A Blockchain-based key Management Scheme for Named Data Networking

Junjun Lou, Qichao Zhang, Zhuyun Qi, Kai Lei*
† Shenzhen Key Lab for Information Centric Networking & Blockchain Technology (ICNLAB)
School of Electronics and Computer Engineering (SECE)
Peking University, Shenzhen 518055, P.R. China
Email: 1601214004@sz.pku.edu.cn, qczhang@pku.edu.cn, qizy@pkusz.edu.cn, Corresponding Author: leik@pkusz.edu.cn

Abstract—**Named Data Networking is built with security which requires each named Data object to be digitally signed by its producer. Thus, the NDN project has proposed a key management model on NDN testbed for verification of the Data packet to be immune to distributing poisoned content. However, in practice, this model poses two challenges for verifying fake content: (1) the centralized architecture easily leads to a single point of failure, especially when the root key fails, its difficult to verify the keys across sites due to the lack of trust between them, and (2) excessive overhead of certificate chain traversal when verifying signature. This paper first proposes a blockchain-based key management scheme in NDN to address the problem of lack of mutual trust between sites without trust anchors. Specifically, all site nodes form a permissioned blockchain for storing public key hashes to ensure the authenticity, and the proxy gateway participates in verifying to reduce excessively frequent communication between the router and the blockchain. In addition, the NDN public key content object and the scheme of their storage, verification, and revocation are redesigned. The result of our analysis and evaluation shows that the proposed scheme is capable of supporting less verification numbers and higher verification efficiency.**

Index Terms—**blockchain, NDN, key management, trust, signature**

I. Introduction

The design of a new network architecture including a suitable security mechanism aims to reduce the cost of computation, storage and transmission as much as possible while guaranteeing the authenticity, integrity, confidentiality of the message. Due to the content-oriented security design principle, Named Data Network, a type of Information-Centric Networking (ICN), is completely different from the traditional IP network based on channel security, which requires SSL/TLS to encrypt the channel at the application layer or the session layer to ensure that data cannot be eavesdropped during transmission. The NDN architecture provides content based security i.e., all applications can sign and verify each packet, which builds data authentication into the network layer.

In order to ensure the authenticity and integrity of the information, the publisher will sign the contents, which effectively and securely binds the names to Data. In this way, consumers (and routers) can verify the signature and determine the provenance of data. This enables Consumer to trust the provenance of data without having to determine how data is obtained or where to get it. At the same time, this supports fine grained trust, and allows Consumer to confirm whether a Public Key owner is a trustworthy publisher of a certain piece of data (Publisher) in a specific case.

The NDN project group has presented a Key trust model on NDN Testbed in its technical report [1], which uses a hierarchical structure, the root key as a common and well-known trust anchor to authenticate the site public key, that is, the site key authenticates the user's public key, and the user key authenticates public keys of applications. The producer then uses the key name to indicate which public key should be retrieved to verify the generated packet signature, and to ensure the integrity and authenticity of the content package. NDN signature and the way of hierarchical management verification avoid the distribution of false content in principle. But in practice, this poses two challenges for verification of signature: (1) As a single center, the root key may be attacked and tampered to cause single point of failure, especially the cross site key validation problem: Each site is a relatively independent trust domain. In the absence of a trust anchor, it is difficult for each site to verify the authenticity of the key issued by each of them; (2) Due to traversing the certificate chain, there is a lot of overhead in signature verification.

Blockchain, through distributed data storage and consensus mechanism, ensures that data can be traced and cannot be easily tampered with. It provides a thought and scheme to establish a trust relationship in the way of decentralization, and has received extensive attention in the field of digital certificates and identity authentication. [2] proposes the first distributed PKI system Certcoin using bitcoin block chain system as framework, which is equivalent to CA, providing efficient key query service. [3] stored the hash of a published or revoked certificate using blockchains. Axon [4] improved the Certcoin scheme in [2] and proposed a PKI authentication system PB-PKI (Privacy-awareness in blockchain-based PKI) for privacy protection. Karen et al. [10] Proposed PKI authentication system based on Ethernet block chain platform, the revocation checking is performed without requiring CRLs or OCSP. Matsumoto [5] introduced the economic incentive method, and proposed a PKI IKP(Instant Karma PKI) based on Ethernet's timely response. All these studies are based on IP networks, which are not completely applicable to the new network NDN. This paper proposes a NDN key authentication and management scheme based on blockchain technology: that

978-1-5386-4870-4/18/$31.00 ©2018 IEEE

is, to set up each site as blockchain node, providing public key hash storage, verification (public key hash is published in the blockchain and is queried through the proxy gateways) and revocation service. This scheme is based on the permissioned blockchain so it is highly scalable.

The main contribution of this paper is summarized as below:

- For the first time, we used blockchain technology for NDN key management and solved the problem of mutual trust between sites. That is, trust between sites is based on cryptography rather than on endorsement.
- This scheme shortens the original multi-layer public key verification chain, reduces the number of signature verification, and improves the verification efficiency.

The remainder of this paper is organized as follows. Section II briefly introduces the background knowledge of NDN, the key trust model on NDN Testbed and the blockchain technology. Section III is devoted to the design of the proposed NDN key management scheme based on blockchain technology. Section IV discusses related work. Section V is about the analysis and assessment of security and efficiency, while Section VI concludes this paper and outlines our future work.

II. Background

A. NDN trust management

NDN is a network architecture based on named data, and there are two types of packets: Interest packets and Data packets. The NDN packet has a hierarchical, readable name that uniquely identifies the content. NDN Communication is a consumer-driven content transfer. The consumer requests the content by sending an Interest package that carries the name of the content. The forwarding mechanism of NDN includes three kinds of data structure: Content Store (CS); Forwarding Interest Base (FIB) and Pending interest Table (PIT). If the data is cached in the local Content Store of the intermediary router, the data is forwarded to the consumer immediately. Otherwise, the router forwards the Interest to the producer. Finally, the requested content will be returned to the consumer via the opposite path of the routing.

A Data consists of four parts: Signature, Name, SignedInfo and Content. ”Signature” is gained through signature by Producer using its public key to bind the ”Name” and ”Content”, this ensures the integrity and the original authentication of the specified data. ”SignedInfo” records information about Producer’s signature. Some of the key fields in SignedInfo element are:

- PublisherPublicKeyDigest: The digest of the publishers public key, used to verify the content source.
- Timestamp: The timestamp when the content is published.
- Type: Type of the content, including DATA, ENCR, GONE, KEY/, LINK, NACK.
- FreshnessSeconds: Used to judge if the content is stale or not.
- KeyLocator: May be the actual key or certificate of the publisher, or a link to the key or certificate.

Verifying the signature of a Data packet requires the application to obtain the appropriate public key. A digital signature is meaningless if the public key cannot be obtained, or if the public key being obtained is false. Therefore, key management, that is, to ensure that the key can be acquired and authentic is critical.

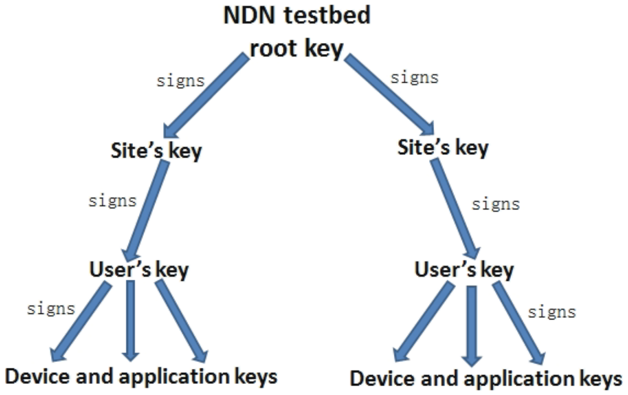


Fig. 1. Key trust model on NDN testbed.

[1] designed a hierarchical trust management model on NDN Testbed. As shown in Figure 1, the NDN testbed has a root key, as a common and well-known anchor, to provide key signature for each site to ensure its authenticity. In the same way, the key of each site makes signatures to the user’s public key under the site, and then user’s key makes further signatures to its device and specific application. The public key can be acquired as an ordinary Data packet and exchanged with Interest. If the authenticity of a public key is to be verified, the upper public key for the public key signature is obtained from the ”KeyLocator/KeyName” field in the public key Data package, while the root key is trusted to retrieve based on the public key chain to verify the authenticity of the public key. As known, because the root key is configured in advance, it can ultimately verify the authenticity of the public key.

B. Blockchain

Blockchain is a distributed ledger technology derived from bitcoin. As shown in Figure 2, its data structure is a list of links in a series of blocks constructed with hash pointer. Each block contains a series of transaction data, which ensures that data will not be tampered with in the way of cryptography [6]. Every blockchain should resolve consensus problem which means all the participator should assent to the order of blocks and have the same capacity of knowledge about the blockchain of the system[12].

An attractive and distinctive feature of blockchain is its decentralized network that enables it to construct trust between nodes in a decentralized system without a trusted third party endorsement, this eliminates single point of failure, and realizes the decentralization and distributed trust mechanism to complete the transfer of information at the same time the transfer of value. The core components of blockchain include:

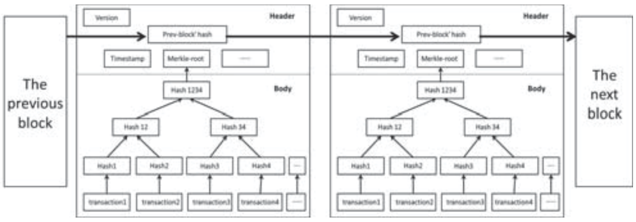


Fig. 2. Data structure of blockchain.

network communication, distributed storage, security mechanism, distributed consensus, smart contract. In this paper, there are two kinds of transactions stored in the Blockchain system: Authentication transaction and Revocation transaction.

III. DESIGN OF BLOCKCHAIN-BASED KEY MANAGEMENT SCHEME FOR NDN

A. Overview

In this section, we will introduce the design of the NDN key management scheme based on blockchain, so as to solve the problem of lack of trust between the sites in the case of no trust anchors, and improve the key verification efficiency of the key management model proposed by the NDN project group.

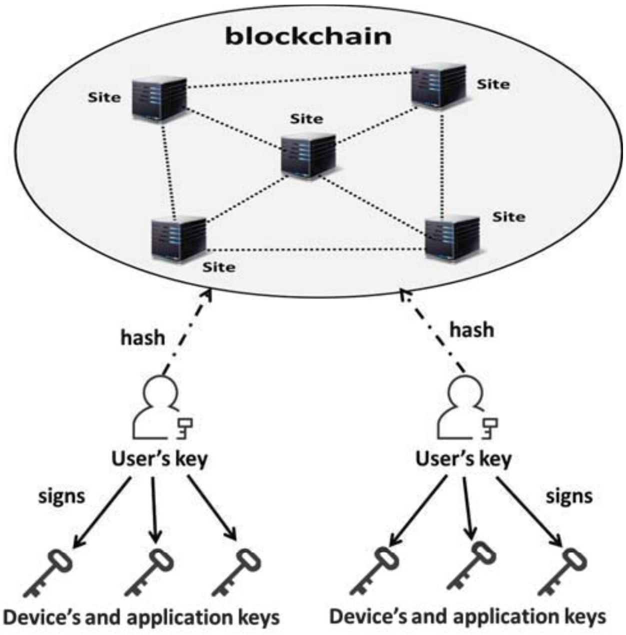
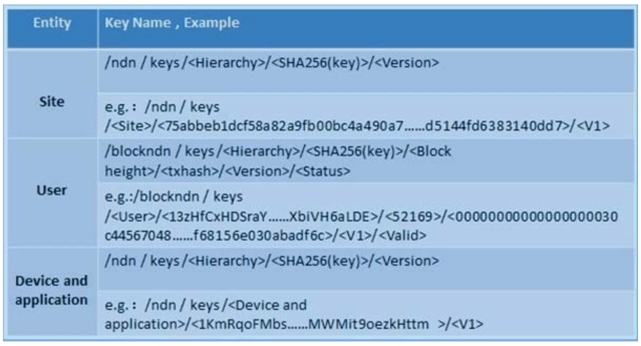


Fig. 3. Blockchain-based key management model for NDN.

Figure 3 shows our key management model and shows the naming scheme of our system. In order to avoid single point of failure, we do not consider setting root key. The blockchain composed of sites is the starting point of trust, and each site is regarded as a consensus node. Since the starting point of trust is crucial to the whole key verification chain, the entry of site nodes needs a certain access mechanism, and the permissioned blockchain becomes our first choice for the sake of higher scalability and efficiency. As a trust anchor, the site is used to store the hash of the user public key in the site to ensure the authenticity of the user's public key, and the user's key will make further signatures to its devices and specific applications. In NDN, when the application creates data packets, it will fill in the full name [1] of the key used to sign this packet in the "KeyLocator / KeyName" field. Therefore, in order to verify the data key, everyone should be able to get the user's public key. We first describe the naming mechanism of all keys, and then design a key storage, authentication and revocation scheme based on blockchain.

B. Naming

The data names in NDN follow a hierarchical naming mechanism. Since the public key can be obtained as an ordinary Data packet, the naming of the key also follows this rule. The naming mechanism of this scheme can be divided into two categories: one is the naming of user keys, the other is the naming for sites, applications and device keys.



Entity	Key Name , Example
Site	/ndn / keys /<Hierarchy>/<SHA256(key)>/<Version>
	e.g. : /ndn / keys /<Site>/<75abbeb1dcf58a82a9fb00bc4a490a7......d5144fd6383140dd7>/<V1>
User	/blockndn / keys /<Hierarchy>/<SHA256(key)>/<Block height>/<txhash>/<Version>/<Status>
	e.g.:/blockndn / keys /<User>/<13zHfCxHDSraY......XbiVH6aLDE>/<52169>/<000000000000000000030c44567048......f68156e030abadf6c>/<V1>/<Valid>
Device and application	/ndn / keys /<Hierarchy>/<SHA256(key)>/<Version>
	e.g. : /ndn / keys /<Device and application>/<1KmRqoFMbs......MWMit9oezkHttm >/<V1>

Fig. 4. Key naming mechanism.

The user key is named as "/blockndn / keys / $< Hierarchy >$ / $< PublicKeyHash >$ / $< BlockHeight >$ / $< TransactionHash >$ / $< Version >$ / $< Status >$. As shown in Figure 4, each of the fields is explained as follows:

- Prefix "/ blockndn / keys": it means that it is the name of the user's public key that needs to be routed to the site blockchain.
- $< Hierarchy >$: /$< site >$ /$< user >$, which means that it is the user layer key.
- $< PublicKeyHash >$: it means the hash of the corresponding public key (SHA256), which will also be stored in the blockchain for verifying the authenticity of the corresponding public key.
- $< BlockHeight >$ / $< TransactionHash >$: this part indicates the location of the public key hash in the blockchain: Block Height $<$ 52169 $>$, Transaction Hash$<$ $...30c445670484d37a3d1f3116a2ff6f68156e0... >$.
- $< Version >$: version number, which represents editions.
- $< Status >$: this part includes "valid" or "invalid" and is used to revoke invalid public key.

The naming format of sites, application and device keys is similar to that in [1]: "/ndn / keys / $< Hierarchy >$ /" $< PublicKeyHash >$ /$< Status >$. Each of the fields is explained as follows:

- Prefix "/ ndn / keys": it is the name of the public key (it may be the site key, or the application and device key).
- $< Hierarchy >$:/$< site >$, which indicates that it is the key of the site layer, and /$< site >$/$< user >$/$< device >$ indicates that it is the device key, and $< site >$/$< user >$/$< application >$ indicates that it is an application key.
- $< PublicKeyHash >$: it represents the hash (SHA256) of the corresponding public key.
- $< Version >$: it is used to revoke invalid public key (introduced in E).

C. Authentication

In order to prove the authenticity of the user's public key, we do not use the way that the site makes signatures to the user's public key, but use the blockchain to store the user's public key hash. For this reason, this paper redesigns the user public key packet, a user public key packet based on blockchain, to adapt to the new key management model (note that this part only refers to the user's key content object). The hash of this public key packet is linked to the blockchain by multiple sites of the permissioned blockchain as a non tampered credential. The comparison between key content object to the key content object based on blockchain designed in this paper is shown in figure 5.

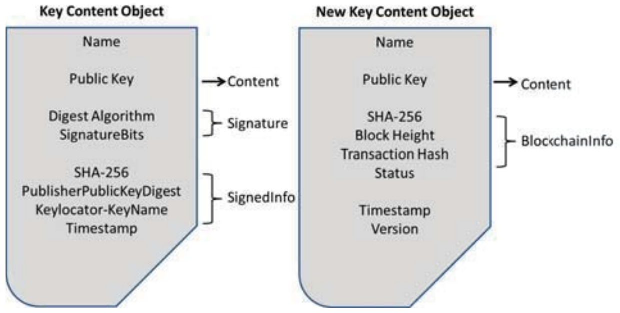


Fig. 5. The comparison of key object and New key object.

The new user's key content object designed in this paper mainly consists of three parts: naming, content and blockchain information.

- Naming represents the name of content object.
- The content is public key.
- Hash $(SHA - 256)$ means the summary value of the public key.
- Block Height and Transaction Hash present the storage location of the user key hash in the blockchain, which is used for fast look-up during validation.
- Status means whether the key has been revoked.
- Other fields include timestamps, version, and so on.

Compared with the original user's key content object, the advantages of key content object based on the blockchain include:

- The signature is omitted. In the original key content object, the name and public key are bond via site key signature to ensure that the user's public key is not tampered with; In this paper, The data on blockchain is not easy to be tampered with, so we decide to make sites form permissioned blockchain to store the hash of the content object as a trust credential to replace the signature process.
- The field KeyLocator / KeyName is omitted. The way to look up hash is used to replace the way to look up the public key through the "KeyLocator / KeyName" field.
- Block height and transaction hash in blocks are added.

After the user public key is authenticated by the site, its hash is stored in the blockchain. Because NDN has a cache mechanism, all the hash of the user public key objects stored can be obtained by other people from intermediate router.The process of authentication is as follows: (i) The user creates an "Authentication transaction" and writes the hash of public key into the transaction (the user doesnt have to sign the public key hash again because the transaction in the blockchain needs to be signed by the user); (ii) The user sends the "Authentication transaction" to his trust domain administrator to apply for authenticate; iiiAfter receiving the user's authentication request, the trust domain administrator signs the transaction if it is judged to be legal after the checking, and broadcasts it to the blockchain network; (iv) After authentication by the consensus node, the "Authentication transaction" is packaged and recorded in the blockchain, so that the public key hash is also stored in the blockchain through the "Authentication Transaction" (This paper uses the Practical Byzantine Fault Tolerance algorithm as the consensus algorithm); (v) A trust domain administrator returns the Block Height of the "Authenticated transaction" to the user, who writes the Block Height and the Transaction Hash to the public key package (i.e., the key object of the user).

D. Verification

Because there are different forms of user public key and application and device public key in this scheme, the way of public key verification also differs.

The way to verify the application and device public key is like this: To check the "KeyLocator" field contained in the application or device public key content object, and find the "key name", the key itself, or the certificate in the "KeyLocator" field. The name is used to find the corresponding user public key to verify the public key of this application or device, if the user's public key is real, then the public key verified successfully by it is real. Otherwise, the situation is not like this.

Therefore, the next step is to verify the authenticity of the user's public key. The process of verification is as follows:

(i) When the consumer (or router) receives the user's public key, checks the $< BlockHeight >$ and the $<$

$TransactionHash >$ field in the Public key packet, which determines the position of the "Authenticated transaction" containing the public key hash in the blockchain so as to facilitate quick lookup; (ii) The consumer quickly finds the "Authentication transaction" by $< BlockHeight >$ $< TransactionHash >$, and obtains the public key hash that has already been written to "Authentication transaction", and calculates the obtained user public key hash by SHA-256 algorithm; (iii) Comparing the hash stored in the blockchain with the computed hash, if the two are the same, the public key hash is proved to be true, otherwise the public key obtained is not the user's public key in the blockchain.

E. Revocation

It is also required to consider such a situation: when the user's private key is leaked, the attacker will perform signatures to the false content with the private key, and the corresponding public key is still verifiable, which will lead to the spread of more false content. Moreover, the cache mechanism of NDN enables key content objects to be distributed in the routers of the network. Therefore, it is necessary to design a revocation mechanism for the public key.

For the application and device public key, if the producer needs to revoke the published public key, then the version number of the public key shall be changed to V0, which indicates that the public key has failed. After receiving the key content object, the consumer needs to check its version number. If it is V0, it shall be discarded.

While for user public key, the way of revocation is different. In the blockchain, each block contains the hash value of the previous block. The consensus mechanism ensures that transaction information can be recognized and recorded without authority. Therefore, the data that has been written to the blockchain can not be changed, which means we cannot make actual deletion on the user public key hash in the blockchain. A new revocation scheme on user public key is proposed in this paper. This scheme requires users to store public key hash and state of public key into blockchain, this means that after the user's public key fails, the producer will again put its public key hash and invalid state together in the blockchain to obtain the Block Height and Transaction Hash of this invalid public key hash. In this way, when the router gets the user's public key, the hash and its latest state of the public key can be found in the blockchain through the fields of $< BlockHeight >$ and $< TransactionHash >$ in the key content package. The revocation process is as follows: (i) The user creates a "Revocation transaction" that will write the public key hash to be revoked and invalid status $< Invalid >$ into the transaction (the transaction in the blockchain needs to be signed by the user so that the user doesnt have to sign additionally the public key hash and invalid state); (ii) The user sends a "Revocation transaction" to his trust domain administrator to apply for revocation; (iii) After receiving the user's revocation request, the trust domain administrator signs the transaction if it is judged to be legal after the checking, and broadcasts it to the blockchain network; (iv) After verification by the consensus node, the "Revocation transaction" is packaged and recorded in the blockchain, so the public key hash and invalid state $< Invalid >$ are also stored in through "Revocation Transaction"; (v) The trust domain administrator returns the $< BlockHeight >$ of the "Revocation transaction" to the user, and the user regenerates an invalid public key packet (e.g., the key object of the user) which contains the $< BlockHeight >$, $< TransactionHash >$, and status $< Invalid >$.

IV. Related works

There are no so many related works on NDN key management, including [7], which proposed a key management system based on endorsement (Endorsement) inspired by the concept of Web-of-Trust (users can build trust to another user through a web consisting of introductions made by others) to protect ChronoChat (a serverless group chat application over NDN). Through this system, users in chat rooms can cooperate to manage the membership of chat rooms and cooperate to authenticate members' membership. [8] proposed new NDN certificate format and discussed several approaches of serving certificates in NDN, as well as how to use new certificates to design revocation certificates. [9] designed a building automation management system based on data, and proposed a hierarchical name-space to simplify user authentication and access control. [10] proposed a hybrid scheme based on public key infrastructure (PKI) and hierarchical identity based encryption (HIBC) to resist attacks. And [11] recommended to use the IKB rule to enable NDN participants (consumers, producers and routers) to alleviate content poisoning, and minimize the trust related complexity of the router synchronously.

V. Analysis and Assessment on Security and Efficiency

A. Analysis and Assessment on Efficiency

Taking public key authentication through public key chain as an example, in [1], it is required to search public keys and verify for 3 times from application and device public key to root key. While in the blockchain based scheme in this paper, it can reduce the level of the public key verification chain. It only needs to search 2 times and verify for 2 times, and one of them is the search and verification of the hash. Same argument between signatures and publishing key, to this end, this scheme reduces the number of digital signatures and public key verification.

NDN uses public key cryptosystem to distribute key. This scheme only uses the public key algorithm when the user public key is signing and verifying the public key of the application and device. In the authentication and verification of user public key, this scheme adopts SHA-256 hash algorithm. As we know, the SHA-256 hash algorithm is more efficient than the public key signature algorithm$(suchasRSA-1024)$. Therefore, compared with the scheme in [1], this scheme has obvious advantages in the computation cost and efficiency of key verification. In [7], the scheme based on the endorsement is a distributed system, while the scheme based on alliance chain in the paper uses the PBFT consensus algorithm which can deal with thousands of transactions per second, so it is more efficient than the completely decentralization framework so as to meet the needs of key management.

B. Analysis and Assessment on Security

The scheme does not set up the root key, and the site constitutes the blockchain as a trust anchor, which guarantees the authenticity of the next layer key by storing hash. The blockchain is a decentralized system, and the confirmation of messages between nodes is based on cryptography instead of trust. The failure of any node in the blockchain does not affect the availability of the whole system. This avoids the single point of failure.

Considering the limited payload size of the block chain, we do not store the entire user public key content object in the blockchain, and the hash used to store user's public key is only 256 bits (256bits). According to the characteristics of blockchains, these stored hash cannot be tampered with easily. At the same time, the blockchain solves the cross site key authentication problem. In the case of no root key, the user key signed and published by the A site cannot be verified by the user of the B site, because the B does not trust the site which lack of "authority" of the endorsement. And this scheme can eliminate this phenomenon.

VI. CONCLUSION AND FUTURE WORK

In this paper, we propose a NDN key management system based on blockchain technology. We do not blindly or mechanically copy the key management mode in the IP network, nor do we grips the blockchain to the trust model of NDN, but use the advantages of blockchain(such as distributed, data cannot be easily tampered with) to propose a key signature and verification method similar to partial decentralization. In this scheme, a more flat hierarchy reduces the number of signatures and authentication keys. The way to store hash is used to replace signature, and the way of query and comparison to hash is used to perform signature verification, this brings less computation cost than public key cryptography. In addition, the redesign of the key content object provides a more efficient way to look up hash in the blockchain. Our future work will focus on the deployment and implementation of the proposed scheme in IOT [13]. We will combine specific applications (such as video playback) to build systems and conduct systematic reviews, including computing, storage overhead, efficiency of key distribution, and so on to implement our key management scheme. We will also explore more efficient public key hash synchronization mechanism and trust credential look-up mode to achieve faster key signing and verification. Moreover, based on this, we will combine with routing, naming mechanism to deploy our scheme in more scenarios to evaluate its feasibility and superiority in mitigating NDN cache pollution and other security issues.

ACKNOWLEDGMENT

This work has been financially supported by Shenzhen Key Fundamental Research Projects (No. JCYJ20160330095313861, JCYJ20170412150946024, JCYJ20170412151008290 and 2014GKXM054).

REFERENCES

[1] NDN Project, Deploying Key Management in NDN Testbed. Named Data Networking Tech. Report 009, Tech, Rep, February 2013.

[2] Fromknecht C, Velicanu D, Yakoubov S. CertCoin: A NameCoin Based Decentralized Authentication System[J]. Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep, 2014, 6.

[3] Lewison K, Corella F. Backing Rich Credentials with a Blockchain PKI. Tech. Rep. Pomian Corella, LLC, 2016.

[4] Axon L. Privacy-awareness in blockchain-based PKI[J]. Oxford University Research Archive, 2015.

[5] Matsumoto S, Reischuk R M. IKP: Turning a PKI Around with Decentralized Automated Incentives[C] Security and Privacy. IEEE, 2017:410-426.

[6] A. Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction. Princeton University Press, 2016.

[7] An Endorsement-based Key Management System for Decentralized NDN Chat Application. NDN, Technical Report, NDN-0023, 2014.

[8] Public Key Management in Named Data Networking. NDN, Technical Report, NDN-0029, 2015.

[9] Shang W, Ding Q, Marianantoni A, et al. Securing building management systems using named data networking[J]. IEEE Network, 2014, 28(3):50-56.

[10] Hamdane B, Serhrouchni A, Fadlallah A, et al. Named-Data security scheme for Named Data Networking[C] Network of the Future. IEEE, 2013:1-6.

[11] Ghali C, Tsudik G, Uzun E. Network-Layer Trust in Named-Data Networking[J]. Acm Sigcomm Computer Communication Review, 2014,44(5):12-19.

[12] Jin T, Zhang X, Liu Y, et al. BlockNDN: A bitcoin blockchain decentralized system over named data networking[C]. Ninth International Conference on Ubiquitous and Future Networks. IEEE, 2017:75-80.

[13] Lei K, Zhong S, Zhu F, et al. A NDN IoT Content Distribution Model with Network Coding Enhanced Forwarding Strategy for 5G[J]. IEEE Transactions on Industrial Informatics, 2017, PP(99):1-1.