SANC: Source Authentication Using Network Coding

Ahmed Fathy, Tamer ElBatt
Wireless Intelligent Network Center, Nile University,
Cairo, Egypt

Moustafa Youssef
Alexandria University and E-JUST,
Alexandria, Egypt

Abstract—**Abstract— In this paper, we explore the security merits of network coding and potential trade-offs with the widely accepted throughput benefits, especially in multicast scenarios. In particular, we propose a novel Source Authentication using Network Coding (SANC) scheme that can either complement state-of-the-art application-layer authentication schemes proposed in the literature or be used as a stand-alone scheme in network coding-based networks. Towards this objective, we propose a general framework for embedding the authentication information within the network coding Global Encoding Vector. This is attained using a mapping function that enforces a structure on the Global Encoding Vector to facilitate authentication at the destination. First, we illustrate the proposed concept using a simple mapping function, namely a parity bit within each network coding coefficient. Second, we present a detailed security analysis that reveals the security merits of the proposed scheme, contrasted against two baseline schemes that solely adopt application-layer authentication. Finally, we present simulation results pertaining to the network coding performance. Simulation results show that, for plausible scenarios, SANC achieves the same throughput as regular network coding. Furthermore, the results reveal that, for the same packet header, stronger security can be attained. This is confirmed for small as well as scalable networks encountered in practice.**

I. Introduction

Network coding has been proposed in the seminal paper by Ahlswede et al. [2] to achieve the multicast capacity of the network and has received considerable attention at the theoretical level, e.g. [14], [15]. Recently, there has been growing interest in exploring the benefits and potential tradeoffs of network coding in practical scenarios [4], [5], [7]. Network coding has shown higher throughput than conventional multicast theoretically [14] and experimentally [5].

In wireless ad hoc networks, passive attacks such as eavesdropping and traffic analysis arise since any malicious entity can sniff the traffic of the victim network. Several approaches were proposed in [1], [3], [6] to combat the traffic analysis and eavesdropping attacks. In [8], Lima et al. show how network coding can be leveraged to provide a free cipher. In [1], Fan et al. studied the potential of homomorphic encryption along with network coding to combat traffic analysis attacks. It hinges on the fact that there is a difference between the number of input packets and the number of output packets attributed to network coding. In addition, network coding processing introduces delays that differ according to the number of packets encoded. The aforementioned factors confuse the attacker and protect the network against timing attacks. Hence, protect the privacy of the communicating nodes.

Combining the authentication and privacy security requirements give rise to a fundamental tradeoff. The former enables the destination to assure that the source is a legitimate peer while the latter hides the identity of the source and intermediate nodes from malicious nodes. The privacy of the source takes different forms and may involve identity, location, etc. We divide our solution to this problem into two stages. In this paper, we focus on the first stage only that deals with the source authentication problem. Privacy preservation which constitutes the second stage is a subject of future research. Without proper authentication, the communicating nodes become more vulnerable to threats and, furthermore, breaking the authentication scheme aggravates the effect of adversaries.

Extensive work has been done to support data integrity (data authentication) in order to protect against active attacks, especially pollution attacks [13-14]. The essence of pollution attacks is to insert malicious packets which are mixed with legitimate packets at the intermediate nodes leading to data corruption at the destination. In this paper, we focus on active attacks that distribute false or misleading information. The adversary is not interested in destroying the network performance. In particular, we consider the problem of Source Authentication.

Encrypting the global encoding vector (GEV[1]), using homomorphic encryption, was proposed in [1], [3], [6] to protect against passive attacks while using network coding. In this paper, we embed authentication information into the GEV via enforcing a structure on the encoding coefficients at the source and maintaining that structure at intermediate nodes.

We argue that network coding can be used to provide security measures against active attacks in addition to its inherent throughput and reliability gains. Our work demonstrates that, by introducing minor modifications to network coding, we should be able to support source authentication with minimal impact on performance and without complicating the intermediate node processing.

Recently, the authentication problem was studied for particular types of networks. On the contrary of this approach, we provide a general framework for an authentication scheme that can either complement state-of-the-art application-layer authentication schemes proposed in the literature or be used as a standalone scheme in network coding-based networks.

We focus on the source authentication problem by providing

[1]We use the word tag interchangeably with the Global Encoding Vector.

978-1-4244-9920-5/11/$26.00 ©2011 IEEE

a simple scheme that embeds authentication information into the GEV of the linear network coding tag. It preserves the structure of the tag, necessary for authenticating the source at the destination, during the packet mixing process at intermediate nodes allowing almost the same decoding probability at the destination as quantified in Section V.

Finally, this work is inspired by the key observation that the packet mixing process inherent to network coding, jointly with homomorphic encryption, constitutes a compelling approach for source authentication with marginal impact on computation complexity and invertability probability.

A. Main Contribution

Our contribution in this paper is three-fold:

1) We leverage network coding packet mixing, along with homomorphic encryption, to authenticate source nodes.
2) We propose a scheme for embedding the authentication information into the network coding GEV using a simple mapping function with minimal impact on the decoding probability.
3) We show the efficiency and effectiveness of the proposed scheme by carrying out exhaustive simulations, security analysis, and discussing the complexity and limitations of our approach.

B. Paper Organization

In Section II, we introduce the necessary background. Section III constitutes the core of the paper as it describes the proposed SANC scheme problem setting, attack model, the basic idea (without homomorphic encryption), and finally, SANC with homomorphic encryption. In Section IV, we validate the proposed scheme. Afterwards, we analyze the proposed scheme and show its effectiveness with the aid of simulation results and security analysis in Section V. Finally, conclusions are drawn in Section VI.

II. Background

A. Linear Network Coding

A network is modeled as a directed graph G consisting of (V, E) where V is the set of vertices and E is the set of edges. e_{ij} is the edge between node i and node j. The capacity of all edges is equal and assumed to be unity. we define h as the capacity of the min-cut in this network. A noiseless communication between source node $s \epsilon V$ and $D \epsilon V$ where D is a set of multicast destinations. s encodes h packets, $\bar{x} = [x_1, ..., x_h]$, h times and sends out the encoded packets, $\bar{y} = [y_1, .., y_k, .., y_h]$, to all neighbors one at a time,

$$y_k^{(s)} = \sum_h \alpha_i x_i$$

where $k = 1, ..., h$, $y_k^{(s)}$ is called the k^{th} encoding of the source packets by node s. x_i and α_i are chosen from a Galois field $F = GF(2^p)$, where $i = 1, ..., h$ and p is a large prime number. s sends $y_k^{(s)}$ to all neighbors along with its encoding coefficients $\bar{\alpha}_k = [\alpha_1 ... \alpha_h]$. In this paper, we refer to the vector of encoding coefficients $\bar{\alpha}_k$ as the Global Encoding Vector (GEV). Intermediate node, i, calculates $y^{(i)}$ such that,

$$y^{(i)} = \sum_j \beta_j y_{ji}$$

where y_{ji} is the packet transmitted over the edge e_{ji} and β_j is a random coefficient chosen from the Galois field, F. Thus, we define the vector $\bar{\beta} = [\beta_1 ... \beta_j ...]$ as the local encoding vector. Each node in D collects h linearly independent (innovative) packets along with their GEV, constructs the global encoding matrix G [16], and decodes $\bar{x}$ such that,

$$\bar{x} = G^{-1} \bar{y}$$

An innovative packet is a packet that increases the rank of the global encoding matrix. This implies that G will be invertible iff the rank of the matrix is h. Hence, the set D needs at least h linearly independent packets to be able to decode.

B. Homomorphic Encryption

Homomorphic Encryption is one type of encryption where the arithmetic operations that takes place on cipher text is reflected on the plain text. Several homomorphic cryptosystems exist such as El Gamal cryptosystem [10] and Paillier cryptosystem [11]. In this paper, we are interested in the following properties of homomorphic encryption.

1) Addition Property : The summation of two cipher texts is equivalent to the encryption of their addition, that is,

$$E(A + B) = E(A) + E(B)$$

2) Multiplication by Scalar Property: The multiplication of a cipher text by a scalar value is equivalent to the encryption of the text multiplied by a scalar value, that is,

$$\alpha E(A) = E(\alpha A)$$

III. Source Authentication Using Network Coding

A. Problem Setting

A network is modeled as a directed graph with one source and multiple destinations, i.e. focus is on multicast sessions. We assume multi-hop routes have already been established and fixed.

In this work, we focus on the use of network coding for wireless ad hoc networks. In particular, we focus on intra-flow network coding where each node mixes packets belonging to the same flow. We assume all packets are of the same size which is a reasonable assumption since all packets belong to the same flow and is controlled by the same source node.

We also assume that a key management scheme exists to handle the assignment of the Authentication and Confidentiality keys between the communication peers.

We define the confidentiality key between the source node (s) and the destination nodes (D) as the key for encrypting the GEV at the source using a homomorphic encryption algorithm. On the other hand, the Authentication key is the key that uniquely identifies the source node to the intended destinations. The length of the authentication key, n, is greater than or equal to the number of coefficients in the GEV, h.

B. Attack Model

In this paper, we consider active and insider attacks. An active attacker may insert, delete or modify packet contents. However, we limit our attention to cases where the attacker does not have any incentive to destroy the network traffic and, hence, pollution attacks are out of the scope of this work. A mobile attacker can sniff different parts of the topology at different times. In addition, an insider attacker may participate in the routing and has access to the packets contents.

Finally, we consider the attacks that aim to send misleading or false information to the destination; namely: *Impersonation Attack***:** The attacker tries to imitate a legitimate user. Afterwards, he/she sends misleading information to the destination. The goal is to trust the information coming from the attacker by stealing the authentication key of a legitimate user.

C. Basic Idea

The main idea behind our scheme is to map the authentication key of length, n, to a certain pattern of the GEV at the source node by enforcing a certain structure on the chosen random coefficients and preserving this structure throughout the packet mixing process at intermediate nodes. Let the authentication key ($\bar{A}_{SD}$) exists between the source and destinations. Although the mapping function could be arbitrarily complex, depending on the desired level of security, we focus in this paper on a simple parity mapping function, denoted by $f(x)$, where x is an arbitrary bit of the authentication key, to illustrate the concept,

$$y = f(x) = \begin{cases} Randn \,\epsilon\, \{2Z\} & , x = 0 \\ Randn \,\epsilon\, \{2Z+1\} & , x = 1 \end{cases}$$

where $Z = 0, 1, 2, ...$ and $Randn$ is a random number generated according to an arbitrary distribution from Galois field.

Accordingly, this mapping function, $f(x)$, returns an odd coefficient if the corresponding authentication bit is one and an even coefficient if the corresponding authentication bit is zero. We preserve the pattern of odd and even coefficients of the GEV at the intermediate nodes and check this pattern at the destination to authenticate the source.

The SANC scheme consists of three main phases; the source phase, the intermediate phase and the destination phase. The source phase consists of two steps.

Step 1: Source node s chooses the encoding coefficient, denoted α_i according to the mapping function $f(x)$, hence,

$$\alpha_i = \begin{cases} Randn \,\epsilon\, \{2Z\} & , A^i_{SD} = 0 \\ Randn \,\epsilon\, \{2Z+1\} & , A^i_{SD} = 1 \end{cases}$$

where A^i_{SD} is the i^{th}authentication bit. The number of source packets (h) is chosen according to the min-cut max-flow theorem such that $n \geq h$. Without loss of generality, we will assume that $n = h$ for simplicity. s encodes a set of packets, $[p_1...p_h]$, via linearly mixing them as follows,

$$p^{(s)} = \textstyle\sum_{j=1}^{h} \alpha_j p_j$$

Step 2: s sends the encoded packet, $p^{(s)}$, accompanied by the GEV, $\bar{\alpha}$, to the next hops.

The source repeats step 1 and 2 at least h times to guarantee h innovative packets that enable the destination to decode.

Our primary objective, in the second phase, is to preserve the structure of the GEV in face of the packet mixing process that takes place as part of random linear network coding at the intermediate nodes. Towards that objective, its essential to mix an odd number of packets at each intermediate node (proven in Section IV). Hence, we propose to mix a packet twice if the number of coefficients is even as shown next,

$$p^{(i)} = \begin{cases} \sum_{j=1}^{m} \beta_j p_j & , m \,\epsilon\, \{2Z+1\} \\ \beta_1 p_1 + \sum_{j=1}^{m} \beta_j p_j & , m \,\epsilon\, \{2Z\} \end{cases} \tag{1}$$

where m is the number of incoming packets to node i, β_j is the j^{th} coefficient of the local encoding vector where,

$$\beta_j \,\epsilon\, \{2Z+1\}$$

and $j = 1, ..., m$. All $\alpha' s$ and $\beta' s$ are chosen randomly from a Galois field $GF(2^p)$.

In the last phase, we define an inverse mapping function $f^{-1}(y)$ such that,

$$x = f^{-1}(y) = \begin{cases} 0 & , y \,\epsilon\, \{2Z\} \\ 1 & , y \,\epsilon\, \{2Z+1\} \end{cases}$$

The inverse mapping function $f^{-1}(y)$ returns zero if the corresponding coefficient is even and returns one if the corresponding coefficient is odd. Each destination authenticates the source by checking whether the pattern in the GEV, after tag decryption, matches the authentication key at hand. Afterwards, it checks if the packet received is innovative with respect to the packets in its current buffer by checking the rank of the global encoding matrix. If the rank of the global encoding matrix reaches h then the destination calculates the matrix inverse and decodes the data. Otherwise, the packet will be added or dropped, depending on whether it is innovative or not, and the destination will wait for new authentic innovative packets.

D. Homomorphic Encryption SANC Scheme

In this section, we discuss the key role homomorphic encryption plays, along with our basic idea, to form the proposed SANC scheme.

Without encryption, it is straightforward for an adversary to sniff the tag and distill the Authentication key out of it (assuming it knows the SANC mapping function). Furthermore, adversaries can gather enough packets to construct the global encoding full-rank matrix.

We propose the use of Homomorphic encryption to the tag to solve the aforementioned two problems. First, it prevents global adversaries from early decoding. Second, it conceals the authentication information, embedded in the GEV, from adversaries as well as nodes participating in the packet mixing operation. Finally, homomorphic encryption permits intermediate nodes to carry out packet mixing operations without having to decrypt at each hop since the mixing operation can be performed blindly on the cipher text.

Next, we show how homomorphic encryption works with our basic scheme. Homomorphic encryption is applied to the GEV tag at the source node, after each of the h packet encoding operations, as explained in SectionIII-C such that,

$$\bar{c}_k = E\left(\begin{array}{cccc} \alpha_1 & \alpha_2 & \dots & \alpha_h \end{array}\right)$$
$$= \left[\begin{array}{cccc} c_1(k) & c_2(k) & \dots & c_h(k) \end{array}\right]$$

where $E(.)$ is a Homomorphic encryption function, $c_i(k) = E(\alpha_i)$, $1 \leq i \leq h$ and k is k^{th} encoding.

Each intermediate node performs the encoding of m packets such that,

$$\bar{c}^{(i)} = \beta_1 \bar{c}_1 + \ldots + \beta_m \bar{c}_m$$

Using homomorphic properties discussed earlier, $\bar{c}^{(i)}$ can be simplified to,

$$\bar{c}^{(i)} = E\left(\begin{array}{ccc} \sum_{i=1}^{m} \beta_i c_1(i) & \dots & \sum_{i=1}^{m} \beta_i c_h(i) \end{array}\right) \quad (2)$$

From (2), it can be seen that the encrypted GEV conceals the authentication information embedded in it since linearly mixing encrypted GEVs of packets incoming to node i yields an outgoing encrypted GEV that is hardly related to the incoming packets. This, in turn, makes it harder for adversaries to sniff the GEV thanks to the joint use of networking coding packet mixing along with encryption. This makes the tag continuously changing as packets proceed from hop to hop en route to the destination.

E. SANC Scalability Challenge

In this section, we study the scalability of the proposed SANC scheme. The scheme works fine for small networks, less than 10 hops, for the Galois Field sizes we used. For large networks, we faced a problem that we refer to as the finite field wrapping problem. This problem is attributed to the fact that elements from a finite field reach their maximum value after a certain number of hops (found to be 10 for $GF(2^8)$) and, hence, tend to wrap around. Although this wrapping phenomena does not cause any problem to network coding schemes in the literature based on random codes, it poses a serious challenge to the proposed SANC scheme as it may alter the structure of the GEV from hop to hop (e.g. change an odd value to an even one or vice versa for the parity mapping function at hand). This hurdle is caused by the structure we enforce on the network coding GEVs to embed authentication information. Further details are eliminated due to space limitations [17]. Finite field wrapping means that a value that exceeds the finite field size will be wrapped to a certain value belonging to the field.

To circumvent this fundamental hurdle, we construct the local encoding vector to be all ones, that is,

$$\bar{\beta} = \left[\begin{array}{cccc} 1 & 1 & \dots & 1 \end{array}\right]$$

The choice of $\bar{\beta}$ above is inspired by the multiplication operation between Galois Field elements for each coefficient in (2) which is the root cause of the unpredictable wrapping. Hence, choosing $\beta_i \, \forall i$ equals one reduces the operation to only summation of Galois Field elements, as shown in (3), which solves the wrapping problem. This, in turn, preserves a certain structure on the GEV for each output packet after the mixing operation of the input packets. The security implications for the above choice for $\bar{\beta}$ are discussed in [17].

Substituting in (2), the encoded packet is given by,

$$\bar{c}(new) = E\left(\begin{array}{cccc} \sum_{i=1}^{k} c_1(i) & \sum_{i=1}^{k} c_i & \dots & \sum_{i=1}^{k} c_h(i) \end{array}\right)$$

In the next section, we will provide an analytical validation of the proposed SANC scheme.

IV. Scheme Validation

To validate the proposed scheme, we need to prove that the authentication information embedded in the GEV will be preserved during the packet mixing process at intermediate nodes.

Since the parities of the GEV coefficients are all based on the bits of a single Authentication Key, the coefficients of the GEVs of all encoded packets belonging to the flow of interest would have the same parity. Next, we prove that the summation of the GEV coefficients from different packets generates an output packet with global encoding coefficients having the same parities as the input packets. This is of paramount importance to preserve the structure (parity) of the GEVs as it flows from the source to destination. We will prove this result for the un-encrypted SANC scheme since the proof of its encrypted counterpart follows directly using the homomorphic encryption properties in Section II. We define the output packet GEV, $\bar{l}$, as,

$$\bar{l} = \left[\sum_{i=1}^{m} \alpha_1(i) \ldots \sum_{i=1}^{m} \alpha_h(i)\right] = [l_1 \ldots l_m]$$

Given m incoming packets with GEVs $\bar{\alpha(i)}$, $1 \leq i \leq m$, then the GEV of the outgoing packet $\bar{l}$ preserves the parity of the incoming packets, that is,

$$l_i \wedge 1 = \begin{cases} 0 & , A_{SD}^{i} = 0 \\ 1 & , A_{SD}^{i} = 1 \end{cases}$$

where $\wedge$ denotes bitwise AND operation. This operation masks all bits except for the least significant bit

Proof: Given the intermediate node mixing equation, (1), in Section III-C, we need to prove the theorem for m odd only, i.e. $m \, \epsilon \, \{\, 2Z + 1 \,\}$ where m is the number of packets to be encoded at the intermediate node.

Given that the sum of m odd numbers (where m is odd) is also odd, i.e.

$$\left[\sum_{i=1}^{m} Randn_i \in \{\, 2Z + 1 \,\}\right] \in \{\, 2Z + 1 \,\}$$

where $1 \leq i \leq m$.

Similarly, the sum of m even numbers (where m is odd) is even, i.e.

$$\left[\sum_{i=1}^{m} Randn_i \in \{\, 2Z \,\}\right] \in \{\, 2Z \,\}$$

where $1 \leq i \leq m$.

Thus, in both cases above, the output coefficient parities will match the input coefficient parities which proves the theorem. ■

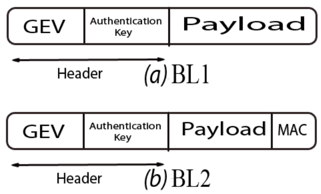

Fig. 1. Baseline Schemes

V. SANC Scheme Analysis

A. Baseline Schemes

In order to demonstrate the effectiveness and merits of SANC, we consider two generic baseline schemes, namely Baseline 1 (BL1) and Baseline 2 (BL2), which represent state-of-the-art authentication schemes that do not leverage network coding. We assume that the length of the authentication key in the two baselines is exactly similar to SANC. Next, we describe the two baselines and explain the rationale behind them.

Baseline 1: is a simple scheme where the authentication key is encrypted with a confidentiality key and is included in the header next to the network coding GEV tag. (Figure 1(a)) However, BL1 has a fundamental problem since there is no correlation between the data, in the payload, and the header and hence, BL1 is vulnerable to impersonation attacks. An adversary can launch an attack through extracting the authentication key from legitimate packets and implanting it into fake packets. On the other hand, SANC incorporates correlation between the GEV and the payload of the packet since altering one of the coefficients, or any part of the payload, will corrupt the entire packet.

Baseline 2: overcomes the limitation of BL1 via incorporating a message authentication code (MAC), at the end of the packet, that ties the GEV to the packet payload. It is important to note that the MAC needs to be recalculated and checked at each hop. Hence, BL2 requires more computation, at intermediate nodes, than SANC. Also, it requires a new MAC key distributed to all nodes on the path. However, using homomorphic MAC [9] solves the problem of per hop MAC decryption, in the same manner as our scheme.

B. Simulation Setup

We contrast the security merits and throughput performance of SANC to BL1 and BL2 with the aid of extensive simulations built using our SANC simulator in C++. We simulate a network with stationary nodes where a source node generates constant bit rate (CBR) traffic at a rate of 25 messages per second and a single randomly selected attacker attempts to launch Impersonation attacks. Nodes are deployed on a square grid and the transmission range of a node is set such that the immediate horizontal and vertical neighbors of a node are only its direct neighbors. The simulation results are averaged over 84 runs per topology. Table I summarizes the network and authentication scheme simulation parameters.

TABLE I
Simulation parameters

Simulation Parameter	Value
Message Length	512 bytes
Authentication Key Length	16 bits
Simulation Time	1000 Secs
Number of Nodes	$6, 12, 14, 18, 20, 25, 35, 57$
Field Size	$2^8 - 2^{12}$

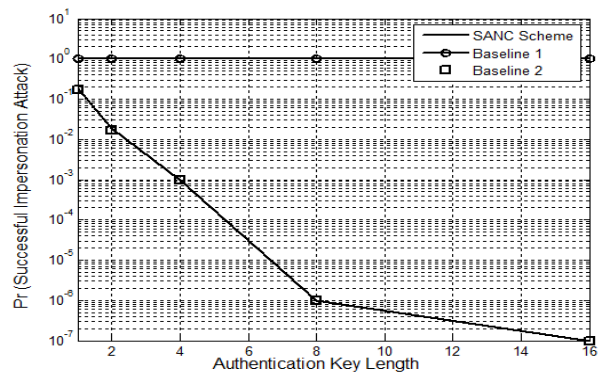

Fig. 2. Probability of successful attack vs. Authentication Key Length (57 Node Network)

C. Performance Evaluation

In this section, we demonstrate, quantitatively, the security and performance merits of SANC compared to the baselines.

We show in Fig. 2 the probability of successful impersonation attack versus the authentication key length for the three authentication schemes under consideration. In this Figure, we have choosed the MAC bits to be equal to the authentication key bits so that BL2 can achieve the same security performance as SANC scheme. A number of key observations can be distilled from this figure. First, we notice that BL1 exhibits very poor security performance attributed to its naive authentication scheme which can be easily broken via sniffing the encrypted authentication key and using it to impersonate legitimate users as explained before. Second, the probability of successful impersonation decreases, for both SANC and BL2, as the authentication key length increases which agrees with intuition. Finally, the security resistance of SANC and BL2, against impersonation attacks, is essentially the same. This somewhat interesting result is attributed to the fact that both schemes create correlation between the authentication key and payload in order to make it harder for an impersonator to sniff the authentication key. However, BL2 requires an extra key between the source and the destination in addition to adding extra bits for the MAC. Therefore, BL2 depends on the decryption probability which achieves the same security as our scheme, yet, with the extra overhead bits to correlate the authentication key and packet header to the packet payload. This correlation is created for free under SANC, thanks to the GEV that already correlates the packet header to the packet payload. Thus, SANC achieves comparable security to BL2 while saving the MAC bits overhead which could be

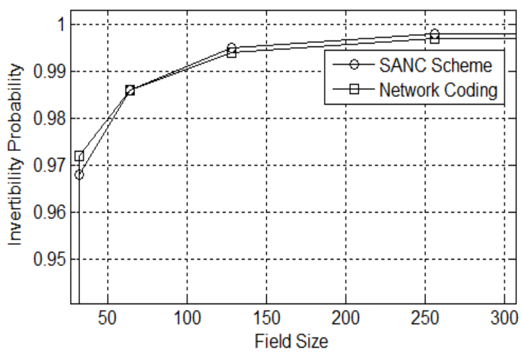

Fig. 3. Invertibility Probability vs. Field Size (20 Node Network)

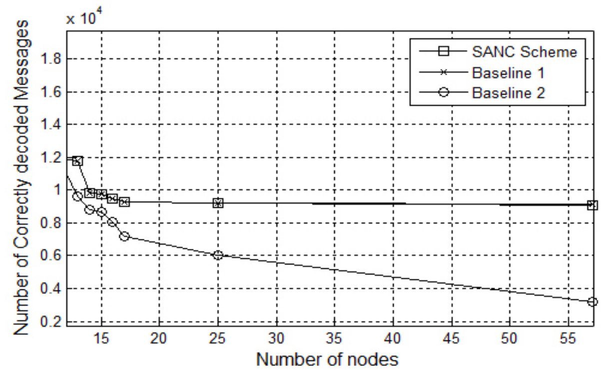

Fig. 4. Number of decoded Messages vs. Number of nodes.

considerable.

Next, we demonstrate in Fig. 3 that SANC does not degrade the invertibility probability, which is an important metric with direct impact on the end-to-end throughput, compared to plain network coding schemes with no security provisions. It is evident that consuming one bit in each GEV coefficient to store authentication data (i.e. key) is equivalent to halving the field size. Decreasing the field size directly affects the invertibility probability. Nevertheless, the results in Fig. 3 show that, beyond a certain field size (i.e. 2^8), the invertibility probability remains almost the same. More importantly, it shows that SANC exhibits invertibility probability similar to plain network coding. Hence, we conclude that halving the Galois field size, to embed the authentication key in the GEV, has hardly any effect on the message decodability probability of SANC. This reveals a compelling feature of SANC, namely providing source authentication provisions with hardly any impact on the network coding throughput performance.

Finally, we show in Fig. 4 the behavior of the number of decodable messages at the destination with the network size. The decreasing trends for all schemes is attributed to the growth of non-innovative packets which, in turn, reduces the number of the linearly independent packets needed to invert the GEM and decode the source's original message. More importantly, the throughput of BL2 decreases dramatically with the increase of network size. Thus, we conclude that SANC is more scalable than BL2 due to the per hop MAC decryption necessary for BL2.

VI. Conclusion

The essence of the proposed SANC scheme is to enforce a structure on the GEV using a mapping function, e.g. parity bit in each linear encoding coefficient that matches the corresponding bit in the authentication key. The major challenge is to preserve this bit pattern, in the GEV, throughout the packet mixing process at intermediate nodes. We proved the correctness of our scheme and showed its effectiveness using analysis and extensive network simulations. Our work can be augmented to provide message authentication (data integrity) service at the destination. It can be extended to tackle the fundamental authentication-privacy trade-off. Adopting more complex mapping functions, within the proposed framework, and the associated complexity-security trade-off is another interesting research direction.

References

[1] Y. Fan, Y. Jiang, H. Zhu and X. Shen, An Efficient Privacy-Preserving Scheme against Traffic Analysis Attacks in Network Coding, IEEE INFOCOM, 2009.

[2] R. Ahlswede, N. Cai, S.-Y.R. Li and R.W. Yeung, Network information flow, IEEE Transactions on Information Theory, vol. 46, no. 4, pp. 1204-1216, 2000.

[3] P. Zhang, Y. Jiang, C. Lin, Y. Fan and X. Shen, P-Coding: Secure Network Coding against Eavesdropping Attacks, IEEE INFOCOM, 2010.

[4] J.S. Park, D.S. Lum, F. Soldo, M. Gerla and M. Medard, Performance of Network Coding in Ad Hoc Networks, IEEE Military Communications Conference (MILCOM), 2006.

[5] C. Campolo, C. Casetti, CF Chiasserini and S. Tarapiah, Performance of network coding for ad hoc networks in realistic simulation scenarios, International Conference on Telecommunications (ICT), 2009.

[6] J.P. Vilela , L. Lima and J. Barros, Lightweight Security for Network Coding, IEEE International Conference on Communications (ICC), 2008.

[7] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard and J. Crowcroft, XORs in the Air: Practical Wireless Network Coding, IEEE/ACM Transactions on Networking, vol. 16, no. 3, pp. 497-510, 2008.

[8] L. Lima, M. Medard and J. Barros, Random Linear Network Coding: A free cipher?, IEEE International Symposium on Information Theory (ISIT), 2007.

[9] S. Agrawal and D. Boneh, Homomorphic MACs: MAC-Based Integrity for Network Coding, 7th International Conference on Applied Cryptography and Network Security, 2009.

[10] T. El Gamal, A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, IEEE Transactions on Information Theory, vol. 31, no. 4, pp469–472, 1985.

[11] P. Paillier, Public-Key Cryptosystems Based on Composite Degree Residuosity Classes, EUROCRYPT 1999.

[12] Z. Yu, Y. Wei, B. Ramkumar and Y. Guan, An Efficient Signature-Based Scheme for Securing Network Coding Against Pollution Attacks, IEEE INFOCOM 2008.

[13] F. Oggier and H. Fathi, An Authentication Code against Pollution Attacks in Network Coding, arXiv:0909.3146, 2009.

[14] R. Koetter and M. Medard, An Algebraic Approach to Network Coding, IEEE/ACM Transactions on Networking, vol. 11, no. 5, pp. 782- 795, Oct. 2003.

[15] C. Fragouli, J.Y. Le Boudec and J. Widmer, Network coding: an instant primer, ACM SIGCOMM Computer Communication Review, 2006.

[16] S.-Y.R. Li, R.W. Yeung and Ning Cai, Linear network coding, Information Theory, IEEE Transactions on , vol.49, no.2, pp.371-381, Feb. 2003.

[17] A. Fathy, T. ElBatt and M. Youssef, SANC: Source Authentication Using Network Coding, Nile University Technical Report, In Preparation.