Hindawi
Security and Communication Networks
Volume 2018, Article ID 9641273, 6 pages
https://doi.org/10.1155/2018/9641273

Research Article

A Homomorphic Network Coding Signature Scheme for Multiple Sources and its Application in IoT

Tong Li[1], **Wenbin Chen**[1], **Yi Tang**[2] **and Hongyang Yan**[3]

[1]*School of Computer Science, Guangzhou University, Guangzhou, China*
[2]*School of Mathematics and Information Science, Guangzhou University, Guangzhou, China*
[3]*College of Computer and Control Engineering, Nankai University, Tianjin, China*

Correspondence should be addressed to Yi Tang; ytang.bjs@139.com

Received 13 January 2018; Revised 21 March 2018; Accepted 15 April 2018; Published 14 June 2018

Academic Editor: Ilsun You

As a method for increasing throughput and improving reliability of routing, network coding has been widely used in decentralized IoT systems. When files are shared in the system, network coding signature techniques can help authenticate whether a modified packet in files is injected or not. However, in an IoT system, there are often multiple source devices each of which has its own authentication key, where existing single-source network coding signature schemes cannot work. In this paper, we study the problem of designing secure network coding signatures in the network with multiple sources and propose the multisource homomorphic network coding signature. We also give construction and prove its security.

1. Introduction

With the rapid development of Internet and communication technologies, the Internet of Things (IoT) has emerged as a leading technology that brings convenience to our daily lives. More and more smart terminals are connected on the Internet, and files, logs, and other real-time contents are shared among these terminals all the times. According to a report of International Data Corporation (IDC), there will be nearly 28 billion installed IoT devices by 2020.

Considering the scale of IoT's expansion, it is very essential to increase its throughput in such a huge decentralized network. When a source device transmits a file to a set of target receivers, an effective way is to split the file into t data packets and send them to its neighbouring nodes by using the network coding technique. In the network coding, each intermediate node linearly combines packets rather than simply storing and forwarding the incoming packets. In other words, an intermediate node that receives a set of packets from its incoming links can modify them and send the modifications to other nodes through its outgoing edges. In some applications, either an ad hoc node or an intermediate device can play a role of an intermediate node. This linear network coding allows receivers to recover the original information with high probability if they collect sufficiently many correct packets. Thus, the throughput for sharing real-time contents in IoT is increased.

However, security is one of the most important requirement of IoT systems, and IoT devices often interact with third-party applications. Without authenticate mechanisms, the inherent flaw of linear network coding would be disturbed by invalid packets injected by third-party applications. Intermediate nodes can later use the invalid incoming vectors in its output, which means that the errors are propagated subsequently and data receivers will not obtain the original information. As a result, adversaries could easily initiate a Denial of Service (DoS) attack to prevent the original file from being recovered. The main idea to mitigate attacks is to provide a way to authenticate valid packets, and Catalano et al. [1] proposed an efficient network coding signature scheme as a solution of the authentication problem. By verifying a modified signature of the corresponding modified packet, any device can easily know whether this packet is valid.

Unfortunately, in an IoT system, origin data are usually collected from various sources (e.g., sensors) each of which could have its own signature for authentication. It is required

that any (intermediate) receiver can perform the combination of incoming packets which are signed by different keys. As a drawback, trivially adopting the existing network coding has to generate signatures are linear in the number t of the sources, and thus the signatures cannot be directly combined when packets are modified. Motivated by this problem, in this paper, we propose a multisource linearly homomorphic network coding signature scheme. The proposed scheme is extended from our previous work [2] and enables a multilayers routing network rather than a 3-layer one, which can be used to implement authentication for transmitting files in the IoT system.

The rest of this paper is organized as follows. Section 2 presents some related works. Section 3 overviews some definitions. In Section 4, we describe our multisource linearly homomorphic network coding signature scheme. Section 5 analyzes the correctness and security of the proposed scheme. In Section 6 we summarize the paper.

2. Related Works

In the traditional network routing, every node simply receives packets and forwards them to neighbour nodes. A routing method called network coding [3, 4] is proposed and developed for increasing throughput in the network. In the network coding, intermediate nodes combine received data packets and transit them and the data receiver still obtains the original data. This technique can be used in IoT applications and cloud systems for broadcast and transmission [5–19].

In the single-source scenario, some schemes were proposed to make sure that there is always a recipient bound to the corresponding for authentication. M. Krohn et al. introduced the homomorphic hash function [20, 21] and extended it to network coding. The linearly homomorphic signature is a more effective authentication for the network coding. Reference [22] proposed the first linearly homomorphic signature scheme. Reference [23–25] found some security flaw and errors, and Yu et al. [26] gave a construction by combining the RSA-based signature with the homomorphic hash function. Reference [27, 28] designed signature schemes for peer-to-peer networks and distributed contents respectively. Reference [29, 30] proposed homomorphic network coding signature schemes based on the bilinear mapping and RSA assumption respectively. In [31], Boneh et al. designed a signature scheme with the property of signing unlimited number of messages. Based on the complexity of lattice problems, [32] introduced the k-SIS problem and constructed a signature scheme over binary fields. For a fine-grain access control, [33, 34] proposed schemes based on the identity-based signature. The schemes above are proven secure in the random oracle model. In the standard model, some homomorphic network coding signature schemes were proposed [1, 34–36]. The security of the scheme in [35] is based on the discrete logarithm assumption. Independent of these works, [37] proposed a method that transforms standard signature schemes to linearly homomorphic signatures in the standard model.

However, in a multisource case which is the common scenario in the IoT system, there is still no linearly homomorphic network coding signature scheme. Our goal is to design a multisource linearly homomorphic network coding signature scheme.

3. Preliminaries

Then, we show some definitions of the linearly homomorphic network coding signature as follows.

Definition 1 (linearly homomorphic network coding signatures adapted from [1]). A linearly homomorphic network coding signature scheme $\mathcal{LS}$ consists of a tuple of probabilistic, polynomial-time algorithms ($NetKG$, $NetSign$, $NetVer$, $NetEval$) with the following functionality.

NetKG$(1^\lambda, m, n) \to (pk, sk)$. Given the security parameter λ and m, n, this algorithm outputs a key pair (sk, pk), where sk is the secret key and pk is the public verification key. Note that m is the dimension of the vector spaces and $n + m$ is an upper bound to the size of the signed vectors.

NetSign$(sk, Id, w) \to \sigma$. The signing algorithm takes a secret key sk, a file identifier $Id \in \mathbb{F}_p$ and a vector $w \in \mathbb{F}_p^{n+m}$ as input and then outputs a signature σ.

NetVer$(pk, Id, w, \sigma) \to accept$. Given the public key pk, a file identifier Id, a vector $w \in \mathbb{F}_p^{n+m}$, and a signature σ, the algorithm outputs a bit *accept* represents accept or reject.

NetEval$(pk, Id, \{(w_i, \alpha_i, \sigma_i)\}_{i=1}^{\mu}) \to \sigma$. Given a public key pk, a file identifier Id, and a set of tuples $(w_i, \alpha_i, \sigma_i)$, this algorithm outputs a new signature σ such that if each σ_i is a valid signature on vector w_i, then σ is a valid signature for w obtained from the linear combination $\sum_{i=1}^{\mu} \alpha_i w_i$.

For *correctness*, it is required that if the key pair (sk, pk) is output by **NetKG**$(1^\lambda, m, n)$, then

(i) let $Id \in \mathbb{F}_p$ and $w \in \mathbb{F}_p^{n+m}$; if $\sigma \leftarrow$ **NetSign**(sk, Id, w), then **NetVer**$(pk, Id, w, \sigma) = 1$;

(ii) for all Id, any $t > 0$ and all sets of triples $\{(w_i, \alpha_i, \sigma_i)\}_{i=1}^{\mu}$; if **NetVer**$(pk, Id, w_i, \sigma_i) = 1$ for all i, then **NetVer**$(pk, Id, \sum_{i=1}^{\mu} \alpha_i w_i$ and **NetEval**$(pk, Id, \{(w_i, \alpha_i, \sigma_i)\}_{i=1}^{\mu})) = 1$.

The definition of unforgeability of linearly homomorphic signature is presented as follows.

Definition 2 (unforgeability adapted from [32]). For a linearly homomorphic network coding signature scheme $\mathcal{LS} = (NetKG, NetSign, NetVer, NetEval)$, the following game is considered.

Setup: The challenger runs $NetKG(1^\lambda, m, n)$ to obtain (sk, pk) and gives pk to $\mathcal{A}$.

Queries: Proceeding adaptively, $\mathcal{A}$ specifies a sequence of data sets $\mathbf{w}_i$. For each i, the challenger chooses Id_i uniformly

from $\mathbb{F}_p$ and gives to $\mathcal{A}$ the tag Id_i and the signatures $\sigma_{ij} \leftarrow NetSign(sk, Id_i, w_{ij})$ for $j = 1, \ldots, k$.

Output: $\mathcal{A}$ outputs a file identifier Id^*, a message m^*, and a signature σ^*. The adversary wins if $NetVer(pk, Id^*, w^*, \sigma^*) = 1$, and either (1) $Id^* \neq Id_i$ for all i or (2) $Id^* = Id_i$ for some i but $w^* \notin \text{span}(\mathbf{w}_i)$, where $\text{span}(\mathbf{w}_i)$ is the subspace generated by all $\mathbf{w}_i$.

The advantage of $\mathcal{A}$ is defined to be the probability that $\mathcal{A}$ wins the security game. $\mathcal{LS}$ is called unforgeable if for any PPT adversary $\mathcal{A}$, the advantage in the game is negligible in λ.

Let $e : \mathbb{G} \times \mathbb{G}' \rightarrow \mathbb{G}_{\mathbb{T}}$ be a bilinear map, where $\mathbb{G}, \mathbb{G}'$ and $\mathbb{G}_T$ are bilinear groups of prime order p. In [38], Boneh and Boyen introduced the definition of the q-Strong Diffie-Hellman Assumption (q-SDH for short).

Definition 3 (q-SDH assumption [38]). Let $k \in \mathbb{N}$ be the security parameter, $p > 2^\lambda$ be a prime, and $\mathbb{G}, \mathbb{G}', \mathbb{G}_{\mathbb{T}}$ be bilinear groups of prime order p. Let g be a generator of $\mathbb{G}$ and g' be a generator of $\mathbb{G}'$, respectively. Then we say that the q-SDH Assumption holds in $\mathbb{G}, \mathbb{G}', \mathbb{G}_{\mathbb{T}}$ if for any PPT algorithm $\mathcal{A}$ and any $q = \text{poly}(k)$, the following probability is negligible in λ:

$$\Pr\left[\mathcal{A}\left(g, g^x, g^{x^2}, \ldots, g^{x^q}, g', \left(g'\right)^x\right) = \left(c, g^{1/(x+c)}\right)\right] \leq \text{negl}(\lambda) \quad (1)$$

4. The Proposed Scheme

4.1. Architecture. Consider an application in practical. A log report of some intelligent terminals is supposed to be jointly published via the linear network coding. To prevent the injection of invalid data packets and make the transmission reliable, each data packet should be suffixed with a valid recipient before forwarding. A network coding signature scheme can help to meet this requirement when all terminal devices have the same key used for signing packets. However, if each device has its own key, a group of signatures cannot be directly combined for the corresponding packets. As a solution for the verification problem, we present this homomorphic network coding signature scheme for multiple sources.

An architecture is shown in Figure 1. A terminal device can be seen as a source, while the receiver wants to get the log report with a correct recipient. Each entity in the scheme is described as follows:

(i) **Source nodes**. After some parameters are generated as public information, the ith source node generates its own key pair (pk_i, sk_i) for signing and verifying. Each node has a part of the original file, and the part can be seen as a data packet. To obtain a signature σ, the ith node signs its packet that belongs to the file with an identifier Id using sk_i. Then, it sends the signed tuple $(Id, \mathbf{w}, \sigma)$ on its outgoing edges.

(ii) **Intermediate nodes**. When an intermediate node receives some packets with the corresponding signatures, it checks whether any one is not valid. Then, it selects μ coefficients for the rest valid packets $\mathbf{w}_1, \ldots, \mathbf{w}_\mu$, and combines the packets and their signatures, respectively. Finally, the combined tuple is forwarded on the outgoing edges.

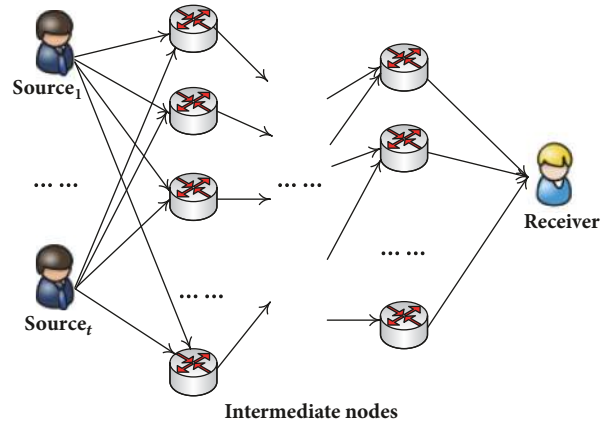

Figure 1: Architecture of the proposed scheme.

(iii) **Receiver**. Once the receiver has collected Id file's packets signed by using all t secret keys, it checks the validity and recovers the original file if the check is passed.

4.2. Scheme Description. In this section, we present our construction of the multisource homomorphic network coding signature scheme. There are t devices as source nodes, any number of intermediate nodes, and several receivers. For simplicity, we assume that each source node holds and forwards a packet of a file. The whole file can be represented as a augmented vector set $\mathbf{w} = \{\mathbf{w}_1, \ldots, \mathbf{w}_t\}$, of which the ith packet can be represented as $\mathbf{w}^{(i)} = (u_1^{(i)}, \ldots, u_m^{(i)}, v_1^{(i)}, \ldots, v_n^{(i)})$. Each packet belongs to a file with the ID Id and some of the packets are encoded together with the same Id.

As described above, each source node has its own key pair in the system. For the ith source, its private key is used to sign $\mathbf{w}^{(i)}$ so that the signed packets can be verified by intermediate nodes which receive the signatures. After receiving several input packets, an intermediate node firstly checks each packet and discards all the packets that cannot pass the check. With a signature σ, the corresponding packet can be verified even though it is linear combinations of vectors originated from different sources. Then, using random or established coefficients, the node makes linear combinations of the remaining data packets and produces a signature for the encoded packet based on the received signatures without accessing the private keys. Briefly, an adversary's attack is successful if it can generate a forged data packet $(Id^*, \mathbf{w}^*, \sigma^*)$ that makes verification algorithm output 1 using public keys, while either Id^* is an invalid file ID or Id^* is valid but $\mathbf{w}^*$ is not in the domain of files.

Then, we describe the algorithms of a multisource homomorphic network coding signature scheme based on the signature scheme CFW in [1] as follows:

Public System Parameters$(\lambda) \to param$. Choose a bilinear map e: $\mathbb{G} \times \mathbb{G}' \to \mathbb{G}_{\mathbb{T}}$, where $\mathbb{G}, \mathbb{G}', \mathbb{G}_{\mathbb{T}}$ are bilinear groups of prime order $p > 2^{\lambda}$, while g is a generator of $\mathbb{G}$ and g' is a generator of $\mathbb{G}'$. Randomly choose elements $h, h_1, \ldots, h_n, g_1, \ldots, g_m$ from $\mathbb{G}$. The algorithm outputs system parameters $param = (p, g, g', h, h_1, \ldots, h_n, g_1, \ldots, g_m)$ as public.

NetKG$(i, m, n) \to (pk_i, sk_i)$. This algorithm is run by each ith source node for setting up its own key pair. The ith source node randomly chooses $a_i \in \mathbb{F}_p$ to set its private key as $sk_i = a_i$ and public key as $k_i = P_i = (g')^{a_i}$. The algorithm outputs the key pair (pk_i, sk_i).

NetSign$(sk_i, Id, \mathbf{w}^{(i)}) \to \sigma$. Each ith source node signs its data packet as if in the single-source signature scheme CFW, but their secret key is different. For the packet vector $\mathbf{w}^{(i)} = (u_1^{(i)}, \ldots, u_m^{(i)}, v_1^{(i)}, \ldots, v_n^{(i)}) \in \mathbb{F}_p^{m+n}$, the ith source computes its signature as follows:

(i) Let $Id \in \mathbb{F}_p^*$ and $s_i \in \mathbb{F}_p$;

(ii) Compute $X_i = (h^{s_i} \prod_{j=1}^{m} h_j^{u_j} \prod_{j=1}^{m} g_j^{v_j})^{1/(a_i+Id)}$ and output its signature $\sigma = (X^{(1)}, \ldots, X^{(i)}, \ldots, X^{(t)}, s)$, where $X^{(i')} = 1$ for each $i' \neq i$.

Note that, for the ith source node, the rest packets in $\mathbf{w} = \{\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(t)}\}$ other than $\mathbf{w}^{(i)}$ are all set as zero vectors before forwarding.

NetEval$(Id, \{\mathbf{w}_j, \alpha_j, \sigma_j\}_{j=1}^{\mu}) \to \sigma$. If an intermediate node has received μ valid packet sets $\mathbf{w}_j = (u_{j,1}^{(i)}, \ldots, u_{j,m}^{(i)}, v_{j,1}^{(i)}, \ldots, v_{j,n}^{(i)})$ from the same Id, of which signature is $\sigma_j = (X_j^{(1)}, \ldots, X_j^{(t)}, s_j)$. Each coefficient $\alpha_j \in \mathbb{F}_p$ is determined by the intermediate node. The combined signature of this node is also a $t+1$-dimension vector $\sigma = (X^{(1)}, \ldots, X^{(t)}, s)$, where $s = \sum_{i=1}^{t} \sum_{j=1}^{\mu} \alpha_j s_j \bmod p$ and each $X^{(i)} = \prod_{j=1}^{\mu} (X_j^{(i)})^{\alpha_j}$. This algorithm outputs the combined signature σ. Then, the modified packet $\mathbf{w}$ is forwarded to other nodes along with its signature σ.

NetVer$(\{pk_i\}_{i=1}^{t}, Id, \mathbf{w}, \sigma) \to accept$. Taking public keys $\{pk_i\}_{i=1}^{t}$, file ID Id, data packet $\mathbf{w}$, and the corresponding signature σ as input, the verification algorithm outputs

1, if $\prod_{i=1}^{t} e(X^{(i)}, P_i(g')^{Id}) = e(h^s \prod_{j=1}^{m} h_j^{u_j} \prod_{j=1}^{n} g_j^{v_j}, g')$ and 0, otherwise.

5. Analysis

5.1. Correctness. According to the definition in Section 3, we analyze the correctness of the proposed scheme.

Theorem 4. *The proposed multisource homomorphic network coding signature scheme is correct.*

Proof. For each $i \in \{1, \ldots, t\}$, the ith original vector is denoted as $\mathbf{w}^{(i)} = (u_1^{(i)}, \ldots, u_m^{(i)}, v_1^{(i)}, \ldots, v_n^{(i)}) \in \mathbb{F}_p^{m+n}$. There is $\prod_{i=1}^{t} e(X^{(i)}, P_i(g')^{Id}) = e(X^{(i)}, P_i(g')^{Id}) = e((h^{s_i} \prod_{j=1}^{m} h_j^{u_j} \prod_{j=1}^{m} g_j^{v_j})^{1/(a_i+Id)}, (g')^{a_i+Id}) = e(h^s \prod_{j=1}^{m} h_j^{u_j} \prod_{j=1}^{n} g_j^{v_j}, g')$. Thus, the verification result on a valid original signature σ is 1.

On the other hand, in a modified packet, $\mathbf{w} = \sum_{j=1}^{\mu} \alpha_j \mathbf{w}_j$ and $s = \sum_{i=1}^{t} \sum_{j=1}^{\mu} \alpha_j s_j \bmod p$. There is $\prod_{i=1}^{t} e(X^{(i)}, P_i(g')^{Id}) = \prod_{i=1}^{t} e((h^{s_i} \prod_{j=1}^{m} h_j^{u_j} \prod_{j=1}^{m} g_j^{v_j})^{1/(a_i+Id)}, (g')^{a_i+Id}) = e(h^s \prod_{j=1}^{m} h_j^{u_j} \prod_{j=1}^{n} g_j^{v_j}, g')$. Thus, The verification result on a combined signature σ is 1.

Therefore, algorithms in the proposed is correct. □

5.2. Security. Then, we give the proof that the signatures in the scheme is unforgeable according to the definition in Section 3.

Theorem 5. *The proposed multisource homomorphic network coding signature scheme is secure under the q-SDH assumption.*

Proof. If an adversary has a PPT algorithm $\mathscr{B}^*$ which can forge the valid signature for a data packet, $\mathscr{B}^*$ can be used to construct an efficient algorithm $\mathscr{B}$ to forge valid signatures the CFW signature scheme.

The public system parameters are chosen as follows: $\mathbb{G}, \mathbb{G}', \mathbb{G}_{\mathbb{T}}$ are bilinear groups of prime order $p > 2^{\lambda}$ and e is a bilinear map: $\mathbb{G} \times \mathbb{G}' \to \mathbb{G}_{\mathbb{T}}$; g is a generator of $\mathbb{G}$ and g' is a generator of $\mathbb{G}'$; $h, h_1, \ldots, h_n, g_1, \ldots, g_m$ are random factors.

The algorithm $\mathscr{B}^*$ takes the public system parameters $param$, the public keys $\{pk_i\}$ a valid identifier Id, and each packet $\mathbf{w}$ with its signature as input. According to the received data signed packets $(Id, \{\mathbf{w}_j, \alpha_j, \sigma_j\}_{j=1}^{\mu})$, algorithm $\mathscr{B}^*$ tries to output a forged signed packet $(Id^*, \{\alpha_j^*\}, \mathbf{w}^*, \sigma^*)$ which makes verification algorithm output 1. That is, either $Id^* \neq Id$ or $Id^* = Id$ and $\mathbf{w}^* \neq \sum \alpha_j^* \mathbf{w}_j$.

Then, based on $\mathscr{B}^*$ algorithm, we construct another PPT algorithm $\mathscr{B}$ which can produce a forged signature for packets in the CFW signature scheme.

We assume that the first-source node uses (a, P) as its private and public keys, i.e., $sk_1 = a$ and $pk_1 = P$. Other $t-1$ source nodes whose private and public keys are generated as follows:

(i) Randomly select $t-1$ numbers from $\mathbb{F}_p$: $x_1, \ldots, x_{t-1}$;

(ii) Set the private key of the ith source $a_i = x_{i-1}(a + Id) - Id$ and its public key $P_i = (g')^{a_i}$;

Using the **NetSign** algorithm, each source node outputs its signature.

A forged signature packet $(Id^*, \{\alpha_j^*\}, \mathbf{w}^*, \sigma^*)$ for CFW is output, where $\sigma^* = (X^{(1)*}, \ldots, X^{(t)*}, s^*)$ and $s^* = \prod_{j=1}^{\mu} \alpha_j^* s_i^* \bmod p$.

It is easy to get

$$
\begin{aligned}
& e\left(\prod_{i=1}^{t} X^{i*}, P\cdot\left(g'\right)^{Id}\right) \\
&= e\left(\left(X^{(1)*}\right)\prod_{i=2}^{t}\left(X^{(i)*}\right)^{\alpha_i^* x_{i-1}}, P\cdot\left(g'\right)^{Id}\right) \\
&= e\left(\left(X^{(1)*}\right)^{\alpha_1^*}, P\cdot\left(g'\right)^{Id}\right) \\
&\quad\cdot e\left(\prod_{i=2}^{t}\left(X^{(i)*}\right)^{\alpha_i^* x_{i-1}}, P\cdot\left(g'\right)^{Id}\right) \\
&= e\left(\left(X^{(1)*}\right)^{\alpha_1^*}, P\cdot\left(g'\right)^{Id}\right) \\
&\quad\cdot e\left(\prod_{i=2}^{t}\left(X^{(i)*}\right)^{\alpha_i^*}, P\cdot\left(g'\right)^{Id}\right)^{x_{i-1}} \\
&= e\left(\left(X^{(1)*}\right)^{\alpha_1^*}, P\cdot\left(g'\right)^{Id}\right) \\
&\quad\cdot e\left(\prod_{i=2}^{t}\left(X^{(i)*}\right)^{\alpha_i^*}, \left(P\cdot\left(g'\right)^{Id}\right)^{x_{i-1}}\right) \\
&= e\left(\left(X^{(1)*}\right)^{\alpha_1^*}, P\cdot\left(g'\right)^{Id}\right) \\
&\quad\cdot e\left(\prod_{i=2}^{t}\left(X^{(i)*}\right)^{\alpha_i^*}, \left(g'\right)^{a_i+Id}\right) \\
&= e\left(\left(X^{(1)*}\right)^{\alpha_1^*}, P\cdot\left(g'\right)^{Id}\right) \\
&\quad\cdot e\left(\prod_{i=2}^{t}\left(X^{(i)*}\right)^{\alpha_i^*}, P_i\cdot\left(g'\right)^{Id}\right) \\
&= e\left(\prod_{i=1}^{t}\left(X^{(i)*}\right)^{\alpha_i^*}, P_i\cdot\left(g'\right)^{Id}\right) \\
&= e\left(h^{s^*}\prod_{j=1}^{n} h_j^{u_j}\prod_{j=1}^{m} g_j^{v_j}, g'\right).
\end{aligned}
\tag{2}
$$

Since either $Id^* \neq Id$, or $\mathbf{w}^* \neq \sum \alpha_j^* \mathbf{w}_j$, the forged signature is valid.

However, the CFW signature scheme is secure under q-SDH assumption. Therefore, the proposed multisource homomorphic network coding signature scheme is secure under the q-SDH assumption. □

6. Conclusion

In this paper, to give a solution for authentication of network coding, we propose the multisource homomorphic network coding signature in the standard model and show that the signature scheme is security under q-SDH assumption holds. The proposed scheme can effectively guarantee the availability in a multisource IoT system.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Natural Science Foundation of Guangdong Province for Distinguished Young Scholars (2014A030306020), Guangzhou Scholars Project for Universities of Guangzhou (no. 1201561613), Science and Technology Planning Project of Guangdong Province, China (2015B010129015), the National Natural Science Foundation of China (no. 61472091), and the National Natural Science Foundation for Outstanding Youth Foundation (no. 61722203).

References

[1] D. Catalano, D. Fiore, and B. Warinschi, "Efficient network coding signatures in the standard model," in *Public key cryptography*, vol. 7293, pp. 680–696, Springer, 2012.

[2] W. Chen, H. Lei, J. Li, C. Gao, F. Li, and K. Qi, "A multisource homomorphic network coding signature in the standard model," in *Proceedings of the International Conference on Green, Pervasive and Cloud Computing*, pp. 66–74, 2017.

[3] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung, "Network information flow," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, 2000.

[4] S. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 49, no. 2, pp. 371–381, 2003.

[5] J. Zhou, M. Tang, Y. Tian et al., "Social network and tag sources based augmenting collaborative recommender system," *IEICE Transaction on Information and Systems*, vol. 98, no. 4, pp. 902–910, 2015.

[6] S. Xie and Y. Wang, "Construction of tree network with limited delivery latency in homogeneous wireless sensor networks," *Wireless Personal Communications*, vol. 78, no. 1, pp. 231–246, 2014.

[7] Z. Huang, S. Liu, X. Mao, K. Chen, and J. Li, "Insight of the protection for data security under selective opening attacks," *Information Sciences*, vol. 412-413, pp. 223–241, 2017.

[8] J. Li, Y. K. Li, X. Chen, P. P. C. Lee, and W. Lou, "A Hybrid Cloud Approach for Secure Authorized Deduplication," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 5, pp. 1206–1216, 2015.

[9] X. Zhang, Y. Tan, C. Liang, Y. Li, and J. Li, "A Covert Channel Over VoLTE via Adjusting Silence Periods," *IEEE Access*, vol. 6, pp. 9292–9302, 2018.

[10] Q. Lin, H. Yan, Z. Huang, W. Chen, J. Shen, and Y. Tang, "An ID-based linearly homomorphic signature scheme and its application in blockchain," *IEEE Access*, vol. 6, pp. 20632–20640, 2018.

[11] Q. Lin, J. Li, Z. Huang, W. Chen, and J. Shen, "A short linearly homomorphic proxy signature scheme," *IEEE Access*, vol. 6, pp. 12966–12972, 2018.

[12] D. Xie, X. Lai, X. Lei, and L. Fan, "Cognitive Multiuser Energy Harvesting Decode-and-Forward Relaying System with Direct Links," *IEEE Access*, vol. 6, pp. 5596–5606, 2018.

[13] L. Fan, X. Lei, N. Yang, T. Q. Duong, and G. K. Karagiannidis, "Secure Multiple Amplify-and-Forward Relaying with Cochannel Interference," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 8, pp. 1494–1505, 2016.

[14] L. Fan, X. Lei, N. Yang, T. Q. Duong, and G. K. Karagiannidis, "Secrecy Cooperative Networks with Outdated Relay Selection over Correlated Fading Channels," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 7599–7603, 2017.

[15] W. Meng, E. W. Tischhauser, Q. Wang, Y. Wang, and J. Han, "When Intrusion Detection Meets Blockchain Technology: A Review," *IEEE Access*, vol. 6, pp. 10179–10188, 2018.

[16] J. Xu, L. Wei, Y. Zhang, A. Wang, F. Zhou, and C. Gao, "Dynamic Fully Homomorphic encryption-based Merkle Tree for lightweight streaming authenticated data structures," *Journal of Network and Computer Applications*, vol. 107, pp. 113–124, 2018.

[17] H. Tian, Z. Chen, C. Chang et al., "Public audit for operation behavior logs with error locating in cloud storage," *Soft Computing*, pp. 1–14, 2018.

[18] J. Shen, Z. Gui, S. Ji, J. Shen, H. Tan, and Y. Tang, "Cloud-aided lightweight certificateless authentication protocol with anonymity for wireless body area networks," *Journal of Network and Computer Applications*, vol. 106, pp. 117–123, 2018.

[19] J. Shen, T. Zhou, X. Chen, J. Li, and W. Susilo, "Anonymous and Traceable Group Data Sharing in Cloud Computing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 4, pp. 912–925, 2018.

[20] M. N. Krohn, M. J. Freedman, and D. Mazières, "On-the-fly verification of rateless erasure codes for efficient content distribution," in *Proceedings of the 2004 IEEE Symposium on Security and Privacy*, pp. 226–239, May 2004.

[21] C. Gkantsidis and P. Rodriguez, "Cooperative security for network coding file distribution," in *Proceedings of the INFOCOM*, vol. 3, 2006.

[22] R. Johnson, D. Molnar, D. Song, and D. Wagner, "Homomorphic signature schemes," in *Topics in cryptology-CT-RSA*, vol. 2271, pp. 244–262, Springer, Berlin, 2002.

[23] S. Agrawal, D. Boneh, X. Boyen, and D. M. Freeman, "Preventing pollution attacks in multi-source network coding," in *Public Key Cryptography-PKC*, vol. 6056, pp. 161–176, Springer, Berlin, Germany, 2010.

[24] A. Yun, J. H. Cheon, and Y. Kim, "On homomorphic signatures for network coding," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 59, no. 9, pp. 1295-1296, 2010.

[25] Y. Wang, "Insecure provably secure network coding and homomorphic authentication schemes for network coding," *IACR Cryptology ePrint Archive*, vol. 2010, p. 60, 2010.

[26] Z. Yu, Y. Wei, B. Ramkumar, and Y. Guan, "An efficient signature-based scheme for securing network coding against pollution attacks," in *Proceedings of the 27th IEEE Communications Society Conference on Computer Communications (INFOCOM '08)*, pp. 2083–2091, April 2008.

[27] H. J. Kang, A. Yun, E. Y. Vasserman, H. T. Lee, J. H. Cheon, and Y. Kim, "Secure network coding for a p2p system," in *Proceedings of the ACM Conference on Computer and Communications Security*, Poster, 2009.

[28] F. Zhao, T. Kalker, M. Médard, and K. J. Han, "Signatures for content distribution with network coding," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '07)*, pp. 556–560, Nice, France, June 2007.

[29] D. Charles, K. Jain, and K. Lauter, "Signatures for network coding," in *Proceedings of the 2006 40th Annual Conference on Information Sciences and Systems, CISS 2006*, pp. 857–863, usa, March 2006.

[30] R. Gennaro, J. Katz, H. Krawczyk, and T. Rabin, "Secure network coding over the integers," in *Public Key Cryptography-PKC 2010*, vol. 6056 of *Lecture Notes in Comput. Sci.*, pp. 142–160, Springer, Berlin, Germany, 2010.

[31] D. Boneh, D. Freeman, J. Katz, and B. Waters, "Signing a linear subspace: signature schemes for network coding," in *Public Key Cryptography-PKC 2009*, vol. 5443 of *Lecture Notes in Comput. Sci.*, pp. 68–87, Springer, Berlin, Germany, 2009.

[32] D. Boneh and D. M. Freeman, "Linearly homomorphic signatures over binary fields and new tools for lattice-based signatures," in *Public Key Cryptography-PKC 2011*, vol. 6571 of *Lecture Notes in Comput. Sci.*, pp. 1–16, Springer, 2011.

[33] A. Lewko and B. Waters, "New techniques for dual system encryption and fully secure HIBE with short ciphertexts," in *Theory of Cryptography*, vol. 5978 of *Lecture Notes in Comput. Sci.*, pp. 455–479, Springer, 2010.

[34] N. Attrapadung and B. t. Libert, "Homomorphic network coding signatures in the standard model," in *Public Key Cryptography-PKC 2011*, vol. 6571, pp. 17–34, Springer, 2011.

[35] D. Catalano, D. Fiore, and B. Warinschi, "Adaptive pseudo-free groups and applications," in *Advances in Cryptology-EUROCRYPT*, vol. 6632 of *Lecture Notes in Comput. Sci.*, pp. 207–223, Springer, 2011.

[36] W. Chen, H. Lei, and K. Qi, "Lattice-based linearly homomorphic signatures in the standard model," *Theoretical Computer Science*, vol. 634, pp. 47–54, 2016.

[37] D. M. Freeman, "Improved Security for Linearly Homomorphic Signatures: A Generic Framework," in *Public Key Cryptography – PKC 2012*, vol. 7293 of *Lecture Notes in Computer Science*, pp. 697–714, Springer, 2012.

[38] D. Boneh and X. Boyen, "Short signatures without random oracles," in *Advances in Cryptology-EUROCRYPT 2004*, vol. 3027, pp. 56–73, Springer, 2004.