototype with a desirable efficiency.

However, there still exit improvement spaces in Cecoin. The storage cost of Cecoin is acceptable to the miners, while for other nodes, storing the decentralized Certificate Library will bring a lot of storage overhead. We will design a lightweight client with Simple Payment Verification (SPV) for users of Certificate owners to let them store the concerned nodes in modified Merkle Patricia tree. Besides, we will design a certificate browser for users of Certificate users to let them authenticate and search certificates quickly without having to join the network of Cecoin.

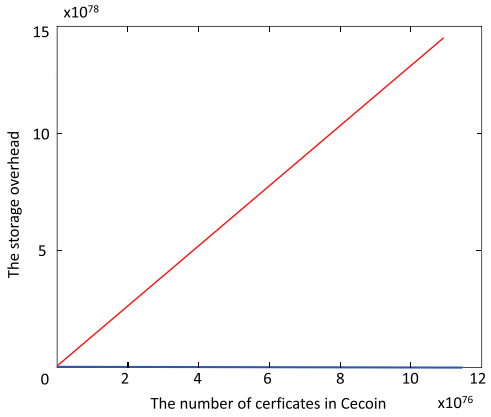
Acknowledgments

This paper is supported by the Natural Science Foundation of China through projects 61772538, 61672083, 61370190, 61532021, 61472429, and 61402029.

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.



(a) Computation overhead curve.



(b) Storage overhead curve.

Fig. 5. The efficient curve and storage overhead curve.

References

[1] C. Adams, S. Lloyd, Understanding Public-key Infrastructure: Concepts, Standards, and Deployment Considerations, Sams Publishing, 1999.
[2] R. Housley, W. Polk, W. Ford, D. Solo, Internet x. 509 public key infrastructure certificate and certificate revocation list (crl) profile, Technical Report (2002).
[3] C.S.R. Murthy, B. Manoj, *Ad hoc* Wireless Networks: Architectures and Protocols, Portable Documents, Pearson Education, 2004.
[4] F. Callegati, W. Cerroni, M. Ramilli, Man-in-the-middle attack to the HTTPS protocol, IEEE Security Privacy 7 (1) (2009) 78–81.
[5] J. Prins, B.U. Cybercrime, Diginotar certificate authority breach. operation black tulip (2011).
[6] C. Comodo, Ltd.: Comodo report of incident-comodo detected and thwarted an intrusion on 26-March-2011, Technical Report (March 2011).
[7] T. SpiderLabs, The web hacking incident database. Semiannual report. July to December 2010, Technical Report, Computer Science in Trustwave Spider Labs (2011).
[8] A. Shamir, Identity-based cryptosystems and signature schemes, in: Workshop on the Theory and Application of Cryptographic Techniques, Springer, 1984, pp. 47–53.
[9] D. Boneh, M. Franklin, Identity-based encryption from the weil pairing, in: Annual International Cryptology Conference, Springer, 2001, pp. 213–229.
[10] S.S. Al-Riyami, K.G. Paterson, Certificateless public key cryptography, in: International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2003, pp. 452–473.
[11] B. Laurie, A. Langley, E. Kasper, Certificate transparency, Technical Report, 2013.
[12] A. Abdul-Rahman, The pgp trust model, EDI-Forum: The Journal of Electronic Commerce 10 (1997) 27–31.
[13] C. Fromknecht, D. Velicanu, S. Yakoubov, CertCoin: A namecoin based decentralized authentication system 6.857 class project.
[14] C. Fromknecht, D. Velicanu, S. Yakoubov, A decentralized public key infrastructure with identity retention, IACR Cryptology EPrint Archive 2014 (2014) 803.
[15] B. Leiding, C.H. Cap, T. Mundt, S. Rashidibajgan, Authcoin: Val- idation and authentication in decentralized networks, arXiv preprint arXiv:1609.04955.
[16] S. Matsumoto, R.M. Reischuk, IKP: Turning a PKI around with blockchains.
[17] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system (2008).
[18] C. Lee, Litecoin (2011).
[19] S. King, Primecoin: Cryptocurrency with prime number proof-of-work, July 7th (2013).
[20] Namecoin, Namecoin: A decentralized open source information registration and transfer system, URL https://namecoin.info/.
[21] H. Kalodner, M. Carlsten, P. Ellenbogen, J. Bonneau, A. Narayanan, An empirical study of namecoin and lessons for decentralized namespace design, in: Workshop on the Economics of Information Security (WEIS), Citeseer, 2015.
[22] C. Cachin, Architecture of the hyperledger blockchain fabric.
[23] V. Buterin, Ethereum: A next-generation smart contract and decentralized application platform, URL https://Github.Com/Ethereum/Wiki/Wiki/%5BEnglish%5D-White-Paper.
[24] G. Wood, Ethereum: A secure decentralised generalised transaction ledger, Ethereum Project Yellow Paper (2014).
[25] S. Haber, W.S. Stornetta, How to time-stamp a digital document, in: Conference on the Theory and Application of Cryptography, Springer, 1990, pp. 437–455.
[26] A. Back, et al., Hashcash-a denial of service counter-measure (2002).
[27] P. Vasin, Blackcoin's proof-of-stake protocol v2 (2014).
[28] J. Kwon, Tendermint: Consensus without Mining, URL http://Tendermint.Com/Docs/Tendermint_V04.Pdf.
[29] M. Castro, B. Liskov, et al., Practical byzantine fault tolerance, OSDI 99 (1999) 173–186.
[30] The Ehereum project, the modified merkle patricia tree, URL https://github.com/ethereum/wiki/wiki/Patricia-Tree.
[31] M.K. Franklin, M.K. Reiter, Fair exchange with a semi-trusted third party, in: Proceedings of the 4th ACM Conference on Computer and Communications Security, ACM, 1997, pp. 1–5.
[32] J. Zhou, D. Gollman, A fair non-repudiation protocol, in: Security and Privacy, 1996. Proceedings., 1996 IEEE Symposium on, IEEE, 1996, pp. 55–61.
[33] I. Bentov, R. Kumaresan, How to use bitcoin to design fair protocols, in: International Cryptology Conference, Springer, 2014, pp. 421–439.
[34] D. Jayasinghe, K. Markantonakis, K. Mayes, Optimistic fair-exchange with anonymity for bitcoin users, in: E-Business Engineering (ICEBE), 2014 IEEE 11th International Conference on, IEEE, 2014, pp. 44–51.
[35] G. Maxwell, The first successful zero-knowledge contingent payment. Bitcoin core https, Bitcoincore. Org/En/2016/02/26/Zero-Knowledge-Contingent-Payments-Announcement (2016).

Bo Qin received her Ph.D. degree in Cryptography from Xidian University in 2008 in China. Since then, she has been with Xi'an University of Technology (China) as a lecturer and with Universitat Rovira i Virgili (Catalonia) as a postdoctoral researcher. She is currently a lecturer in the Renmin University in China. Her research interests include pairing based cryptography, data security and privacy, and VANET security. She has been a holder/coholder of 5 China/Spain funded projects. She has authored over 60 publications and served in the program committee of several international conferences in information security.

Jikun Huang received her B.S. degree in Information Security from Renmin University of China in 2014. She is currently working towards an M.S. degree in Information Security at School of Information, Renmin University of China. Her research interests include computation security, identity authentication and block chain.

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.

Qin Wang received his Bachelor of Engineering degrees in Northwestern Polytechnical University in 2015. He is now pursuing a master's degree in information and communication engineering in Beijing University of Aeronautics and Astronautics. His research interests include distributed computing system security, biometric information security, block chain and cryptographic currencies.

Xizhao Luo is an associated professor with the School of Computer and Technology, Soochow University, Suzhou 215006, China. He received his Ph.D. from Soochow University in 2010 in China. He spent three years as a postdoctor with the center of Cryptography and Code in the School of Mathematical Science, Soochow University. His main fields of interest are Cryptography and Computational Complexity.

Liang Bin received his doctor degree from Institute of Software Chinese Academy of Sciences. He had been with Tsinghua University as postdoctoral fellow and now with Renmin University of China as a professor. His research interests include security Analysis of software system, information security countermeasures and system software security mechanism. He has published tens of publications in these areas.

Wenchang Shi received his Bachelor from Peking University, Master and Ph.D. from Chinese Academy of Science. He is not now a professor of Renmin University of China and an adjunct professor of Institute of Information Engineering, Chinese Academy of Sciences. His research interests include information system security and operation system security. He has published over 100 publications in these areas.

Please cite this article in press as: B. Qin, et al., Cecoin: A decentralized PKI mitigating MitM attacks, Future Generation Computer Systems (2017), https://doi.org/10.1016/j.future.2017.08.025.