Hop-by-Hop Message Authentication and Source Privacy in Wireless Sensor Networks

Jian Li, Yun Li, Jian Ren, *Senior Member, IEEE*, and Jie Wu, *Fellow, IEEE*

Abstract—Message authentication is one of the most effective ways to thwart unauthorized and corrupted messages from being forwarded in wireless sensor networks (WSNs). For this reason, many message authentication schemes have been developed, based on either symmetric-key cryptosystems or public-key cryptosystems. Most of them, however, have the limitations of high computational and communication overhead in addition to lack of scalability and resilience to node compromise attacks. To address these issues, a polynomial-based scheme was recently introduced. However, this scheme and its extensions all have the weakness of a built-in threshold determined by the degree of the polynomial: when the number of messages transmitted is larger than this threshold, the adversary can fully recover the polynomial. In this paper, we propose a scalable authentication scheme based on elliptic curve cryptography (ECC). While enabling intermediate nodes authentication, our proposed scheme allows any node to transmit an unlimited number of messages without suffering the threshold problem. In addition, our scheme can also provide message source privacy. Both theoretical analysis and simulation results demonstrate that our proposed scheme is more efficient than the polynomial-based approach in terms of computational and communication overhead under *comparable* security levels while providing message source privacy.

Index Terms—Hop-by-hop authentication, symmetric-key cryptosystem, public-key cryptosystem, source privacy, simulation, wireless sensor networks (WSNs), distributed algorithm, decentralized control

1 Introduction

Message authentication plays a key role in thwarting unauthorized and corrupted messages from being forwarded in networks to save the precious sensor energy. For this reason, many authentication schemes have been proposed in literature to provide message authenticity and integrity verification for wireless sensor networks (WSNs) [1], [2], [3], [4], [5]. These schemes can largely be divided into two categories: public-key based approaches and symmetric-key based approaches.

The symmetric-key based approach requires complex key management, lacks of scalability, and is not resilient to large numbers of node compromise attacks since the message sender and the receiver have to share a secret key. The shared key is used by the sender to generate a message authentication code (MAC) for each transmitted message. However, for this method, the authenticity and integrity of the message can only be verified by the node with the shared secret key, which is generally shared by a group of sensor nodes. An intruder can compromise the key by capturing a single sensor node. In addition, this method does not work in multicast networks.

To solve the scalability problem, a secret polynomial based message authentication scheme was introduced in [3]. The idea of this scheme is similar to a threshold secret sharing, where the threshold is determined by the degree of the polynomial. This approach offers information-theoretic security of the shared secret key when the number of messages transmitted is less than the threshold. The intermediate nodes verify the authenticity of the message through a polynomial evaluation. However, when the number of messages transmitted is larger than the threshold, the polynomial can be fully recovered and the system is completely broken.

An alternative solution was proposed in [4] to thwart the intruder from recovering the polynomial by computing the coefficients of the polynomial. The idea is to add a random noise, also called a *perturbation factor*, to the polynomial so that the coefficients of the polynomial cannot be easily solved. However, a recent study shows that the random noise can be completely removed from the polynomial using error-correcting code techniques [6].

For the public-key based approach, each message is transmitted along with the digital signature of the message generated using the sender's private key. Every intermediate forwarder and the final receiver can authenticate the message using the sender's public key [7], [8]. One of the limitations of the public-key based scheme is the high computational overhead. The recent progress on elliptic curve cryptography (ECC) shows that the public-key schemes can be more advantageous in terms of

- *J. Li and J. Ren are with the Department of Electrical & Computer Engineering, Michigan State University, East Lansing, MI 48824-1226. E-mail: {lijian6, renjian}@egr.msu.edu.*
- *Y. Li is with the SPD Department, Microsoft, Redmond, WA 98052. E-mail: yunl@microsoft.com.*
- *J. Wu is with the Department of Computer & Information Sciences, Temple University, Philadelphia, PA 19122. E-mail: jiewu@temple.edu.*

Manuscript received 17 Oct. 2012; revised 18 Feb. 2013; accepted 6 Apr. 2013; date of publication 16 Apr. 2013; date of current version 21 Mar. 2014.
Recommended for acceptance by W. Lou.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPDS.2013.119

computational complexity, memory usage, and security resilience, since public-key based approaches have a simple and clean key management [9].

In this paper, we propose an unconditionally secure and efficient source anonymous message authentication (SAMA) scheme based on the optimal modified ElGamal signature (MES) scheme on elliptic curves. This MES scheme is secure against adaptive chosen-message attacks in the random oracle model [10]. Our scheme enables the intermediate nodes to authenticate the message so that all corrupted message can be detected and dropped to conserve the sensor power. While achieving compromise-resiliency, flexible-time authentication and source identity protection, our scheme does not have the threshold problem. Both theoretical analysis and simulation results demonstrate that our proposed scheme is more efficient than the polynomial-based algorithms under comparable security levels.

The major contributions of this paper are the following:

1. We develop a source anonymous message authentication code (SAMAC) on elliptic curves that can provide unconditional source anonymity.
2. We offer an efficient hop-by-hop message authentication mechanism for WSNs without the threshold limitation.
3. We devise network implementation criteria on source node privacy protection in WSNs.
4. We propose an efficient key management framework to ensure isolation of the compromised nodes.
5. We provide extensive simulation results under ns-2 and TelosB on multiple security levels.

To the best of our knowledge, this is the first scheme that provides hop-by-hop node authentication without the threshold limitation, and has performance better than the symmetric-key based schemes. The distributed nature of our algorithm makes the scheme suitable for decentralized networks.

The remainder of this paper is organized as follows: Section 2 presents the terminology and the preliminary that will be used in this paper. Section 3 discusses the related work, with a focus on polynomial-based schemes. Section 4 describes the proposed source anonymous message authentication scheme on elliptic curves. Section 5 discusses the ambiguity set (AS) selection strategies for source privacy. Section 6 describes key management and compromised node detection. Performance analysis and simulation results are provided in Section 7. We conclude in Section 8.

2 Terminology and Preliminary

In this section, we will briefly describe the terminology and the cryptographic tools that will be used in this paper.

2.1 Threat Model and Assumptions

The WSNs are assumed to consist of a large number of sensor nodes. We assume that each sensor node knows its relative location in the sensor domain and is capable of communicating with its neighboring nodes directly using geographic routing. The whole network is fully connected through multi-hop communications. We assume there is a security server (SS) that is responsible for generation, storage and distribution of the security parameters among the network. This server will never be compromised. However, after deployment, the sensor nodes may be captured and compromised by attackers. Once compromised, all information stored in the sensor nodes can be accessed by the attackers. The compromised nodes can be reprogrammed and fully controlled by the attackers. However, the compromised nodes will not be able to create new public keys that can be accepted by the SS and other nodes.

Based on the above assumptions, this paper considers two types of attacks launched by the adversaries:

- *Passive attacks*. Through passive attacks, the adversaries could eavesdrop on messages transmitted in the network and perform traffic analysis.
- *Active attacks*. Active attacks can only be launched from the compromised sensor nodes. Once the sensor nodes are compromised, the adversaries will obtain all the information stored in the compromised nodes, including the security parameters of the compromised nodes. The adversaries can modify the contents of the messages, and inject their own messages.

2.2 Design Goals

Our proposed authentication scheme aims at achieving the following goals:

- *Message authentication.* The message receiver should be able to verify whether a received message is sent by the node that is claimed, or by a node in a particular group. In other words, the adversaries cannot pretend to be an innocent node and inject fake messages into the network without being detected.
- *Message integrity.* The message receiver should be able to verify whether the message has been modified en-route by the adversaries. In other words, the adversaries cannot modify the message content without being detected.
- *Hop-by-hop message authentication.* Every forwarder on the routing path should be able to verify the authenticity and integrity of the messages upon reception.
- *Identity and location privacy.* The adversaries cannot determine the message sender's ID and location by analyzing the message contents or the local traffic.
- *Node compromise resilience.* The scheme should be resilient to node compromise attacks. No matter how many nodes are compromised, the remaining nodes can still be secure.
- *Efficiency.* The scheme should be efficient in terms of both computational and communication overhead.

2.3 Terminology

Privacy is sometimes referred to as anonymity. Communication anonymity in information management has been discussed in a number of previous works [11], [12], [13], [14], [15], [16]. It generally refers to the state of being

unidentifiable within a set of subjects. This set is called the AS. *Sender anonymity* means that a particular message is not linkable to any sender, and no message is linkable to a particular sender.

We will start with the definition of the unconditionally secure SAMA.

Definition 1 (SAMA). *A* SAMA *consists of the following two algorithms:*

- Generate $(m, Q_1, Q_2, \ldots, Q_n)$. *Given a message m and the public keys $Q_1, Q_2, \ldots, Q_n$ of the AS $\mathcal{S} = \{A_1, A_2, \ldots, A_n\}$, the actual message sender $A_t, 1 \leq t \leq n$, produces an anonymous message $\mathcal{S}(m)$ using its own private key d_t.*
- Verify $\mathcal{S}(m)$. *Given a message m and an anonymous message $\mathcal{S}(m)$, which includes the public keys of all members in the AS, a verifier can determine whether $\mathcal{S}(m)$ is generated by a member in the AS.*

The security requirements for SAMA include:

- Sender ambiguity. *The probability that a verifier successfully determines the real sender of the anonymous message is exactly $1/n$, where n is the total number of members in the AS.*
- Unforgeability. *An anonymous message scheme is unforgeable if no adversary, given the public keys of all members of the AS and the anonymous messages $m_1, m_2, \ldots, m_n$ adaptively chosen by the adversary, can produce in polynomial time a new valid anonymous message with non-negligible probability.*

In this paper, the user ID and the user public key will be used interchangeably without making any distinctions.

2.4 Modified ElGamal Signature Scheme

Definition 2 (MES). *The modified ElGamal signature scheme [17] consists of the following three algorithms:*

Key generation algorithm. *Let p be a large prime and g be a generator of $\mathbb{Z}_p^*$. Both p and g are made public. For a random private key $x \in \mathbb{Z}_p$, the public key y is computed from $y = g^x \bmod p$.*

Signature algorithm. *The MES can also have many variants [18], [19]. For the purpose of efficiency, we will describe the variant, called* optimal scheme. *To sign a message m, one chooses a random $k \in \mathbb{Z}_{p-1}^*$, then computes the exponentiation $r = g^k \bmod p$ and solves s from:*

$$s = rxh(m, r) + k \bmod (p - 1), \tag{1}$$

where h is a one-way hash function. The signature of message m is defined as the pair (r, s).

Verification algorithm. *The verifier checks whether the signature equation $g^s = ry^{rh(m,r)} \bmod p$. If the equality holds true, then the verifier* `Accepts` *the signature, and* `Rejects` *otherwise.*

3 Related Work

In [1], [2], symmetric key and hash based authentication schemes were proposed for WSNs. In these schemes, each symmetric authentication key is shared by a group of sensor nodes. An intruder can compromise the key by capturing a single sensor node. Therefore, these schemes are not resilient to node compromise attacks. Another type of symmetric-key scheme requires synchronization among nodes. These schemes, including TESLA [5] and its variants, can also provide message sender authentication. However, this scheme requires initial time synchronization, which is not easy to be implemented in large scale WSNs. In addition, they also introduce delay in message authentication, and the delay increases as the network scales up.

A secret polynomial based message authentication scheme was introduced in [3]. This scheme offers information-theoretic security with ideas similar to a threshold secret sharing, where the threshold is determined by the degree of the polynomial. When the number of messages transmitted is below the threshold, the scheme enables the intermediate node to verify the authenticity of the message through polynomial evaluation. However, when the number of messages transmitted is larger than the threshold, the polynomial can be fully recovered and the system is completely broken. To increase the threshold and the complexity for the intruder to reconstruct the secret polynomial, a random noise, also called a perturbation factor, was added to the polynomial in [4] to thwart the adversary from computing the coefficient of the polynomial. However, the added perturbation factor can be completely removed using error-correcting code techniques [6].

For the public-key based approach, each message is transmitted along with the digital signature of the message generated using the sender's private key. Every intermediate forwarder and the final receiver can authenticate the message using the sender's public key. The recent progress on ECC shows that the public-key schemes can be more advantageous in terms of memory usage, message complexity, and security resilience, since public-key based approaches have a simple and clean key management [9].

The existing anonymous communication protocols are largely stemmed from either mixnet [11] or DC-net [12]. A mixnet provides anonymity via packet re-shuffling through a set of mix servers (with at least one being trusted). In a mixnet, a sender encrypts an outgoing message, and the ID of the recipient, using the public key of the mix. The mix accumulates a batch of encrypted messages, decrypts and reorders these messages, and forwards them to the recipients. Since mixnet-like protocols rely on the statistical properties of the background traffic, they cannot provide provable anonymity. DC-net [12], [16] is an anonymous multi-party computation scheme. Some pairs of the participants are required to share secret keys. DC-net provides perfect (information-theoretic) sender anonymity without requiring trusted servers. However, in DC-net, only one user can send at a time, so it takes additional bandwidth to handle collision and contention.

Recently, message sender anonymity based on ring signatures was introduced [20]. This approach enables the message sender to generate a source anonymous message

signature with content authenticity assurance. To generate a ring signature, a ring member randomly selects an AS and forges a message signature for all other members. Then he *uses his trap-door information to glue the ring together*. The original scheme has very limited flexibility and very high complexity. Moreover, the original paper only focused on the cryptographic algorithm, and the relevant network issues were left unaddressed.

4 Proposed Source Anonymous Message Authentication on Elliptic Curves

In this section, we propose an unconditionally secure and efficient SAMA. The main idea is that for each message m to be released, the message sender, or the sending node, generates a source anonymous message authenticator for the message m. The generation is based on the MES scheme on elliptic curves. For a ring signature, each ring member is required to compute a forgery signature for all other members in the AS. In our scheme, the entire SAMA generation requires only three steps, which link all non-senders and the message sender to the SAMA alike. In addition, our design enables the SAMA to be verified through a single equation without individually verifying the signatures.

4.1 Proposed MES Scheme on Elliptic Curves

Let $p > 3$ be an odd prime. An elliptic curve E is defined by an equation of the form:

$$E: \ y^2 = x^3 + ax + b \bmod p,$$

where $a, b \in \mathbb{F}_p$, and $4a^3 + 27b^2 \not\equiv 0 \bmod p$. The set $E(\mathbb{F}_p)$ consists of all points $(x, y) \in \mathbb{F}_p$ on the curve, together with a special point $\mathcal{O}$, called the point at infinity.

Let $G = (x_G, y_G)$ be a base point on $E(\mathbb{F}_p)$ whose order is a very large value N. User A selects a random integer $d_A \in [1, N-1]$ as his private key. Then, he can compute his public key Q_A from $Q_A = d_A \times G$.

Signature generation algorithm. For Alice to sign a message m, she follows these steps:

1. Select a random integer k_A, $1 \le k_A \le N-1$.
2. Calculate $r = x_A \bmod N$, where $(x_A, y_A) = k_A G$. If $r = 0$, go back to step 1.
3. Calculate $h_A \xleftarrow{l} h(m, r)$, where h is a cryptographic hash function, such as SHA-1, and $\xleftarrow{l}$ denotes the l leftmost bits of the hash.
4. Calculate $s = rd_A h_A + k_A \bmod N$. If $s = 0$, go back to step 2.
5. The signature is the pair (r, s).

Signature verification algorithm. For Bob to authenticate Alice's signature, he must have a copy of her public key Q_A, then he:

1. Checks that $Q_A \neq \mathcal{O}$, otherwise invalid
2. Checks that Q_A lies on the curve
3. Checks that $nQ_A = \mathcal{O}$

After that, Bob follows these steps to verify the signature:

1. Verify that r and s are integers in $[1, N-1]$. If not, the signature is invalid.
2. Calculate $h_A \xleftarrow{l} h(m, r)$, where h is the same function used in the signature generation.
3. Calculate $(x_1, x_2) = sG - rh_A Q_A \bmod N$.
4. The signature is valid if $r = x_1 \bmod N$, invalid otherwise.

In fact, if the signature is correctly generated, then:

$$\begin{aligned}(x_1, x_2) &= sG - rh_A Q_A \\ &= (rd_A h_A + k_A)G - rh_A Q_A \\ &= k_A G + rh_A Q_A - rh_A Q_A \\ &= k_A G.\end{aligned}$$

Therefore, we have $x_1 = r$, and the verifier should Accept the signature.

4.2 Proposed SAMA on Elliptic Curves

Suppose that the message sender (say Alice) wishes to transmit a message m anonymously from her network node to any other nodes. The AS includes n members, $A_1, A_2, \ldots, A_n$, for example, $\mathcal{S} = \{A_1, A_2, \ldots, A_n\}$, where the actual message sender Alice is A_t, for some value $t, 1 \le t \le n$. In this paper, we will not distinguish between the node A_i and its public key Q_i. Therefore, we also have $\mathcal{S} = \{Q_1, Q_2, \ldots, Q_n\}$.

Authentication generation algorithm. Suppose m is a message to be transmitted. The private key of the message sender Alice is $d_t, 1 \le t \le N$. To generate an efficient SAMA for message m, Alice performs the following three steps:

1. Select a random and pairwise different k_i for each $1 \le i \le n-1, i \neq t$ and compute r_i from $(r_i, y_i) = k_i G$.
2. Choose a random $k_i \in \mathbb{Z}_p$ and compute r_t from $(r_t, y_t) = k_t G - \sum_{i \neq t} r_i h_i Q_i$ such that $r_t \neq 0$ and $r_t \neq r_i$ for any $i \neq t$, where $h_i \xleftarrow{l} h(m, r_i)$.
3. Compute $s = k_t + \sum_{i \neq t} k_i + r_t d_t h_t \bmod N$.

The SAMA of the message m is defined as:

$$\mathcal{S}(m) = (m, \mathcal{S}, r_1, y_1, \ldots, r_n, y_n, s).$$

4.3 Verification of SAMA

Verification algorithm. For Bob to verify an alleged SAMA $(m, \mathcal{S}, r_1, y_1, \ldots, r_n, y_n, s)$, he must have a copy of the public keys $Q_1, \ldots, Q_n$. Then he:

1. Checks that $Q_i \neq \mathcal{O}, i = 1, \ldots, n$, otherwise invalid
2. Checks that $Q_i, i = 1, \ldots, n$ lies on the curve
3. Checks that $nQ_i = \mathcal{O}, i = 1, \ldots, n$

After that, Bob follows these steps:

1. Verify that $r_i, y_i, i = 1, \ldots, n$ and s are integers in $[1, N-1]$. If not, the signature is invalid.
2. Calculate $h_i \xleftarrow{l} h(m, r_i)$, where h is the same function used in the signature generation.
3. Calculate $(x_0, y_0) = sG - \sum_{i=1}^{n} r_i h_i Q_i$
4. The signature is valid if the first coordinate of $\sum_i (r_i, y_i)$ equals x_0, invalid otherwise.

In fact, if the SAMA has been correctly generated without being modified, then we compute:

$$
\begin{aligned}
(x_0, y_0) &= sG - \sum_{i=1}^{n} r_i h_i Q_i \\
&= \left(k_t + \sum_{i \neq t} k_i + r_t d_t h_t\right) G - \sum_i r_i h_i Q_i \\
&= \sum_{i \neq t} k_i G + \left(k_t G - \sum_{i \neq t} r_i h_i Q_i\right) \\
&= \sum_{i \neq t} (r_i, y_i) + (r_t, y_t) \\
&= \sum_i (r_i, y_i).
\end{aligned}
$$

Therefore, the verifier should always `Accept the SAMA.`

Remark 1. It is apparent that when $n = 1$, SAMA becomes a simple signature algorithm.

4.4 Security Analysis

In this section, we will prove that the proposed SAMA scheme can provide unconditional source anonymity and provable unforgeability against adaptive chosen-message attacks.

4.4.1 Anonymity

In order to prove that the proposed SAMA can ensure unconditional source anonymity, we have to prove that: 1) for anybody other than the members of $\mathcal{S}$, the probability to successfully identify the real sender is $1/n$, and 2) anybody from $\mathcal{S}$ can generate SAMAs.

Theorem 1. *The proposed SAMA can provide unconditional message sender anonymity.*

Proof. The identity of the message sender is unconditionally protected with the proposed SAMA scheme. This is because, regardless of the sender's identity, there are exactly $(N-1)(N-2)\ldots(N-n)$ different options to generate the SAMA. All of them can be chosen by any members in the AS during the SAMA generation procedure with equal probability without depending on any complexity-theoretic assumptions. The proof for the second part, that anybody from $\mathcal{S}$ can generate the SAMA, is straightforward. This finishes the proof of this theorem. □

4.4.2 Unforgeability

The design of the proposed SAMA relies on the ElGamal signature scheme. Signature schemes can achieve different levels of security. Security against existential forgery under adaptive-chosen message attacks is the maximum level of security.

In this section, we will prove that the proposed SAMA is secure against existential forgery under adaptive-chosen message attacks in the random oracle model [21]. The security of our result is based on ECC, which assumes that the computation of discrete logarithms on elliptic curves is computationally infeasible. In other words, no efficient algorithms are known for non-quantum computers.

We will introduce two lemmas. Lemma 1 is the Splitting Lemma, which is a well-known probabilistic lemma from reference [10]. The basic idea of the Splitting Lemma is that when a subset Z is "large" in a product space $X \times Y$, it will have many "large" sections. Lemma 2 is a slight modification of the Forking Lemma presented in [10]. The proofs of the two lemmas are mainly probability theory related. We will skip the proofs of these two lemmas here.

Lemma 1 (The Splitting Lemma). *Let $Z \subset X \times Y$ such that $\Pr[(x, y) \in Z] \geq \varepsilon$. For any $\alpha < \varepsilon$, define $W = \{(x, y) \in X \times Y | \Pr_{y' \in Y}[(x, y') \in Z] \geq \varepsilon - \alpha\}$, and $\overline{W} = (X \times Y) \backslash W$, then the following statements hold:*

1. $\Pr[W] \geq \alpha$.
2. $\forall (x, y) \in W, \Pr_{y' \in Y}[(x, y') \in Z] \geq \varepsilon - \alpha$.
3. $\Pr[W|Z] \geq \alpha/\varepsilon$.

Lemma 2 (The Forking Lemma). *Let $\mathcal{A}$ be a Probabilistic Polynomial Time (PPT) Turing machine. Given only the public data as input, if $\mathcal{A}$ can find, with non-negligible probability, a valid SAMA $(m, \mathcal{S}, r_1, y_1, \ldots, r_n, y_n, h_1, \ldots, h_n, s)$ within a bounded polynomial time T, then with non-negligible probability, a replay of this machine, which has control over $\mathcal{A}$ and a different oracle, outputs another valid SAMA $(m, \mathcal{S}, r_1, y_1, \ldots, r_n, y_n, h'_1, \ldots, h'_n, s)$, such that $h_i = h'_i$, for all $1 \leq i \leq v, i \neq j$ for some fixed j.*

Theorem 2. *The proposed SAMA is secure against adaptive chosen-message attacks in the random oracle model.*

Proof. (Sketch) If an adversary can forge a valid SAMA with non-negligible probability, then according to the Forking Lemma, the adversary can obtain two valid SAMAs $\mathcal{S}(m) = (m, \mathcal{S}, r_1, y_1, \ldots, r_n, y_n, h_1, \ldots, h_n, s)$, and $\mathcal{S}(m) = (m, \mathcal{S}, r_1, y_1, \ldots, r_n, y_n, h'_1, \ldots, h'_n, s)$, such that for $1 \leq i \leq n, i \neq j, h_i = h'_i, h_j \neq h'_j$ and $sG - \sum_{i=1}^{n} r_i h_i Q_i = \sum_i (r_i, y_i)$, $s'G - \sum_{i=1}^{n} r_i h'_i Q_i = \sum_i (r_i, y_i)$.

Subtracting these two equations, we obtain $(s - s')$ $G = r_j(h_j - h'_j)Q_j$. Equivalently, we have:

$$Q_j = \frac{s - s'}{r_j(h_j - h'_j)} G.$$

Therefore, we can compute the elliptic curve discrete logarithm of Q_j in base G with non-negligible probability, which contradicts the assumption that it is computationally infeasible to compute the elliptic discrete logarithm of Q_j in base G. Therefore, it is computationally infeasible for any adversary to forge a valid SAMA. □

5 AS Selection and Source Privacy

The appropriate selection of an AS plays a key role in message source privacy, since the actual message source node will be hidden in the AS. In this section, we will discuss techniques that can prevent the adversaries from tracking the message source through the AS analysis in combination with local traffic analysis.

Before a message is transmitted, the message source node selects an AS from the public key list in the SS as its choice. This set should include itself, together with some other nodes. When an adversary receives a message, he can possibly find the direction of the previous hop, or even the real

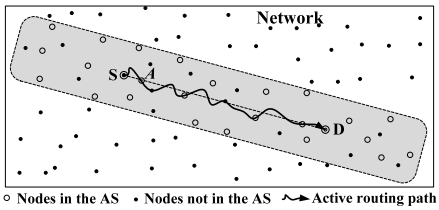


Fig. 1. Anonymous set selection in active routing.

node of the previous hop. However, the adversary is unable to distinguish whether the previous node is the actual source node or simply a forwarder node if the adversary is unable to monitor the traffic of the previous hop. Therefore, the selection of the AS should create sufficient diversity so that it is infeasible for the adversary to find the message source based on the selection of the AS itself.

Some basic criteria for the selection of the AS can be described as follows:

- To provide message source privacy, the message source needs to select the AS to include nodes from all directions of the source node. In particular, the AS should include nodes from the opposite direction of the successor node. In this way, even the immediate successor node will not be able to distinguish the message source node from the forwarder based on the message that it receives.
- Though the message source node can select any node in the AS, some nodes in the AS may not be able to add any ambiguity to the message source node. For instance, the nodes that are apparently impossible or very unlikely to be included in the AS based on the geographic routing. Therefore, these nodes are not appropriate candidates for the AS. They should be excluded from the AS for energy efficiency.
- To balance the source privacy and efficiency, we should try to select the nodes to be within a predefined distance range from the routing path. We recommend selecting an AS from the nodes in a band that covers the active routing path. However, the AS does not have to include all the nodes in the routing path.
- The AS does not have to include all nodes in that range, nor does it have to include all the nodes in the active routing path. In fact, if all nodes are included in the AS, then this may help the adversary to identity the possible routing path and find the source node.

As an example, suppose we want to transmit a packet from source node S to destination node D in Fig. 1. We select the AS to include only nodes marked with $\circ$, while nodes marked as $\bullet$ will not be included in the AS. Of all these $\circ$ nodes, some of them are on the active routing path, while others are not. However, all these nodes are located within the shaded band area surrounding the active routing path. Suppose node A is compromised, unless node A collaborates with other nodes and can fully monitor the traffic of the source node S, it will not be able to determine whether S is the source node, or simply a forwarder. Similar analysis is also true for other nodes.

Any node in the active routing path can verify the contents' authenticity and integrity. However, anybody who receives a packet in the transmission can possibly exclude some of the nodes in the WSNs as the possible source node. Inclusion of these nodes in the AS will not increase the source privacy. Nevertheless, the more the nodes included in the AS are, the higher the energy cost will be. Therefore, the selection of the AS has to be done with care so that the energy cost and the source privacy can both be optimized.

In addition, to balance the power consumption between authenticity and integrity verification, and the possibility that corrupted messages are being forwarded, the verification service may not have to take place in every hop; instead, it may be configured to take place in every other hop, for instance.

6 Key Management and Compromised Node Detection

In our scheme, we assume that there is an SS whose responsibilities include public-key storage and distribution in the WSNs. We assume that the SS will never be compromised. However, after deployment, the sensor node may be captured and compromised by the attackers. Once compromised, all information stored in the sensor node will be accessible to the attackers. We further assume that the compromised node will not be able to create new public keys that can be accepted by the SS.

For efficiency, each public key will have a short identity. The length of the identity is based on the scale of the WSNs.

6.1 Compromised Node Detection

As a special scenario, we assume that all sensor information will be delivered to a sink node, which can be co-located with the SS. As described in Section 5, when a message is received by the sink node, the message source is hidden in an AS. Since the SAMA scheme guarantees that the message integrity is untampered, when a bad or meaningless message is received by the sink node, the source node is viewed as compromised. If the compromised source node only transmits one message, it would be very difficult for the node to be identified without additional network traffic information. However, when a compromised node transmits more than one message, the sink node can narrow the possible compromised nodes down to a very small set.

As shown in Fig. 2, we use the circle to represent an AS. When only one message is transmitted, the sink node can only obtain the information that the source node will be in a set, say AS_1. When the compromised source node transmits two messages, the sink node will be able to narrow the source node down to the set with both vertical lines and horizontal lines. When the compromised source node transmits three messages, the source node will be further narrowed down to the shaded area. Therefore, if the sink node keeps tracking the compromised message, there is a high probability that the compromised node can be isolated.

If the compromised nodes repeatedly use the same AS, it makes traffic analysis of the compromised nodes feasible,

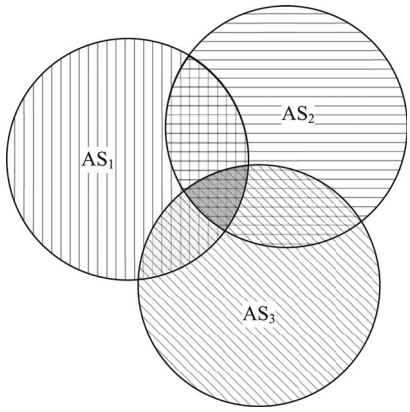


Fig. 2. Compromised node detection.

which will increase the likelihood for the compromised nodes to be identified and captured.

When a node has been identified as compromised, the SS can remove its public key from its public key list. It can also broadcast the node's short identity to the entire sensor domain so that any sensor node that uses the stored public key for an AS selection can update its key list. Once the public key of a node has been removed from the public key list, and/or broadcasted, any message with the AS containing the compromised node should be dropped without any process in order to save the precious sensor power.

7 Performance Analysis

In this section, we will evaluate our proposed authentication scheme through both theoretical analysis and simulation demonstrations. We will compare our proposed scheme with the bivariate polynomial-based symmetric-key scheme described in [3], [4]. *A fair comparison between our proposed scheme and the scheme proposed in [4] should be performed with* $n = 1$.

7.1 Theoretical Analysis

Key management is one of the major issues for secret-key based authentication schemes. This is especially true for large scale WSNs. While many of these schemes are designed to provide node authentication, they can only provide end-to-end node authentication using the secret key shared between the two nodes, which implies that only the receiver can verify the authenticity of the messages enroute. This means that no intermediate node can authenticate the message in general. The intermediate nodes may have to forward a manipulated message for many hops before the message can finally be authenticated and dropped by the receiving node. This not only consumes extra sensor power, but also increases the network collision and decreases the message delivery ratio. In addition to performance improvement, enabling intermediate node authentication will thwart adversaries from performing denial-of-service attacks through message manipulation to deplete the energy and communication resources of the wireless network. Therefore, developing a protocol that can provide hop-by-hop intermediate node authentication is an important research task.

Most of the authentication schemes are based on symmetric-key schemes, including the polynomial evaluation based threshold authentication scheme [4]. The secret bivariate polynomial is defined as [3]:

$$f(x,y) = \sum_{i=0}^{d_x} \sum_{j=0}^{d_y} A_{i,j} x^i y^j,$$

where each coefficient $A_{x,y}$ is an element of a finite field $\mathbb{F}_p$, and d_x and d_y are the degrees of this polynomial. d_x and d_y are also related to the message length and the computational complexity of this scheme. From the performance aspect, d_x and d_y should be as short as possible.

On the other hand, it is easy to see that when either more than $d_y + 1$ messages transmitted from the base station are received and recorded by the intruders, or more than $d_x + 1$ sensor nodes have been compromised, the intruders can recover the polynomial $f(x,y)$ via Lagrange interpolation. In this case, the security of the system is totally broken and the system cannot be used anymore. This property requires that both d_x and d_y be very large for the scheme to be resilient to node compromise attacks.

An alternative approach based on perturbation of the polynomial was also explored. The main idea is to add a small amount of random noise to the polynomial in the original scheme so that the adversaries will no longer be able to solve the coefficients using Lagrange interpolation. However, this technique is proved to be vulnerable to security attacks [6], since the random noise can be removed from the polynomial using error-correcting techniques.

While hop-by-hop authentication can be achieved through a public-key encryption system, the public-key based schemes were generally considered as not preferred, mainly due to their high computational overhead. However, our research demonstrates that it is not always true, especially for elliptic curve public-key cryptosystems.

In our scheme, each SAMA contains an AS of n randomly selected nodes that dynamically changes for each message. For $n = 1$, our scheme can provide at least the same security as the bivariate polynomial-based scheme. For $n > 1$, we can provide extra source privacy benefits. Even if one message is corrupted, other messages transmitted in the network can still be secure. Therefore, n can be much smaller than the parameters d_x and d_y. In fact, even a small n may provide adequate source privacy, while ensuring high system performance.

In addition, in the bivariate polynomial-based scheme, there is only one base station that can send messages. All the other nodes can only act as intermediate nodes or receivers. This property makes the base station easy to attack, and severely narrows the applicability of this scheme. In fact, the major traffic in WSNs is packet delivery from the sensor nodes to the sink node. In this case, our scheme enables every node to transmit the message to the sink node as a message initiator.

The recent progress on ECC has demonstrated that the public-key based schemes have more advantages in terms of memory usage, message complexity, and security resilience, since public-key based approaches have a simple and clean key management [9].

TABLE 1
Performance Comparison of the Bivariate Polynomial-Based Scheme in Two Different Scenarios: (a) The Original Implementation under 8 MHz Mica2 Platform, and (b) Our Implementation under 4 MHz Telosb

(a). Original implementation [4]							
$d_x, d_y = 3$				$d_x, d_y = 4$			
ROM (KB)	RAM (B)	Sign (ms)	Verf (ms)	ROM (KB)	RAM (B)	Sign (ms)	Verf (ms)
14.78	1938	5.8	57.89	15.04	2211	7.59	70.8
(b). Our implementation							
$d_x, d_y = 3$				$d_x, d_y = 4$			
ROM (KB)	RAM (B)	Sign (ms)	Verf (ms)	ROM (KB)	RAM (B)	Sign (ms)	Verf (ms)
13.61	1938	9	108	13.65	2302	11.73	126.93

7.2 Experimental Results

In this section, we implement the bivariate polynomial-based scheme and our proposed scheme in a real world comparison. The comparison is based on comparable security levels.

The implementation in [4] was carried out on Mica2 platform, which is 8 MHz, while our implementation is carried out on Telosb platform, which is 4 MHz. We first provide simulation in Table 1 to compare and justify our parameter selections. From the table, we can see that our results is comparable with the original paper. This justifies that the performance comparisons between our scheme and the algorithm proposed in [4] using different parameters are consistent and reasonable.

7.2.1 Simulation Parameter Setup

The bivariate polynomial-based scheme is a symmetric-key based implementation, while our scheme is based on ECC. This requires us to determine the comparable key sizes. If we choose the key size to be l for the symmetric-key cryptosystem, then the key size for our proposed ECC should be $2l$ according to [22], which is much shorter than the traditional public-key cryptosystem. This progress facilitates the implementation of the authentication scheme using ECC.

In our simulation setting, we choose five security levels, which are indicated by the symmetric-key sizes l: 24, 32, 40, 64, and 80 bits, respectively. The comparable key sizes of our scheme are 48, 64, 80, 128, and 160 bits, respectively.

We also need to determine d_x and d_y for the bivariate polynomial-based scheme, and the n for our scheme. In our simulation, we select d_x equal to d_y and choose three values for them: 80, 100, and 150. We assume that WSNs do not contain more than 2^{16} nodes in our simulation, which is reasonably large. For size n of the AS, we choose three values in the simulation: 10, 15 and 20.

We will compare the *computational overhead*, *communication overhead*, *delivery ratio*, *energy consumption*, *transmission delay*, and *memory consumption* of our proposed scheme with the bivariate polynomial-based scheme.

7.2.2 Computational Overhead

For a public-key based authentication scheme, computational overhead is one of the most important performance measurements. So we first performed simulation to measure the process time. The simulations were carried out in 16-bit, 4 MHz TelosB mote.

Table 2 shows the process time of our scheme and the bivariate polynomial-based scheme for both authentication generation and verification. In the simulations, we assume that the key length of our scheme is $2l$.

From the table, we have the following findings:

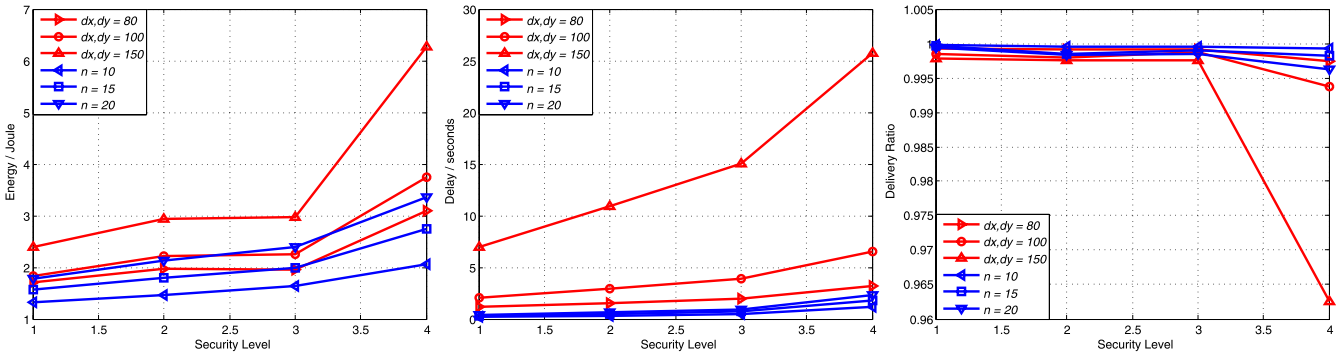
- For the bivariate polynomial-based scheme, the authentication generation time is much longer than the verifying time; while for our proposed scheme, the verifying time is about half of the authentication generation time, except when $n = 1$, the generation time is shorter than the verification time.
- Comparing bivariate polynomial-based scheme with our proposed scheme for $n = 1$, we find that the generation time of our scheme is less than 5 percent of the bivariate polynomial-based scheme for all d_x, d_y, but the verifying time is slightly longer when d_x, d_y is less than 100. When d_x, d_y is longer than 150, the verifying times of the two schemes are comparable.
- The memory consumption of our proposed scheme is slightly less than the bivariate polynomial-based scheme in all scenarios.
- For our proposed scheme, to provide source privacy, the cost of generation time and verifying time increase linearly with n.

7.2.3 Communication Overhead and Message Transmission Delay

The communication overhead is determined by the message length. For the bivariate polynomial-based scheme, each message is transmitted in the form of $<m, MAF_m(y)>$, where $MAF_m(y)$ is defined as: $MAF_m(y) = f(h(m), y) = \sum_{j=0}^{d_y} M_j y^j$. $MAF_m(y)$ is represented by its

TABLE 2
Process Time (s) for the Two Schemes (16-bit, 4 MHz TelosB Mote)

	Polynomial-based approach						Proposed approach							
	$d_x, d_y = 80$		$d_x, d_y = 100$		$d_x, d_y = 150$		$n = 1$		$n = 10$		$n = 15$		$n = 20$	
	Gen	Verify	Gen	Verify	Gen	Verify	Gen	Verify	Gen	Verify	Gen	Verify	Gen	Verify
l = 24	9.31	0.25	14.45	0.31	31.95	0.46	0.24	0.53	4.24	2.39	6.16	3.51	8.38	4.44
l = 32	12.95	0.33	20.05	0.41	44.60	0.62	0.34	0.80	5.99	3.32	8.92	5.05	12.19	6.42
l = 40	13.32	0.35	20.57	0.44	45.73	0.65	0.46	1.05	8.03	4.44	11.94	6.71	16.18	8.50
l = 64	21.75	0.57	33.64	0.71	74.85	1.06	1.18	1.77	20.53	11.03	30.12	16.41	41.44	21.10
l = 80	26.40	0.70	41.03	0.88	90.86	1.30	1.46	2.22	25.58	13.90	37.66	20.96	50.96	26.18



(a) Energy consumption (b) Message delay (c) Delivery ratio

Fig. 3. Performance comparison of our proposed scheme and bivariate polynomial-based scheme.

TABLE 3
Memory (KB) for the Two Schemes (TelosB) (F Stands for Flash Memory)

	Polynomial-based approach									Proposed approach											
	$d_x, d_y = 80$			$d_x, d_y = 100$			$d_x, d_y = 150$			$n = 1$			$n = 10$			$n = 15$			$n = 20$		
	ROM	RAM	F	ROM	RAM	F	ROM	RAM	F	ROM	RAM	F	ROM	RAM	F	ROM	RAM	F	ROM	RAM	F
$l = 24$	21	3	26	21	4	40	26	4	90	21	1	0	21	2	0	21	2	0	21	2	0
$l = 32$	21	4	39	21	5	60	26	6	135	21	2	0	21	2	0	21	2	0	21	2	0
$l = 40$	21	4	39	21	5	60	26	6	135	21	2	0	21	2	0	21	2	0	21	3	0
$l = 64$	21	6	64	21	7	100	26	9	225	21	2	0	22	3	0	22	3	0	22	3	0
$l = 80$	21	7	77	21	8	120	26	10	270	20	2	0	21	3	0	21	3	0	21	4	0

$d_y + 1$ coefficients $M_i \in \mathbb{Z}_p, 0 \leq i \leq d_y$, where $p \in (2^{l-1}, 2^l)$ is a large prime number. The total length of the message is $l(d_y + 1)$.

For our scheme, the message format is: $(m, \mathcal{S}, r_1, y_1, \ldots, r_n, y_n, s)$, where m, s, r_i, y_i are all numbers with length $L = 2l$. $\mathcal{S}$ is the ID list for all the nodes included in the AS. Assuming the network is composed of λ nodes in total, each ID will be of the length: $\lceil \log_2 \lambda \rceil$. When n nodes are included in the AS, the length of $\mathcal{S}$ is $n\lceil \log_2 \lambda \rceil$. Therefore, the total length of one message for our scheme is: $4l(n+1) + n\lceil \log_2 \lambda \rceil$.

The large communication overhead of the polynomial-based scheme will increase the energy consumption and message delay. The simulation results in Figs. 3a and 3b demonstrate that our proposed scheme has a much lower energy consumption and message transmission delay. These simulations were carried out in ns-2 on RedHat Linux system. The security levels 1, 2, 3, 4 correspond to symmetric key sizes 24, 32, 40, and 64 bits, and elliptic curves key size 48, 64, 80, and 128 bits, respectively.

We also conduct simulations to compare the delivery ratios using ns-2 on RedHat Linux system. The results show that our scheme is slightly better than the bivariate polynomial-based scheme in delivery ratio. The results are given in Fig. 3c.

Our simulation on memory consumption derived in TelosB, see Table 3, shows the overall memory consumption for bivariate polynomial-based scheme is at least five times larger than our proposed scheme.

8 Conclusion

In this paper, we first proposed a novel and efficient SAMA based on ECC. While ensuring message sender privacy, SAMA can be applied to any message to provide message content authenticity. To provide hop-by-hop message authentication without the weakness of the built-in threshold of the polynomial-based scheme, we then proposed a hop-by-hop message authentication scheme based on the SAMA. When applied to WSNs with fixed sink nodes, we also discussed possible techniques for compromised node identification. We compared our proposed scheme with the bivariate polynomial-based scheme through simulations using ns-2 and TelosB. Both theoretical and simulation results show that, in comparable scenarios, our proposed scheme is more efficient than the bivariate polynomial-based scheme in terms of computational overhead, energy consumption, delivery ratio, message delay, and memory consumption.

Acknowledgments

This research was supported in part by NSF grants CNS-0845812, CNS-1117831, CNS-1217206, ECCS-1232109.

References

[1] F. Ye, H. Lou, S. Lu, and L. Zhang, "Statistical En-Route Filtering of Injected False Data in Sensor Networks," *Proc. IEEE INFOCOM*, Mar. 2004.

[2] S. Zhu, S. Setia, S. Jajodia, and P. Ning, "An Interleaved Hop-By-Hop Authentication Scheme for Filtering False Data in Sensor Networks," *Proc. IEEE Symp. Security and Privacy*, 2004.

[3] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-Secure Key Distribution for Dynamic Conferences," *Proc. Advances in Cryptology (Crypto '92)*, pp. 471-486, Apr. 1992.

[4] W. Zhang, N. Subramanian, and G. Wang, "Lightweight and Compromise-Resilient Message Authentication in Sensor Networks," *Proc. IEEE INFOCOM*, Apr. 2008.

[5] A. Perrig, R. Canetti, J. Tygar, and D. Song, "Efficient Authentication and Signing of Multicast Streams over Lossy Channels," *Proc. IEEE Symp. Security and Privacy*, May 2000.
[6] M. Albrecht, C. Gentry, S. Halevi, and J. Katz, "Attacking Cryptographic Schemes Based on 'Perturbation Polynomials'," Report 2009/098, http://eprint.iacr.org/, 2009.
[7] R. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Comm. ACM*, vol. 21, no. 2, pp. 120-126, 1978.
[8] T.A. ElGamal, "A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *IEEE Trans. Information Theory*, vol. IT-31, no. 4, pp. 469-472, July 1985.
[9] H. Wang, S. Sheng, C. Tan, and Q. Li, "Comparing Symmetric-Key and Public-Key Based Security Schemes in Sensor Networks: A Case Study of User Access Control," *Proc. IEEE 28th Int'l Conf. Distributed Computing Systems (ICDCS)*, pp. 11-18, 2008.
[10] D. Pointcheval and J. Stern, "Security Proofs for Signature Schemes," *Proc. Advances in Cryptology (EUROCRYPT)*, pp. 387-398, 1996.
[11] D. Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms," *Comm. ACM*, vol. 24, no. 2, pp. 84-88, Feb. 1981.
[12] D. Chaum, "The Dinning Cryptographer Problem: Unconditional Sender and Recipient Untraceability," *J. Cryptology*, vol. 1, no. 1, pp. 65-75, 1988.
[13] A. Pfitzmann and M. Hansen, "Anonymity, Unlinkability, Unobservability, Pseudonymity, and Identity Management a Proposal for Terminology," http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.31.pdf, Feb. 2008.
[14] A. Pfitzmann and M. Waidner, "Networks without User Observability—Design Options.," *Proc. Advances in Cryptology (EUROCRYPT)*, vol. 219, pp. 245-253, 1985.
[15] M. Reiter and A. Rubin, "Crowds: Anonymity for Web Transaction," *ACM Trans. Information and System Security*, vol. 1, no. 1, pp. 66-92, 1998.
[16] M. Waidner, "Unconditional Sender and Recipient Untraceability in Spite of Active Attacks," *Proc. Advances in Cryptology (EUROCRYPT)*, pp. 302-319, 1989.
[17] D. Pointcheval and J. Stern, "Security Arguments for Digital Signatures and Blind Signatures," *J. Cryptology*, vol. 13, no. 3, pp. 361-396, 2000.
[18] L. Harn and Y. Xu, "Design of Generalized ElGamal Type Digital Signature Schemes Based on Discrete Logarithm," *Electronics Letters*, vol. 30, no. 24, pp. 2025-2026, 1994.
[19] K. Nyberg and R.A. Rueppel, "Message Recovery for Signature Schemes Based on the Discrete Logarithm Problem," *Proc. Advances in Cryptology (EUROCRYPT)*, vol. 950, pp. 182-193, 1995.
[20] R. Rivest, A. Shamir, and Y. Tauman, "How to Leak a Secret," *Proc. Advances in Cryptology (ASIACRYPT)*, 2001.
[21] M. Bellare and P. Rogaway, "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols," *Proc. ACM First Conf. Computer and Comm. Security (CCS '93)*, pp. 62-73, 1993.
[22] "Cryptographic Key Length Recommendation," http://www.keylength.com/en/3/, 2013.

Jian Li received the BE and MA degrees in electrical engineering both from Tsinghua University, Beijing, China, in 2005 and 2008, respectively. He is currently working toward the PhD degree in electrical and computer engineering at Michigan State University, East Lansing. His research interests include cyber security, network coding and distributed storage.

Yun Li received the BE degree from Xidian University, Xi'an, China, in 2005, and the PhD degree in electrical and computer engineering from Michigan State University, East Lansing, in May 2010. He joined the Network Operating System Technology Group of Cisco System in 2010. He also joined the Server & Tools Division of Microsoft in 2012. His current research interests include wireless sensor networks, network security and cloud based data services.

Jian Ren received the BS and MS degrees both in mathematics from Shaanxi Normal University, China, and the PhD degree in electrical engineering from Xidian University, China. He is an associate professor in the Department of Electrical Communication Engineering at Michigan State University, East Lansing. His current research interests include cryptography, network security, energy efficient sensor network security protocol design, privacy-preserving communications, and cognitive networks. He received the US National Science Foundation Faculty Early Career Development (CAREER) Award in 2009. He is a senior member of the IEEE.

Jie Wu is the chair and a professor in the Department of Computer and Information Sciences, Temple University, Philadelphia, Pennsylvania. Prior to joining Temple University, he was a program director at National Science Foundation. His research interests include wireless networks and mobile computing, routing protocols, fault-tolerant computing, and interconnection networks. He serves in the editorial board of the *IEEE Transactions on Computers* and *Journal of Parallel and Distributed Computing*. He is a program cochair for IEEE INFOCOM 2011. He was also the general cochair for IEEE MASS 2006, IEEE IPDPS 2008, and DCOSS 2009. He currently serves as an ACM distinguished speaker and is the chairman of the IEEE Technical Committee on Distributed Processing. He is a fellow of the IEEE.

www.redpel.com
+917620593389