A MAC-based Scheme for Multi-Generation Content Distribution with Network Coding

Yu Zhang*

*Shanghai Key Laboratory of Intelligent Information Processing
School of Computer Science, Fudan University, Shanghai, 200433, P. R. China
11210240041@fudan.edu.cn

Abstract— **The system that uses network coding is highly susceptible to pollution attacks, where a malicious node may pollute a small number of packets with the purpose of preventing the recipient nodes from reconstructing the original messages properly. Some schemes that use Message Authentication Code (MAC) have been proposed for resisting this attack. However, these schemes could be broken with probability *1/q*, where *q* is the size of the underlying field. Although the trace function has already been used for constructing MACs for a higher security, it can only be used for single-generation distribution. This paper proposes a novel MAC-based scheme that also employs trace function. However, different from prior work, our scheme can be immediately used for secure multi-generation distribution.**

Keywords— **Homomorphic MAC, Pollution Attacks, Network Cod-ing, Authentication, Repetitive Attacks**

I. INTRODUCTION

Network coding [1][2] has been applied to network for achieving the optimal throughput. However, it is susceptible to *pollution attacks*, where a malicious node injects corrupted packets into network, aiming at preventing the recipient from reconstructing the original file. Due to the way the packets are combined and transmitted, a small number of polluted packets can cause large-scale pollution propagation. To solve the problem, several public-key based schemes [5]-[8] and hybrid [14][15] schemes are proposed. In these schemes, the source node signs the packet using a private key. The recipients use the public key known to all the nodes to check the integrity of the packet. However, these schemes are based on expensive paring operations [5]-[7] or exponentiation operations over a large field [8], which makes these schemes not fast enough for online communication.

To address the inefficiency of public-key based scheme, some efficient symmetric-key based [9]-[13] schemes are introduced. Le *et al.* [13] proposed an efficient *message authentication code* (MAC) based inter-session scheme which supports multi-generation communication. However, its security relies on the size of the underlying field, which means in standard network coding based environment, where the field size is usually 2^8, the scheme is not security enough. Cheng *et al.* [12] proposed a homomorphic MAC scheme which can achieve a higher security level using the same field size. However, it does not support multi-generation transmission and is susceptible to repetitive attacks [14]. A repetitive attack is when transmissions contains multiple generations [3], a malicious node collect the legitimate packets from previous generations and use them to forge packets for current or subsequence generations.

In this paper, we propose a novel homomorphic MAC based scheme named MtMac. Our scheme has the following advantages:

(1) Unlike the scheme described in [10][12], our scheme supports multi-generation transmission.

(2) Our scheme can prevent the repetitive attack while [10][12] cannot.

(3) Our scheme can achieve the same level of security using smaller field size.

II. PRELIMINARIES

In this section, we'll introduce some definitions and notations which will be used in the construction of MtMac.

A. System Model

We propose a model of network based on *linear network coding* [2]. There are three kinds of nodes in the network: source node, intermediate node and recipient. The source node intends to send a file (i.e. some packets) to the recipient through the intermediate nodes. To achieve this, the source node divides the packets into g generations [3]. Each generation consist of m packets with n symbols. Then the source transforms m packets into a sequence of vectors $\overline{v}_1,\ldots,\overline{v}_m$ in an n-dimensional vector space over a finite field $\mathbb{F}_q$, where q is a prime. Then the source augments these m vectors to produce *m augmented packets* $v_1, v_2, \ldots, v_m$, defined as:

$$v_i = (\overbrace{\underbrace{0,\ldots,0}_{i-1},1,\underbrace{0,\ldots,0}_{m-i}}^{m},\overline{v}_i) \in \mathbb{F}_q^{n+m} \quad (1)$$

which means the first m coordinates of v_i is a unit vector with a "1" in the position. If $x \in \mathbb{F}_q^{n+m}$ is a linear combination of $v_1, v_2, \ldots, v_m$, then the first m coordinates of x is exactly the coefficients of the linear combination. The source transmits these augmented packets through the network generation by generation. Intermediate nodes in the network perform random network coding [4]. These intermediate nodes linearly

combine packets from the same generation. For instance, when an intermediate node receives j packets $p_1,\ldots,p_j$ of the *kth* generation from its incoming links, it sends $y=\sum_{i=1}^{j} c_i p_i$ to its outgoing links, where each $c_i \in \mathbb{F}_q$ is random chosen from $\mathbb{F}_q$. If there is no error during transmission, all packets in the network are linear combination of the m original augmented packets $v_1, v_2, \ldots, v_m$. When a recipient receives m linear independent packets, it may recover the original vectors $v_1, v_2, \ldots, v_m$ by using Gaussian elimination on a $m \times (m+n)$ matrix which is formed by the m received linearly independent packets.

B. Threat Model

In our scheme, we assume that the source node and the recipients are trusted. Other than that, all the other nodes could be malicious. Malicious nodes may create polluted packets and inject it into a network, or distort the packet passing by. For instance, a malicious node may tamper with the tag carried by the packet, or collect legitimate packet from the previous generation to fake the packets for subsequent generation. We assume that the adversaries know the construction of our scheme and have the computation power to perform probabilistic polynomial time algorithms. We define a polluted packet as follows:

Definition 1: We denote V as a linear span of m vectors $v_1, v_2, \ldots, v_m$ of the *gth* generation. We say a packet $y=(y_1, y_2, \ldots, y_m, y_{m+1}, \ldots, y_{m+n})$ is a polluted packet with respect to the *gth* generation, if $y \notin V$, i.e. $y \neq \sum_{i=1}^{m} y_i v_i$.

C. Trace Functions and Finite Fields

In this section, we introduce some notations about the trace functions over finite field which will be used in our construction to replace the normal inner product operation.

In network coding, every source augmented packet $v_1, v_2, \ldots, v_m$ can be viewed as an element in $\mathbb{F}_q^{m+n}$, which is a finite extension of $\mathbb{F}_q$. So any element v in the linear span of $v_1, v_2, \ldots, v_m$ can be represented as:

$$v=\sum_{i=1}^{m+n} c_i \alpha_i \tag{2}$$

where $c_i \in \mathbb{F}_q$ and $\{\alpha_1,\ldots,\alpha_{m+n}\} \in \mathbb{F}_q$ is a basis of $\mathbb{F}_q^{m+n}$ over $\mathbb{F}_q$.

We let $ls=m+n$, where $s>m$. Then the field $\mathbb{F}_q^{m+n}$ can be represented as a finite extension of $\mathbb{F}_q^l$, and every element in the linear span of $v_1, v_2, \ldots, v_m$ can be represented as:

$$v=\sum_{i=1}^{s} c_i \gamma_i \tag{3}$$

where $c_i \in \mathbb{F}_q$ and $\{\gamma_1,\ldots,\gamma_s\} \in \mathbb{F}_q^l$ is a basis of $\mathbb{F}_q^{m+n}$ over $\mathbb{F}_q^l$. This form of representation will be used in the sign and verification process of our scheme.

Definition 2: We suppose K is a finite extension of field E, and the dimension of K over E is n, the size of E is q. For $u \in K$, the trace function $Tr_{K/E}(u)$ is defined as follows:

$$Tr_{K/E}(u)=\sum_{i=0}^{n-1} u^{q^i} \tag{4}$$

where $Tr_{K/E}(u)$ is an element of E. If the field is clear, we can denote the trace function by Tr.

Definition 3: [17] K is a finite extension of E. Then the two basis $\{\alpha_1,\ldots,\alpha_n\}$, $\{\beta_1,\ldots,\beta_n\}$ of K over E are said to be dual if:

$$Tr(\alpha_i \beta_j)=\begin{cases}1 & i=j\\ 0 & i\neq j\end{cases}$$

where $1 \le i, j \le n$.

Proposition 1: [17]

(1) $Tr(u+v)=Tr(u)+Tr(v)$.

(2) $Tr(cv)=cTr(v)$ for all $c \in E$, $v \in K$.

(3) For any basis $\{\alpha_1,\ldots,\alpha_n\}$ of K over E, there uniquely exists a dual basis $\{\beta_1,\ldots,\beta_n\}$ of K over E.

Lemma 1: [12] Let $\{\alpha_1,\ldots,\alpha_s\}$ be a basis of $K=\mathbb{F}_q^n$ over $E=\mathbb{F}_q^l$, from Proposition 1.(3), we know that there exists a unique dual basis $\{\beta_1,\ldots,\beta_s\}$. If u, v can be represented as $x=u_1\alpha_1+\ldots+u_s\alpha_s$ and $v=v_1\beta_1+\ldots+v_s\beta_s$, $v_i, u_j \in \mathbb{F}_q^l$, $1 \le i, j \le s$, then we have:

$$Tr(uv)=\langle u, v\rangle$$

where the inner product arithmetic operates over $\mathbb{F}_q^l$.

III. A HOMOMORPHIC MAC SCHEME (MTMAC)

In this section, we present the construction of MtMac. Let vector space V denotes the linear span of $v_1, v_2, \ldots, v_m$. Each vector space has a unique identifier *id*, which is an element randomly sampled from a set $\mathcal{I}$ [6]. The source chooses a secret key k at random from the key space $\mathcal{K}$ and shares it securely with all the recipients. Then the source uses the secret key with a PRF to generate the key used in the MAC stage to compute a MAC tag t_i for every basis v_i. Then the source transmits the packet (id, v_i, t_i) into the network. When intermediate nodes received the packets from their parents, they employ the homomorphic property to create valid tags. Recipients use the tag and the secret key k to verify the packet and drop polluted packets. We let $m+n=ls$, where $s>m$ and $l>1$. We denote $\mathcal{I}$ as the set of vector space identifiers, $\mathcal{K}$ as the key space and define our scheme as follows:

Definition 4: A (q,n,m,s,l) $(m+n=ls,\ s>m,\ l>1)$ homomorphic MAC cryptographic scheme for network coding is defined by a tuple of four probabilistic polynomial time (PPT) algorithms (**Generate**, **Sign**, **Combine**, **Verify**):

Generate(id, k, V):

Input: a vector space identifier *id*, a vector space *V*, a secret key *k*.

Output: a private key k_p.

Sign(id, k_p, v):

Input: a vector space identifier id, a private key k_p and a source packet $v \in \mathbb{F}_q^{m+n}$.

Output: a tag $t \in \mathbb{F}_q^l$.

Combine($id, (y_1, t_1, c_1), \ldots, (y_r, t_r, c_r)$):

Input: a vector space identifier *id*, *r* vectors $y_1, \ldots, y_r \in \mathbb{F}_q^{m+n}$, their tags $t_1, \ldots, t_r \in \mathbb{F}_q^l$, coefficients $c_1, \ldots, c_r \in \mathbb{F}_q$.

Output: a tag $t \in \mathbb{F}_q^l$ of $y = \sum_{i=1}^{r} c_i y_i$.

Verify(id, k, y, t):

Input: a vector space identifier *id*, a secret key *k*, a vector $y \in \mathbb{F}_q^{m+n}$ and its tag *t*.

Output: 1 (accept) or 0 (reject).

Correctness: The scheme must satisfy the following requirements:

Let t_i = **Sign**(*id*,**Generate**(id, k, V),*v*), and $c_i \in \mathbb{F}_q$ for all $1 \le i \le m$. Let t = **Combine**($id, (v_1, t_1, c_1), \ldots, (v_m, t_m, c_m)$). Then **Verify**($id, k, \sum_{i=1}^{m} c_i v_i, \sum_{i=1}^{m} c_i t_i$) = 1.

Security: The security of a homomorphic MAC cryptographic scheme is defined using following game:

Definition 5: Let $\mathcal{T}$ = (**Generate, Sign, Combine, Verify**) be a homomorphic MAC cryptographic scheme. We denote the adversary by $\mathcal{A}$ and challenger by $\mathcal{C}$. We say that $\mathcal{T}$ is secure if the probability that the PPT adversary $\mathcal{A}$ wins the following game is negligible:

Attack Game:

Setup: $\mathcal{C}$ chooses a secret key *k* randomly from $\mathcal{K}$.

Queries: $\mathcal{A}$ adaptively queries $\mathcal{C}$. Each query is of the form (id_g, V_g), where V_g is a linear space spanned by a basis of *m* vectors $v_1, \ldots, v_m \in \mathbb{F}_q^{m+n}$, and id_g is the space identifier. $\mathcal{C}$ Computes $t_i \leftarrow$ **Sign**(id_g ,**Generate**(id_g ,*k*, V_g), v_i) and sends $(t_1, \ldots, t_m)$ to $\mathcal{A}$.

Output: $\mathcal{A}$ outputs a triple (id_*, y_*, t_*). $\mathcal{A}$ wins the game if **Verify** (id_*, y_*, t_*) =1 and either (1) $id_* \neq id_g$ for all *g* and $y_* \neq 0$ (Forgery 1) or (2) $id_* = id_g$ for some *g*, $y_* \notin V_g$ (Forgery 2).

We denote by Adv[$\mathcal{A}, \mathcal{T}$] as the probability that $\mathcal{A}$ wins the above game.

Construction of MtMac: Let F be a *pseudo random function* (PRF): $\mathcal{K} \times \mathcal{I} \rightarrow \mathbb{F}_q^{m+n}$. Let $m + n = ls$, $s > m$. Then $\mathbb{F}_q^{m+n}$ is a finite extension of $\mathbb{F}_q^l$. Recall from Equation (3), any packet $v \in \mathbb{F}_q^{m+n}$ can be viewed as a vector $(a_1, \ldots, a_s)$. We know that $i \in [1, s]$ $a_i \in \mathbb{F}_q^l$.

Generate(id, k, V):

Input: a vector space identifier $id \in \mathcal{I}$, a vector space *V*, a secret key *k*. Let $v_1, \ldots, v_m \in \mathbb{F}_q^{m+n}$ be the packets that span *V*. The source computes: $k_p \leftarrow F(k, id)$, $k_p \in \mathbb{F}_q^{m+n}$.

Output: a key k_p.

Sign(id, k_p, v):

Input: a vector space identifier *id*, a private key k_p and a source packet *v*.

Recall that any element $v \in \mathbb{F}_q^{m+n}$ can be viewed as a vector $(a_1, \ldots, a_s), i \in [1, s], a_i \in \mathbb{F}_q^l$. For simplicity we choose a set of unit basis $\{\alpha_1, \ldots, \alpha_s\}$ of $K = \mathbb{F}_q^{m+n}$ over $E = \mathbb{F}_q^l$ .Then we compute the dual basis $\{\beta_1, \ldots, \beta_s\}$ .Then we can present k_p as $k_p = \sum_{i=1}^{s} a_i \alpha_i$ and v as $v = \sum_{i=1}^{s} b_i \beta_i$. Additional with the conclusion of Lemma 1, we have:

$$v = \sum_{i=1}^{s} b_i \beta_i \tag{5}$$

The multiplications are operated on $E = \mathbb{F}_q^l$. The source then computes $\langle k_p, v \rangle$.

Output: $t \leftarrow Tr_{K/E}(vk_p) = \langle k_p, v \rangle, t \in \mathbb{F}_q^l$.

Combine($id, (y_1, t_1, c_1), \ldots, (y_r, t_r, c_r)$):

Input: a vector space identifier *id*, *r* vectors $y_1, \ldots, y_r \in \mathbb{F}_q^{m+n}$, their tags $t_1, \ldots, t_r$, coefficients $c_1, \ldots, c_r \in \mathbb{F}_q$.

Output: $t = \sum_{i=1}^{r} c_i t_i$. The computation of this stage is performed on $\mathbb{F}_q^l$.

Verify(id, k, y, t):

Input: a vector space identifier *id*, a secret key *k*, a vector $y \in \mathbb{F}_q^{m+n}$ and its tag *t*. The recipient computes a basis the same as the **Sign** stage, then computes $t' \leftarrow Tr_{K/E}(yF(id, k))$.

Output: If $t' = t$, output 1 (accept). Otherwise output 0 (reject).

Correctness: Recall from the correctness requirement that:

$$t = \sum_{i=1}^{m} c_i t_i = \sum_{i=1}^{m} c_i Tr(k_p v_i)$$

The tag t' computed by the verification algorithm is

$$t' = Tr(k_p \sum_{i=1}^{m} c_i v_i) = \sum_{i=1}^{m} Tr(c_i k_p v_i) = \sum_{i=1}^{m} c_i Tr(k_p v_i)$$

The above equation comes from property (1) and (2) of Proposition 1. As computed $t = t'$.

Security: We prove the security of MtMac by assuming the PRF F is secure. For a probabilistic polynomial time adversary $\mathcal{B}$, we denote $\mathcal{B}$'s advantage in winning a PRF security game with respect to F as PRF-Adv[$\mathcal{B}$, F] [16].

Theorem 1: For any fixed q,n,m,s,l such that $m+n = ls$, (where $s > m$, $l > 1$), MtMac is a secure homomorphic cryptographic MAC scheme, assuming F is a secure PRF. For every homomorphic MAC adversary $\mathcal{A}$, there is a PRF adversary $\mathcal{B}$ with the same running time as $\mathcal{A}$, such that $\text{Adv}[\mathcal{A}, \text{MtMac}] \le \text{PRF-Adv}[\mathcal{B}, F] + 1/q^l, l > 1$.

Proof: This proof is by using two games denoted as Game 0 and 1.Let W_0, W_1 denote the events that $\mathcal{A}$ wins the homomorphic MAC security Game in Game 0 and 1. We let Game 0 exactly the same as the Attack Game. Thus,

$$\Pr[W_0] = \text{Adv}[\mathcal{A}, \text{MtMac}] \tag{6}$$

In Game 1, the PRF F is replaced by a true random string, i.e., in respond to the queries, the challenger $\mathcal{C}$ random choose k_p from $\mathbb{F}_q^{m+n}$ .Everything else remains the same. Then there exists a PRF adversary $\mathcal{B}$ such that:

$$|\Pr[W_0] - \Pr[W_1]| = \text{PRF-Adv}[\mathcal{B}, F] \tag{7}$$

The challenger in Game 1 works as follows:

Queries: $\mathcal{A}$ adaptively submits queries (id, V), where V is the linear span of $v_1, \ldots, v_m$. $\mathcal{C}$ randomly chooses k_p from $\mathbb{F}_q^{m+n}$. The challenger then computes tags for the packets, i.e., $t_i = Tr(k_p v_i) \in \mathbb{F}_q^l, i \in [1, m]$. Finally $\mathcal{C}$ sends all the tags to $\mathcal{A}$.

Output: Finally $\mathcal{A}$ outputs a triple (id_*, y_*, t_*). We say that $\mathcal{A}$ wins the game, i.e., W_1 happens, if the following equation holds:

$$t_* = Tr(y_* k_p) \tag{8}$$

Through Lemma 1 we know that Equation (8) is equivalence to Equation (9):

$$t_* = \langle y_*, k_p \rangle \tag{9}$$

Let E_1 denote the event that $\mathcal{A}$ outputs a Forgery 1. We denote this by $\Pr[W_1 \bigcap E_1]$. Then we know that $id_* \neq id_g$ for some g and $y_* \neq 0$. Since k_p is random chosen from $\mathbb{F}_q^{m+n}$, which is also a random vector $(k_1, \ldots, k_s)$ over $\mathbb{F}_q^l$, so for any fixed y_*, the distribution of $\langle y_*, k_p \rangle$ is the same. So when E_1 happens, the probability that Equation (9) holds is at most $1/q^l$. Therefore, we have:

$$\Pr[W_1 \bigcap E_1] = \Pr[W_1 \mid E_1] \Pr[E_1] \le 1/q^l \Pr[E_1].$$

Let E_2 denote the event that $\mathcal{A}$ outputs a Forgery 2. We denote this by $\Pr[W_1 \bigcap E_2]$. Then we know that $id_* = id_g$ for some g and $y_* \notin V_g$, V_g is the linear span of $v_1, \ldots, v_m$. Consider the following system of linear equations:

$$\begin{cases} Tr(v_1 k_p) = t_1 \\ \cdots \\ Tr(v_m k_p) = t_m \\ Tr(y_* k_p) = t_* \end{cases}$$

By Lemma 1, we have:

$$\begin{cases} \langle v_1, k_p \rangle = t_1 \\ \cdots \\ \langle v_m, k_p \rangle = t_m \\ \langle y_*, k_p \rangle = t_* \end{cases}$$

which is equivalence to the previous system of linear equations. Recall that all these computations are operated over $\mathbb{F}_q^l$. All but the last equation represent all information that $\mathcal{A}$ gets from the queries. Let r be the rank of the coefficient matrix of the first m equations, we know that $r \le m$. From the construction of MtMac we know that $m < s$, hence we have $r < s$, and the system of the first m equations has solutions of cardinality $q^{l(s-r)}$. Since $y_* \notin V_g$, y_* and all the v_i s are linear independent. So the system is consistent in spite of the choice of t_*, since the coefficient matrix has the rank $r + 1$. The solution space has the size $q^{l(s-r-1)}$. Because k_p is randomly chosen from $\mathbb{F}_q^{m+n}$, and all solutions to the above system is in $\mathbb{F}_q^{m+n}$, so for a fixed y_*, its valid tag could be any element in the solution space of the first m equations with the same probability. Hence, when E_2 happens, the probability that $\mathcal{A}$ has chosen a correct t_* is at most $q^{l(s-R-1)} / q^{l(s-R)} = 1/q^l$. Therefore, we have:

$$\Pr[W_1 \bigcap E_2] = \Pr[W_1 \mid E_2] \Pr[E_2] \le (1/q^l) \Pr[E_2].$$

We calculate $\Pr[W_1]$ as follows:

$$\begin{aligned} &\Pr[W_1] \\ &= \Pr[W_1 \bigcap E_1] + \Pr[W_1 \bigcap E_2] \\ &\le (1/q^l) \Pr[E_1] + (1/q^l) \Pr[E_2] \\ &= 1/q^l \end{aligned} \tag{10}$$

Combine Equation (6),(7) and (10), we completed the proof.

IV. Performance Evaluation

This section studies the performance of our MtMac cryptographic scheme in terms of bandwidth and computation overhead.

A. Bandwidth Overhead

We ignore the bandwidth consumed by the **Generate** stage, since it can be done offline. The online bandwidth overhead per packet includes 1 vector space identifier and 1 tag. Since our tags are elements of $\mathbb{F}_q^l$, so it has size $\log_2^{q^l} = l \log_2^q$. Next we determine the size of the vector space identifier. Assume the size of $\mathcal{I}$ is I, so its size is $\log_2^I$. The size of a packet in our scheme is $m+n$ symbols over $\mathbb{F}_q$, so the size of the source

augmented packet is $(m+n)\log_2^q$. So the total bandwidth overhead is $\frac{l\log_2^q+\log_2^l}{(m+n)\log_2^q}$.

B. Computational Overhead

For the same reason as the last subsection, we will only consider the online computation overhead for the next stages. The basis computation can be done off-line, so the source and the recipient can do the computation in advance.

Sign: Recall from Lemma 1, for each packet, the source node need to compute s multiplications over F_q^l to calculate the MAC tag.

Combine: Let ρ denote the number of packets combined in the network each round. To combine the tags, it needs about ρ multiplications over $\mathbb{F}_q^l$. The computation overhead will not lower the performance very much since the combination of ρ packets needs $\rho(m+n)$ multiplications over $\mathbb{F}_q$ in standard network coding, where $(m+n)$ is much larger than 1, so the $\rho(m+n)$ multiplications over $\mathbb{F}_q$ is much slower than ρ multiplications over $\mathbb{F}_q^l$.

Verify: Unlike the public-key based homomorphic scheme, MtMac doesn't need the exponentiation operation in the finite field. To verify a packet, our scheme needs \$s\$ computations over $\mathbb{F}_q^l$.

V. CONCLUSIONS

In this paper, we proposed a novel homomorphic MAC cryptographic scheme that used network coding. Since the probability that the scheme is broken is $1/q^l$ (q is the size of the underlying field, $q>1$), we can use small fields to achieve the same level of security. Meanwhile, our scheme can be used in multi-generation content distribution network. Moreover, our scheme can prevent repetitive attacks and can be proven secure on the low-level cryptographic assumptions without random oracles. Thus, our scheme is desirable for the network coding based communication system or content distribution system. We are working on extend the approach proposed in this paper to multi-source network coding, space-time code [20]-[23], convolutional code [19] and standard network coding [18].

REFERENCES

[1] R. Ahlswede, N. Cai, S.Y.R. Li, and R.W. Yeung, "Network information flow, "*IEEE Trans. Inform. Theory*, vol. 46, issue. 4, pp. 1204-1216, Jul.2000.

[2] S.-Y. R. Li, W.Yeung, and N. Cai, "Linear network coding,"*IEEE Trans.Inform. Theory*,vol. 49, issue. 2, pp. 371-381, Feb. 2003.

[3] P. A. Chou, Y. Wu, and K. Jain, "Practical Network Coding," in *Proc.41st Annual Allerton Conf. Commun. Control and Computing*, Oct.2003, pp. 40-49.

[4] T. Ho, R. Koetter, M. Medard, D. Karger, and M. Effros, "The Benefits of Coding over Routing in a Randomized Setting," in *Proc. 2003 IEEE Int. Symp. Inf. Theory*, Sept. 2003, pp. 442.

[5] D. Charles, K. Jain, and K. Lauter, "Signatures for network coding," in *Proc. 40th Annual Conf. Informa. Sciences and Systems*, Mar.2006, pp.857-863.

[6] D. Boneh, D. Freeman, J. Katz, and B. Waters, "Signing a Linear Subspace: Signature Schemes for Network Coding," in *Proc. PKC 2009*, Jun. 2009, LNCS 5443, pp. 68-87.

[7] Y. Jiang, H. Zhu, M. Shi, X. Shen, and C. Lin, "An efficient dynamic Identity based signature scheme for secure network coding, " *Computer Networks*, vol. 54, no. 1, pp. 28-40, Jan. 2010.

[8] Z. Yu, Y. Wei, B. Ramkumar, and Y. Guan, "An Efficient Signature-Based Scheme for securing Network Coding Against Pollution Attacks," in *Proc. 2008 IEEE INFOCOM, 27th Conf. Computer Commun*, Apr.2008, pp. 1409-1417.

[9] S. Agrawal, and D. Boneh, "Homomorphic MACs: MAC-Based Integrity for Network Coding," in *Proc. ACNS 2009*, LNCS 5536, Jun. 2009, pp.292-305.

[10] E. Kehdi, and B. Li, "Null keys: Limiting malicious attacks via null space properties of network coding," in *Proc. 2009 IEEE INFOCOM, 28th Conf. Computer Commun,* 2009, pp. 1224-1232.

[11] Y. Li, H. Yao, M. Chen, S. Jaggi, and A. Rosen, "RIPPLE Authentication for Network Coding, " in *Proc. 2010 IEEE INFOCOM, 29th Conf.Computer Commun,* Mar. 2010, pp. 1-9.

[12] C. Cheng, and T. Jiang, "A Novel Homomorphic MAC Scheme for Authentication in Network Coding, "*Communications Letters, IEEE*, vol. 15, no. 11, pp. 1228-1230, Nov. 2011.

[13] A. Le, and A. Markopoulou, "On Detecting Pollution Attacks in Inter-session Network Coding," in *Proc. 2012 IEEE INFOCOM, 31th Conf. Computer Commun*, Mar. 2012, pp. 343-351.

[14] P. Zhang, Y. Jiang, C. Lin, H. Yao, A. Wasef, and X. S. Shen, "Padding for Orthogonality: Efficient Subspace Authentication for Network Coding, " in *Proc. 2011 IEEE INFOCOM, 30th Conf. Computer Commun,* 2011, pp. 1026-1034.

[15] M. Liang, and H. Kan, "An Efficient Hybrid Cryptographic Schemefor Wireless Sensor Network with Network Coding, " *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E96-A, No. 9, pp. 1889-1894.

[16] J. Katz, and Y. Lindell, *Introduction to Modern Cryptography*, 1st ed., Boca Raton, FL, United States:Chapman & Hall/CRC Press, 2008.

[17] R. Lidl, and H. Niederreiter, *Finite fields*, 2nd ed., The Pitt Building, Trumpington Street, Cambridge, United Kingdom: Cambridge University Press, 2000

[18] C. Yuan and H. Kan, "A characterization of solvability for a kind of networks," *Science in China(F)*, Vol.55, No.4, 747-754, 2012

[19] S. Liang and H. Kan, "Practically Feasible Design for Convolutional Network Code," *IEICE Transactions on Fundamentals*, Vol.E96-A, No.9, Sep. 2013.

[20] Yuan Li and Haibin Kan, "Complex Orthogonal Designs with Forbidden 2×2 submatrices", *IEEE Trans. Inform. Theory*, Vol. 58, No. 7, July 2012.

[21] H. Kan and H. Shen, "Lower bounds of the minimal delays of complex orthogonal designs with maximal rates," *IEEE Transactions on Communications*, Vol. 54, No. 3, March 2006.

[22] H. Kan and H. Shen, "A relation between the characteristic generators of a linear code and its dual," *IEEE Trans. Inform. Theory*, Vol. 51, No. 3, March 2005.

[23] H. Kan and H. Shen, "A counterexample for the open problem on the minimal delay of orthogonal designs with maximal rates," *IEEE Trans. Inform. Theory*, Vol. 51, No. 1, January 2005

Yu Zhang. Yu Zhang received the B.S degree of Computer Science in Fudan University, Shanghai, China, in 2010. He is now a graduate student at School of Computer Science, Fudan University. His major/interests are Information Security and Network Coding, Computational Complexity and the defense against Pollution Attacks.