A Survey of Applications of Identity-Based Cryptography in Mobile Ad-Hoc Networks

Shushan Zhao, Akshai Aggarwal, Richard Frost, Xiaole Bai

***Abstract*—Security in mobile ad-hoc networks (MANETs) continues to attract attention after years of research. Recent advances in identity-based cryptography (IBC) sheds light on this problem and has become popular as a solution base. We present a comprehensive picture and capture the state of the art of IBC security applications in MANETs based on a survey of publications on this topic since the emergence of IBC in 2001. In this paper, we also share insights into open research problems and point out interesting future directions in this area.**

***Index Terms*—Identity-based Cryptography, Mobile Ad-hoc Networks**

I. Introduction

RESEARCH on security of MANETs remains active, in spite of years of exploration, in both academia and industry. It is partially due to the fact that no mature solution is widely accepted and the growing availability of small, personalized mobile devices with peer to peer communication capability through wireless channels.

General security requirements for MANETs include [1]: *Data Confidentiality* that keeps data secret to outsiders, *Data Integrity* that prevents data from being altered, *Data Freshness* that keeps data in the correct order and up-to-date, *Data Availability* that ensures data to be available on request, *Data & Identity Authentication* that verifies that the data or request came from a specific, valid sender, and *Non-repudiation* that ensures a node cannot deny sending a message.

Security mechanisms that are widely used and proven to be effective in wired networks are not always applicable to MANETs. Attacks that can be effectively detected and prevented in wired networks have been big security challenges in MANETs. Examples include, but are not limited to, identity/address spoofing, message tampering and forgery, message replay, etc. Compared to wired networks, the combination of the following characteristics of MANETs make it especially difficult to achieve security requirements:

- Lack of a network infrastructure and online administration.
- Network topology and node membership dynamics.
- The potential insider attacks.
- Computing and communicational capacity constrained resources.
- Wireless link vulnerabilities.

Security proposals in early research are typically attack-oriented. They often first identify several security threats and then enhance the existing protocol or propose a new protocol to thwart them. Such solutions are designed explicitly against limited attack models. They work well in the presence of designated attacks but may collapse under combined or unanticipated attacks [2].

Cryptography is then used to provide a general design framework. Cryptography techniques used in MANETs can be classified into two categories, namely, *Symmetric Key based* and *Asymmetric Key based*. In symmetric key based schemes, if an attacker compromises the symmetric key of a group of users, then all encrypted messages for that group will be exposed. Asymmetric key based schemes can provide more functionalities than symmetric ones, e.g., key distribution is much easier, authentication and non-repudiation are available, compromise of a private key of a user does not reveal messages encrypted for other users in the group. However, they are generally computationally expensive.

Traditional asymmetric cryptography widely and effectively used in the Internet relies on a Public Key Infrastructure (PKI). The success of PKI depends on the availability and security of a Certificate Authority (CA), a central control point that everyone trusts. In general MANETs, applying PKIs by maintaining a central control point is clearly not always feasible. Another obstacle that impedes PKI's employment in MANETs is the heavy overhead of transmission and storage of public key certificates (PKCs).

Identity-based cryptography (IBC) is a special form of public key cryptography. It is an approach to eliminate the requirement of a CA and PKCs. Since 2001, IBC has attracted more and more attention from security researchers. Some properties of IBC make it especially suitable for MANETs. Fang et al. [3], [4] summarize the advantages of IBC to MANETs:

- Easier to deploy without any infrastructure requirement. This saves certificate distribution, while bringing "free" pairwise keys without any interaction between nodes.
- Its resource requirements, regarding process power, storage space, communication bandwidth, are much lower.
- The public key of IBC is self-proving and can carry much useful information.

We believe that IBC, with its fast development in recent years, is a promising solution for MANET security issues. This has motivated us to write this survey. We present a

Manuscript received 23 April 2010; revised 31 August 2010 and November 27 2010. This work was supported in part by the National Science Foundation of China (NSFC) under grant No. 61070245. Any opinions, findings, conclusions, and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

S. Zhao, A. Aggarwal and R. Frost are with School of Computer Science, University of Windsor, Canada (e-mail: {zhao114, akshaia, rfrost}@uwindsor.ca).

X. Bai is with Department of Computer and Information Science, University of Massachusetts Dartmouth, U.S. (e-mail: xbai@umassd.edu)

Digital Object Identifier 10.1109/SURV.2011.020211.00045

comprehensive picture and have identified the state of the art of important IBC security applications in MANETs by conducting a survey on publications over the recent decade from 2001 to 2010. We also share insights into open research problems and point out interesting future directions in this area. Since difficulty of MANET security lies on differences between MANETs and wired infrastructured networks in network and lower layers, identity-based cryptosystems are mostly employed in network layer, i.e. in routing protocols. Hence, most of previous publications, and we, focus on key management and routing protocols. A non-trivial point of this survey is that we review the proposals in the literature from a system engineering perspective as to how a practical system works with these existing proposals, e.g., how to set up a secure routing among a set of nodes. In this perspective, we identify some weaknesses of these protocols which cannot be found if we look at them separately.

The survey is organized as follows: Section II briefly reviews the background of research on security of MANETs and IBC, and summarizes important publications in the development of IBC which have had a great influence on security of MANETs. Sections III to V review and summarize schemes applying IBC to MANETs, in sub-areas of key management, secure routing, and other applications. Section VI presents some remaining issues and potential research directions of applying IBC to MANETs. Section VII identifies the suitable market of IBC in MANETs and concludes the survey.

II. Background

A. A Brief History of Identity-based Cryptography

Identity-based cryptography schemes are in the category of "*Asymmetric Key based*" cryptography. Identity-based cryptography specifies a cryptosystem in which both public and private keys are based on the identities of the users. The idea of IBC was first proposed by Shamir [5] in 1984. Such a scheme has the property that a user's public key is an easily calculated function of his identity, while a user's private key can be calculated for him by a trusted authority, called a Private Key Generator (PKG). The identity-based public key cryptosystem can be an alternative for certificate-based PKI, especially when efficient key management and moderate security are required. Compared to traditional PKI, it saves storage and transmission of public keys and certificates, which is especially attractive for devices forming MANETs. Thus, application of IBC in MANETs is an important research topic in areas of both cryptography and MANETs.

For a long time after Shamir published his idea, the development on IBC was very slow. Joux [6], in 2000, showed that Weil pairing can be used for "good" by using it in a protocol to construct three-party one-round Diffie-Hellman key agreement. This was one of the breakthroughs in key agreement protocols. After this, Boneh and Franklin [7] presented at Crypto 2001 an identity-based encryption scheme based on properties of bilinear pairings on elliptic curves, which is the first fully functional, efficient and provably secure identity-based encryption scheme. In Asiacrypt 2001, Boneh, Lynn and Shacham proposed a basic signature scheme using pairing, the BLS scheme [8], that has the shortest length among signature schemes in classical cryptography.

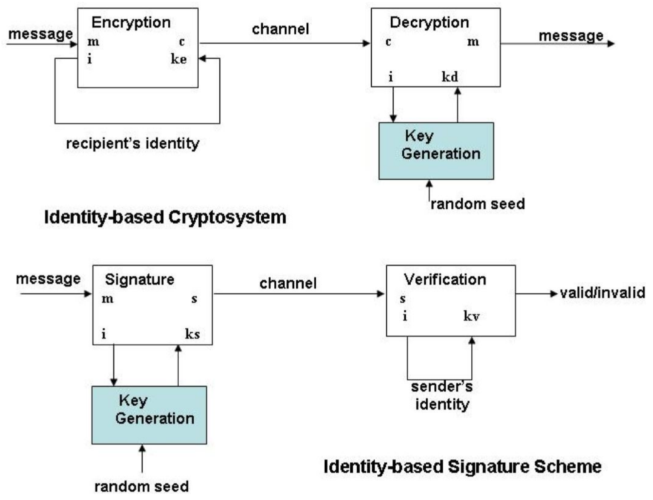


Fig. 1. Shamir's Identity-based Cryptosystem and Signature Scheme ([5, pp. 52])

Subsequently, a number of cryptographic schemes based on the work of [7] and [8] were proposed. This type of identity-based cryptography is also named Pairing-based Cryptography (PBC). There are also a few IBC schemes using other approaches, e.g., Cocks' scheme is based on the quadratic residuosity problem [9]. Most of proposals for MANETs in the literature use PBC.

B. Preliminaries of Identity-based Cryptography

Unless otherwise stated, in this and following sections we use the same notations as in this section, which are summarized in Table I.

In [5], Shamir introduced a novel type of cryptographic scheme, the so-called identity-based cryptosystem, which enables any pair of users to communicate securely and to verify each other's signatures without exchanging private or public keys, without keeping key directories, and without using the services of a third party.

Shamir stated that "The scheme is based on a public key cryptosystem with an extra twist: instead of generating a random pair of public/secret keys and publishing one of these keys, the user chooses his name and network address as his public key. Any combination of name, social security number, street address, office number or telephone number can be used provided that it uniquely identifies the user in a way he cannot later deny, and that it is readily available to the other party. The corresponding secret key is computed by a PKG and issued to the user when he first joins the network." Figure 1 illustrates his idea: In an identity-based cryptosystem, the recipient's identity i is used to generate the encryption key, and the decryption key is derived from i and a random seed k. In an identity-based signature scheme, the signature key is generated from sender identity i and a random seed k, and the verification key is derived from sender's identity i.

In his paper, Shamir specifies the requirements of an implementation of such a scheme and lists the implementation principals:

- The choice of keys is based on a truly random seed k. When the seed k is known, secret keys can be easily computed for a non-negligible fraction of the possible public keys.

- The problem of computing the seed k from specific public/secret key pairs generated with this k is intractable.

Based on these requirements, he states that the RSA scheme is not capable for his scheme.

He states that at that stage they have concrete implementation proposals only for identity-based signature schemes, but conject that such cryptosystems exist and encourage the readers to look for such systems.

Currently, most of IBC schemes, and all PBC schemes, are based on assumptions of hard problems in elliptic curves. The most frequently used assumptions are [10, pp. 7]:

- **Computational Diffie-Hellman (CDH) problem in** $\mathbb{G}_1$: there is no efficient algorithm to compute $\hat{e}(P,P)^{ab}$ from $P, aP, bP \in \mathbb{G}_1$ for $a, b \in \mathbb{Z}_q^*$.
- **Weak Diffie-Hellman (W-DH) problem in** $\mathbb{G}_1$: there is no efficient algorithm to compute sQ from $P, Q, sP \in \mathbb{G}_1$ and $s \in \mathbb{Z}_q^*$. (W-DH problem is no harder than CDH problem).
- **Bilinear Diffie-Hellman (BDH) problem in** $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$: there is no efficient algorithm to compute $\hat{e}(P,P)^{abc} \in \mathbb{G}_2$ from $P, aP, bP, cP \in \mathbb{G}_1$ where $a, b, c \in \mathbb{Z}_q^*$.
- **Decisional Bilinear Diffie-Hellman (DBDH) problem in** $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$: there is no efficient algorithm to decide if $r = \hat{e}(P,P)^{abc}$ given $r \in \mathbb{G}_2$ and $a, b, c \in \mathbb{Z}_q^*$

Boneh and Franklin's scheme, published in [7], was the first fully functional IBC scheme. The paper refers to Shamir's idea of the identity-based Encryption (IBE) scheme [5], and several proposals for IBE schemes [11], [12], [13], [14]. They consider none of them fully satisfactory due to unrealistic requirements, such as users not colluding, the long time required for private key generation, and tamper-resistant hardware.

Security of their system is based on the BDH problem, an analogue of the computational Diffie-Hellman assumption on elliptic curves. They build the IBE system from a symmetric bilinear map and use the Weil pairing on elliptic curves as an example of such a map. A *Symmetric Bilinear Map* is denoted $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ between two cyclic groups $\mathbb{G}_1, \mathbb{G}_2$ of order q for some large prime q, where $\mathbb{G}_1$ is the group of points of an elliptic curve over $\mathbb{F}_p$ and $\mathbb{G}_2$ is a subgroup of $\mathbb{F}_{p^2}^*$ [1].

A cryptographic bilinear map satisfies the following properties [10, pp. 6]:

1) **Bilinear**: $\hat{e}(aP, bQ) = \hat{e}(P,Q)^{ab}$ for all $P, Q \in \mathbb{G}_1$ and all $a, b \in \mathbb{Z}_q^*$. This can be restated in the following way. For $P, Q, R \in \mathbb{G}_1, \hat{e}(P+Q, R) = \hat{e}(P,R)\hat{e}(Q,R)$ and $\hat{e}(P, Q+R) = \hat{e}(P,Q)\hat{e}(P,R)$.
2) **Non-degenerate**: $\hat{e}(P,P) \in \mathbb{F}_{p^2}^*$ is an element of order q, and in fact a generator of $\mathbb{G}_2$. In other words, $\hat{e}(P,P) \neq 1$
3) **Computable**: Given $P, Q \in \mathbb{G}_1$ there is an efficient algorithm to compute $\hat{e}(P,Q)$.

Their scheme is specified by four randomized algorithms [7, pp. 215]:

- Setup: The algorithm maps arbitrary string identities to points on an elliptic curve. Set the system public key P_{pub} as sP where s is a random number in $\mathbb{Z}_q^*$, and P is an arbitrary point in $E/\mathbb{F}_p$ of order q. Choose a cryptographic hash function $H : \mathbb{F}_{p^2} \rightarrow \{0,1\}^n$ for some n. Choose a cryptographic hash function $G : \{0,1\}^* \rightarrow \mathbb{F}_p$. The system parameters are $params = \langle p, n, P, P_{pub}, G, H \rangle$. The master-key is $s \in \mathbb{Z}_q$.
- Extract: For a given string $ID \in \{0,1\}^*$, the algorithm builds public key for ID: $Q_{ID} = G(ID)$, a point in $E/\mathbb{F}_p$ mapped from ID, and the private key d_{ID} as $d_{ID} = sQ_{ID}$.
- Encrypt: Choose a random $r \in \mathbb{Z}_q$, and set the ciphertext to be $C = \langle rP, M \oplus H(g_{ID}^r) \rangle$ where $g_{ID} = \hat{e}(Q_{ID}, P_{pub}) \in \mathbb{F}_{p^2}$
- Decrypt: Let $C = \langle U, V \rangle$ be a ciphertext encrypted using the public key of ID, decrypt C using the private key d_{ID}: $V \oplus H(\hat{e}(d_{ID}, U)) = M$

TABLE I
NOTATIONS USED IN THIS SURVEY

Symbols	Meanings
$\mathbb{Z}$	set of integers
$\mathbb{Z}_n$	set of integers mod n
$\mathbb{F}_q$	the finite field with q elements
$\mathbb{Z}_q^*$	the multiplicative group of integers modulo prime number q. $\mathbb{Z}_q^* = \{a \mid 1 \leq a \leq q-1\}$
$E/\mathbb{F}_p$	elliptic curve over $\mathbb{F}_p$
$\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$	a bilinear map between two cyclic groups $\mathbb{G}_1, \mathbb{G}_2$
P	an arbitrary point in $E/\mathbb{F}_p$
d_{ID}	private key of ID
Q_{ID}	public key of ID
s	master secret key
P_{pub}	system public key
$H_{(i)}$	a hash function

Further, they analyze the security of their scheme, and state that the scheme has chosen ciphertext security in the random oracle model assuming Weak Diffie-Hellman.

The scheme proposed in their paper is subsequently improved by many other researchers, and widely adopted in many identity-based security schemes.

Following Boneh and Frankin's scheme [7], many PBC schemes have been proposed. Modified Weil Pairing and Tate Pairing are examples of cryptographic bilinear maps. Currently, active research is being carried out to obtain efficient algorithms to compute pairings.

C. Threshold Cryptography and Key Management in MANETs

Many IBC schemes use threshold cryptography which originated from Shamir [15], for their key management. Shamir gives a solution to the problem of sharing a secret among a number of users in [15]. In his paper, he identifies the problem of how to divide data D into n pieces in such a way that D is easily reconstructable from any t pieces, but even complete knowledge of $t-1$ pieces reveals absolutely no information about D.

Shamir proposes a (t, n) threshold scheme to solve this problem based on polynomial interpolation: given t points in the dimensional plane $(x_1, y_1) \ldots (x_t, y_t)$, with distinct x_i's, there is one and only one polynomial $q(x)$ of degree $t-1$ such that $q(x) = y_i$ for all i. To divide the secret D

[1] The general form of a bilinear map is like: $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_3$, where $\mathbb{G}_1$ and $\mathbb{G}_3$ are cyclic, and $\mathbb{G}_2$ is not necessarily cyclic.

into n pieces, he suggests picking a random $t-1$ degree polynomial $q(x) = a_0 + a_1x + \cdots + a_tx^{t-1}$ in which $a_0 = D$, and each piece is the value of the polynomial at the n points: $D_1 = q(1), \ldots, D_i = q(i), \ldots, D_n = q(n)$. Thus any subset of t of the pieces can determine the coefficients of the polynomial (using e.g. Lagrange interpolation) and thus the secret data at a certain point. He suggests the use of modular arithmetic instead of real arithmetic. The set of integers modulo a prime number p forms a field in which interpolation is possible.

This scheme was later employed by many researchers to construct a distributed PKG in IBC and to solve security problem in MANETs.

Zhou et al. [16] suggest the use of Shamir's threshold scheme to secure ad hoc networks. The authors identify the problem to establish a key management service using a single CA in ad hoc networks. They suggest distributing this service to an aggregation of nodes.

Zhou et al. refer to the work of [17], [18] and indicate that they use the theory of threshold cryptography as a basis for their work. The authors propose a distributed CA architecture and PKI used in ad hoc networks. The CA service, as a whole, has a public/private key pair K/k. The public key K is known to all nodes in the network, whereas the private key k is divided into n shares $s_1, s_2, ..., s_n$, one share for each server. To provide the certificate signing service, "threshold" cryptography algorithm is used — for a message m, server i can generate a partial signature $PS(m, s_i)$ using its share s_i and forward the signature to a combiner. If t out of n partial signatures are collected by the combiner, they can jointly perform the operation correctly.

The idea of distributed CA has been subsequently adopted for distributed PKG in many IBC proposals in MANETs later.

III. Key Management Using IBC

Cryptographic techniques are often at the center of solving security problems in MANETs and hence need key management. Key management in IBC requires key generation and distribution methods, and ideally key protection and revocation. This section reviews and discusses proposals for IBC key management in MANETs.

A. Master Key and Private Key Generation

Most of the master key and private key generation schemes are derived from and are variants of [7]. The criteria to judge this type of scheme is use of their four primitive algorithms. In this section, we first provide some examples based on traditional threshold cryptography of [16] and discuss the limitations of these schemes, and then discuss some proposals that attempt to improve traditional threshold cryptography. We give some key generation schemes tweaked for specific purposes: e.g. high privacy, compromise-tolerance, or lightweight.

1) Key Generation Using Traditional Threshold Cryptography: PKG plays a fundamental role in an identity-based cryptosystem, but it is not trivial to have a robust PKG in a MANET environment. As Zhou et al. have suggested [16], a CA service of PKI can be distributed to multiple nodes in a MANET environment. This idea is also applicable to IBC.

Khalili et al. [19] propose to use IBC to secure ad hoc networks. The authors refer to the work of [16], [20] and identify the problem that all proposed key management solutions assume either pre-existing shared secrets among nodes or the presence of a common PKI. They propose to combine efficient techniques from identity-based and threshold cryptography to provide a mechanism that enables flexible and efficient key distribution while respecting the constraints of ad-hoc networks. At the time of network formation, the participating nodes form a threshold PKG, and generate—in a distributed fashion—a master public key. The master secret key is shared in a t-out-of-n threshold manner by this initial set of n nodes. All nodes in the network can use their identities as their public keys. The secret key, corresponding to the public key, is computed by having the node obtain t shares of their key from t-out-of-n of the original nodes. All subsequent communications are encrypted and decrypted using the master public key and the ID of the recipient. The authors based their proposal on Boneh's identity-based cryptosystem algorithms [7].

As a detailed implementation of Khalili's idea, Deng et al. [21], [22] propose an identity-based key management and authentication system for MANET, using identity-based and threshold cryptography. The proposed approach consists of two components: distributed key generation and identity-based authentication. This paper describes algorithms for master key generation, distributed private key generation, new master key share creation. The system was built on the assumption that each mobile node has a mechanism to discover its one-hop neighborhood and to get the identities of other nodes in the network. The key generation component provides the network master key pair and the public/private key pair to each node in a distributed way. The system public key/master key pair is computed collaboratively by the initial network nodes without constructing the master key at any single node, as Shamir and Zhou suggested [15], [16] [2]. The public key of node ID can be computed as $Q_{ID} = H(ID||Expire - time)$.

When a new node with public key Q_{ID} joins a network, it presents its identity, self-generated temporary public key, and some other required physical proofs (depending on key issuing policy) to t neighbor nodes and requests PKG service. Each node in the coalition, with share of master private key — s_i, verifies the validity of the identity of the new node and generates a secret share of a new private key d_{ID} encrypted with the temporary public key and sends to the requesting node. By collecting t shares of its new private key, the requesting node would compute its new private key $d_{ID} = \sum_{i=1}^{t} s_iQ_{ID}$. It discards its temporary public/private key pair, and keeps the new key pair in its memory for the later authentication

[2] Each node C_i randomly chooses a secret x_i and a polynomial $f_i(z)$ over $\mathbb{Z}_q$ of degree $t-1$, such that $f_i(0) = x_i$. Node C_i computes his sub-share for node C_j as $ss_{ij} = f_i(j)$ for $j = 1, 2...n$ and sends ss_{ij} securely to C_j. After receiving $n-1$ sub-shares, node C_j can compute its share of master private key as $S_j = \sum_{i=1}^{n} ss_{ij} = \sum_{i=1}^{n} f_i(j)$. Any coalition of t shareholders can jointly recover the secret as in basic secret sharing: $s = \sum_{i=1}^{t} S_il_i(z)mod\ q$, where $l_i(z)$ is the Lagrange coefficient. Due to the homomorphic property of share refreshing, the jointly generated master key is equal to $\sum_{i=1}^{n} f_i(0)$.

and communication. After this key generation process, the requesting node obtains its new private key d. To initialize the share of master key for the requesting node, each coalition node C_i generates the partial share $ss_{i,p} = S_i \cdot l_i(p)$ for node C_p ($l_i(p)$ is the Lagrange term). Node C_p obtains its new share by summing up the partial shares as $S_p = \sum_{j=1}^{t} ss_{j,p}$.

Another implementation of Deng's scheme is described in [23]. The authors implemented a scheme with distributed master key generation, private key generation, secret share update, and secret share generation for a new joining node. One thing they did not mention is how secret shares are distributed to other nodes from one node.

Xia's scheme [24] is also very similar to Deng's scheme: A set of Distributed PKG nodes collaboratively generate system public key and master key in a fully distributed manner; Shares can be updated among PKGs; New nodes can get their shares from PKGs and become new PKG nodes.

Differences from Deng's scheme are:

1) This scheme does not use temporary PKI for secret share distribution as in Deng's scheme. Instead, it employs a self-generated public/private key pair in the following way: each DPKG node computes a temporary public key and sends it to other DPKG nodes. Secret shares are encrypted and decrypted using this temporary public key.
2) The paper applies their scheme in OLSR routing protocol, particularly use HELLO messages and TC messages in OLSR to select and mark DPKG nodes, while Deng et al. apply their scheme in DSR routing protocol.

These differences lead to the following problems:

1) Each DPKG node has to store in memory the temporary public keys of other DPKG nodes.
2) System public key and master key collection process is not secure, because only public channels are available at this stage.
3) The keys generated are not guaranteed secure, because it does not provide any security protection for OLSR routing protocol it relies on.

All of these schemes use threshold cryptography to distribute the functionality of PKG to multiple nodes. Due to threshold cryptography, these schemes have the following weaknesses:

1) Interdependency Cycle between Secure Routing and Security Services: These scheme rely on some existing routing or online administration mechanisms (e.g. out-of-band communicant, side channel) to distribute secret shares among the distributed PKG nodes. Thus, they cannot be used in secure routing protocols that would require secure keys. This is noted as the problem of interdependency cycle between security services and secure routing [25], [26].
2) Proximity-caused Insecurity: In some circumstances where a node can move in order to access to more nodes, one way to avoid the routing-security interdependency cycle problem is to have a threshold number of authorized users that are physically close to each other (i.e., within one-hop communication distance so that routing is eased). This incurs another related problem—the proximity-caused insecurity: it is possible that an adversary compromises these nodes within a short period of time (e.g., by capturing the nodes and/or compromising them one by one) [26]. Furthermore, the proximity-based solution is not applicable to fully distributed key generation schemes where all nodes participate in and contribute to the key generation, and thus routing connecting all nodes (not only among a threshold number of nodes) is still required.
3) Mobile Attacks: Threshold cryptography is subject to mobile attacks, in which a mobile adversary could move to compromise multiple nodes and reveal the secret shares of them in order to recover the secret. To counter mobile attacks, the above proposals use secret refreshing mechanism in which secret shares are updated in intervals and new shares cannot be combined with old ones to recover the secret. They assume a mobile adversary cannot compromise enough authentic nodes within the share refreshing period. Merwe et al. in [25] do not think this assumption is practical. We have a separate paper analyzing this issue and proposing solutions [27].

We will recall and discuss this problem further in Section IV shortly.

2) Multicast Group for Threshold PKG: Li et al. [28] point out that share refreshing in [16] needs a secure channel for delivering subshares, of which Zhou et al. did not provide the implementation. They propose a signcryption scheme that exactly provides a way for secure transmission, by using periodic private keys, multicast group of PKGs, and key proxy. Their work is based on papers [5], [16], [29].

Li et al. introduce a key proxy for key generation. A key proxy is selected from a group of server nodes: all server nodes form and maintain a few multicast groups according to location. A node floods its Routing REQuest (RREQ) to find a route to the server nodes group. When it receives Routing REPlies (RREPs) from server nodes, it selects a server node, say u, which has the shortest path to itself as its key proxy. The routing information to the node u is stored. When it wants to update its private key later, it sends its Private key update REQuest (PREQ) to u and u multicasts the PREQ to all server nodes. The private key of a node is updated periodically. Server node computes a partial private key of the client ($d_{A,i}$) using its master key share, then signcrypts and sends it in a Private key update REPly (PREP) message to A.

In order to check malicious server nodes, at the initial time of the network, PKG publishes a piece of verification information consisting of $s_i \cdot P$ for each server node i. To check the validity of partial key it receives from i, node A needs only to check whether the equation $\hat{e}(Q_A, s_i \cdot P) = \hat{e}(d_{A,i}, P)$ holds.

Li et al. use "proactive threshold" similar to Zhou et al.'s [16], with two modifications: replacing secure channel with multicast, and replacing a secret share with a vector. The share vector is encrypted and multicast to the server nodes group. Every server node can only decrypt its own share.

This scheme distributes partial private keys of PKG server nodes to the network before starting for future secure communication, in a way like certificates in PKI. This is against IBC advantages. The multicast group of PKGs is fundamental in the scheme, but a critical question remaining open in this work is how the multicast group is formed. Secure multicast routing cannot be established without secure keys. Thus the security-routing interdependency cycle problem is not addressed.

3) Offline Threshold PKG: Zhang et al. [30] propose a distributed PKG (D-PKG) scheme to distribute PKG of IBC to multiple nodes, based on the work of [5], [16], [7]. The master key of the IBC system is distributed to D-PKGs in an offline manner, and then a threshold number of D-PKG's can function as PKG. In each D-PKG, the Trusted Authority (TA) supplements the network bootstrapping process with the following operations [30, pp. 3517]:

1) Determine a $(t-1)$-degree $(1 \leq t \leq N)$ polynomial, $h(x) = s + a_1x + a_2x^2 + \cdots + a_{t-1}x^{t-1} (mod\ q)$.
2) Select n $(t \leq n \leq N)$ nodes as D-PKGs (denoted by SH). Each node in SH gets a share of s as $s_k = h(k)$.
3) Calculate a set of share commitments as $SC = \{P_k = s_k \cdot P \in \mathbb{G}_1 | 1 \leq k \leq n\}$.

SH and SC are appended to the public system parameters and sent to all nodes. Similar to schemes mentioned above (refer to III-A1), any combination of t D-PKGs can collectively reconstruct the system master-key s.

These D-PKG's collaboratively provide the PKG service: Node B sends them a private-key sub-request containing its public key ID_B. Upon receiving the request, each chosen D-PKG sends back a sub-reply containing a partial private key: $d_{B,i} = s_i H_1(ID_B || otherInfo)$. B can verify its authenticity using P_i: $\hat{e}(d_{B,i}, P) = \hat{e}(H_1(ID_B || otherInfo), P_i)$ [3]. After obtaining t authentic private-key pieces, B can calculate the complete private key in the same way computing the master-key.

This scheme is similar to the one mentioned in III-A1, but differs in the following ways: this scheme distributes secret shares offline, and thus does not require on-line secure channels for secret share distribution; the secret shares of this scheme are not refreshed or updated, thus it is more subject to mobile attacks. Although the master key generation does not require secure channels, the private key generation still needs them; thus, routing-security interdependency cycle is not addressed. Also, the share commitments of each D-PKG are used like certificates which are distributed to the network nodes before network starts. This is against IBC advantages.

4) Public Channels for Threshold PKG: Ren et al. [31] propose another D-PKG scheme. The scheme eliminates the secure channel requirement by using mutual authentication in public channels.

The key generation and issuing works as follows: A user U_{ID} chooses a password pwd and computes $H_1(ID), H_1(pwd), H_2(pwd)$. Then it publishes the tuple $\langle ID, H_1(ID), H_1(pwd), H_2(pwd) \rangle$. The D-PKGs store them in their database. User selects a random number r and computes a request and sends the request to D-PKGs. D-PKGs checks the validity of the request and computes blinded partial private key and sends it to the user. The user upon receiving blinded partial private keys verifies them and unblinds the private using the proprietary knowledge of r.

The paper claims that the protocol does not require any secure channel to issue the private key and is secure. However, D-PKGs have to store a password for each user, in a way as the distributed CA works in PKC mechanisms. This violates the advantages of identity-based cryptosystems, and requires online service from D-PKGs. Also, the paper did not mention how requests and secret shares are transmitted in public channel. We assume they use broadcast in the discussion below. If that is the case, the interdependency cycle between secure routing and security services is addressed, but it is not efficient, because of communication and computational overhead of broadcast.

5) A Threshold Key Generation Scheme with Compromise-tolerant Key-update Parameters: Fang et al. [4] propose a key generation scheme that provides compromise-tolerant feature for private keys. This is achieved by dividing public/private keys into node-specific and phase-specific components, and predistributed key-update parameters.

The cryptographic materials distributed to each node before network deployment include: pairing parameters: $\langle p, q, \hat{e}, H_1, P, s_1P, s_2P \rangle$, public and private keys: $\langle Q_{ID,0}, d_{ID,0} \rangle$, phase salt: $salt_1$, key-update parameters: $\langle \{v_i(x), l_i(ID)\}_{i=1,\ldots,m} \rangle$, where m is the maximum possible phase index, H_1 is a hash function that maps a string to a non-zero element in $\mathbb{G}_1$, s_1 and s_2 are two distinct master keys. PKG distributes s_2 to D-PKGs using threshold secret sharing, each D-PKG $V \in \Omega$ holds a secret share s_{2V} and a set of values $\{P_{2V} = s_{2V} \cdot P | V \in \Omega\}$ where Ω is the D-PKG set, and $|\Omega| = n$.

Each public/private key pair is both node-specific and phase-specific. At phase-i, node A's public key is $Q_{A,i} = \langle H_1(ID_A), H_1(salt_i) \rangle$, private key is $d_{A,i} = \langle s_1 \cdot H_1(ID_A), s_2 \cdot H_1(salt_i) \rangle$. The first element of each key is node-specific, and the second element is phase-specific. Initially, the PKG issues $Q_{A,1}$ and $d_{A,1}$ to node A. A can acquire phase-specific element $Q_{i+1} = H_1(salt_{i+1})$ and $d_{i+1} = (s_2 \cdot H_1(salt_i))$, where $salt_{i+1} = salt_i + 1$, from the D-PKG set through key update. In the key update, a D-PKG node Z contacts $t-1$ D-PKG, and collects t shares of d_{i+1} and generates d_{i+1} using a $t-out-of-n$ threshold cryptography. Z then broadcasts d_{i+1} to unrevoked nodes securely using a variant of the self-healing group key distribution scheme by Liu et al. [32].

The key update parameters also faciliate key revocation feature, which we will discuss in a later section.

This scheme employs threshold cryptography for generation of phase-specific componenets of private keys online. It is not clearly stated how D-PKGs communicate with each other to exchange secret shares. This process either relies on secure routing which leads to routing-security interdependency cycle problem, or relies on broadcasting which incurs extra traffic overhead. In addition, the scheme does not have good scalability because the size of key-update parameters to be distributed to nodes before network deployment is proportional to number of phases and number of D-PKGs, both of which can become very large.

6) A Non-threshold Key Issuing Scheme for High Key Privacy: Threshold PKG key generation allows redundant PKGs for high availability of master key. The opposite way

[3] The verification process is same as in subsection III-A2

is to use a chain of key privacy authorities (KPAs) to protect master key for high privacy. Lee et al. [33] propose a secure key issuing protocol in which a private key is issued by a key generation center (KGC) and then its privacy is protected by multiple key privacy authorities (KPAs). For all $i = 1, \cdots, n$, KPA_i chooses his master key s_i and computes his public key $P_i = s_i P$. Then KPAs cooperate sequentially to compute the system public key $Y = s_0 s_1 ... s_n P$.

A user ID gets its private key in three stages [33, pp. 73]:

1) In key issuing stage, a node sends its identity ID and blinding factor $X = xP$ to the KGC and requests him to issue a partial private key. The KGC issues a partial private key to the user in a blinded manner: $Q'_0 = H_3(\hat{e}(s_0 X, P_0)) s_0 Q_{ID}$, together with a signature: $Sig_0(Q'_0) = s_0 Q'_0$. Here $H_3(\hat{e}(s_0 X, P_0))$ is a blinding factor. User can unblind it using his knowledge of x [4].
2) In key securing stage, the user requests multiple KPAs in a sequential manner to provide key privacy service by sending ID, X, Q'_{i-1} and $Sig_{i-1}(Q'_{i-1})$. Then KPAs return the private key shares: $Q'_i = H_3(\hat{e}(s_i X, Pi)) s_i Q'_{i-1}$ and signature $Sig_i(Q'_i) = s_i Q'_i$ in a blinded manner.
3) Finally, in key retrieving stage, the user unblinds it to retrieve the real private key: $d_{ID} = \frac{Q'_n}{H_3(\hat{e}(P_0,P_0)^x) \cdots H_3(\hat{e}(P_n,P_n)^x)} = s_0 s_1 \cdots s_n Q_{ID}$. The user can verify the correctness of his private key by $\hat{e}(d_{ID}, P) = \hat{e}(Q_{ID}, Y)$.

The authors have analyzed the security of this scheme and state that since the private key of a user is computed cooperatively by the KGC and n KPAs, the privacy of user's private key is kept if at least one authority remains honest. Only the legitimate user who knows the blinding parameter can unblind the message to retrieve the private key.

This scheme was not originally designed for MANETs. In a MANET environment, it has the following weaknesses: first, all KPAs are required to be online and available, which is not feasible in MANETs; second, secure routing is required to get partial key and signature, which is in routing-security interdependency cycle.

7) A Non-PBC Lightweight IBC Key Generation Scheme: Saxena [34] proposes a scheme of public key cryptography for MANET analogous to identity-based cryptography with some claimed advantages. This scheme can be viewed as a lightweight IBC. This work is based on the work of [16], [15], [35] on threshold cryptography, and on the work of [7] on IBC.

The paper suggests the use of Feldman's *Verifiable Secret Sharing (VSS)* [35] to generate private keys and public keys. In order to setup the system, a dealer (or a set of co-founding members) first chooses appropriate parameters (p, q, g) for the group, and selects a polynomial $f(z) = a_0 + a_1 z + \cdot\cdot\cdot + a_t z^t$ in Z_q, where a_0 is the group secret. The dealer keeps the polynomial secret and publishes commitments to the coefficients of the polynomial, as $w_i = g^{a_i}(mod\ p)$, for $i = 0, \cdot\cdot\cdot, t$. To join the group, a user M_i sends its unique identifier id_i to the dealer who issues it its secret share $x_i = f(id_i)(mod\ q)$ as the private key for M_i. The public key $y_i = g^{x_i}(mod\ p)$ of M_i can be computed by M_j as $y_i = \prod_{j=0}^{t}(w_j)^{id_i^j}(mod\ p)$. Also M_i can compute M_j's public key as: $y_j = \prod_{i=0}^{t}(w_i)^{id_j^i}(mod\ p)$, and pairwise shared key as: $k_{ij} = y_j^{x_i} = g^{x_j x_i} = k_{ji}(mod\ p)$. With these keys, they define the sign/verify and encrypt/decrypt methods as counterparts to Boneh's (see [34, pp. 382] for detail).

The paper points out that the proposed scheme can be viewed as an IBC based on threshold assumption. Knowing the identifier of a particular user and also the public key of the trusted center, one can send encrypted messages and verify signatures. This is equivalent to identity-based encryption and signature. The paper further states that unlike other IBC schemes, the proposal is based on standard (discrete logarithm) assumptions, and thus is much more efficient than these prior IDC schemes.

According to Xu et al. [26], Saxena's scheme is arguably subject to Sybil attacks. Besides, the scheme publishes per-node parameter w_i to all nodes to compute public key of user i, which is similar to certificate-based schemes and against advantages of IBC.

8) A PKI-IBC Hybrid Key Management Scheme: Traditional PKI is based on PKC. In MANETs, because the computational and communication resources required by PKC operations are very limited, and also a centralized CA is not reliable, traditional PKI is regarded as being unsuitable. By applying IBC, new hybrid PKIs can be setup and adapted to MANETs.

In [36], [37], Lin et al. identify the difficulty of applying traditional PKI security architecture to MANET. They suggest the use of a hybrid architecture that combines the good sides of both traditional PKI and IBC, and propose a cluster-organized key management scheme.

Based on former work of [7], [38], [16], [5], they propose a key management scheme and integrate it into secure routing protocols. The proposed network framework is a two-layer hierarchical structure performing key generation, key distribution, and storage. The bottom layer is responsible for internal cluster domain authentication using IBC, and the upper layer, root CA, is responsible for external cluster domain authentication.

In every cluster domain, cluster heads only maintain identities of members, without needs to store and distribute public keys. The cluster head serves as the PKG for cluster members. When a node joins the network, it is given a master public-key belonging to a cluster domain. Furthermore, each node also applies for a personal private-key from its cluster domain head, and uses it to achieve routing packets and messages encryption/decryption capability. The identity-based key generation and distribution use Boneh's algorithms.

The authors state that the simulation results demonstrate that the scheme can reduce computing loads of central CA and key repositories. However, at the same time, the scheme adds much additional overhead to inter-cluster communication.

9) Techniques to Improve Threshold Key Generation: As there is no infrastructure in MANETs, and only error-prone and dynamically changing wireless links for communication exist, high availability is of great importance in MANETs. Threshold is employed to eliminate single-point of failure, and enable PKG service in a dynamically changing environment.

[4] $H_3(\hat{e}(s_0 X, P_0)) = H_3(\hat{e}(s_0 x P, P_0)) = H_3(\hat{e}(P_0, P_0)^x)$

There are proposals in the literature to improve security and availability of threshold cryptography.

- Improved Sigature Scheme for Threshold Secret Shares: To prevent forgery and ensure integrity of secret shares exchange messages in threshold key generation, Crescenzo et al. in [39] propose a scheme to sign those messages using modified "BLS Signature Scheme' from Boneh et al. [8]. In their scheme, they suggest hashing the concatenation of the message and various other parameters; specifically, the threshold parameter, the group size and the indices associated with the parties taking part in this execution of a threshold signature protocol: Let $\langle \mathbb{V}, \mathbb{E} \rangle$ denote the connection graph over network nodes, $\mathbb{T} = \{i_1, ..., i_l\}$ a subset of $\mathbb{V}$ that is requested by a node to provide a threshold signature for message M, and t the positive integer the threshold requested by the node, set $m' = M \| t \| l \| i_1, ..., i_l \| c$, $m = H(m')$. Then feed m as the input to BLS singature scheme. Thus, not only the message but also the secret share parameters are protected by the signature.
- Verifiable Secret Sharing (VSS): Many proposals employ Feldman's *Verifiable Secret Sharing (VSS)* [35] to verify integrity of secret shares of threshold cryptography. For example, in [39], Crescenzo et al. employ a share verification process in their distributed key generation protocol: each party P_i randomly chooses $a_{i0}, ..., a_{it} \in \mathbb{Z}_q$, defines polynomial $p_i(x) = a_{i0} + a_{i1}x + ... + a_{it}x^t$ (where the operations are performed over $\mathbb{Z}_q$), computes $s_{ij} = p_i(j)$ mod q for $j = 1, ..., n$, and computes $A_{ik} = g^{a_{ik}}$ for $k = 0, ..., t$. Each P_i sends A_{ik}, for $k = 0, ..., t$, to all parties and s_{ij} secretly to participant P_j . Then each party P_j verifies the shares received from other parties by checking that, for $i = 1, ..., n$, $g^{s_{ij}} = A_{i0}A_{i1}^{j}A_{i2}^{j^2}...A_{it}^{j^t}$.
- Partial Secret Shuffling: Kong et al. [40] notice that in traditional threshold cryptography, partial secret shares are broadcasted as $SS_{ij} = S_i \cdot l_i(j)$, and S_i can be derived from the share since Lagrange coefficients $l_i(j)$ can be publicly calculated. They suggest a random nonce be exchanged between any two members in the coalition (PKG coalition in our context). The entity with larger ID treats the nonce as a positive number while the other side treats it as a negative number. Sum of partial secret shares and nonces are sent instead of actual partial secret shares. The nonces are canceled out at the real destination node and partial secret shares are recovered.

By properly combining the above methods, one can obtain a more reliable threshold cryptosystem scheme for key generation.

B. Group Key Generation and Agreement

In cases when a message is intended for every node in a group, using public/private keys and pairwise communication generates trenmendous traffic. A symmetric group key minimizes the traffic bandwidth, and is more efficient. The advantage of the group broadcast key is that it at most needs only n private keys to be generated and distributed to n nodes, whereas pairwise communication schemes need $n(n-1)/2$ and $n(n-1)$ respectively.

A group key can be generated by one member of the group and distributed to other members. Group key can also be contributed and agreed by multiple members. A group key can be either dynamic, which means in each broadcast message the group key is different; or static, which means the group key does not change in each broadcast message once it is determined. In this subsection, we classify group key generation and agreement schemes based on these criteria.

1) Dynamic Group Key Generation Based on Node-specific Broadcast Secret: If the members of a group of nodes share a secret that is unknown to non-members, it is intuitive that they can generate a share group key based on this secret. Many group key generation schemes are based on this idea. The differences only lie in how the shared secret is generated and how it is distributed to members.

Bohio et al. [41] propose a non-probabilistic method for computing unique broadcast keys for different groups. Based on the work of [42], they use identity-based pairwise symmetric keys as the building block for their broadcast scheme. They state such keys are computed non-interactively by the nodes, which reduces communication overhead and simplifies key management in pairwise communication.

The group key is generated in this way: Let K_{1N} be the broadcast secret of node 1 for any group of N nodes. Node 1 computes its broadcast parameter $P_{1-brdcst}$ as: $P_{1-brdcst} = K_{1N} \cdot Q_{id_1}$, and distributes it to all candidate nodes using respective pairwise encryption. To sign and encrypt a message M, node 1 computes:
$h = H_3(M)$, where $H_3 : \{0,1\}^* \rightarrow \{0,1\}^*$;
$K_{1-brdcst} = H_2(\hat{e}(Q_{id_1}, P)^{(r+h)})$, where $r \in Z_q^*, H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^m$; $C = M \oplus K_{1-brdcst}$, $U = rP$, $V = K_{1N}^{-1}(r+h)P$.
The broadcast message is: $\langle C, U, V \rangle$. Every node in the group can compute the same broadcast key $K_{1-brdcst}$ as node 1 from $H_2(\hat{e}(P_{1-brdcst}, V))$ and decrypt the message from the cipher text C as: $M = C \oplus K_{1-brdcst}$; After decrypting message , its hash can be computed as: $h = H_3(M)$, and authentication is verified by checking if $\hat{e}(K_{1N}Q_{id_1}, V) = \hat{e}(Q_{id_1}, U + hP)$ holds.

In [43], Bohio et al. continue their work and indicate that the use of pairwise communication creates additional bandwidth overhead in case of broadcast messages. They propose an authenticated broadcast scheme based on symmetric keys and a corresponding signature scheme. Based on work of [7] and their former work [41], the authors extend pairwise shared key generation method proposed in [44] – $K_{AB} = K_{BA} = \hat{e}(Q_{id_A}, sQ_{id_B})$, and propose a method for computing collision-free broadcast keys that can be used for different groups in the network and changed as the group membership varies. Such keys can be useful in the context when it is important to have all the broadcast keys unique without causing additional handshake between the nodes.

Compared to [41], the authors simplify the scheme as: Node 1 computes its broadcast parameter $P_{1-brdcst}$ as: $P_{1-brdcst} = K_{1N} \cdot P$, and distribute it to all candidate nodes using respective pairwise encryption. Every node will then compute the broadcast key of node 1 as $K_{1-brdcst}$ using the hash function $H_3 : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow (0,1)^m$. The key $K_{1-brdcst} = H_3(P_{1-brdcst})$. To generate unique broadcast secret K_{1N} for node 1, let $D_{1N} = \hat{e}(sQ_{id_1}, Q_{id_2} + Q_{id_3} + \cdots + Q_{id_n}) = \hat{e}((sQ_{id_1}, Q_{id_2}) \cdot \hat{e}((sQ_{id_1}, Q_{id_3}) \cdots \hat{e}((sQ_{id_1}, Q_{id_n})$ and $K_{1N} = H_2(D_{1N})$. Further, the authors use this group key to sign group messages M: $\langle U, V \rangle = \langle rQ + id_1, K_{1N}^{-1}(r+h)Q_{id_1} \rangle$ where $r \in Z_q^*$, $h = H_4(M)$. And the receiver can verify if $\hat{e}(P_{1-brdcst}, V) = \hat{e}(P, U + hQ_{id_1})$ holds.

The authors point out one potential problem of this scheme is that it might be possible for malicious nodes to generate computational overhead for other nodes by sending unnecessary broadcast messages. The countermeasure is the non-repudiation and authentication provided by the signature in the scheme.

In [45] – the extended version of [41] and [43] – the authors reiterate their scheme to generate collision-free broadcast keys for different groups and an authenticated broadcast scheme based on symmetric keys and a corresponding signature scheme. On the basis of the former two papers, the authors present two varieties of their former scheme to generate group keys hidden to the TA:

The first scheme is based on group identity. A group public key Q_{GRP-ID} is to be generated by the TA based on any group identity or arbitrary string. The TA, using its master keys, then computes the initial group key $D = s \cdot Q_{GRP-ID}$. Every node i will then receive the point D from the TA and will generate its private key k_i, a random secret, and compute the corresponding public key as $D_{i-pub} = k_i \cdot D$. All such individual public keys should be available from the TA. The participating nodes then get the public key of every node from the TA.

For the broadcast key, parameter $P_{1-brdcst} = K_{1N} \cdot P$ is computed as in the basic scheme with K_{1N} being any random secret. The signature scheme would be used as in the basic model.

The second scheme is based on individual identity. The TA will compute the partial private key of any node i as $D_i = s \cdot Q_{id-i}$. Node i computes its private key as $k_i = H_3(x_i \cdot D_i)$, where x_i is a random secret chosen by node i. It computes public key as $D_{i-pub} = k_i \cdot P$, and submits it to the TA. The pairwise and broadcast keys will be computed similarly as the first scheme does.

It has been pointed out in [46] that the above signature scheme is vulnerable to the universal forgery attack that an adversary can forge signatures on any message.

For group key generation scheme based on broadcast secret, one issue is how the broadcast secret is distributed to other nodes in the group. If it is distributed by broadcasting, the issue turns to be scalability problem. Each node generates a group key secret and broadcasts it to other nodes. The number of messages and storage space are both $O(n)$, the broadcast traffic is $O(n^2)$ (each of n nodes relays n messages). If n is too big, the scheme does not work. If the group broadcast secret is distributed using respective pairwise communication, it requires an existing secure routing mechanism. The issue turns to be interdependency cycle problem between secure routing and security services. Another issue is that each node generates a broadcast secret and distributes it to other nodes in the group. This is against the advantages of IBC schemes.

2) Static Group Key Agreement Based on Diffie-Hellman Key Exchange: An approach to address the scalability problem is "recursive subgrouping" – dividing a large group to subgroups again and again, each subgroup contains a small number of sub-subgroups until a small number of members reached. For these small number of members, there are already key exchange protocols ready to use, e.g., 2-party or 3-party Diffie-Hellman key exchange protocol.

Chien et al. in, [47] and [46], propose a group key agreement protocol in this approach. The authors base their work on [48], [49], [45], and apply IBC to these schemes. In their scheme, they divide the whole group into several cell groups and a control group, and each cell group is managed by its cell group controller independently of the other cell groups. Nodes within the same cell group share a cell group key, which can be generated by a distributive or contributory way.

They provide two versions of pair-wise key agreement: one is static and the other is dynamic. The static one uses the same static pair-wise key as Bohio-Miri's scheme [45]. The dynamic one, contrary to Bohio-Miri's scheme, is certificateless. The protocol works as follows: $A \rightarrow B : P_A = aP$, $B \rightarrow A : P_B = bP$, where a, b are random numbers. Then A and B independently compute a common session key based on P_A and P_B.

On the basis of the pair-wise communication, they propose a *Tripartite key agreement protocol* which allows three parties establish their session keys. The scheme is modified from Hess' signature [50] for traditional public key setting. The protocol has two rounds. In the first round, the entities broadcast their ephemeral public keys, e.g. $A \rightarrow B, C$: $\langle sid, ID_A, ID_B, ID_C, P_A, P'_A \rangle$, Node A computes $P_A = aP, P'_A = a'P$, where a and a' are random numbers chosen by node A, sid is session id. In the second round, the entities broadcast their confirmation (signatures) on the session and ephemeral public keys, e.g. $A \rightarrow B, C$: $\langle sid, v_A, u_A \rangle$, Node A computes $m_A = H_3(sid, ID_A, ID_B, ID_C, P_A, P'_A, P_B, P'_B, P_C, P'_C), r_A = \hat{e}(P, P)^{K_A}, v_A = H_4(m_A, r_A)$ and $u_A = v_A S_A + k_A P$, where K_A is a random number chose by node A. B and C broadcast similar messages. Then A checks whether the following two equations hold: $v_B = H_4(m_B, \hat{e}(u_B, P) \cdot \hat{e}(Q_B, P_{pub})^{-v_B})$ and $v_C = H_4(m_C, \hat{e}(u_C, P) \cdot \hat{e}(Q_C, P_{pub})^{-v_C})$ [5]. After authenticating the message from the other two nodes, A, B, and C share these session keys: $K^1_{A,B,C} = \hat{e}(P_B, P_C)^a, K^2_{A,B,C} = \hat{e}(P_B, P'_C)^a, K^3_{A,B,C} = \hat{e}(P'_B, P_C)^a, K^4_{A,B,C} = \hat{e}(P'_B, P'_C)^a$, $K^5_{A,B,C} = \hat{e}(P_B, P_C)^{a'}$, $K^6_{A,B,C} = \hat{e}(P_B, P'_C)^{a'}$, $K^7_{A,B,C} = \hat{e}(P'_B, P_C)^{a'}$, $K^8_{A,B,C} = \hat{e}(P'_B, P'_C)^{a'}$.

The tripartite key agreement scheme can be easily extended to share n^3 keys by sending n ephemeral public values per node. The scheme then uses the ternary tree and bilinear map to establish the cell group key. Hierarchical ternary tree is a hierarchical tree, where the degree of a node is at most three. The keys corresponding to the key nodes are generated iteratively from bottom up to the root node, and the key

[5] $\hat{e}(u_B, P) \cdot \hat{e}(Q_B, P_{pub})^{-v_B} = \hat{e}(v_B sQ_B + k_B P, P) \cdot \hat{e}(Q_B, sP)^{-v_B} = \hat{e}(sQ_B, P)^{v_B} \cdot \hat{e}(k_B P, P) \cdot \hat{e}(sQ_B, P)^{-v_B} = \hat{e}(P, P)^{k_B}$

corresponding to the root node is taken as the group key. If a node has three child nodes, then the tripartite key agreement scheme is adopted; otherwise, the two-party key agreement scheme is adopted.

This scheme addresses the scalability issue by subgrouping, but is subject to these weaknesses: first, each node generates an ephemeral key and distributes it to group members, which is against advantages of IBC. Fortunately, the group is always of size 2 or 3 members; second, key exchange messages use respective pairwise communication, which requires an existing secure routing mechanism.

3) Static Group Key Agreement Based on Broadcast Ephemeral Keys: Characteristics of MANETs make it difficult to generate a group key. Zhang's constant-round contributory key agreement scheme [51] avoids the two obstacles for contributory key agreement in MANETs: authenticating the exchanged information without an online Trusted Third Party (TTP), and resistance to unstable links.

Using the IBC scheme of Boneh et al. [7], the authors revised the constant-round key agreement scheme proposed by Lee et al. [52] that was on password-based. In round 1 of the new scheme, each node generates an ephemeral key $N_i \in \mathbb{Z}_q^*$, computes $z_i = N_i P$, and signs it using the signature scheme of Du et al. [53]: $T_i = H(z_i)sQ_i + N_i P_{pub}$. The node then broadcasts them with its ID:$\langle z_i, T_i, ID_i \rangle$.

In round 2, each of the group member firstly verifies $\hat{e}(\sum_{j\in\{1,\cdots,n\}\backslash\{i\}} T_j, P) = \hat{e}(\sum_{j\in\{1,\cdots,n\}\backslash\{i\}}(H(z_j)Q_j + z_j, P_{pub})$. Then group members are divided into two subgroups. Only one subgroup broadcasts messages, and two subgroup keys are generated once a time. Each node computes a group key based on two sub-group keys. In short, for every group key's information exchange at round 2, it only needs about half of group members to take part in, while all members can compute out the same session keys according to the broadcasted messages. This group is divided into two subgroups, and as long as one of these two subgroups does not meet with the link failures, this scheme will succeed.

This scheme requires an ephemeral key for each node which is stored on all other nodes. This is a drawback inherited from certificate-based cryptography, and is against the advantages of IBC.

4) Static Group Key Generation Based on Identity-based Broadcast Encryption: Zhang et al. [54] propose another group key generation protocol that is quite different from the above schemes. The scheme is based on Identity-based broadcast encryption (IBBE) scheme [55]. In IBBE, one public key can be used to encrypt a message to any possible group of identities.

The proposed scheme only requires each group member to broadcast one message to set up the group key. Compared to Bohio's scheme, this scheme does not require a node to store any temporary or pseudo public key of other nodes. Compared to above schemes, the scheme does not require secure routing for key exchange message, because all messages are broadcasted. However, the group key generation is static and not suitable for dynamic networks, such as MANETs, because it requires all members be determined before protocol starts. In case of membership changes, for example, one member leaves or one new member joins, all members must start the process again. Besides, like other group key schemes discussed above, IBBE group keys are symmetric keys; but unlike them, IBBE is not integrated with any asymmetric private/public key scheme. A different set of parameters and algorithms is needed for asymmetric private/public keys generation which is indispensable for authentication and non-repudiation.

C. Discussion and Comments

Table II summarizes the main characteristics and weaknesses of the master key and private key generation and distribution schemes.

Table III summarizes the main characteristics and weaknesses of group key generation and agreement schemes.

Key management is an essential and fundamental service for ad hoc networks. Secure keys should be set up before other services can start. This can be achieved by pre-distribution of keys in network initialization phase. One advantage of IBC key management is that it saves storage and transmission of public keys and certificates. Many IBC key management proposals suggest generating master key and private keys online. There is a problem in this case. Consider the following scenario: we need to find a key management scheme to design a secure routing protocol. Since there is no routing for unicast, the only way to distribute keys or key shares is broadcast which is not secure. It turns out to be a group key agreement problem, and the group key agreement protocol cannot use unicast routing at that time. Thus key management should not rely on any other online service if keys are generated online. Unfortunately, many IBC key management schemes in the literature do not comply with this rule—they rely on secure routing or online administration mechanisms (e.g. out-of-band communicant, side channel) to generate or distribute keys. We will recall and discuss this problem further in Section IV shortly.

Another issue that needs to be noted for schemes in which a master key is generated in a distributed manner (e.g. [19]) is Byzantine attacks. These schemes need an initial policy negotiation process that is a potential target for Byzantine or active adversaries. The system may be totally taken over by adversaries. For other schemes in which a TA is responsible for the master key generation, this issue does not exist.

For group keys, static group keys are less secure than dynamic group keys, while the latter takes more communication bandwidth in each message. In group key generation/agreement proposals, some use pairwise communication and unicast routing. Key generation/agreement messages are distributed via pairwise communication which relies on unicast routing. This leads to the problem of interdependency cycle between security services and secure routing, e.g., in [45], the group broadcast key is distributed to all candidates using respective pairwise encryption. This process requires an existing secure routing mechanism.

In both master key and group key generation proposals, one problem is the use of temporary or ephemeral public keys: One node generates a temporary or ephemeral public key and distributes it to other nodes. Other nodes then need to store it for later use. This process is more like the way a certificate-based cryptosystem works. It is inconsistent with the essence of IBC, and offsets the advantages of IBC.

TABLE II
SUMMARY OF MASTER KEY AND PRIVATE KEY GENERATION AND DISTRIBUTION SCHEMES

Year	Publication(s)	Main Idea & Contribution(s)	Online /Off-line TA	PKG	Key Share Distribution	Weaknesses
2003	[19]	Idea of applying IBC and threshold cryptography to secure ad hoc networks	No	Fully distributed	Secure channel	1. Technical details of key generation are not given. 2. Routing-security interdependency cycle. 3. Threshold cryptography weaknesses. 4. The network initialization stage is vulnerable to Byzantine failures.
2004	[21], [22]	A complete implemention of Khalili's Scheme	No	Fully distributed	Temporary PKI	1. Routing-security interdependency cycle. 2. Threshold cryptography weaknesses. 3. The network initialization stage is vulnerable to Byzantine failures.
2004	[33]	Secure Key Issuing Protocol Using Key Privacy Authorities	Offline	Partially distributed	Not mentioned	1. All KPAs are required to be online and available, which is not feasible in MANETs. 2. Secure routing is required to get partial key and signature, which is in routing-security interdependency cycle.
2005	[28]	Multicast group of PKGs; Key proxy.	Offline	Partially distributed	Encrypted Multicast	1. Routing-security interdependency cycle. 2. Distributes partial private keys of PKG server nodes to the network.
2005	[30]	Offline threshold D-PKG	Offline	Partially distributed	Pre-distribution	1. Routing-security interdependency cycle. 2. More subject to mobile attacks. 3. Distributes share commitments of D-PKGs
2005	[34]	Lightweight IBC	Yes	Partially distributed	Not mentioned	1. Subject to Sybil attacks [26]. 2. Routing-security interdependency cycle
2006	[4]	Compromise-tolerant Key Generation	Yes	Partially distributed	Not mentioned	1. Routing-security interdependency cycle problem, or extra broadcasting traffic overhead. 2. Poor scalability.
2007	[31]	Use of the blind signature to ensure the secure issuing of the private key shares in public channel	Yes	Partially distributed	Public channel	1. Distribution and storage of password for each node. 2. Additional traffic of broadcasting.
2008	[23]	Another IBC and threshold cryptography implementation.	No	Fully distributed	Not mentioned	Routing-security interdependency cycle
2008	[24]	Implementation of Deng's scheme in OLSR routing protocol	No	Fully distributed	Self-generated public/private key pair	1. Each DPKG node has to store in memory the temporary public keys of other DPKG nodes 2. Master public key and master private key collection process is not secure, because only public channels are available at this stage. 3. Does not provide any security protection for OLSR routing protocol it relies on. 4. Routing-security interdependency cycle
2006	[36], [37]	A PKI-IBC hybrid key management scheme	Yes	Fixed on cluster head	PKI	Additional overhead for inter-cluster communication.

TABLE III
SUMMARY OF GROUP KEY AGREEMENT SCHEMES

Year	Publication(s)	Main Idea & Contribution(s)	Using unicast routing	Static/ Dynamic	Rounds	Weaknesses
2004	[41], [45], [43]	A method for computing collisionfree broadcast keys; Use of signatures in broadcast messages.	Yes	Dynamic	2	1. Distribution of the broadcast secret leads to either routing-security interdependency cycle or scalability problem. 2. Node specific secret is against IBC advantages.
2005	[51]	Authenticating the exchanged information without online TTP; Resistance to unstable links	No	Static	2	In round 1, each node generates an ephemeral key and broadcast it.
2008	[47], [46]	Subgrouping a 2-party/3-party Key Agreement	Yes	Static	2	1. Each node generates an ephemeral key; 2. Key exchange messages use respective pairwise communication, which requires an existing secure routing mechanism.
2008	[54]	Set up the group key in one round based on IBBE	No	Static	1	1. Not suitable for dynamic membership. 2. Not integrated with any asymmetric private/public key scheme.

IV. SECURE ROUTING PROTOCOLS USING IBC

Routing in MANETs enables packet delivery from one node to another by way of intermediate nodes. It is the fundamental issue considered in MANETs, thus secure routing is a fundamental issue in MANET security. Secure routing ensures successful routing among authentic nodes with adversary nodes existing around or inside the network, and forms the bedrock of a secure MANET system. An important application of IBC in MANETs is to design secure routing protocols. Generally, compared to traditonal cryptosystems, IBC provides the following advantages in terms of secure routing:

- IBC improves efficiency of secure routing. Once secure keys are avaiable, IBC can be applied to either on-demand routing protocols like DSR, or link state routing protocols like OLSR. The routing messages encrypted and signed by the sender and decrypted and verified by the receiver using IBC. To protect routing messages, on same security level, IBC encryption/decryption schemes are faster, and IBC signature is shorter.

- IBC eases the process of key distribution. Key exchange messages can be spared. Pairwise keys are available with only a few security parameters distributed at the network deployment phase, which is not possible with traditional symmetric key or PKI cryptosystems. The sender and receiver share a default pairwise key $K_{AB} = \hat{e}(d_A, Q_B) = \hat{e}(d_B, Q_A) = K_{BA}$ without any extra distribution and storage of keys. This is critical to routing protocols because until routing is set up there seems no way to distribute or negotiate secret keys among nodes. Traditional symmetric or asymmetric cryptography requires a large amount of keys to distribute and store.

Depending on what encryption/decryption and signature/ verification schemes are used, and what routing protocols are used, there are various secure routing proposals using IBC.

A. Securing On-demand Routing Protocols

Lee, Kim, Chung and Yoon [56] apply previous IBC schemes [7], [57] to a DSR routing protocol.

In their routing protocol, the format of a route request packet is $\langle RReq, SourceID, DestinationID, seq, Sign_S(M), (IntermediateIDList), W, U, V\rangle$, where $M = \langle RReq||SourceID||DestinationID||seq||W\rangle$, and $Sign_S(M)$ is a signature algorithm from [57]. Assume $Q_i = n_i \cdot P$ is the public key of a node (n_S for the source node, n_D for the destination node.[6]), and $d_i = s \cdot Q_i$ is its private key, the source node computes W, U, V as follows: It generates a random string $\sigma_S \in \{0,1\}^n$, and computes $r = H_3(ID_{Source}, \sigma_S)$; Using r and its private key $d_S = s \cdot n_S \cdot P$, it computes: $g = \hat{e}(P,P), \hat{e}(rP, d_S) = g^{r \cdot s \cdot n_S}$. Then $W = rP$, $U = g^{r \cdot s \cdot n_S} \times \sigma_S$, $V = (\hat{e}(sP, Q_{Dest}))^r \oplus r = g^{r \cdot s \cdot n_D} \oplus r$.

An intermediate node i that receives route request packet verifies the signature value. If it is correct, node i adds ID_i to the $intermediateIDList$, computes the new value of U by: $U = U \times \hat{e}(rP, d_i) = U \times g^{r \cdot s \cdot (n_S + \ldots + n_i)}$, and then rebroadcasts the packets generated.

A destination node D that receives routing request packet and whose ID is matched to value of $DestinationID$ field in the packet performs the following procedure [7]: computes r' using private key of D and the values of packet received: $r' = V \oplus (\hat{e}(sP, Q_{Dest}))^r = V \oplus \hat{e}(W, s \cdot n_D \cdot P)$, gets the public key $Q_i = H_2(ID_i)$ of ID_i that are described in $intermediateIDList$ and computes $A = \hat{e}(sP, \sum_{i=1}^{k} Q_i)^{r'} = \hat{e}(sP, \sum_{i=1}^{k}(n_i \cdot P))^{r'} = g^{r' \cdot s \cdot \sum_{i=1}^{k} n_i}$. Using A value, D computes $\sigma' = U \times A^{-1}$, and compares r' and $H_3(ID_S, \sigma')$. If the two values are equal, D makes route reply packet as $\langle RRep,\ seq,\ (ID_S, ID_1,\ \ldots,\ ID_k, ID_D), W, V \oplus \sigma',\ Sign_D(M)\rangle$, where $M = \langle RRep||$ $seq||ID_S||ID_1||\ \ldots\ ||ID_k||ID_D||W||V \oplus \sigma'\rangle$.

After receiving the route reply packet, the intermediate nodes in routing path and source node S verify the signature of D. And if it is correct, they add the path in the packet to their route cache.

The authors then analyze the security of their protocol. They point out a weakness of their new protocol: An attacker can do resource consumption attacks using invalid packets. They suggest that an attack be prevented by using other network features such as counting number of packets per some duration and additional policy.

The weaknesses of the scheme are: it is subject to wormhole attacks [58], and misses a key management scheme.

B. Concatenated Signature for Intermediate Node List in On-demand Routing Protocols

Park, Myung and Lee [59] base their work on [7], [60], and apply IBC to on-demand routing protocols.

Their protocol is similar to [56], but the signature and verification procedures are different:

When the source node sends $RReq$ to intermediate nodes, the packet format is: $\langle RReq||ID_S||(r_S, Z_S)||Sign_S(H(M))\rangle$, where $M = \langle RReq||ID_S||(r_S, Z_S)\rangle$, $r_S = H(\hat{e}(P, sP)^x||Q_S||RReq)$, $Z_S = xP_{pub} - r_S d_S = xsP - r_S s Q_S$, x is a random number.

An intermediate node X_i computes $k' = \hat{e}(P, Z_S) \cdot \hat{e}(sP, Q_S)^{r_S} = \hat{e}(P,P)^{xs}$ for the authentication of the node that sends the message, and it checks $r_S = H(k'||Q_S||RReq)$. If the verification is successful, the intermediate node can trust the received message and then it computes r_X and Z_S similarly, and broadcasts the message to the next node as: $\langle RReq||ID_S||ID_X||(r_S, Z_S)||(r_X, Z_X)||Sign_S(H(M))\rangle$.

When the destination node receives this message, it checks the destination address. If the destination address is the same as its address, it verifies the signature, (r_S, Z_S) and (r_X, Z_X). If the verification process is successful, it is ready to reply a message. The destination node sends a $RREP$ message to the source node. After passing intermediate nodes the reply message is like:
$\langle RRep||ID_D||ID_X||(r_D, Z_D)||(r_X, Z_X)||Sign_S(H(M'))\rangle$.

Park and Lee [61], Park, Myung and Lee in [62], Lee and Sriborrirux [52] present similar results separately.

These schemes have the following weaknesses: A key management scheme is missing; They do not have good scalability, since message signature is concatenated and can be quite large.

C. Aggregated Signature for Intermediate Node List in On-demand Routing Protocols

The concatenated signature of an intermediate node list can be very large, Song et al. [63] apply identity-based multi-signature to routing protocols and propose an authentication mechanism with aggregation signature, based on the work of [64], [42].

In their scheme, an aggregate signature can be generated on distinct messages: assume $\sigma = (U, V)$ is the signature on messages $M_1, \cdots, M_{i-1}$, and $\sigma = (U', V')$ is the signature on message M_i, $U = rQ_{ID_i}, h = H_1(M_i), V = (r+h)d_{ID_i}$. The aggregator verifies that M_i is different from any other messages. If it is true, it computes: $U = U + U' \in \mathbb{G}_1$,

[6] n_i is only a helper for explanation purpose here, and is unknown to any node.

[7] with correction to the original paper

$V = V + V' \in \mathbb{G}_1$. Then $\sigma = (U, V)$ becomes the aggregate signature on $M_1, \cdots, M_i$. The destination can verify the validity of the aggregation signature: Given identities $ID_1, ..., ID_n$, distinct messages $M_1, ..., M_n$, and an aggregate signature $\sigma = (U, V)$, the verifier computes $h_i = H_1(M_i)$ for all $1 \leq i \leq n$. Then it checks whether $\hat{e}(\sum_{i=1}^{n} h_i Q_{ID_i} + U, P_{pub}) = \hat{e}(\sum_{i=1}^{n} [(h_i + r_i)Q_{ID_i}], P_{pub}) = \hat{e}(\sum_{i=1}^{n} [(h_i + r_i)d_{ID_i}], P) = \hat{e}(V, P)$ holds. If it is true, all the signatures are valid.

They then demonstrate in the paper the use of this scheme in on-demand routing protocols such as DSR and AODV, which is similar to [56].

This scheme is subject to wormhole attacks [58], and misses a key management scheme.

D. A Security Architecture to Secure OLSR

Adjih et al. [65] propose a security architecture to secure OLSR using IBC.

Their proposal is based on the work of [42], [8]. In their scheme, an (offline) TA is in charge of certifying or assigning keys of each node participating in the trusted network. Each node joining the network will have the public key of the TA. This key is denoted the global key. Later, any node entering the ad-hoc network could diffuse its public keys, with a specific key exchange protocol, with proper parameters and signatures. The key which is used later to sign message is called the local key, and can be either its global key, or newly generated private/public keys. A node would start originating OLSR control messages, signing them using the local key with a specific extension which prepends a special signature message.

Technical details of the scheme are not given in the paper, e.g. how keys are generated and distributed, how packets are signed and encrypted.

E. A Key Management Integrated OLSR Routing Protocol

Most routing protocols do not consider key management issues. In [66], Zhao and Aggarwal propose a secure routing protocol integrated with key management. Based on previous work of [7], [67], [29], and using proposed proactive security approach, they design a secure routing protocol for pre-planned MANETs.

The network starts with initial nodes. The first phase is routing setup. Initial nodes contact and get system secret from an off-line official administrator. With the system secret, the nodes communicate with each other securely and set up routing table. The second phase is secret update. Since routing is already set up, initial nodes can communicate with each other securely using pair-wise session key and contribute to a new secret. System secret can be updated periodically or when necessary.

When node A sends a routing message, including *HELLO* and *TC* message, it encrypts and authenticates the message as follows:

1) *Encrypting the message:* The entire message, M, is encrypted using a symmetric encryption function E. The symmetric encryption key is calculated as: $k = H_1(g^r)$, where $g = \hat{e}(d_A, P)$, r is a random number in $\mathbb{Z}_q^*$ $(r \cdot Q_A \neq \infty)$. g can be stored in the node's memory for future use until secret update. The encrypted message $E_k(M)$ is put in the message field.
2) *Signing the message and message header:* A signature is calculated over the message header except the *Time To Live (TTL)* and *Hop Count (HC)* fields, and the encrypted message. Assume the encrypted message to be signed is M_1, the secret authentication key is calculated as $K_1 = H_2(M_1, g^r)$. The authentication code $\sigma = HMAC(K_1, M_1)$ and $r \cdot Q_A$ are appended at the end of the message.

OLSR packet is also signed and encrypted, and verified at each hop with *TTL* and *HC* fields recalculated. Authentic intermediate nodes and destination node can decrypt and verify the packets by computing the key $g^r = \hat{e}(Q_A, P_{pub})^r = \hat{e}(r \cdot Q_A, P_{pub}), k = H_1(g^r)$.

This scheme addresses routing-security interdependency cycle, by way of secret pre-distribution, which is not a problem in pre-planned, or so-called authority-based, MANETs.

F. Discussion and Comments

A routing protocol must satisfy basic security requirements mentioned in Section I. There have been some routing protocols for MANETs in environment with adversary nodes, which do not rely on secure keys, e.g.: Marti et al. [68] uses a watchdog to monitor behavior of nodes and a pathrater to find routes among nodes trustworthy; Buchegger et al. proposes CONFIDANT protocol [69] that rewards nodes forwarding packets and punishes nodes not forwarding packets; Michiardi et al. proposes a reputation mechanism that extends pathrater [70] to more protocols and improves security by disallowing negative rating. These routing protocols mainly aim at improving routing availability, and do not provide authentication of node's identity, confidentiality, integrity, freshness, and non-repudiation of routing messages, which rely on use of secure keys. To meet all of these requirements, a cryptosystem with a unique private key for each entity is required. However, from this and the previous sections, we can see that many key management schemes assume a secure routing is available; at the same time, many secure routing schemes assume secure keys are already available. This chicken-and-egg-like paradox is noted as routing-security interdependency cycle. The right way to break the cycle is to have a key management not relying on secure routing, as suggested by Hegland et al. in [71], because secure routing should not be working without secure keys.

Besides the attacks mentioned in Section I, routing protocols are subject to many other attacks especially targeted to the network layer, e.g.:

- Wormhole: An adversary receives packets at one point in the network, "tunnels" them to another point in the network, and then replays them into the network from that point.
- Blackhole: An adversary uses the routing protocol to advertise itself as having the shortest path to the node whose packets it wants to intercept. A forged route can then be created.

IBC provides many advantages in terms of secure routing. Many simulation works from above publications show that

IBC secure routing schemes improve efficiency over counterparts using traditional cryptosystems. However, as many of the above schemes fail to note, we summarize missing points from above proposals which need to be noted in a IBC routing protocol:

- Secure keys should be available before a routing protocol starts working.
- To prevent the above routing attacks, the protocol must encrypt and authenticate every message and packet, not only end-to-end, but also hop-by-hop.
- Each routing protocol has its own weakness. When applying IBC to the routing protocol, pay attention to this. For example, the above routing protocols based on AODV are all subject to wormhole attacks.

Table IV summarizes the main characteristics and weaknesses of IBC routing protocols in MANETs. As an aside, in the network layer, no cryptography-based routing protocol is immune to denial-of-service (DoS) attacks. The adversary can bring the system down by hijacking packets and garbling messages which leads to receivers consuming limited resources on wastes.

V. Applications of IBC in Special-purpose MANETs

Besides key management and secure routing, there are also some other applications of IBC in special MANETs, such as multi-domain or multi-TA coalition networks. These applications are not relevant to general MANETs, thus we deliberately leave out much of the detail in the following.

In [72], Balfe et al. envisage that in IBC infrastructures, entities from multiple TAs might be present within a larger coalition structure, with each TA issuing cryptographic keys to entities in its own security domain. Based on the work of [73], [74], [19], [75], they propose a lightweight, generic and broadly applicable framework enabling the refreshing of privates keys in coalition-forming situations. They point out their contribution is the improvement upon the obvious approach of simply distributing new private keys by encrypting them using the old public keys.

The basic idea is to use a Refresh algorithm to generate a new private key for a given identity and sends it in a Refresh message via a secure way; and use a Recover algorithm for a node to recover its new private key. The Refresh algorithm takes old ID ID_1, new ID ID_2, old private key SK_{ID_1}, time periods t_1 and t_2 for old and new IDs respectively, and new private key SK_{ID_2} as input, and outputs a blinded key: $R = H(ID_1, ID_2, t_1, t_2, SK_{ID_1}) + SK_{ID_2}$. The Recover algorithm recovers the correct new private key as: $SK_{ID_2} = R - H(ID_1, ID_2, t_1, t_2, SK_{ID_1})$.

The authors claim that their scheme is secure and state that the framework is applicable to enable secure interoperation between entities with different trusted authorities in dynamic coalitions environments, and is particularly well-suited to coalition forming in computation and bandwidth-limited MANETs.

In [76], Li et al. consider cross-domain key agreement in multi-domain ad hoc networks. They propose a new IBC scheme based on multiple PKGs, which is more suitable for multi-domain ad hoc networks.

They assume that there are two PKGs—PKG_1 and PKG_2 for two domains, which share the same system paremeters, but have different master private keys. In this situation, the scheme provides encryption/decryption, sign/verify functions between the two domains.

Cai et al. [77] apply IBC to peer collaboration in MANETs. They identify the problem of peer collaboration in ad hoc networks, especially when some peers are autonomous, selfish, or malicious in large-scale, heterogeneous networks. Payment-incited mechanism is an approach for this problem, but most existing electronic payment schemes either rely on online, interactive authorities, or are too heavy for MANETs.

The authors design a lightweight and cheat-resistant micropayment scheme to stimulate and compensate collaborative peers that sacrifice their resources to relay packets for other peers. They base their work on [78], [7]. Their scheme uses identity-based signature and verification mechanisms to achieve authentication and non-repudiation of commitment proposal messages and commitment confirmation messages, and uses hash-chain to count data volume transmitted.

The authors conducted simulations of their schemes. Through simulation results, they claim that when security and collaboration measures are properly enforced, profitable collaboration is a preferable strategy for all peers in MANETs; and with profitable collaboration, system utility increases when peers have maximized their potential profit.

In this section, we have studied applications of IBC in special-purpose MANETs. These applications are only applicable to very limited scenarios, and are not popularly useful.

VI. Weaknesses of and Issues with IBC

We have mentioned many properties of IBC which make it especially attractive for MANETs. However, there are still some problems not completely addressed which impedes application of IBC in MANETs. In this section, we will study “key escrow”, “identity disclosure”, and “identity revocation” problems, and proposals to address them. We deliberately omit those explained in Section III or Section IV.

A. Addressing Identity Disclosure

The main advantage of IBC is that the public key of an entity is its identity that is piggybacked and explicit in the message. This leads to the problem of identity exposure — the identity of any node is exposed to all others. In some MANET systems, this is not desirable, e.g. for those used in battlefield, this may expose the identity of a commander to the enemy, which then enables traffic analysis and incurs great danger.

1) MASK for Anonymous Communications: Zhang et al. apply IBC to anonymous communications in MANETs [79]. The authors identify the problem of malicious traffic analysis in MANETs due to the broadcast nature of radio transmission, and propose an anonymous on-demand routing protocol termed MASK. Derived from work of [7], [80], the protocol enables anonymous communications by allowing neighboring nodes to authenticate each other without revealing their identities.

The PKG pre-calculates a large set of collision-resistant pseudonyms and a corresponding secret point set. During

TABLE IV
SUMMARY OF SECURE ROUTING SCHEMES

Year	**Protocol(s)**	**Main Contribution(s)**	**Routing protocol based on**	**Require-ment not satisfied**	**Weaknesses**
2003	ODSRP [56]	A secure DSR routing protocol using IBC	DSR	Confidentiality, authenticity	1. Missing a key management scheme. 2. Subject to wormhole attacks.
2005	LSRP [59]	Concatenated Signature and verification of routing messages in on-demand routing protocols	On-demand routing protocols	Confidentiality	1. Missing a key management scheme. 2. Message signature is concatenated and can be large.
2005	Multi-signature Routing Protocol [63]	A authentication mechanism with aggregation signature	On-demand routing protocols	Confidentiality	1. Missing a key management scheme. 2. Subject to wormhole attacks.
2005	A Security Architecture to Secure OLSR[65]	The security issues of OLSR, and an architecture including multiple securing mechanisms.	OLSR	Not clear	Details not given.
2009	PAPA-UIC [66]	A secure routing protocol in proactive approach with key management integrated	OLSR	No	Not clear

the bootstrapping phase, a TA distributes system public parameters. Moreover, the TA furnishes each node ID_i with a sufficiently large set PS_i of collision-resistant pseudonyms and a corresponding secret point set. No one but the PKG can link a given pseudonyms to a particular node or identity, or deduce the corresponding secret point with non-negligible probability. Using PS_i and nonces n_1, n_2, A and B can calculate γ pairs of shared session key ($SKey$) and link identifier ($LinkID$) as: $K^{\gamma}_{AB} = H_2(K_{AB}||n1||n2||2 \cdot \gamma)$, $L^{\gamma}_{AB} = H_2(K_{AB}||n1||n2||2 \cdot \gamma + 1)$ (see [79, pp. 1943] for details). Such $\langle SKey, LinkID\rangle$ pairs are unique due to collision-resistant hash functions H_1 and H_2. The $LinkID$s will be used to identify the packets transmitted between A and B and the $SKey$ can be used to encrypt, integrity-protect, or authenticate the content of the packets if needed.

Based on this anonymous neighborhood authentication scheme, the authors propose an improved AODV routing protocol which enables communication between nodes without disclosing the real identity of the node.

The authors evaluate the computation costs of the critical cryptographic operations in their scheme. In this implementation, the routing information is not authenticated, they plan to combine MASK with other secure routing schemes to provide an anonymous yet secure routing protocol.

Weaknesses of this scheme are: First, each node maintains a large set of pseudonyms and the corresponding private keys for each pseudonym. This is resource consuming, and against advantages of IBC. Second, it can only be used in their own routing protocol and not in any other protocol or any higher layer application, because it uses link identifier to transport packets among nodes without using real identities, but there seems no way to convert link identifier's back to identities.

2) General-purpose Identity Hiding Schemes: In [81], Zhao and Aggarwal reiterate the importance of identity protection. The authors propose requirements of an identity hiding scheme for MANETs, and propose three general-purpose identity hiding schemes. The basic idea of the schemes is to encrypt the source and destination IP addresses with a random number using some popular cryptosystem, and transmit the random numbers in the IP header option field.

1) AES-based Scheme (Only show scheme for IPv4 addresses here): Generate a 96-bit random number r for each packet. Append the random number to the IP address and encrypt it with a 128-bit AES cryptosystem: $c = AES_E(m + r, k)$. The first 32 bits of resulted 128-bit output are placed in the IP address, and the rest in the option field. An authentic node decrypts the original 128-bit plaintext as $m + r = AES_D(c, k)$, and gets the 32-bit address.
2) RSA-based Scheme:
   a) The system administrator chooses secret primes p and q and computes $n = pq$, $\phi(n) = (p-1)(q-1)$. n and $\phi(n)$ are distributed to authentic nodes before dispatching.
   b) For each packet, node A chooses a random e with $gcd(e, \phi(n)) = 1$, and encrypts IP address m in the packet as $c \equiv m^e \pmod{n}$. c and e are sent in the packet.
   c) An authentic node computes $d \equiv e^{-1} \pmod{\phi(n)}$, upon receiving the packet, and decrypts c by $m \equiv c^d \pmod{n}$.
3) ElGamal-based Scheme:
   a) The system administrator chooses a large primes n, a primitive root g, and a random integer k (less than n). n, g and k are distributed to authentic nodes before dispatching.
   b) For each packet, node A chooses a random integer r, and encrypts IP address m in the packet as $c \equiv m \cdot g^{k^r} \pmod{n}$. c and g^r are sent in the packet.
   c) An authentic node, upon receiving the packet, decrypts c by $m \equiv c/g^{rk} \pmod{n}$.

The limitation of this scheme is that it requires pre-distribution of identity-hiding parameters.

B. Addressing Key Revocation

Due to the weak physical protection of nodes, node compromises including key disclosures are very likely in MANETs. Meanwhile, the infrastructure for certificate or public key revocation does not exist in MANETs. Frequent key renewals to prevent such compromises are either computationally challenging in solution with distributed on-line key generation or infeasible in solutions with off-line key generation.

1) Appending "Expire Time" to the Identity: In the very beginning of IBC, the public key of node ID was computed as $D_{ID} = H(ID||ExpireTime)$ to allow identity revocation [7], [22], [21], [19].

Hoeper et al. [82] propose a scheme for key revocation and key renewal using an IBC scheme in MANET. This work is based on their former work in [73], and the work of [83], [84]. To enable key renewal in IBC schemes, they introduce a new format for ID-based public keys: $D_{ID} = H(ID||t_i||v_i)$, where t_i denotes the expiration date, and v_i is the version number. The version number always starts with 1 for every new expiry date and is incremented with each key renewal for the same date.

New keys can be issued for the same identity after the previous key has been revoked. And new nodes that join the network can learn about past accusations and revocations. Upon receiving a new key pair and re-joining the network, a node only needs to broadcast its new public key to m-hop neighborhood. The receivers update the version number in their revocation lists accordingly and set all accusation values for this node to zero. The level of security can be chosen as performance trade-off.

These proposals append extra information to an identity to generate a public key. This seemingly tiny change in public keys leads to some complications in MANETs: It was first intended for Internet applications where arbitrary identities are accepted, e.g. email services, and works well there. As in the network layer of MANETs where identities are usually fixed, such as MAC addresses or IP addresses, a public key can no longer be derived directly from the identity of a packet; so that a separate field in each packet is needed to indicate the public key. Furthermore, this scheme requires precise synchronization among all network nodes, which is difficult to achieve in a MANET environment.

2) Long-term and Short-term Identities: Zhao et al. [66] notice the weaknesses of previous solutions, and propose to subdivide identities (IP addresses) into two groups: long-term addresses that do not expire and are implicitly valid until explicit revocation, and short-term addresses that expire after a while until explicit renewal. The first type address ends with bit 0; the second type ends with 1. Each address contains a *Validity counter* that indicates the count of packets sent from this address. *Validity counter* decrements each time when a packet is sent from an address. Correspondingly, each node en route maintains in the routing table the value of *validity counter* for each short-term address, and checks this value in each packet. If this value is not decremented or reaches 0, the packet is discarded.

When a short-term identity expires, the administrator has the right to renew its validity by broadcasting an "*Identity Renewal Message*". The network nodes update their validity counter table according to the *Identity Renewal Message*.

When a node with short-term or long-term identity is compromised or dismissed, or a node with short-term has finished its task before schedule, the administrator can revoke its identity at any time needed by broadcasting an "*Identity Revocation Message*". Each network node maintains a revocation list according to the *Identity Revocation Message*. The node checks the identity of each packet it receives against the revocation list and discards the packet with an identity in the list.

The authors state that by dividing identities to long-term and short-term ones, they minimize the need of identity renewal and identity revocation, which provides a higher reliability and saves much overhead traffic. With the limitation of "validity counter", the benefit that an adversary can get by stealing an identity is limited. Thus the significance of revocation is also decreased—even the identity is stolen by adversaries, there is very limited room they can make use of it. However, difficulty lies in predicting the validity counter.

3) Key-update Parameters: Zhang et al. [4] propose to use key-update parameters to revoke voided public and private keys, using a variant of the self-healing group key distribution scheme by Liu et al. [32]. The key generation was explained in Section III-A5.

Before network deployment, key-update parameters: $\langle\{v_i(x), l_i(ID)\}_{i=1,\dots,m}\rangle$, where m is the maximum possible phase index, are distributed to all nodes. The PKG generates private keys d_i for a node for all phases $i = 1, ..., m$ (strictly speaking, the phase-specific components of private keys which are then combined with node-specific components to generate a node's private key). The PKG calculates the differences v_i between d_i series and a polynomial series u_i, and distribute the difference series v_i to all nodes. At a later phase, online D-PKGs only provide the u_i series to unrevoked nodes. In this way, revoked nodes cannot update their private keys.

Key-update parameters are generated in this way: the PKG picks m distinct $2t^c$-degree polynomials, denoted by $\{l_i(x) = \sum_{j=0}^{2t^c} l_{i,j}x^j(mod\ \ q)\}_{i=1,\dots,m}$ with $l_{i,j} \in \mathbb{Z}_q^*$, and m distinct t^c-degree polynomials, denoted by $\{u_i(x) = \sum_{j=0}^{t^c} u_{i,j}x^j(mod\ q)\}_{i=1,\dots,m}$ with $u_{i,j} \in \mathbb{Z}_q^*$. The PKG then constructs $\{v_i(x) = d_{iy} - u_i(x)\}_{i=1,\dots,m}$, where d_{iy} denotes y-coordinate of the elliptical curve point d_i represents.

At phase i, a D-PKG node, say Z, collects secret shares and generates private key d_i. Z broadcasts the following message: $B_i := \{ID_X\}_{X\in\Lambda}\bigcup\{U_j(x) = \xi_j(x)u_j(x) + l_j(x)\}_{j=1,\dots i}$, where Λ denotes the set of nodes revoked until phase i, $\xi_j(x) = \prod_{x\in\Lambda}(x - ID_X)$. A unrevoked node B can derive $U_i(ID) = \xi_i(ID_B)u_i(ID_B) + l_i(ID_B)$, and then get $u_i(ID_B) = \frac{U_i(ID_B)-l_i(ID_B)}{\xi_i(ID_B)}$ and then $d_{iy} = v_i(ID_B) + u_i(ID_B)$, while a revoked one X cannot get $u_i(ID_X)$ because $\xi_i(ID_X) = 0$.

Though this scheme is novel and sound, there exists a possible drawback: The scheme does not have good scalability, since the phase-specific components of all phases need to be calculated before network deployment to get key-update parameters and furnish all nodes with them, and size of parameters to be distributed to D-PKGs is also proportional to number of D-PKGs.

C. Addressing Key Escrow

Key escrow is inherent in IBC. The PKG or the TA that generates private keys for nodes know the private key of each node and can eavesdrop the traffic or impersonate it. Although it may be a desirable feature in some cases (e.g. in military hierarchy), it is a problem with some MANETs. Traditional solutions for general IBC include: using additional private/public key pairs [85]; assigning an expiry date to the system's master secret key, or using threshold cryptography to distribute the secret key to multiple nodes [7], [33], [86], [87], [88], [57]. These solutions are widely used in many routing protocols in MANETs mentioned in previous section. However, all of them have some limitation: additional private/public key pairs are against advantages of IBC; expiry date needs an additional field for public key in packets and requires synchronization of all nodes; threshold cryptography has weaknesses mentioned in Section III.

1) Key Exchange Protocols without Key-escrow: Hoeper and Gong [89] identify the key escrow problem, and propose a set of key exchange protocols without key-escrow, based on the work of [7], [50].

In these protocols, a TTP computes the private key for each node using a master key and node's public key Q_{ID}, and distributes the key over a secure channel during network initialization. After initialiazation, the TTP is not needed, and any two nodes share a pairwise secret key: $K_{AB} = \hat{e}(d_A, Q_B) = \hat{e}(Q_A, d_B) = \hat{e}(d_B, Q_A) = K_{BA}$. To provide forward security and prevent the TTP from being a key escrow, the authors propose some protocols. A basic form of these protocols is: First, K_{AB} is divided into two parts K_e and K_a. Encryption under K_e prevents all other nodes from reading the messages, whereas K_a is used in a message authentication code (MAC) to enable mutual authentication. Then, $A{\rightarrow}B : A, E_{K_e}(K_1)$, $A{\leftarrow}B \quad : \quad B, E_{K_e}(K_2), MAC_{K_a}(A, E_{K_e}(K_1), E_{K_e}(K_2))$, $A{\rightarrow}B : MAC_{K_a}(B, E_{k_e}(K_2), E_{k_e}(K_1))$. A shared session key can be set up as $K_{ses} = f(K_1, K_2)$.

By replacing K_1 and K_2 with different forms, different properties can be obtained. For example, using Elliptic Curve Diffie-Hellman protocol, A and B can select ephemeral private keys r_A and r_B, generates and sends public keys $T_A = r_A P$ and $T_B = r_B P$. A shared session key can be obtained as $K_{ses} = h(r_A T_B) = h(r_B T_A)$ which is unknown to the TTP and achieves perfect forward secrecy. A revised version of this protocol can be found in their later work [90].

The authors analyze what kind of security properties can be achieved by each protocol. They claim the presented protocols resist most of the common attacks, such as impersonation, replay, known-key, unknown-key share and key compromise impersonation. However, they state that these protocols cannot resist active attacks launched by TTP using the system's master key. Furthermore, the scheme seems not applicable to routing protocols, because it assumes secure routing is ready.

2) Static and Dynamic Components of Keys: Zhang et al. [4] propose to use node-specific and phase-specific components to generate private keys of nodes for key revocation problem. Zhao et al. [66] employ the same notion to address key escrow issue. The difference is that in the former scheme, both node-specific and phase-specific components are generated by PKG before network deployment, so the PKG knows both of them; in the latter scheme, the PKG only knows the static component, but does not know the dynamic component, so that PKG cannot get private keys of nodes.

In Zhao's scheme, before the network starts, the system parameters are distributed among nodes. Using these system parameters, a secure routing can be established. Then the master key is updated among nodes. The new master key is updated with two parts: static part—s_{sta} that is always equal to initial master key s generated by the offline PKG, and dynamic part—s_{dyn} generated by all nodes contributively. New system public key and private keys are determined by the new master key. The PKG only knows the static part of the master key, the online nodes only know the dynamic part of the master key. The routing-security interdependency cycle problem of threshold cryptography is avoided by secure routing based on initial master key. The mobile attack problem is avoided by static component kept offline by PKG.

3) Adversary Models for Dishonest TAs: In [91] and [90], Hoeper and Gong propose three adversary models for dishonest TAs, and analyze the probabilities of successful attack for each model. Further, they suggest some countermeasures against this type of attacks: aborting protocols if a node receives messages of different contents that belong to the same protocol flow; two nodes establishing a shared key as soon as they are close to each other; using mobility to enable the use of different routing paths for different protocol flows.

The authors also studied the problem of utilizing key escrow. They conclude that increasing the number of deployed spy nodes, giving them more communication power and placing them at strategic places, can significantly improve the ability of a PKG to act as key escrow.

D. Security Concerns of IBC

The greatest concern of applying IBC in MANETs is the reliability of its security. In [92], Granger et al. state that it is still hard to say whether pairing-based cryptosystems (the mainstream of IBC) will be able to provide satisfactory security and efficiency as the desired level of security rises. They state that as the security requirements increase, the price one has to pay for the extra functionality will increase sharply.

They also identify some theoretical concern on the pairing-based systems – the BDHP (bilinear Diffiee-Hellman problem) is a new problem that has not been widely studied. It is closely related to the Diffiee-Hellman Problem (DHP) in the elliptic curve group. It follows that if one has an algorithm for the DHP on the curve, one can immediately solve the BDHP as well. Hence it is a source of concern that security depends on the presumed intractability of the DHP rather than the more natural and more extensively studied Discrete Log Problem (DLP).

Verheul [93] shows an example in which the DHP is efficiently solvable. The author states if a Verheul homomorphism might some day be constructed, even if it were constructed just for the class-VI supersingular elliptic curves, that would be enough to render all pairing-based cryptosystems completely insecure. From the literature, it seems that up to now, Verheul's guess has not been proven positive or negative. In a more recent article [94], Moody reviews some of the problems that the security of elliptic curve cryptosystems are based upon, and discusses in detail the theorem of Verheul (including its generalization), and its consequences. The author tries to

TABLE V
SUMMARY OF ADDRESSING WEAKNESSES OF IBC IN MANETS

Year	Proposal(s)	Main Idea and Contribution(s)	Main Problem Solved	Other Problems Considered	Weaknesses
2005	MASK [79]	An anonymous routing protocol.	Identity disclosure	No	1. Each node maintains a set of pseudonyms and the corresponding private keys for each pseudonym. 2. It can only be used in their own routing protocol and not in any other protocol or any higher layer application.
2009	General-purpose identity hiding schemes [81]	Requirements of an identity hiding scheme for MANETs and three general-purpose identity hiding algorithms.	Identity disclosure	No	Pre-distribution of secret
2003	Using threshold cryptography [7] etc.	System private key is divided to multiple PKGs	Key escrow	No	Weaknesses mentioned in Section III.
2003	Appending "Expire Time" and version to the Identity [73] etc.	The validity of identity is explicit to check.	Key revocation & key escrow	No	1. A separate field in each packet is needed to indicate the public key. 2. Requires precise synchronization among all network nodes.
2006	Compromise-tolerant key update [4]	Using predistributed key-update parameters	Key revocation	Identity disclosure	Number of phases is hard to predict; Scalability is not good.
2009	Long-term and Short-term Identities [66]	Decreases the frequency of key revocation and key renewal, and the risk of key compromise	Key revocation	Key escrow & identity disclosure	It is difficult to predict the validity count of identities in some situations.
2003	Additional public/private key pairs [85]	Using additional keys pairs to prevent PKG knowing session key.	Key escrow	No	Against IBC advantages.
2005	Key Exchange Protocols without key-escrow [89]	A set of key exchange protocols without key-escrow	Key escrow	No	Only applicable on high layer communication, not in routing protocols.
2005	Adversary models for dishonest TAs [91], [90]	Three adversary models for dishonest TAs, the probabilities of successful attack for each model, and some countermeasures	Key escrow	No	Not clear
2009	Static and dynamic components of keys [66]	Addresses key escrow and avoids threshold cryptography problems	Key escrow	No	Not clear

generalize Verheul's theorem to more ordinary curves. As a conclusion, the author leaves it as an open question to generalize some form of Verheul's theorem to ordinary curves with low embedding degree, and states that this work would require new methods.

To achive high security and counter attacks towards IBC, researchers suggest putting more strict restraints on its mathematical basis and choosing the elliptic curve and finite field it uses meticulously. Researches on these security concerns and challenges will be the future work on IBC schemes and their applications.

E. Summary and Comments

In this section, we studied proposals to address main problems of identity-based cryptography in MANETs: identity disclosure, identity revocation, and key escrow. Although each proposal addresses some of the problems, no one addresses all the problems, and these proposals are not compatible with each other.

Table V summarizes the main characteristics and weaknesses of schemes overcoming problems of IBC in MANETs. We also show that security of IBC schemes is also a concern, and research on this topic is still going on.

VII. CONCLUSIONS

In this survey, we have studied major developments in IBC, and the applications of IBC in MANETs in various areas. We have identified the drawbacks and challenges of IBC which impose difficulties on its application to MANETs.

In the field of MANETs' security IBC has already been widely applied. However, we notice there are many issues unaddressed in these applications.

To apply IBC better in MANETs, we must look at properties of IBC and identify its pros and cons. On the one hand, some properties lend IBC attractions to MANETs: private keys are short and easy to generate and store, public keys are implicitly carried by their identities, so there is no need to distribute and store certificates of partners or public key of CA. On the other hand, its other properties appear awkward in MANETs, e.g., the problems mentioned in Section VI. Another thing that is not mentioned there (because there is no way to work around it) but a problem for many MANETs is that: from the nature of IBC, it requires the system parameters be distributed to all communicating parties before any messages can be encrypt/decrypted. This requirement excludes the so called "truly ad hoc" networks out of its scope. In those networks, a group of strangers come together without any central node in charge of the administration and the organization of the network. The master key can only be generated online contributively by untrusted peers. Thus, they are inevitably subject to Byzantine attacks, and may be totally taken over by adversaries.

Considering properties on both sides of IBC in MANETs, we find a type of MANETs that is most suitable for IBC: there is an administrator that generates and distributes initial system parameters to all nodes; the administrator can authenticate the identity of a node, and assign initial private key to it. For those MANETs that meet these requirements, e.g., sensor networks, military networks such as moving soldiers with wearable computers, portable communication systems for future public safety, emergency and disaster applications, IBC is the most promising security solution, but there seem no perfect solutions yet. We suggest future research be focused on this type of MANETs.

References

[1] L. Abusalah, A. Khokhar, and M. Guizani, "A survey of secure mobile ad hoc routing protocols," *IEEE Commun. Surveys & Tutorials, IEEE*, vol. 10, no. 4, pp. 78–93, 2008.

[2] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang, "Security in mobile ad hoc networks: challenges and solutions," *IEEE Wirel. Commun.*, vol. 11, no. 1, pp. 38–47, 2004.

[3] Y. Fang, X. Zhu, and Y. Zhang, "Securing resource-constrained wireless ad hoc networks," *Wireless Commun.*, vol. 16, no. 2, pp. 24–29, 2009.

[4] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Securing mobile ad hoc networks with certificateless public keys," *IEEE Trans. Dependable Secur. Comput.*, vol. 3, no. 4, pp. 386–399, 2006.

[5] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. Crypto 1984*, 1984.

[6] A. Joux, "A one round protocol for tripartite diffie-hellman," in *ANTS IV*, ser. LNCS, vol. 1838. Springer-Verlag, 2000, pp. 385–394.

[7] Boneh and Franklin, "Identity-based encryption from the weil pairing," in *Proc. Crypto 2001*, ser. LNCS, vol. 2139. Springer, 2001, pp. 213–219.

[8] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in *Proc. ASIACRYPT*, ser. LNCS, vol. 2248. Springer-Verlag, 2001, pp. 514–532.

[9] C. Cocks, "An identity based encryption scheme based on quadratic residues," in *IMA: IMA Conference on Cryptography and Coding, LNCS lately (earlier: Cryptography and Coding II, Edited by Chris Mitchell, Clarendon Press, 1992)*, 2001.

[10] R. Dutta, R. Barua, and P. Sarkar, "Pairing-based cryptographic protocols: A survey," Cryptology ePrint Archive, Report 2004/064, Jun. 24 2004.

[11] Desmedt and Quisquater, "Public-key systems based on the difficulty of tampering (is there a difference between DES and RSA?) (extended abstract)," in *Proc. Crypto*, 1986.

[12] H. Tanaka, "A realization scheme for the identity-based cryptosystem," in *Proc. CRYPTO '87*, ser. LNCS, vol. 293. Springer-Verlag, 1988, 16–20 Aug. 1987, pp. 340–349.

[13] S. Tsujii and T. Itoh, "An ID-based cryptosystem based on the discrete logarithm problem," *IEEE J. Sel. Areas Commun.*, vol. 7, no. 4, May 1989.

[14] Maurer and Yacobi, "Non-interactive public-key cryptography," in *EUROCRYPT: Advances in Cryptology*, 1991.

[15] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, 1979.

[16] L. Zhou and Z. J. Haas, "Securing ad hoc networks," *IEEE Network*, vol. 13, no. 6, pp. 24–30, 1999.

[17] Y. Desmedt and Y. Frankel, "Threshold cryptosystems," in *Proc. Crypto 1989*, 1989.

[18] Y. Desmedt, "Threshold cryptography," *European Transactions on Telecommunications*, vol. 5, no. 4, pp. 449–457, Jul. – Aug. 1994.

[19] A. Khalili, J. Katz, and W. A. Arbaugh, "Toward secure key distribution in truly ad-hoc networks," in *SAINT Workshops*. IEEE Computer Society, 2003, pp. 342–346.

[20] R. Bobba, L. Eschenauer, V. Gligor, and W. Arbaugh, "Bootstrapping security associations for routing in mobile ad-hoc networks," in *IEEE Global Telecommunications Conference 2003*. IEEE Computer Society Press, 2003.

[21] H. Deng, A. Mukherjee, and D. P. Agrawal, "Threshold and identity-based key management and authentication for wireless ad hoc networks," in *ITCC (1)*. IEEE Computer Society, 2004, pp. 107–111.

[22] H. Deng and D. P. Agrawal, "TIDS: threshold and identity-based security scheme for wireless ad hoc networks," *Ad Hoc Networks*, vol. 2, no. 3, pp. 291–307, 2004.

[23] Y. Zhang, J. Liu, Y. Wang, J. Han, H. Wang, and K. Wang, "Identity-based threshold key management for ad hoc networks," *Pacific-Asia Workshop on Computational Intelligence and Industrial Application, IEEE*, vol. 2, pp. 797–801, 2008.

[24] P. Xia, M. Wu, K. Wang, and X. Chen, "Identity-Based Fully Distributed Certificate Authority in an OLSR MANET," in *4th Wireless Communications, Networking and Mobile Computing*. IEEE, 2008, pp. 1–4.

[25] J. V. D. MERWE, D. DAWOUD, and S. McDONALD, "A survey on peer-to-peer key management for mobile ad hoc networks," *ACM Comput. Surv.*, vol. 39, no. 1, pp. 1–45, 2007.

[26] S. Xu and S. Čapkun, "Distributed and secure bootstrapping of mobile ad hoc networks: Framework and constructions," *ACM Trans. Inf. Syst. Secur.*, vol. 12, no. 1, pp. 1–37, 2008.

[27] S. Zhao and A. Aggarwal, "Against mobile attacks in ad-hoc networks," in *Proc. IEEE International Conference on Information Theory and Information Security*, 2010.

[28] G. Li and W. Han, "A new scheme for key management in ad hoc networks," in *Proc. 4th International Conference on Networking Proceedings*, ser. LNCS, vol. 3421. Springer, 2005, pp. 242–249.

[29] X. Boyen, "Multipurpose identity-based signcryption: A swiss army knife for identity-based cryptography," in *Proc. Crypto 2003*, 2003.

[30] Y. Zhang, W. Liu, W. Lou, Y. Fang, and Y. Kwon, "Ac-pki: anonymous and certificateless public-key infrastructure for mobile ad hoc networks," in *Proc. International Conference on Communications*. IEEE Computer Society Press, 2005.

[31] Y. Ren, J. Wang, Y. Zhang, and L. Fang, "Identity-based key issuing protocol for ad hoc networks," *Computational Intelligence and Security, International Conference on*, vol. 0, pp. 917–921, 2007.

[32] D. Liu, P. Ning, and K. Sun, "Efficient self-healing group key distribution with revocation capability," in *Proc.10th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2003, pp. 231–240.

[33] B. Lee, C. Boyd, E. Dawson, K. Kim, J. Yang, and S. Yoo, *Secure Key Issuing in ID-based Cryptography*, ser. Conferences in Research and Practice in Information Technology. ACS, 2004, vol. 32.

[34] N. Saxena, "Public key cryptography sans certificates in ad hoc networks," in *Applied Cryptography and Network Security, 4th International Conference Proceedings*, ser. LNCS, vol. 3989, 2006, pp. 375–389.

[35] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in *28th Symposium on Foundations of Computer Science*. IEEE, 1987, pp. 427–437.

[36] H.-Y. Lin, Y.-M. Huang, and T.-I. WANG, "Resilient cluster-organizing key management and secure routing protocol for mobile ad hoc networks," in *Proc. IEICE Transactions on Communications*, 2005.

[37] Y.-M. Huang, H.-Y. Lin, and T.-I. Wang, "Inter-cluster routing authentication for ad hoc networks by a hierarchical key scheme," *J. Comput. Sci. Technol.*, vol. 21, no. 6, pp. 997–1011, 2006.

[38] Y.-M. Huang and H.-Y. Lin, "Information service on scalable ad-hoc mobile wireless networks," in *Proc. Computer Networks and Mobile Computing 2003*, 2003.

[39] D. Crescenzo, Arce, and Ge, "Threshold cryptography in mobile ad hoc networks," in *International Conference on Security in Communication Networks, SCN, LNCS*, vol. 4, 2004.

[40] J.Kong, P. Zerfos, H. Luo, and L. Zhang, "Providing robust and ubiquitous security support for mobile ad-hoc networks," in *Proc. IEEE ICNP*, 2001.

[41] M. J. Bohio and A. Miri, "An authenticated broadcasting scheme for wireless ad hoc network," in *Proc. CNSR 2004*. IEEE Computer Society, 2004, pp. 69–74.

[42] J. Cha and J. Cheon, "An identity-based signature from gap diffie-hellman groups," in *PKC: International Workshop on Practice and Theory in Public Key Cryptography*, vol. 2567. LNCS, 2003.

[43] M. Bohio and A. Miri, "Authenticated secure communications in mobile ad hoc networks," in *Proc. Canadian Conference on Electrical and Computer Engineering*. IEEE Computer Society Press, 2004.

[44] R. Sakai, K. Ohgishi, and M. Kasahara, "Cryptosystems based on pairing," in *Symposium on Cryptography and Information Security*, 2000.

[45] M. J. Bohio and A. Miri, "Efficient identity-based security schemes for ad hoc network routing protocols," *Ad Hoc Networks*, vol. 2, no. 3, pp. 309–317, 2004.

[46] H.-Y. Chien and R.-Y. Lin, "Improved id-based security framework for ad hoc network," *Ad Hoc Netw.*, vol. 6, no. 1, pp. 47–60, 2008.

[47] ——, "Identity-based key agreement protocol for mobile ad-hoc networks using bilinear pairing," in *Proc. Sensor Networks, Ubiquitous, and Trustworthy Computing*. IEEE Computer Society, 2006, pp. 520–529.

[48] K. H. Rhee, Y.-H. Park, and G. Tsudik, "A group key management architecture for mobile ad-hoc wireless networks," *J. Inf. Sci. Eng*, vol. 21, no. 2, pp. 415–428, 2005.

[49] J. Kong, H. Luo, K. Xu, D. L. Gu, M. Gerla, and S. Lu, "Adaptive security for multilevel ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 533–547, 2002.

[50] F. Hess, "Efficient identity based signature schemes based on pairings," in *SAC: Annual International Workshop on Selected Areas in Cryptography*, vol. 2595. LNCS, 2003.

[51] P. Zhang, C. Ye, X. Li, Y. Cheng, and X. Ma, "Constant-round contributory group key agreement for ad hoc networks," in *Proc. IEEE Wireless Communications, Networking and Mobile Computing*. IEEE Computer Society Press, 2005.

[52] W. Lee and W. Sriborrirux, "Optimizing authentication mechanisms using ID-based cryptography in ad hoc wireless mobile networks," in *Proc. Information Networking, Networking Technologies for Broadband and Mobile Networks*, ser. LNCS, vol. 3090. Springer, 2004, pp. 925–934.

[53] X. Du, Y. Wang, J. Ge, and Y. Wang, "Id-based authenticated two round multi-party key agreement," Cryptology ePrint Archive, Report 2003/247, 2003, http://eprint.iacr.org/.

[54] L. Zhang, Y. Hu, and N. Mu, "An identity-based broadcast encryption protocol for ad hoc networks," *Young Computer Scientists, International Conference for*, vol. 0, pp. 1619–1623, 2008.

[55] C. Delerablée, "Identity-based broadcast encryption with constant size ciphertexts and private keys," in *Proc. ASIACRYPT'07*. Springer-Verlag, 2007, pp. 200–215.

[56] Y.-H. Lee, H. Kim, B. Chung, J. Lee, and H. Yoon, "On-demand secure routing protocol for ad hoc network using id based cryptosystem," in *Proc. 4th ICPDCAT*. IEEE, 2003, pp. 211–215.

[57] K. G. Paterson, "ID-based signatures from pairings on elliptic curves," Cryptology ePrint Archive, Report 2002/004, Jan. 2002.

[58] E. A. Panaousis, L. Nazaryan, and C. Politis, "Securing aodv against wormhole attacks in emergency manet multimedia communications," in *Proc. 5th International ICST Mobile Multimedia Communications Conference*, ser. Mobimedia '09, 2009, pp. 1–7.

[59] B.-N. Park, J. Myung, and W. Lee, "LSRP: A lightweight secure routing protocol with low cost for ad-hoc networks," in *Proc. International Conference on Convergence in Broadband and Mobile Networking*, ser. LNCS, vol. 3391. Springer, 2005, pp. 160–169.

[60] J. jacques Quisquater, "New identity based signcryption schemes from pairings," Cryptology ePrint Archive, Report 2003/023, Feb. 24 2003. [Online]. Available: http://eprint.iacr.org/2003/023.ps.gz

[61] B.-N. Park and W. Lee, "ISMANET: A secure routing protocol using identity-based signcryption scheme for mobile ad-hoc networks," *IEICE Trans. Communications*, 2005.

[62] B.-N. Park, J. Myung, and W. Lee, "ISSRP: A secure routing protocol using identity-based signcryption scheme in ad-hoc networks," in *Proc. 5th International Conference on Parallel and Distributed Computing*, ser. LNCS, vol. 3320. Springer, 2004, pp. 711–714.

[63] J. Song, H. Kim, S. Lee, and H. Yoon, "Security enhancement in ad hoc network with id-based cryptosystem," in *Proc.7th International Conference on Advanced Communication Technology*. IEEE Computer Society Press, 2005.

[64] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," in *Proc. EUROCRYPT*, ser. LNCS, vol. 2656. Springer-Verlag, 2003, pp. 416–432.

[65] C. Adjih, D. Raffo, and P. Muhlethaler, "Attacks against olsr: Distributed key management for security," in *Proc. OLSR Interop and Workshop*, 2005.

[66] S. Zhao and A. Aggarwal, "PAPA-UIC: a design approach and a framework for secure mobile ad hoc networks," *Security and Communication Networks, John Wiley & Sons*, vol. 1, pp. 371–383, 2010.

[67] B. Lynn, "Authenticated identity-based encryption," Cryptology ePrint Archive, Report 2002/072, Jul. 11 2002.

[68] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proc. 6th annual international conference on Mobile computing and networking*, ser. MobiCom '00. New York, NY, USA: ACM, 2000, pp. 255–265.

[69] S. Buchegger and J.-Y. Le Boudec, "Performance analysis of the confidant protocol," in *Proc. 3rd ACM international symposium on Mobile ad hoc networking & computing*, ser. MobiHoc '02. New York, NY, USA: ACM, 2002, pp. 226–236.

[70] P. Michiardi and R. Molva, "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Proc. IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security*, 2002, pp. 107–121.

[71] A. M. Hegland, E. Winjum, S. F. Mjolsnes, C. Rong, O. Kure, and P. Spilling, "A survey of key management in ad hoc networks," *IEEE Commun. Surveys & Tutorials, IEEE*, vol. 8, no. 3, pp. 48–66, 2006.

[72] S. Balfe, K. D. Boklan, Z. Klagsbrun, and K. G. Paterson, "Key refreshing in identity-based cryptography and its applications in manets," in *Military Communications Conference, 2007. MILCOM 2007. IEEE*. IEEE, 2007, pp. 1–8.

[73] K. Hoeper and G. Gong, "Bootstrapping security in mobile ad hoc networks using identity-based schemes with key revocation," University of Waterloo, Report 2006-04, 2006. [Online]. Available: http://www.comsec.uwaterloo.ca/~khoeper/IBCrevocation_hoeper.pdf

[74] D. Carman, "New directions in sensor network key management," *International Journal of Distributed Sensor Networks*, vol. 1, pp. 3–15, 2004.

[75] S. Balfe, K. D. Boklan, Z. Klagsbrun, and K. G. Paterson, "Toward hierarchical identity-based cryptography for tactical networks," in *Military Communications Conference, 2004. MILCOM 2004. IEEE*. IEEE, 2004, pp. 1–8.

[76] F. Li, Y. Hu, and C. Zhang, "An identity-based signcryption scheme for multi-domain ad hoc networks," in *Proc. 5th international conference on Applied Cryptography and Network Security*. Springer-Verlag, 2007, pp. 373–384.

[77] L. Cai, J. Pan, X. Shen, and J. W. Mark, "Peer collaboration in wireless ad hoc networks," in *Proc. 4th International IFIP-TC6 Networking Conference*, ser. LNCS, vol. 3462. Springer, 2005, pp. 840–852.

[78] G. Gentry and A. Silverberg, "Hierarchical ID-based cryptography," in *Proc. ASIACRYPT*. LNCS, Springer-Verlag, 2002.

[79] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "MASK: anonymous on-demand routing in mobile ad hoc networks," in *Wireless Communications*. IEEE, 2006, pp. 2376–2385.

[80] D. Balfanz, D. K. Smetters, P. Stewart, and H. C. Wong, "Talking to strangers: Authentication in ad-hoc wireless networks," in *Proc. Symp. Network and Distributed Systems Security*. Internet Society, 2002, pp. 23–35.

[81] S. Zhao and A. Aggarwal, "General-purpose identity hiding schemes for ad-hoc networks," in *Proc. International Symposium on Intelligent Ubiquitous Computing and Education*. IEEE, 2009, pp. 349–353.

[82] K. Hoeper and G. Gong, "Key revocation for identity-based schemes in mobile ad hoc networks," in *ADHOC-NOW*, ser. LNCS, vol. 4104. Springer, 2006, pp. 224–237.

[83] C. Crepeau and C. Davis, "A certificate revocation scheme for wireless ad hoc networks," in *ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN)*, vol. 1, 2003.

[84] H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang, "Self-securing ad hoc wireless networks," in *ISCC*. IEEE Computer Society, 2002, pp. 567–574.

[85] Gentry, "Certificate-based encryption and the certificate revocation problem," in *Proc. EUROCRYPT*, 2003.

[86] Boyd, Mao, and Paterson, "Key agreement using statically keyed authenticators," in *International Conference on Applied Cryptography and Network Security (ACNS), LNCS*, vol. 2, 2004.

[87] L. Chen and C. Kudla, "Identity based authenticated key agreement protocols from pairings," Hewlett Packard Laboratories, Tech. Rep. HPL-2003-25, Feb. 12 2003.

[88] J. Oh, K. Lee, and S.-J. Moon, "How to solve key escrow and identity revocation in identity-based encryption schemes," in *ICISS 2005 Proceedings*, ser. LNCS, vol. 3803. Springer, 2005, pp. 290–303.

[89] K. Hoeper and G. Gong, "Identity-based key exchange protocols for ad hoc networks," in *Proc. Canadian Workshop on Information Theory*, 2005.

[90] ——, "Preventing or utilizing key escrow in identity-based schemes employed in mobile ad hoc network," *Int. J. of Security and Networks (IJSN)*,, Jul. – Aug. 2007.

[91] ——, "Limitations of key escrow in identity-based schemes in ad hoc networks," in *Proc. First International Conference on SecureComm 2005*, 2005.

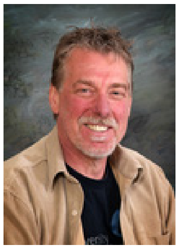
[92] R. Granger, D. Page, and N. Smart, "High security pairing-based cryptography revisited," in *Algorithmic Number Theory Symposium VII*. Springer-Verlag LNCS 4076, Jul. 2006, pp. 480–494.

[93] E. Verheul, "Evidence that XTR is more secure than supersingular elliptic curve cryptosystems," *JCRYPTOL: Journal of Cryptology*, vol. 17, 2004.

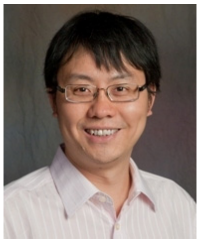
[94] D. Moody, "The diffie—hellman problem and generalization of verheul's theorem," *Des. Codes Cryptography*, vol. 52, no. 3, pp. 381–390, 2009.

Shushan Zhao received his B.Sc degree in Computer Science from Shandong University, China, and M.Sc degree in Computer Science from Helsinki University of Technology, Finland. He is currently a Ph.D. candidate in the School of Computer Science at the University of Windsor, Canada. His research interests include network systems, network security, cryptography, and telecommunication systems. In mobile ad-hoc networks area, he participated in WIDENS project (http://www.comlab.hut.fi/projects/WIDENS/) and multiple industry projects. He has rich experience and actively involves in telecommunication and software industry in China, Finland, and Canada.

Akshai Aggarwal (M 1966, SM 1992) is working as Vice Chancellor of Gujarat technological University, India. Till May 31, 2010, he was working as the Director, School of Computer Science, University of Windsor, Canada. Before coming to Canada, he had worked as Professor and Head of Department of Computer Science at Gujarat University for about 10 years and as Professor and Head, Department of EE at M.S. University of Baroda. He has been actively associated with the IEEE platform in India. He was Chairman of IEEE India Council for two years. He was also the founder Chairman of IEEE Gujarat Section, the IEEE Computer Society Chapter and the IEEE Joint Chapter of Industry Applications, Industrial Electronics and Power Electronics. The Section conducted two International Conferences and one national Seminar during his Chairmanship. He graduated with a B.Sc.(EE) from Punjab Engineering College and studied at MS University of Baroda for his Master's and Doctoral work.



Richard Frost has a degree in Physics from Queen Mary Collge London, a Masters degree in Medical Physics from the University of Aberdeen, and a doctorate in Computer Science from Strathclyde University in Glasgow. He previously held faculty positions at Strathclyde and Glasgow Universities. He is currently the Acting Director of the School of Computer Science at the University of Windsor, where he has been Professor of Computer Science for 23 years.



Xiaole Bai is an assistant professor in Department of Computer and Information Science at University of Massachusetts Dartmouth. He received his B.S. degree in 1999 at Southeast University, China, and the M.S. degree in 2003 at Networking Laboratory, Helsinki University of Technology, Finland. He then joined The Ohio State University in 2004 and received his Ph.D. degree in 2009 from Department of Computer Science and Engineering. His research interests include network science and engineering, cyber space security, and distributed computing.