Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2010, Article ID 534712, 10 pages
doi:10.1155/2010/534712

Research Article

SPM: Source Privacy for Mobile Ad Hoc Networks

Jian Ren, Yun Li, and Tongtong Li

Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48864-1226, USA

Correspondence should be addressed to Jian Ren, renjian@egr.msu.edu

Received 2 September 2009; Accepted 21 September 2009

Academic Editor: Benyuan Liu

Source privacy plays a key role in communication infrastructure protection. It is a critical security requirement for many mission critical communications. This is especially true for mobile ad hoc networks (MANETs) due to node mobility and lack of physical protection. Existing cryptosystem-based techniques and broadcasting-based techniques cannot be easily adapted to MANET because of their extensive cryptographic computation and/or large communication overhead. In this paper, we first propose a novel unconditionally secure source anonymous message authentication scheme (SAMAS). This scheme enables message sender to transmit messages without relying on any trusted third parties. While providing source privacy, the proposed scheme can also provide message content authenticity. We then propose a novel communication protocol for MANET that can ensure communication privacy for both message sender and message recipient. This protocol can also protect end-to-end routing privacy. Our security analysis demonstrates that the proposed protocol is secure against various attacks. The theoretical analysis and simulation show that the proposed scheme is efficient and can provide high message delivery ratio. The proposed protocol can be used for critical infrastructure protection and secure file sharing in mobile ad hoc networks where dynamic groups can be formed.

1. Introduction

The decentralized nature of mobile ad hoc networks (MANETs) makes rapid deployment of independent mobile users practical. MANETs are suitable for many applications, such as establishing survivable, dynamic communication for emergency/rescue operations, disaster relief efforts, and military networks. MANETs consist of autonomous collection of mobile users that communicate over bandwidth constrained wireless links. All these issues make security, jamming protection, and even node capture significant concerns. Without privacy protection, adversaries can easily learn the identities of the communication parties and the relevant information that two users are communicating. For example, the adversaries can track your on-line orders, the web sites that you access, the doctors that you visit and many more. Adversaries can also easily overhear all the messages, passively eavesdrop on communications and perform traffic analysis, routing monitoring, and denial-of-service (DoS) attacks.

For a tactical military communication networks, communication privacy is becoming an essential security requirement. As an example, an abrupt change in traffic pattern or volume may indicate some forthcoming activities. The exposure of such information could be extremely dangerous in that adversaries can easily identify critical network nodes and then launch direct DoS attacks on them. Communication privacy is also an indispensable security requirement for applications such as e-voting, e-cash and so on.

In the past two decades, originated largely from Chaum's mixnet [1] and DC-net [2], a number of privacy-preserving communication protocols have been proposed, including for example, onion routing [3], K-anonymous message transmission [4], Web MIXes [5], Mixminion [6], Mixing email [7], Mixmaster Protocol [8], Crowds [9], and Buses seat allocation [10], to name a few. The mixnet family protocols use a set of "mix" servers that mix the received packets to make the communication parties (including the sender and the recipient) ambiguous. They rely on the statistical properties of background traffic that is also referred to as *cover traffic* to achieve the desired source privacy. The DC-net family protocols [2, 4, 11, 12] on the other hand, utilize secure multiparty computation techniques. They provide provable source privacy without relying on

trusted third parties. However, to broadcast a message, each party of the group that the message sender hides, called the *ambiguity set (AS)*, needs to choose a random position. Even if all parties are honest, there are no effective noninteractive means that can enable players to select distinct message positions. This means that multiple parties may transmit messages in the same slot. This is called the *transmission collision problem*. There is no existing practical solution to solve this problem [12].

As the computing, communicating, and cryptographic techniques progress rapidly, increasing emphasis has been placed on developing new efficient and secure anonymous communication schemes and network protocols without relying on trusted third parties and free of collision.

In this paper, we first propose a novel unconditionally secure source anonymous message authentication scheme (SAMAS). While providing source privacy, the proposed scheme can also provide message content authenticity without relying on any trusted third parties. We then propose a novel communication protocol for MANET that can ensure communication privacy for both communication parties and their end-to-end communications. The proposed network protocol can be used for critical information distribution, infrastructure protection, and secure file sharing. Our security analysis demonstrates that the proposed protocol is secure against various attacks. The theoretical analysis and simulation results show that the proposed scheme is efficient and can ensure high message delivery ratio.

The rest of this paper is organized as follows. In Section 2, the terminology, assumptions, and previous related works are briefly reviewed. The proposed unconditionally secure source anonymous message authentication scheme (SAMAS) and security analysis are described in Section 3. In Section 4, we propose an anonymous communication protocol in detail along with security analysis and simulation results in Section 5. Finally, Section 6 concludes this paper.

2. Terminology and Preliminary

In this section, we will briefly describe the terminology that will be used in this paper. Then we will introduce some cryptographic tools that will be used in this paper. Finally, we will present a brief overview of the related works in this area.

2.1. Terminology. Privacy is sometimes referred to as *anonymity*. Communication anonymity in information management has been discussed in a number of previous works [1, 2, 9, 13–15]. It generally refers to the state of being not identifiable within a set of subjects. This set is called the *ambiguity se*t (AS). Three types of anonymity were defined [13]: *sender anonymity*, *recipient anonymity,* and *relationship anonymity*. *Sender anonymity* means that a particular message is not linkable to any sender and no message if linkable to a particular sender. *Recipient anonymity* similarly means that a message cannot be linked to any recipient and that no message is linkable to a recipient. *Relationship anonymity* means that the sender and the recipient are unlinkable. In other words, sender and recipient cannot be identified as communicating with each other, though it may be clear they are participating in some communications. Relationship anonymity is a weaker property than that of sender anonymity and recipient anonymity. The above anonymities are also referred to as the *full anonymities*, since they guarantee that an adversary cannot infer anything about the sender, the recipient, or the communication relationship from a transmitted message.

Table 1: Notation.

AS or $\mathcal{S}$	Ambiguity set
SAMAS	source anonymous message authentication code
MAC_i	ith message authentication code
MES	modified ElGamal signature
PPT	probabilistic polynomial time
p	a large prime number
$\mathbb{Z}_p$	integer field module p
g	a primitive element in $\mathbb{Z}_p$
x_i	private key of the ith user
y_i	public key of the ith user for SAMAC generation
r_i	ith signature component
s	ElGamal signature component
m	Message
n	number of users in the AS
A_i	ith member in the AS
h	hash function
h_i	hash value $h(m, r_i)$
$pk_i(m)$	encrypt m using public key y_i
sk_i	A_i's secret key for symmetric encryption
$sk_i(m)$	encrypt m using secret key sk_i
N_i	ith nonce
mF_i	ith message flag
rF_i	ith recipient flag
$\|$	message concatenation
$\mathcal{M}(i, j)$	message to send from node i to node j
$\mathcal{S}(m)$	SAMAC of message m

We will start with the definition of unconditionally secure source anonymous message authentication scheme (SAMAS).

Definition 1 (SAMAS). An SAMAS consists of the following two algorithms:

(i) *generate* $(m, y_1, y_2, \ldots, y_n)$: Given a message m and the public keys $y_1, y_2, \ldots, y_n$ of the ambiguity set *(AS)* $\mathcal{S} = \{A_1, A_2, \ldots, A_n\}$, the actual message sender $A_t, 1 \le t \le n$, produces an anonymous message $\mathcal{S}(m)$ using her own private key x_t;

(ii) *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 SAMAS include

(i) *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 AS;

(ii) *Unforgeability:* An anonymous message scheme is unforgeable if no adversary, given the public keys of all members of the AS and the anonymous messages $m_1, m_2, \ldots, m_l$ adaptively chosen by the adversary, can produce in polynomial time a new valid anonymous message with nonnegligible probability.

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

2.2. Modified ElGamal Signature Scheme (MES).

Definition 2 (MES). The modified ElGamal signature scheme [16] consists of the following three algorithms:

(i) *Key generation algorithm:* Let p be a large prime, 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 $y = g^x \bmod p$;

(ii) *Signature algorithm:* The MES can also have many variants [17, 18]. 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);

(iii) *Verification algorithm:* The verifier checks 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.

2.3. Previous Work.

The existing anonymous communication protocols are largely stemmed from either mixnet [1] or DC-net [2]. A mixnet provides anonymity via packet reshuffling through (at least one trusted) "mix". 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. An eavesdropper cannot link a decrypted output message with any particular (encrypted) input message. The mixnet thus protects the secrecy of users' communication relationships. Recently, Möler presented a secure public-key encryption algorithm for mixnet [19]. This algorithm has been adopted by Mixminion [6]. However, since mixnet-like protocols rely on the statistical properties of background traffic, they cannot provide provable anonymity.

DC-net [2, 15] is an anonymous multiparty computation amongst a set of participants, some pairs of which share secret keys. DC-net provides perfect (information theoretic) sender anonymity without requiring trusted servers. In a DC-net, users send encrypted broadcasts to the entire group, thus achieving receiver anonymity. However, all members of the group are made aware of when a message is sent, so DC-net does not have the same level of sender-receiver anonymity. Also, in DC-net, only one user can send at a time, so it takes additional bandwidth to handle collisions and contention. Lastly, a DC-net participant fixes its anonymity versus bandwidth trade off when joining the system, and there are no provisions to rescale that trade off when others join the system.

Crowds [9] extends the idea of anonymizer and is designed for anonymous web browsing. However, Crowds only provides sender anonymity. It does not hide the receivers and the packet content from the nodes en route. Hordes [20] builds on the Crowds. It uses multicast services and provides only sender anonymity.

Recently, message sender anonymity based on ring signatures was introduced [21]. This approach can enable message sender to generate source anonymous message signature with content authenticity assurance, while hiding the real identity of the message sender. The major idea is that the message sender (say Alice) randomly selects n of ring members as the AS on her own without awareness of these members. To generate a ring signature, for each member in the ring other than the actual sender (Alice), Alice randomly selects an input and computes the one-way output using message signature forgery. For the trapdoor one-way function corresponding to the actual sender Alice, she needs to solve the "message" that can "glue" the ring together, and then signs this "message" using her knowledge of the trap-door information. The original scheme has very limited flexibility and the complexity of the scheme is quite high. Moreover, the original paper only focuses on the cryptographic algorithm, the relevant network issues were totally left unaddressed.

In this paper, we first propose an unconditionally secure and efficient source anonymous message authentication scheme based on the modified ElGamal signature scheme. This is because the original ElGamal signature scheme is existentially forgeable with a generic message attack [22, 23]. While the modified ElGamal signature (MES) scheme is secure against no-message attack and adaptive chosen-message attack in the random oracle model [24].

2.4. Threat Model and Assumptions.

We assume the participating MANET nodes voluntarily cooperate with each other to provide the service. All nodes are potential message originators of anonymous communications. The adversaries can collaborate to passively monitor and eavesdrop every MANET traffic. In addition, they may compromise any node in the target network to become an internal adversary, which could be the internal perpetrators. In this paper, we assume that passive adversaries can only compromise a fraction of the nodes. We also assume that the adversaries are

computationally bounded so that inverting and reading of encrypted messages are infeasible. Otherwise, it is believed that there is no workable cryptographic solution.

An agent of the adversary at a compromised node observes and collects all the information in the message, and thus reports the immediate predecessor and successor node for each message traversing the compromised node. Assume also that the adversary collects this information from all the compromised nodes, and uses it to derive the identity of the sender of a message. The sender has no information about the number or identity of nodes being compromised. The adversary collects all the information from the agents on the compromised nodes, and attempts to derive the true identity of the sender.

3. Unconditionally Secure Source Anonymous Message Authentication Scheme (SAMAS)

In this section, we propose an unconditionally secure and efficient source anonymous message authentication scheme (SAMAS). The main idea is that for each message m to be released, the message sender, or the sending node, generates a source anonymous message authentication for the message m. The generation is based on the MES scheme. Unlike ring signatures, which requires to compute a forgery signature for each member in the AS separately, our scheme only requires three steps to generate the entire SAMAS, and link all nonsenders and the message sender to the SAMAS alike. In addition, our design enables the SAMAS to be verified through a single equation without individually verifying the signatures.

3.1. The Proposed SAMAS Scheme. Suppose that the message sender (say Alice) wishes to transmit a message m anonymously from her network node to any other node. 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, $1 \le t \le n$.

Let p be a large prime number and g be a primitive element of $\mathbb{Z}_p^*$. Then g is also a generator of $\mathbb{Z}_p^*$. That is $\mathbb{Z}_p^* = \langle g \rangle$. Both p and g are made public and shared by all members in $\mathcal{S}$. Each $A_i \in \mathcal{S}$ has a public key $y_i = g^{x_i} \bmod p$, where x_i is a randomly selected private key from $\mathbb{Z}_{p-1}^*$. In this paper, we will not distinguish between the node A_i and its public key y_i. Therefore, we also have $\mathcal{S} = \{y_1, y_2, \ldots, y_n\}$.

Suppose m is a message to be transmitted. The private key of the message sender Alice is x_t, $1 \le t \le n$. To generate an efficient SAMAS for message m, Alice performs the following three steps:

(1) Select a random and pairwise different k_i for each $1 \le i \le n, i \ne t$ and compute $r_i = g^{k_i} \bmod p$;

(2) Choose a random $k \in \mathbb{Z}_p$ and compute $r_t = g^k \prod_{i \ne t} y_i^{-r_i h_i} \bmod p$ such that $r_t \ne 1$ and $r_t \ne r_i$ for any $i \ne t$, where $h_i = h(m, r_i)$;

(3) Compute $s = k + \sum_{i \ne t} k_i + x_t r_t h_t \bmod (p-1)$.

The SAMAS of the message m is defined as

$$\mathcal{S}(m) = (m, \mathcal{S}, r_1, \ldots, r_n, s), \tag{2}$$

where $g^s = r_1 \cdots r_n y_1^{r_1 h_1} \cdots y_n^{r_n h_n} \bmod p$, and $h_i = h(m, r_i)$.

3.2. Verification of SAMAS. A verifier can verify an alleged SAMAS $(m, \mathcal{S}, r_1, \ldots, r_n, s)$ for message m by verifying whether the following equation

$$g^s = r_1 \cdots r_n y_1^{r_1 h_1} \cdots y_n^{r_n h_n} \bmod p \tag{3}$$

holds. If (3) holds true, the verifier Accepts the SAMAS as valid for message m. Otherwise the verifier Rejects the SAMAS.

In fact, if the SAMAS has been correctly generated, then we have

$$\begin{aligned} & r_1 \cdots r_n y_1^{r_1 h_1} \cdots y_n^{r_n h_n} \bmod p \\ &= g^{k_1} \cdots g^{k_n} y_1^{r_1 h_1} \cdots y_n^{r_n h_n} \bmod p \\ &= g^{\sum_{i \ne t} k_i} \left(g^k \prod_{i \ne t} y_i^{-r_i h_i} \right) \left(\prod_{i \ne t} y_i^{r_i h_i} \right) y_t^{r_t h_t} \bmod p \\ &= g^{k + \sum_{i \ne t} k_i + x_t r_t h_t} \bmod p \\ &= g^s \bmod p. \end{aligned} \tag{4}$$

Therefore, the verifier should always Accept the SAMAS if it is correctly generated without being modified.

Remark 1. As a trade-off between computation and transmission, the SAMAS can also be defined as $\mathcal{S}(m) = (m, \mathcal{S}, r_1, \ldots, r_n, h_1, \ldots, h_n, s)$. In case $\mathcal{S}$ is also clear, it can be eliminated from the SAMAS.

3.3. Security Analysis. In this subsection, we will prove that the proposed SAMAS scheme is unconditionally anonymous and provably unforgeable against adaptive chosen-message attack.

3.3.1. Anonymity. In order to prove that the proposed SAMAS is unconditionally anonymous, we have to prove that (i) for anybody other than the members of $\mathcal{S}$, the probability to successfully identify the real sender is $1/n$, and (ii) anybody from $\mathcal{S}$ can generate SAMAS.

Theorem 1. *The proposed source anonymous message authentication scheme (SAMAS) can provide unconditional message sender anonymity.*

Proof. The identity of the message sender is unconditionally protected with the proposed SAMAC scheme. This is because that regardless of the sender's identity, there are exactly $(p-1)(p-2)\cdots(p-n)$ different options to generate the SAMAC, and all of them can be chosen by the SAMAC generation procedure and by any of the members in the AS with

equal probability without depending on any complexity-theoretic assumptions. The proof for the second part, that anybody from $\mathcal{S}$ can generate the SAMAC is straightforward. This finishes the proof of this theorem. □

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

In this section, we will prove that the proposed SAMAS is secure against existential forgery under adaptive-chosen message attacks in the random oracle model [25]. The security of our result is based on the well-known discrete logarithms problem (DLP), which assumes that the computation of discrete logarithm in $\mathbb{Z}_p$ for large p is computationally infeasible. In other words, no efficient algorithms are known for non-quantum computers.

We will introduce two lemmas first. Lemma 2, or the Splitting Lemma, is a well-known probabilistic lemma from reference [24]. 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 3 is a slight modification of the Forking Lemma presented in [24]. The proof of this theorem is mainly probability theory related. We will skip the proof of these two lemmas here.

Lemma 2 (The Splitting Lemma). *Let $Z \subset X \times Y$ such that $Pr[(x, y) \in Z] \geq \varepsilon$. For any $\alpha < \varepsilon$, define $W = \{(x, y) \in X \times Y \mid Pr_{y' \in Y}[(x, y') \in Z] \geq \varepsilon - \alpha\}$, and $\overline{W} = (X \times Y) \setminus W$, 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) $Pr[WZ] \geq \alpha/\varepsilon$.

Lemma 3 (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 nonnegligible probability, a valid SAMAS $(m, \mathcal{S}, r_1, \ldots, r_n, h_1, \ldots, h_n, s)$ within a bounded polynomial time T, then with nonnegligible probability, a replay of this machine which has control over $\mathcal{A}$ and a different oracle, outputs another valid SAMAS $(m, \mathcal{S}, r_1, \ldots, r_n, h'_1, \ldots, h'_n, s)$, such that $h_i = h'_i$, for all $1 \leq i \leq n, i \neq j$ for some fixed j.*

Theorem 4. *The proposed SAMAS is secure against adaptive chosen-message attack in the random oracle model.*

Proof. (Sketch) If an adversary can forgery a valid SAMAS with nonnegligible probability, then according to the Forking Lemma, the adversary can get two valid SAMACs

$$\begin{aligned} &(m, \mathcal{S}, r_1, \ldots, r_n, h_1, \ldots, h_n, s), \\ &(m, \mathcal{S}, r_1, \ldots, r_n, h'_1, \ldots, h'_n, s'), \end{aligned} \tag{5}$$

such that for $1 \leq i \leq n, i \neq j, h_i = h'_i$, and $h_j \neq h'_j$. That is

$$g^s = r_1 \cdots r_n y_1^{r_1 h_1} \cdots y_n^{r_n h_n} \bmod p, \tag{6}$$

$$g^{s'} = r_1 \cdots r_n y_1^{r_1 h'_1} \cdots y_n^{r_n h'_n} \bmod p. \tag{7}$$

Divide equations (6) and (7), we obtain

$$g^{s-s'} = y_t^{r_t(h_t - h'_t)} \bmod p. \tag{8}$$

Equivalently, we have

$$y_t = g^{s-s'/r_t(h_t - h'_t)} \bmod p. \tag{9}$$

Therefore, we can compute the discrete logarithm of y_t in base g with nonnegligible probability, which contradicts to the assumption that it is computationally infeasible to compute the discrete logarithm of y_j in base g. Therefore, it is computationally infeasible for any adversary to forge a valid SAMAC. □

4. The Proposed Privacy-Preserving Communication Protocol

4.1. Network Model. Keeping confidential who sends which messages, in a world where any physical transmission can be monitored and traced to its origin, seems impossible. To solve this problem, in this paper, we consider networks with multiple MANETs. That is, the participating nodes are divided into a set of small subgroups. We classify the network nodes into two categories, *normal nodes* and *super nodes*. A normal node is a network node that may not be able to communicate direct with the nodes in other MANETs. A super node can be a normal node that can also provide message forward services to other MANET nodes. It can also be a special node dedicated to providing message forward services to other MANET nodes. For energy optimization, the normal nodes can take turn to be the super nodes (Figure 1).

Prior to network deployment, there should be an administrator. The administrator is responsible for selection of security parameters and a group-wise master key $s_G \in \mathbb{Z}_p^*$. The group master key should be well safeguarded from unauthorized access and never be disclosed to the ordinary group members. The administrator then chooses a collision-resistant cryptographic hash function h, mapping arbitrary inputs to fixed-length outputs on $\mathbb{Z}_p$, for example, SHA-1 [26].

The administrator assigns each super node a sufficiently large set of collision-free pseudonyms that can be used to substitute the real IDs in communications to defend against passive attacks. If a super node uses one pseudonym continuously for some time, it will not help to defend against possible attacks since the pseudonym can be analyzed in the same way as its real ID. To solve this problem, each node should use dynamic pseudonyms instead. This requires each super node to sign up with the administrator, who will assign

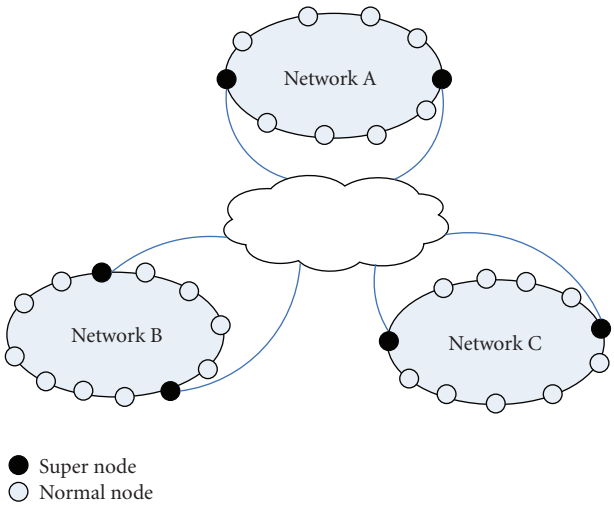


Figure 1: An illustration of network topology of the proposed scheme.

each super node a list of random and collision-resistant pseudonyms:

$$\mathcal{N}_A = \left\{\mathrm{id}_1^A, \ldots, \mathrm{id}_\tau^A\right\}. \tag{10}$$

In addition, each super node will also be assigned a corresponding *secret set*: $\mathcal{S}_A = \{g^{s_G h(\mathrm{id}_1^A)}, \ldots, g^{s_G h(\mathrm{id}_\tau^A)}\}$.

4.2. Anonymous Local MANET Communication. To realize anonymous network layer communications, obviously there should be no explicit information (such as the message sender and recipient addresses) in the message content. All of the information related to addresses, including the destination MANET where the recipient resides, should be embedded into the anonymizing message payload.

Prior to network deployment, the administrator needs to select a set of security parameters for the entire system, including a large prime p and a generator g of $\mathbb{Z}_p^*$. The network nodes $A_1, A_2, \ldots, A_n$ and the corresponding public keys $y_1, y_2, \ldots, y_n$ of the n participating network nodes, where $x_i \in \mathbb{Z}_p$, is a randomly selected private key of node A_i, and y_i is computed from $y_i = g^{x_i} \bmod p$.

A normal node only communicates to other nodes in the same MANET. The communication between two normal nodes in different MANETs has to be forwarded through the supper nodes in the respected local MANETs. Each message contains a nonce (N), a message flag (mF), a recipient flag (rF), and a secret key. The nonce is a random number that is used only once to prevent message replay attack. The recipient flag enables the recipient to know whether he is the targeted receiver or a forwarding node. The secret key is used to encrypt the message payload through symmetric encryption algorithm.

More specifically, for a node A_i to transmit a message m anonymously to a node A_j in the same MANET, through the nodes $A_{i+1}, \ldots, A_{j-1}$, where $j > i+1$, node A_i generates a new message $\mathcal{M}(i, j)$ defined in (11),

$$\begin{aligned}
&\mathcal{M}(i,j)\\
&\quad = pk_{i+1}(N_{i+1}, mF_{i+1}, rF_{i+1}, sk_{i+1}) \| sk_{i+1}(\mathcal{M}(i+1,j))\\
&\mathcal{M}(i+1,j)\\
&\quad = pk_{i+2}(N_{i+2}, mF_{i+2}, rF_{i+2}, sk_{i+2}) \| sk_{i+2}(\mathcal{M}(i+2,j))\\
&\qquad\qquad \vdots\\
&\mathcal{M}(j-1,j)\\
&\quad = pk_j\left(N_j, mF_j, rF_j, sk_j\right) \| sk_j(\mathcal{S}(m)),
\end{aligned} \tag{11}$$

where for $l = i+1, \ldots, j$, N_l is a nonce, mF_l is a message flag, rF_l is a recipient flag, sk_l is the secret key used for one time message encryption, and $\|$ stands for message concatenation.

When the node A_{i+1} receives the message packet, the node decrypts the first block of the received message using its private key corresponding to y_{i+1}. After that, the node will get the recipient flag and message flag with the instruction for the subsequent actions.

When a message reaches the targeted recipient, to ensure traffic balance, the node will generate a dummy message to its subsequent nodes. Only the super nodes can terminate or initiate a dummy message. In this way, the amount of traffic flow that a node creates as the initiator is concealed in the traffic that it forwards since the overall traffic that it receives is the same as the traffic that it forwards. In addition, the message is encrypted with the private key that only the recipient can recover. While the intermediate nodes can only view the instruction of the message allowed. The sender's message is indistinguishable by other nodes. The sender and the recipient are thus hidden amongst the other nodes. It is infeasible for the adversary to correlate messages using traffic analysis and timing analysis due to message encryption. Therefore, perfect obscure of its own messages can be assured. Detailed security analysis will be presented later.

Remark 2. When the message is delivered to the recipient's local MANET, if the super node is close enough to the recipient node, then the super node can simply broadcast the message. In this case, the message format in (11) can be adjusted accordingly.

4.3. Dynamic Local MANET Formation. Due to node mobility in the MANET, the local MANET will dynamically change over time. This makes reforming of the local MANET an essential part of our proposed scheme. The dynamic updating of the MANET can be characterized through mobility of each individual node, that can leave and join a local MANET.

4.3.1. Process for a Node to Join a Local MANET. When a node, say node A_j, wishes to join a local MANET, it needs

to send a request message to the local super node in the form of:

$$\texttt{Join\_Request}\ \left\|\mathrm{y_j}\right\|\mathrm{T}, \tag{12}$$

where y_j is the public key of node A_j, and T is a timestamp. After receiving this request message, the super node has to determine the relative location of this node according to the direction and strength of the request signal provided by nodes that also received this message. The super node will determine where the node should be located in the local MANET logically. Then the super node will broadcast a message in the following format to inform the local MANET that node y_j will be joining the local MANET in between node y_i and node y_{i+1}:

$$\texttt{Join\_Confirm}\ \left\|y_j\right\|y_i\|y_{i+1}\|T, \tag{13}$$

where T is a timestamp.

4.3.2. Process for a Node to Leave a Local MANET. A node can leave a local MANET either *positively* or *passively*. For positive leaving, the node, say node A_j, is aware that it is leaving the local MANET. It will send a request message to the local super node in the format of:

$$\texttt{Leaving\_Request}\ \left\|y_j\right\|T, \tag{14}$$

where y_j is the public key of node A_j, and T is a timestamp. For passive leaving, the node will just leave the local MANET without informing anyone. The super node will discover a node's leaving through message transmission failure and *Hello* message detection.

When a super node is aware of a node's leaving through either of the two manners, it will inform all of the local MANET members through broadcasting a message:

$$\texttt{Leaving\_Confirm}\ \left\|y_j\right\|T, \tag{15}$$

which means a node with public key y_j has left the local MANET, and it should be removed from the local MANET.

4.4. Anonymous Communications between Two Arbitrary Super Nodes. In the previous subsections, we present the mechanism that allows two arbitrary nodes to communicate anonymously within the same MANET. This includes communications between two super nodes in the same MANET. For any two arbitrary super nodes in different MANETs to communicate anonymously, we will first introduce the concept of anonymous authentication or secret handshake by Balfanz et al. [27]. Anonymous authentication allows two nodes in the same group to authenticate each other *secretly* in the sense that each party reveals its group membership to the other party only if the other party is also a group member. Nonmembers are not able to recognize group members.

The scheme consists of a set of super nodes and an administrator who creates groups and enrolls super nodes in groups. For this purpose, the administrator will assign each super node A a set of pseudonyms $\mathrm{id}_1^A, \ldots, \mathrm{id}_\tau^A$, where τ is a large security parameter. In addition, the administrator also calculates a corresponding *secret set* $\{g^{s_G h(\mathrm{id}_1^A)} \bmod p, \ldots, g^{s_G h(\mathrm{id}_\tau^A)} \bmod p\}$ for super node A, where s_G is the group's secret and h is a hash function. The pseudonyms will be dynamically selected and used to substitute the real IDs for each communication. This means that two super nodes A and B can know each other's group membership only if they belong to the same group.

When the super node A wants to authenticate to the super node B, the following secret handshake can be conducted:

(1) $A \rightarrow B$: Super node A randomly selects an unused pseudonym id_i^A and a random nonce N_1, then sends id_i^A, N_1 to super node B;

(2) $B \rightarrow A$: Super node B randomly selects an unused pseudonym id_i^B and a random nonce N_2, then sends $\mathrm{id}_j^B, N_2, V_0 = h(K_{BA}\|\mathrm{id}_i^A\|\mathrm{id}_j^B\|N_1\|N_2\|0)$ to super node A, where $K_{BA} = g^{s_G h(\mathrm{id}_i^A)\cdot h(\mathrm{id}_j^B)} \bmod p$;

(3) $A \rightarrow B$: Super node A sends $V_1 = h(K_{AB}\|\mathrm{id}_i^A\|\mathrm{id}_j^B\|N_1\|N_2\|1)$ to super node B, where $K_{AB} = g^{s_G h(\mathrm{id}_j^B)\cdot h(\mathrm{id}_i^A)} \bmod p$.

Since $K_{BA} = K_{AB}$, A can verify V_0 by checking whether $V_0 \stackrel{?}{=} h(K_{AB}\|\mathrm{id}_i^A\|\mathrm{id}_j^B\|N_1\|N_2\|0)$. If the verification succeeds, then A knows that B is an authentic group peer. Similarly, B can verify A by checking whether $V_1 \stackrel{?}{=} h(K_{BA}\|\mathrm{id}_i^A\|\mathrm{id}_j^B\|N_1\|N_2\|1)$. If the verification succeeds, then B knows that A is also an authentic group peer. However, in this authentication process, neither super node A, nor super node B can get the real identity of the other node. In other words, the real identities of super node A and super node B remain anonymous after the authentication process.

4.5. Anonymous Communication between Two Arbitrary Normal Nodes. As mentioned before, there should be no explicit exposure about the addresses of the message sender and recipient. To transmit a message, the sender first randomly selects a local super node and transmits the message to the super node according to the mechanism described before. On receiving the message, the local super node first determines the destination MANET ID by checking the message recipient flag rF, either 0 or 1. If it is 0, then the recipient and the super node are in the same MANET. The message can be forwarded in the recipient node using the previously described mechanism. If rF is 1, then the recipient is in a different MANET, The super node forwards the message to a super node in the destination MANET as described in the previous subsection. Finally, when the super node in the recipient's local MANET receives the message, the communication again becomes local MANET communications. The message can now be transmitted in the same way that the sender and the recipient are in the same MANET.

While providing message recipient anonymity, the message can also be encrypted so that only the message recipient can decrypt the message. The proposed anonymous

communication is quite general and can be used in a variety of situations for communication anonymity in MANET, including anonymous file sharing.

4.6. Security Analysis. In this subsection, we will analyze anonymity, impersonation attack, and replay attack of the proposed anonymous communication protocol.

4.6.1. Anonymity. We will first prove that the proposed communication protocol can provide both message sender and recipient anonymity in the local MANET communications.

Theorem 5. *It is computationally infeasible for an adversary to identify the message sender and recipient in the local MANET. Therefore, the proposed anonymous communication protocol provides both sender and recipient anonymity in the local MANET.*

Proof. (Sketch) First, since the number of message packages that each node receives from its immediate predecessor is the same as the number of packets that it forwards to its immediate successor, so the adversaries cannot determine the message source based on the traffic volume or the number of message packets. Second, since the message packages are encrypted using either the public keys or the shared secret keys of the intermediate nodes. No adversary is able to distinguish the real meaningful message from the dummy message in the transmission in any of the network nodes due to the traffic balance property and message content encryption. Therefore, the adversary cannot distinguish the initiator traffic from the indirection traffic and learn whether the node is a recipient, a receiver, or simply a node that provides message forward service. Consequently, both the message sender and recipient information is anonymous for the adversary attack. □

For any two normal nodes in different MANETs to communicate anonymously, the communication can be broken into three segments: the communication between the sender and a local super node in the message sender's local MANET, the communication between two super nodes in the corresponding MANETs, and the communication between the recipient super node and the recipient. Theorem 5 has assured the communication anonymity between a super node and a normal node in the local MANETs. Therefore, we only need to ensure anonymity between two super nodes in different MANETs in order to achieve full anonymity between the sender and recipient.

We already described before that each super node is being assigned a large set of pseudonyms. A dynamically selected pseudonym will be used for each communication. The pseudonyms do not carry the user information implicitly. Therefore, the adversary cannot get any information of the super nodes from the network. This result can be summarized into the following theorem.

Theorem 6. *The proposed communication protocol can provide both message sender and recipient anonymity between any two super nodes.*

Corollary 7. *The proposed anonymous communication protocol can provide full anonymity for any sender and recipient in the MANETs.*

4.6.2. Impersonation Attacks. For an adversary elected to perform impersonation attack to a normal node, he needs to be able to conduct forgery attack. We already proved in Theorem 4 that this is infeasible. Therefore, we only need to consider whether it is feasible for an adversary to forge a super node.

For an adversary to impersonate as a super node, he needs to be able to authenticate himself with a super node A. This requires the adversary $\mathcal{A}$ to compute $g^{s_G \mathrm{id}^{\mathcal{A}} \cdot \mathrm{id}_i^A} \bmod p$, where $\mathrm{id}^{\mathcal{A}}$ is the identity of the adversary and id_i^A is the ith pseudonym of the super node A. However, since the adversary does not know the master secret s_G, he is unable to compute $g^{s_G h(\mathrm{id}^{\mathcal{A}}) \cdot h(\mathrm{id}_i^A)} \bmod p$ and impersonate as a super node. Therefore, we have the following theorem.

Theorem 8. *It is computationally infeasible for a PPT adversary $\mathcal{A}$ to impersonate as a super node.*

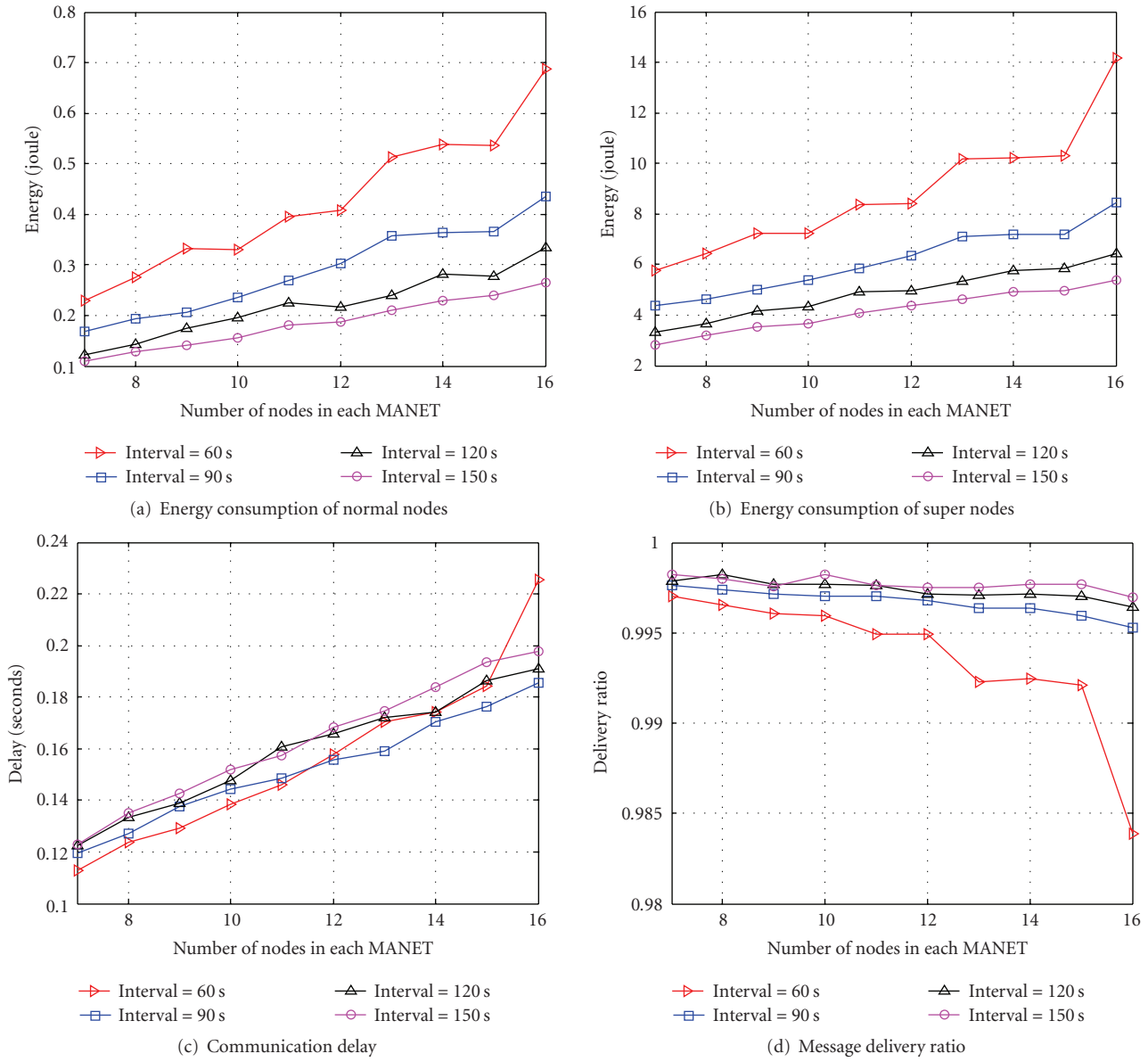
Like all other network communication protocols, in our proposed protocol, an adversary may choose to drop some of the messages. However, if the immediate predecessor and the successor nodes are honest and willing to cooperate, then the messages being dropped, and the substitution of the valid messages with the dummy messages can be effectively tracked using the provided message flags.

An adversary that is elected as a super node may refuse to forward messages across the MANETs and thus block the anonymous communications between the sender and the receiver. This attack can be hard to detect if the sender does not have the capability to monitor all network traffic. However, the sender can randomly select the super nodes for each data transmission. If the nonce is properly generated, when a packet is lost, the recipient should be able to know.

4.6.3. Message Replay Attacks. The message replay attack occurs when an adversary can intercept the communication packet, correlate the message to the corresponding sender and recipient, and retransmit it. We have the following theorem.

Theorem 9. *It is computationally infeasible for an adversary to successfully modify/reply an (honest) node's message.*

Proof (Sketch). According to (11), each message package in communication has a unique one-time session ID (nonce) to protect the message package from being modified or replayed. In addition, these fields are encrypted using the intermediate receiver nodes' public key so that only the designated receiver nodes can decrypt the message. In this way, each packet transmitted across different MANETs bears different and uncorrelated IDs and content for PPT adversaries. Therefore, it is computationally infeasible for the adversary to modify or replay any messages in the MANET. This includes the case that even if the same message is being



(a) Energy consumption of normal nodes

(b) Energy consumption of super nodes

(c) Communication delay

(d) Message delivery ratio

Figure 2: Simulation results of the proposed secure routing scheme.

transmitted multiple times, the adversary still cannot link them together without knowing all the private keys of the intermediate nodes. □

5. Performance Analysis and Simulation Results

In this section, we will provide simulation results of our proposed protocol on energy consumption, communication delay, and message delivery ratio. For energy consumption, we provide simulations for both the normal nodes and the super nodes. For wireless communications, due to collision and packet drop, it is very challenging to assure high messages delivered ratio. However, our simulation results demonstrate that the proposed protocol can achieve high message delivery ratio (Figure 2).

Our simulation was performed using ns-2 on Linux system. In the simulation, the target area is a square field of size 2000 × 2000 meters. There are 64 rings located in this area. The number of the nodes on each ring, that is, the ring length, is set to be from 7 to 16 in our simulation. The message generation interval is set to be four different values: 60 seconds, 90 seconds, 120 seconds, and 150 seconds in our simulation for comparison. The messages transmitted in the network are 512 bytes long.

6. Conclusion

In this paper, we first propose a novel and efficient source anonymous message authentication scheme (SAMAS) that can be applied to any messages. While ensuring message

sender privacy, SAMAS can also provide message content authenticity. To provide provable communication privacy without suffering from transmission collusion problem, we then propose a novel privacy-preserving communication protocol for MANETs that can provide both message sender and recipient privacy protection. Security analysis shows that the proposed protocol is secure against various attacks. Our performance analysis and simulation results both demonstrate that the proposed protocol is efficient and practical. It can be applied for secure routing protection and file sharing.

Acknowledgments

This research is supported in part by NSF grants CNS 0845812, CNS 0848569, CNS 0716039 and CNS 0746811.

References

[1] D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, pp. 84–88, 1981.

[2] D. Chaum, "The dining cryptographers problemml: unconditional sender and recipient untraceability," *Journal of Cryptology*, vol. 1, no. 1, pp. 65–75, 1988.

[3] M. Reed, P. Syverson, and D. Goldschlag, "Anonymous connections and onion routing," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 482–494, 1998.

[4] L. von Ahn, A. Bortz, and N. Hopper, "*k*-anonymous message transmission," in *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS '03)*, pp. 122–130, Washingtion, DC, USA, October 2003.

[5] O. Berthold, H. Federrath, and S. Köpsell, "Web MIXes: a system for anonymous and unobservable internet access," in *Proceedings of the Workshop on Design Issues in Anonymity and Unobservability*, vol. 2248 of *Lecture Notes in Computer Science*, pp. 115–129, 2001.

[6] G. Danezis, R. Dingledine, and N. Mathewson, "Mixminion: design of a type III anonymous remailer protocol," in *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, pp. 2–15, Oakland, Calif, USA, May 2003.

[7] C. Gülcü and G. Tsudik, "Mixing email with babel," in *Proceedings of the Symposium on Network and Distributed System Security*, San Diego, Calif, USA, February 1996.

[8] U. Möller, L. Cottrell, P. Palfrader, and L. Sassaman, "Mixmaster protocol," Version 2, July 2003.

[9] M. Reiter and A. Rubin, "Crowds: anonymity for web transaction," *ACM Transactions on Information and System Security*, vol. 1, no. 1, pp. 66–92, 1998.

[10] A. Beimel and S. Dolev, "Buses for anonymous message delivery," *Journal of Cryptology*, vol. 16, no. 1, pp. 25–39, 2003.

[11] S. Goel, M. Robson, M. Polte, and E. Sirer, "Herbivore: a scalable and efficient protocol for anonymous communication," Tech. Rep. 2003-1890, Cornell University, Ithaca, NY, USA, 2003.

[12] P. Golle and A. Juels, "Dining cryptographers revisited," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (Eurocrypt '04)*, Lecture Notes in Computer Science, pp. 456–473, Interlaken, Switzerland, May 2004.

[13] A. Pfitzmann and M. Hansen, "Anonymity, unlinkability, unobservability, pseudonymity, and identity management a proposal for terminology," February 2008, http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.31.pdf.

[14] A. Pfitzmann and M. Waidner, "Networks without user observability-design options," in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques (Eurocrypt '85)*, vol. 219 of *Lecture Notes in Computer Science*, pp. 245–253, Linz, Austria, April 1985.

[15] M. Waidner, "Unconditional sender and recipient untraceability in spite of active attacks," in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques (Eurocrypt '89)*, vol. 434 of *Lecture Notes in Computer Science*, pp. 302–319, Houthalen, Belgium, April 1989.

[16] D. Pointcheval and J. Stern, "Security arguments for digital signatures and blind signatures," *Journal of Cryptology*, vol. 13, no. 3, pp. 361–396, 2000.

[17] L. Harn and Y. Xu, "Design of generalised ElGamal type digital signature schemes based on discrete logarithm," *Electronics Letters*, vol. 30, no. 24, pp. 2025–2026, 1994.

[18] K. Nyberg and R. A. Rueppel, "Message recovery for signature schemes based on the discrete logarithm problem," in *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques (Eurocrypt '95)*, vol. 950 of *Lecture Notes in Computer Science*, pp. 182–193, Saint-Malo, France, May 1995.

[19] B. Möller, "Provably secure public-key encryption for length-preserving chaumian mixes," in *Proceedings of the Cryptographer's Track at the RSA Conference (CT-RSA '03)*, vol. 2612 of *Lecture Notes in Computer Science*, pp. 244–262, San Francisco, Calif, USA, April 2003.

[20] C. Shields and B. N. Levine, "A protocol for anonymous communication over the Internet," in *Proceedings of the 7th ACM Conference on Computer and Communication Security*, D. Gritzalis, Ed., ACM Press, Athens, Greece, November 2000.

[21] R. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT '01)*, vol. 2248 of *Lecture Notes in Computer Science*, Springer, Gold Coast, Australia, December 2001.

[22] T. A. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

[23] S. Goldwasser, S. Micali, and R. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks," *SIAM Journal on Computing*, vol. 17, pp. 281–308, 1988.

[24] D. Pointcheval and J. Stern, "Security proofs for signature schemes," in *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT '96)*, vol. 1070 of *Lecture Notes in Computer Science*, pp. 387–398, Saragossa, Spain, May 1996.

[25] M. Bellare and P. Rogaway, "Random oracles are practical: a paradigm for designing efficient protocols," in *Proceedings of the 1st ACM Conference on Computer and Communications Security (CCS '93)*, pp. 62–73, Fairfax, Va, USA, November 1993.

[26] F. P. 180-1, "Secure hash standard," April 1995, http://www.itl.nist.gov/fipspubs/fips180-1.htm.

[27] D. Balfanz, G. Durfee, N. Shankar, D. Smetters, J. Staddon, and H. C. Wong, "Secure handshakes from pairing-based key agreements," in *Proceedings of the IEEE Symposium on Security & Privacy*, Oakland, Calif, USA, May 2003.