Survey on Security in Wireless Sensor

Zhijun Li,[1] and Guang Gong[2]
University of Waterloo, Waterloo, Ontario, Canada

Abstract

Advances in electronics and wireless communication technologies have enabled the development of large-scale wireless sensor networks (WSNs). There are numerous applications for wireless sensor networks, and security is vital for many of them. However, WSNs suffer from many constraints, including low computation capability, small memory, limited energy resources, susceptibility to physical capture, and the lack of infrastructure, all of which impose unique security challenges and make innovative approaches desirable. In this paper, we present a survey on security issues in wireless sensor networks. We address several network models for security protocols in WSNs, and explore the state of the art in research on the key distribution and management schemes, typical attacks and corresponding countermeasures, entity and message authentication protocols, security data aggregation, and privacy. In addition, we discuss some directions of future work.

Key words : Wireless Sensor Networks, Security, Key Predistribution

I. Introduction

Wireless sensor networks (WSNs) are innovative large-scale wireless networks that consist of distributed, autonomous, low-power, low-cost, small-size devices using sensors to cooperatively collect information through infrastructureless ad-hoc wireless network. The development of wireless sensor networks was originally motivated by military applications such as battlefield surveillance. However, wireless sensor networks are now used in many civilian application areas, including environment and habitat monitoring, healthcare applications, home automation, and traffic control.

Security plays a fundamental role in many wireless sensor network applications. Due to WSNs' unique characteristics, the security techniques used in the conventional networks cannot be directly applied to WSNs. First, sensor nodes are very sensitive of production cost since sensor networks consist of a large number of sensor nodes. Akyildiz et al. [1] argued that the cost of a sensor node should be less than one dollar in order for sensor networks to be feasible. Therefore, most sensor nodes are resource-restrained in terms of energy, memory, computation, and communication capabilities. Normally sensor nodes are powered by batteries, and recharging batteries are infeasible in many circumstances. Then energy consumption becomes a key consideration for most sensor network protocols. Second, Sensor nodes may be deployed in public hostile locations, which makes sensor nodes vulnerable to physical attacks. Generally, adversaries are assumed to be able to undetectably take control of a sensor node and extract all secret data in the node. Furthermore, the scale of sensor networks is considerably large, and the network topology is dynamically adjusted, because some nodes may die out of running out of energy or failure, and new nodes may join the network to maintain desirable functionality. At last, sensor networks use insecure wireless communication channel and lack infrastructure. As a result, the existing security mechanisms are inadequate, and new approaches are desired.

1.1. Security Goals

Similar to other communication systems, WSNs have the following general security goals:

- *Confidentiality*: protecting secret information from unauthorized entities
- *Integrity*: ensuring messages have not been altered by malicious nodes
- *Data Origin Authentication*: authenticating the source of message
- *Entity Authentication*: authenticating the user/node/base-station is indeed the entity whom it claims to be
- *Access control*: restricting access to resources to privileged entities

Availability: ensuring desired services may be available whenever required

Manuscript received November, 2008.
Manuscript revised November, 2008.

[1] Zhijun Li, University of Waterloo, leezj@engmail.uwaterloo.ca.

[2] Guang Gong, University of Waterloo, ggong@calliope.uwaterloo.ca

In addition, WSNs have the following specific security objectives:

- *Forward secrecy*: preventing nodes from decrypting any secret messages after they left the network
- *Backward secrecy*: preventing joining nodes from decrypting any previously transmitted secret message
- *Survivability*: providing a certain level of service in the presence of failures and/or attacks
- *Freshness*: ensuring that the data is recent and no adversary can replay old messages
- *Scalability*: supporting a great number of nodes
- *Efficiency*: storage, processing and communication limitations on sensor nodes must be considered

1.2. Applications

There are extensive applications of wireless sensor networks [2,3,4], such as Great Duck (bird observation on Great Duck island), Cattle Herding, Bathymetry, ZebraNet, Glacier Monitoring, Ocean Water Monitoring, Cold Chain Management, Grape monitoring, Rescue of Avalanche Victims, Vital Sign Monitoring, Power monitoring, Parts Assembly, Tracking Military Vehicles, and Self-healing Mine Field and Sniper Localization. According to the deployment areas, the WSN applications can be categorized in the following fields: military, environmental, industrial, agricultural, location oriented, public safety oriented, airport oriented, automotive, emergency handling, medical and oceanic.

Military and medical solutions are two of the most security-oriented application fields of wireless sensor networks. Military sensing networks are designed to detect and gain as much information as possible about enemy movements, explosions, and other phenomena. Typically, wireless sensor nodes are integrated with military command, control, communications, computing, intelligence, surveillance, reconnaissance and targeting systems. Examples of military wireless sensor network applications are battlefield surveillance, guidance systems for intelligent missiles, detection of attacks by weapons of mass destruction such as nuclear, biological, or chemical, and other monitoring applications. Due to the nature of the military, it is apparent that those applications could not be mounted without appropriate security assurance.

Recently, many medical systems are equipped with a large number of tiny, non-invasive sensors, located on or close to the patient's body, for health monitoring purposes. Such systems have been designed to measure diverse physiological values, including blood pressure, blood oxygen level, heart activities, activity recognition, etc., and are available in many different forms, including wrist wearable, ambulatory devices and as part of biomedical smart clothes. The term of body sensor network (BSN) [5] is coined to represent this kind of application. A number of intelligent physiological sensors are integrated into a wearable wireless body sensor network, which can be used for computer-assisted rehabilitation and even early detection of medical conditions. Those applications imply that outpatients can be monitored from their homes, freeing space in hospital beds. As the physiological patient data is legally required to be kept private, the implemented network must invoke strong security protocols.

1.3. Network Models

Typically, a wireless sensor network consists of a few base stations and hundreds and thousands of sensor nodes. Sensor nodes are powered by battery, equipped with sensors, data processing units of limited computation capability, limited memory space, and short-range radio communication. Base stations are the gateways to other networks, with powerful data processing/storage centers, or access points for human interface. In general, base stations have many orders of magnitude more powerful than ordinary sensor nodes. As a rule, base stations are assumed to be trusted and to be tamper resistant. Sensor nodes are usually deployed at random in targeted fields. Each of these scattered sensor nodes has the capabilities to collect data and route data back to base stations via infrastructureless wireless architecture. Base stations issue task commands, collect sensor readings, perform costly operations on behalf of sensor nodes and manage the network. WSNs are dynamic in the sense that radio range and network connectivity change over time; some sensor nodes die and new sensor nodes may be added to the networks.

There are different settings about WSN architectures.

Hierarchical Model vs. Distributed Model

In some scenarios, sensor nodes are organized as a hierarchical structure. They are grouped into a number of clusters controlled by some of the nodes which play a particular role denoted as cluster heads. Member nodes are associated with a cluster via a one-hop or multi-hop link and these member nodes perform sensing and forwarding. After gathering or aggregating localized sensing information from their cluster members, the cluster heads send packets to the base station. In contrast, there is no concept of cluster or group in the distributed model. All nodes play similar roles in the network. Once nodes are deployed, they scan their radio coverage area to figure out neighbors and manage to form fully distributed networks.

Sensor nodes collaboratively collect, aggregate, and forward information.

Homogeneous Model vs. Heterogeneous Model

In the homogeneous system model, all nodes are similar in terms of communication, computation, and storage capabilities. By contrast, heterogeneous wireless nodes can be equipped with different transport mediums with different ranges of coverage and distinct specifications including CPU, memory, and power supply to meet specific needs.

The remainder of this paper is organized as follows. In Section 2, we discuss various key distribution schemes. Section 3 describes attacks, countermeasures, intrusion detection, and intrusion tolerance. Section 4 presents a number of authentication protocols, including broadcast authentication and entity authentication. In Section 5, we introduce secure data aggregation protocols. And we address some privacy-protection protocols in Section 6. Finally, we conclude this paper in Section 7.

II. Key Distribution and Management

Security of large scale, densely deployed and infrastructure-less wireless networks of resource limited sensor nodes calls for efficient key distribution and management mechanisms. This is one of the most popular research fields in the secure sensor networks, and plenty of approaches are proposed.

2.1. Straightforward Approaches

The simplest method of key distribution is to preload a single network-wide key into all nodes before deployment. Apparently, this scheme suffers a severe drawback in that the compromise of a single node would cause the collapse of the entire network security. An alternative key distribution scheme is fully pairwise keys, i.e., every node in the sensor network shares a distinct key with every other node in the network. The main problem with this pairwise key scheme is its poor scalability. The number of keys that must be stored in each node is proportional to the total number of nodes in the network. Since sensor nodes are resource-constrained, this brings significant overhead, which limits the scheme's applicability except it can be effectively used only in smaller networks.

The method of Kerberos-like key distribution is widespread in the environment of many networks. In the sensor network, we can use a trusted, secure base station as an arbiter to provide link keys between sensor nodes. Sensor nodes authenticate themselves to the base station, after which the base station generates a link key and sends it to both parties securely. An example of such approach is the SNEP protocol, a part of the SPINS security infrastructure [6]. However, this kind of scheme suffers high energy consumption, which makes it inapplicable in most sensor network applications.

2.2. Schemes based on Initial Trust Model

In LEAP [7], Zhu, Setia, and Jajodia proposed a key distribution scheme based on an initial trust model. All nodes share a common master key K and a keyed one-way hash function H. Upon deployment, nodes begin to discover all neighbor nodes and establish pairwise keys using K and H. For example, the pairwise key between nodes u and v can be computed by $H_{H_K(u)}(v)$ or $H_K(u \| v)$. After establishing the pairwise keys, all nodes eliminate the master key. LEAP assumes that the time necessary for an adversary to compromise a sensor node is larger than the maximum time for nodes to complete the key establishment. If this initial trust assumption holds, LEAP is secure. However, sensors may be deployed in different phases, and new sensors may need to be added. A major disadvantage of LEAP is not supporting multi-phase deployment: new nodes cannot establish pairwise keys with nodes of previous phases.

The proposal in [8] partially solves this problem by generating many phase master keys, each of which is for one phase. Every node u in phase i stores its phase master key K_i and all other $H_{K_j}(u)$, where $j > i$. Every two adjacent nodes in the same phase can establish the pairwise key like LEAP. If node u in phase i and node v in phase j want to establish the pairwise key, supposing $i < j$, they both can compute $H_{H_{K_j}(u)}(v)$ and get the pairwise key. Every node only eliminates its phase master key, and keeps the rest. A drawback of this scheme appears when an adversary that compromises one node can duplicate many nodes which can establish pairwise keys with nodes of later phases. Furthermore, the number of phases has to be determined prior to the first deployment.

Another initial-trust-like scheme is addressed in [9], and is further enhanced in [10]. The authors assume that adversaries can monitor only a small portion of sensor nodes, due to random deployment of sensor networks. Initially, each pair of neighbor nodes just broadcast their pairwise key in plaintext. Afterwards, they can utilize multi-hop and multi-path indirect secure links to exchange other secret data, which results in higher security.

2.3. Random Probabilistic Key Distribution Scheme

Eschenauer and Gligor [11] first proposed a random key probabilistic distribution scheme (EG scheme) based on random graph theory [12]. A random graph is a graph that is generated by starting with a set of n vertices and adding edges between them at random. In the Erdös-Rényi model, a random graph is denoted by $G(n, p)$, in which every possible edge occurs independently with probability p. Erdös and Rényi [13] showed that, to achieve almost one hundred percent graph connectivity, every two vertices only need to have relatively lower probability P' of existence of direct link. Frequently, sensor nodes are randomly deployed, and the number of nodes in a sensor network is massive. We may think of a wireless sensor network as a graph, nodes as vertices, and links as edges. Using random graph theory, we can theoretically analyze the connectivity of sensor networks and design WSN-specific security protocols. Since the proposal of the EG scheme, the random probabilistic approaches have gained much attention in secure wireless sensor networks, and many interesting protocols were proposed. Pietro *et al.* [14] questioned the realistic assumption of the random graph model in WSNs, and proposed another geometric random model for WSNs. Wu and Stinson [15] further discussed these models and validated the use of the random graph model in computing the connectivity of WSNs. Nevertheless, random-graph-based analysis is still prevalent in protocols of WSNs.

The EG scheme works as follows.

(1) *Key initialization stage.* Let m denote the number of distinct cryptographic keys that can be stored into a sensor node. Before sensor nodes are deployed, an offline trusted key distribution server generates a *key pool* of S random keys. For each node, m keys are randomly selected from the key pool and stored in the node's memory. This set of m keys is called the node's *key ring*. The number of keys in the key pool, S, is determined by satisfying that two random subsets of size m in S will share at least one key with probability P' such that the whole network can achieve almost full connectivity probability P_c.

(2) *Directly shared key discovery stage*. After deployment, each node tries to discover its neighbors with which it shares common keys. There are many ways to determine whether two nodes share common keys or not. The simplest way is to make the nodes broadcast their key identity lists to other nodes. If a node finds out that it shares at least one common key with a neighborhood node, it can use the first common key for secure communication. Alternatively, the set of keys in the key ring of a node could be bound to the node's ID via a pseudorandom function. In this case, each node only needs to broadcast its ID to its neighbors.

(3) *Path key establishment stage*. A secure link exists between two nodes only if they share a key, but the path key establishment stage facilitates provision of the link between two nodes when they do not share a common key directly. Nodes can set up path keys with nodes in their vicinity that they did not happen to share keys with in their key rings. If the graph is connected, a path can be found from a source node to its neighbor. The source node can then generate a path key and send it securely via the path to the target node.

Chan, Perrig, and Song [16] introduced two variations of the EG Scheme: q-composite random key predistribution and multipath key reinforcement. The *q*-composite scheme requires that two nodes have at least q common keys to set up a link and use all common keys instead of the first one to establish the pairwise key. As the number of overlapped keys between two nodes increases, it becomes harder for an adversary to break their communication link. On the other hand, to maintain the probability that two nodes establish a link with q common keys, it is necessary to reduce the size of the key pool, which poses a possible security breach in the network as the adversary now has to compromise only a few nodes to recover a large portion of key pool. Therefore, the challenge of the q-composite scheme is to choose an optimal value for q while ensuring that security is not sacrificed. However, the optimal value for q is strictly related to the number of nodes that adversaries may capture, which is dynamic and cannot be precisely determined while network parameters are designated. Therefore, the benefit of *q*-composite ($q > 1$) mode might be trivial. The multipath reinforcement scheme is similar to [9], using multipath indirect secure links to exchange secret data to offer better security with additional communication overhead, suitable for occasions where security is more of a concern than bandwidth or power drain.

Liu, Ning, and Li [17] proposed a key predistribution scheme which combines the EG scheme with polynomial-based key predistribution protocol in [18]. During the key initialization stage, a setup server generates a set of bivariate *t*-degree symmetric polynomials $f_l(x, y) = \sum_{i,j=0}^{t} a_{ij} x^i y^j$, where $a_{ij} = a_{ji}, l \in [0, S-1]$ over a finite field $GF(q)$. For each node, say u, a subset of these polynomials are randomly picked up by the server and all specific polynomials $g_{u,i}(y) = f_i(u, y)$ of one variate y are computed and placed into the memory of node u. In the key discovery stage, sensor node u finds every adjacent node, e.g. node v, with which it shares the same original bivariate polynomial and then both nodes can establish a common pairwise key, because $f(u,v)=f(v,u)$. Du *et al.* [19] independently presented a technique which is equivalent to Liu-Ning-Li's scheme. Huang *et al.* [20] further analyzed the performance of polynomial-based key predistribution scheme. In general, the security performance of this kind of scheme overweighs that of the original EG scheme. On the other hand, it should be noticed that operations in those schemes are in finite field $GF(q)$, where q is necessary to be the minimal prime integer greater than the length of secret key k, typically 2^{128}. Sometimes the costly finite field operations may be not very suitable for some extremely resource-constrained sensor nodes.

In general, sensor nodes are randomly scattered into targeted area, thus it is difficult to obtain deployment knowledge of nodes. As a matter of fact, it is a general assumption for characteristic of sensor networks. However, some proposals argued that some information on deployment knowledge can be achieved if the deployment of nodes follows some particular pattern. For example, if sensor nodes are scattered by an airplane, these nodes might be grouped or placed in a particular order before deployment and, based on this pattern, an approximate knowledge of node positions can be acquired. In these scenarios, combined with random key distribution scheme, several schemes were proposed. Those nodes which are more likely to become neighbors are allocated more same source material such that bigger size of key pool still suffice to maintain the same connectivity of global network, which strengthens resilience to node capture. The major issue in those schemes is how to develop suitable node deployment models. Deployment knowledge in [21] is modeled using non-uniform probability density functions (pdfs), which assumes the positions of sensor nodes to be at certain areas. Generally nodes are deployed in groups; therefore the pdfs of the final resident points of all the sensors in a group is highly likely to be the same as the group of sensors deployed in a single deployment point. Other models are addressed in [22, 23]. A reasonable doubt to those schemes is whether or how precisely their models reflect the actual node deployment.

Traynor *et al.* [24] proposed a random key distribution scheme based on the heterogeneous sensor network model. Instead of a homogeneous composition of nodes, this kind of network now consists of a mix of nodes with different capabilities and missions. The Level 1 (L1) nodes are assumed to be very resource-limited in terms of memory and processing capability, and are responsible to perform the task of data collection. By constant, the Level 2 (L2) nodes have more memory, processing ability, and additional radios (e.g., 802.11). These nodes are equipped with additional keys and take on the role of routers and gateways between networks. In addition to tamper-resistant casings, the L2 nodes are assumed to be equipped with a fast encryption/deletion algorithm to protect their supplementary keys from compromise if they are captured. Under this assumption, a scheme for the unbalanced distribution of keys throughout a wireless sensor network builds upon the EG scheme. Intuitively, with more powerful nodes in the sensor network, it definitely can achieve better security performance.

2.4. Key Distribution Using Combinatorial Design

Çamtepe and Yener [25] first proposed deterministic methods using combinatorial design in key distribution of wireless sensor networks. They showed how to map from two classes of combinatorial designs—balanced incomplete block designs and generalized quadrangles—to deterministic key distribution schemes. Chakrabarti, Maitra, and Roy [26] presented a randomized block merging strategy for key predistribution in WSNs. Wei and Wu [27] provided two key predistribution schemes using difference families and all k-subsets of a set. Lee and Stinson [28] discussed how to employ two types of transversal designs, the set of all linear polynomials and the set of quadratic polynomials, to improve the performance of key predistribution schemes by carefully choosing a certain class of set systems as “key ring spaces”.

2.5. Group Key Distribution

Wireless sensor networks are inherently collaborative environments in which sensor nodes often communicate in groups that typically are dynamic. Efficient group key management schemes are demanded for secure communications under this collaborative model. General speaking, many traditional binary-tree-based group key management schemes and broadcast approaches, such as logical key hierarchy, one-way function chain tree, and subset-cover broadcast encryption, can be adapted into wireless sensor networks. Currently many proposed group key management schemes in WSNs are based on exclusion basis systems (EBS), presented by Eltoweissy *et al.* [29], which is a combinatorial formulation of the group key management problem that produces optimal results with respect to the parameters n, k and m, where n is the size of the group, k is the number of keys stored in each member, and m is the exact number of re-key messages to exclude one member.

Exclusion Basis System: Let n, k and m be positive integers, where $1<k,m<n$. An Exclusion Basis System

of dimension (n,k,m), denoted by $EBS(n,k,m)$, is a collection Γ of subsets of $[1,n]$ such that for every integer $t \in [1,n]$, the following two properties hold:

(a) t is in at most k subsets of Γ

(b) There are exactly m subsets, say $A_1, A_2, \ldots, A_m$, in Γ such that $\bigcup_{i=1}^{m} A_i$. is $[1,n]-\{t\}$. (That is, each element t can be excluded by a union of exact m subsets in Γ)

In a collusion-free environment, using EBS for key management guarantees forward and backward secrecy. Eltoweissy *et al.* [29] proved that there exists a positive solution to the $EBS(n,k,m)$ problem, where $k+m$ is equal to the total number of keys, if and only if $\binom{k+m}{k} \geq n$. Apparently, we can trade off between the number of rekeying messages and the number of keys known to each user. Moreover, it suggests that, in general, for arbitrarily large numbers of users, n, there are systems satisfying the properties of $EBS(n,k,m)$ with k and m smaller than the corresponding values of k and m for a binary tree system. However, the binary-tree-based approaches ensure that collusion between users is not possible, whereas an arbitrary EBS needs an external technique to safeguard security through collusion attacks.

Eltoweissy *et al.* [30,31] applied EBS to sensor networks using specific network models. In GKIP [30], all sensor nodes in the network are anonymous and are preloaded with identical state information. This scheme leverages a location-based virtual network infrastructure, combined with EBS. GKIP implements group keys at the granularity of a set of nodes. The set granularity allows for an efficient peer monitoring mechanism within a particular set that enables detecting nodes that infiltrate the network or exhibit suspicious behavior. LOCK [31], localized combinatorial keying, is another dynamic key management scheme based on EBS. The assumed network model consists of a three-level hierarchy, i.e., base station, cluster heads, and sensor nodes. LOCK does not use location information in the generation of keys. When the nodes are initially released into the environment, they create a set of backup keys. These sets of backup keys are only shared with the base station, not with the local cluster leader nodes. If a node is captured, other nodes are rekeyed locally so that the compromised node is unable to communicate with them. If a cluster leader is compromised, the base station initiates a rekeying phase at the cluster head level. Likewise, nodes within the group governed by the compromised cluster leader rekey with the base station. Therefore, if an adversary compromises any node in LOCK, it does not have any effect on the operations of other nodes in other clusters.

In order to reduce the potential of collusion among compromised sensor nodes in the standard EBS system, Younis, Ghumman, and Eltoweissy [30] proposed the SHELL scheme, using node location information to compute keys with the help of clusters and gateways. SHELL gathers node locations after employment and uses this information for assigning keys. Nodes that are located closer to each other share a higher number of keys than nodes that are located longer distance from each other. The clusters in this scheme track key assignments but not the keys themselves. The actual keys are stored in the gateways of other clusters. SHELL exploits the physical proximity of nodes so that a node would share most keys with reachable nodes, and thus very few additional keys would be revealed when compromised nodes collude.

2.6. Public Key Feasibility

The common perception of public key cryptography is that it is complex, slow, power hungry, and not at all suitable for use in ultra-low power environments like wireless sensor networks. Gaubatz, Kaps and Sunar [32] first challenged the basic assertion of public key cryptography infeasibility in sensor networks, based on a traditional software based approach. They proposed a custom hardware assisted approach for which they claim that it makes public key cryptography available in such environments, provided they use the right selection of algorithms and associated parameters, careful optimization, and low-power design techniques.

In the family of public key algorithms, Elliptic Curve Cryptosystem (ECC) and Hyper Elliptic Curve Cryptosystem (HECC) are widely thought of achieving the best balance in terms of speed, memory requirement and security level. Malan, Welsh, and Smith [33] presented the first implementation of elliptic curve cryptography over $GF(2^p)$ for sensor networks based on the 8-bit, 7.3828MHz MICA2 mote. Although the public-key infrastructure has been thought impractical, they argue, through analysis of their implementation for TinyOS of multiplication of points on elliptic curves, that the public-key infrastructure is, in fact, viable for sensor network key distribution, even on the MICA2. They demonstrated that public keys could be generated within 34 seconds and that shared secrets could be distributed among nodes in a sensor network within the same time, using just over 1 kilobyte of SRAM and 34 kilobytes of ROM.

Bertoni, Breveglieri, and Venturi [34] proposed two coprocessor architectures suitable for sensor networks: a 12K gate processor able to perform one $k \cdot P$ operation (i.e., the ECC primitive) over the finite field $GF(2^{163})$ in 17.05 ms, consuming 1.1 mJ of energy, and a 18.5 K gate coprocessor performing the same operation in 14.68 ms but consuming only 0.66 mJ.

Doyle *et al.* [35] examined the practicality of using efficient elliptic curve algorithms and identity-based encryption to deploy a secure sensor network infrastructure. They evaluated the potential for realizing this on low-power, long-life devices by measuring power consumption of the operations needed for key management in a sensor network and provided further evidence for the feasibility of the approach. However, their platform based on ARM7TDMI processor is considerably more powerful than any of the devices that are used in WSNs at the moment.

The applicable implementation of public-key cryptography in typical sensor nodes platform comes with TinyECC [36], NanoECC [37], and TinyPBC [38]. TinyECC is quite useful since it is a configurable library for ECC operations in wireless sensor networks. TinyECC provides a number of optimization switches, which can turn specific optimizations on or off according to developers' needs. Different combinations of the optimizations cost different execution time and resource consumptions, giving developers flexibility in integrating TinyECC into sensor network applications. Liu and Ning presented the design, implementation, and evaluation of TinyECC on several common sensor platforms, including MICAz, Tmote Sky, and Imote2 in [36]. In NanoECC [37], point multiplication in a curve takes 1.27s at 7.3828MHz on MICA2 mote. Pairing-based cryptography (PBC) is an emerging field related to ECC, which has been attracting the interest of international cryptography community, since it enables the design of original cryptographic schemes (such as, identity-base encryption) and makes well-known cryptographic protocols more efficient. TinyPBC is able to compute pairings, the PBC primitive, in around 5.5s on an ATmega128L clocked at 7.3828MHz. Although it appears not very practical, it does show the applicability of pairing-based cryptography in WSNs.

2.7. Discussion

Random key distribution approaches are prevailing at present. However, little analysis about communication overload in these schemes has been conducted. Especially, finding a secure path in a random graph is a NP-complete problem. Most of those schemes just ignored this problem. Rekeying and perfect backward secrecy are also serious issues for those random predistribution schemes. From the practical point of view, group key distribution and public key based might be the tendency. The progress in efficiently implementation of ECC and HECC and advances in sensor hardware would make public key cryptosystem practicable in a few years.

III. Attacks and Countermeasures

Like any wireless ad hoc network, WSNs are suffering from many attacks. In this section, we introduce the major attacks to WSNs and countermeasures.

3.1. Secure Routing

Routing is a basic functionality of any network, and there are various attacks and countermeasures for WSNs. Sybil attack and wormhole attack are two of major routing attacks specifically for WSNs.

Karlof and Wagner [39] first considered routing security in wireless sensor networks systematically. They addressed security goals for routing in sensor networks, showed how attacks against ad-hoc and peer-to-peer networks can be adapted into powerful attacks against sensor networks, introduced two classes of new attacks against sensor networks—sinkholes and HELLO floods, and analyzed the security of all major sensor network routing protocols. Sink is an alias of base station in sensor networks. In a sinkhole attack, the adversary's goal is to lure nearly all the traffic from a particular area through a compromised node, creating a metaphorical sinkhole with the adversary at the center. Because nodes along or near the path that packets follow have many opportunities to tamper with application data, sinkhole attacks can enable many other attacks. HELLO floods attack can be thought of as one-way, broadcast wormholes. If a laptop-class adversary has a powerful transmitter, it can use a HELLO flood attack to broadcast a routing update loud enough to reach the entire network, causing every node to attempt to use this route, but those nodes sufficiently far away from the adversary would be sending packets into oblivion. They also described crippling attacks against all of them and suggest countermeasures and design considerations.

Sybil Attack:

Sybil attack is a harmful threat to sensor networks, in which a malicious node illegally forges an unbounded number of identities. The Sybil attack can disrupt normal functioning of the sensor network, such as the multipath routing, used to explore the multiple disjoint paths between source-destination pairs. Douceur [40] first presented the Sybil attack problem in the peer-to-peer distributed systems. He pointed out that it could defeat the redundancy mechanisms of the distributed storage systems. Newsome *et al.* [41] analyzed the threat posed by the Sybil attack to wireless sensor networks. They established a classification of different types of Sybil attack, proposed several techniques to defend against the Sybil attack, and analyzed their effectiveness quantitatively.

Zhang *et al.* [42] proposed a lightweight identity certificate method used to thwart Sybil attack. This method utilizes a two-level Merkle hash tree to create certificates. Each sensor node is pre-assigned a unique secret key to derive one-way key chains. An identity certificate is also distributed to each node, which associates the node's identity with its one-way key chain. To securely demonstrate its identity, a node first presents its identity certificate, and then proves that it possesses or matches the associated unique information. An extension of this method exploits node deployment knowledge to reduce the computation overhead at each node. However, the scalability problem of this method adversely affects its use in the large-scale sensor network.

Yin and Madria [43] proposed a lightweight Sybil attack detection method based on a hierarchical architecture in sensor networks. This method also uses a two-level Merkle hash tree to create certificates. A high-level certificate allows a node's identity to be proved to other nodes that it needs to communicate with. A node creating a false identity will not be able to easily forge an identity certificate because the result of a identity verification calculation must match commitment, which is publicly known, according to the properties of the Merkle hash tree. The low-level identity certificate makes this proof specific to a single receiving node.

Wormhole Attack:

Since sensors use a radio channel to send information, malicious nodes can eavesdrop the packets, tunnel them to another location in the network, and re-transmit them. This generates a false scenario that the original sender is in the neighborhood of the remote location. The tunneling procedure forms a wormhole attack. In [44], Wang and Bhargava proposed a mechanism, MDS-VOW, to detect wormholes in sensor networks. MDS-VOW first reconstructs the layout of the sensors using multi-dimensional scaling. Then MDS-VOW detects the wormhole by visualizing the anomalies introduced by the attack. The anomalies, which are caused by the fake connections through the wormhole, bend the reconstructed surface to pull the sensors that are far away to each other. Through detecting the bending feature, the wormhole is located and the fake connections are identified. Yun *et al.* [45] proposed another countermeasure named WODEM against the wormhole attack. In WODEM, a few detector nodes equipped with location-aware devices and longer-lasting batteries are responsible to discover wormholes, and normal sensor nodes are only required to forward control packets from the detector nodes. Then a pair of detectors can detect the wormhole attack between them.

3.2. DoS Attack

Denial of service (DoS) attack is a pervasive threat to most networks. Due to the characteristics of energy-sensitiveness and resource-limitedness, sensor networks are exceedingly vulnerable to DoS attack. Wood and Stankovic [46] explored various DoS attacks that may happen to every network layer of sensor networks. In [47], Kim, Doh, and Chae proposed a DoS detection method via entropy estimation on hierarchical sensor networks reflecting resource constraints of sensors. In order to enhance the accuracy of detection even in the various deployments of attack agents, they deployed hierarchically entropy estimators according to network topology, and a main estimator synthesizes localized computation. This entropy estimator is simplified by only multiplication calculation instead of logarithm, in addition to providing higher estimation precision of entropy compared to the conventional entropy estimation.

3.3. Node Clone Attack

Sensor nodes deployed in hostile environments are vulnerable to capture and compromise. An adversary may extract secret information from these sensors, clone and intelligently deploy them in the network to launch a variety of insider attacks. Chan and Perrig [48] cataloged a number of attacks that can be launched using replicated nodes .

Parno, Perrig, and Gligor [49] provided two probabilistic algorithms to detect node clone. They assumed that every node is aware of its geographic coordinate's location and broadcasts the information to specific witnesses. Randomized multicast distributes node location information to randomly-selected witnesses, exploiting the birthday paradox to detect replicated nodes, while line-selected multicast uses the topology of the network to detect replication, that is, in addition to witness nodes, the nodes within the multicast path check the node replication. Apparently, both of them are very communication-consuming.

SET, proposed by Choi, Zhu, and Porta [50], is to detect node replication by computing set operations (intersection and union) of exclusive subsets in the network. First, SET securely forms exclusive unit subsets among one-hop neighbors in the network in a distributed way. This secure subset formation also provides the authentication of nodes' subset membership. SET then employs a tree structure to compute non-overlapped set operations and integrates interleaved authentication to prevent unauthorized falsification of subset information during forwarding. Randomization is used to further make the exclusive subset and tree formation unpredictable to an adversary.

Brooks *et al.* [51] propose a clone detection protocol in the context of random key predistribution (Section 2.3). The basic idea is that keys that are present on the cloned nodes are detected by looking at how often they are used to authenticate nodes in the network. First, each node makes a counting Bloom filter of the keys it uses to communicate with neighboring nodes and appends a nonce. Then Bloom filter and nonce are transferred to base station, which will count the number of times each key is used in the network. Keys used above a threshold value are considered cloned.

Bekara and Laurent-Maknavicius [52] describe a deterministic node clone detection protocol based on the initial trust assumption (Section 2.2). They also suppose that nodes are not mobile. Therefore, the cloned nodes of former generations cannot request for key establishment.

3.4. Intrusion Detection and Intrusion Tolerance

Agah *et al.* [53] proposed an intrusion detection framework of sensor networks using game theory. They applied three different schemes for defense. The main concern in all three schemes is to find the most vulnerable node in a sensor network and protect it. In the first scheme, they formulated attack-defense problem as a two-player, nonzero-sum, non-cooperative game between an attacker and a sensor network. This game achieves Nash equilibrium [53] and thus leading to a defense strategy for the network. In the second scheme, they used Markov decision process to predict the most vulnerable senor node. In the third scheme, they utilized an intuitive metric (node's traffic) and protected the node with the highest value of this metric.

Based on the DESERT tool, which has been proposed for component-based software architectures, Inverardi, Mostarda, and Navarra [54] derived a framework that permits to dynamically enforce a set of properties of the sensors behavior. This is accomplished by an IDS specification that is automatically translated into few lines of code installed in the sensors. This realizes a distributed system that locally detects violation of the sensors interactions policies and is able to reduce the information sent among sensors in order to discover attacks over the network.

Deng, Han, and Mishra [55] described an INtrusion-tolerant routing protocol for wireless SEnsor NetworkS (INSENS). INSENS constructs forwarding tables at each node to facilitate communication between sensor nodes and base stations. It decreases computation, communication, storage, and bandwidth requirements at the sensor nodes with the expense of increased computation, communication, storage, and bandwidth requirements at the base station. A desired property of INSENS is that while a malicious node may be able to compromise a small number of nodes in its vicinity, it should not cause widespread damage in the network.

3.5. Discussion

The attacks, detection, tolerance and countermeasures are highly application-specific. To effectively resist and detect those attacks, security consideration must be taken in account into various protocols of sensor networks from the very beginning. Many current security protocols have good performance to resist those attacks. For instance, in [56], the authors demonstrated the efficacy of their LBKs scheme in counteracting several notorious attacks against sensor networks such as Sybil, identity replication, wormhole, and sinkhole attacks.

Time synchronization and sensor location are critical to many sensor network applications. There are considerable schemes regarding secure time synchronization and secure localization in wireless sensor networks. Most of them, however, are not cryptographic approaches. Many schemes in these two topics are discussed in [57].

IV. Authentication

Authentication is one of the most important security primitives. Simply speaking, authentication is a mechanism by which some means is provided to guarantee that entities are who they claim to be, or that information has not been manipulated by unauthorized parties. In fact, authentication is specific to the security objective which one is trying to achieve. Examples of specific objectives are message authentication (data origin authentication), entity authentication (identification), access control, data integrity, non-repudiation, and key authentication. In this section, we cover two principal categories of authentication in wireless sensor networks, broadcast message authentication and entity authentication.

4.1. Broadcast Message Authentication

A broadcast message authentication scheme permits any targeted node to verify the authenticity of the source of broadcasted messages. This can be achieved using digital signatures if public key cryptography is used, or if only symmetric cryptography is used, by appending verifiable authentication data, consisting of multiple shared-secrets based message authentication codes (MAC). Due to the properties of sensor networks, broadcast authentication is more pervasive than one-to-one message authentication, and there are a number of schemes to achieve broadcast authentication in sensor networks.

The μTESLA protocol, proposed by Perrig *et al.* [6], is the "micro" version of TESLA (Timed Efficient Stream Loss-tolerant Authentication) [58]. It emulates asymmetry through a delayed disclosure of symmetric keys and serves as the broadcast authentication service of SNEP[6]. μTESLA requires that the base station and the nodes are loosely time-synchronized, and that all nodes know an upper bound on the maximum synchronization error. For an authenticated packet to be sent, the base station computes a MAC on the packet with the key that is secret at that point of time. When a node receives a packet, it can confirm that the base station has not yet disclosed the corresponding MAC key, according to its loosely synchronized clock, maximum synchronization error and the time at which the keys are to be disclosed. The node stores the packet in its buffer. When the keys are to be disclosed, the base station broadcasts the key to all receivers. Each MAC key is a member of a key chain, which has been generated by a one-way function H. In order to generate this chain, the base station chooses the last key K_n of the chain randomly, and applies F repeatedly to compute all other keys: $K_i = H(K_{i+1}), i \in [1, n-1]$. The nodes, which hold K_1, can verify the correctness of the key and use it to authenticate the packet stored in the buffer.

Liu and Ning [59] presented a series of techniques to extend the capabilities of μTESLA. The basic idea is to predetermine and broadcast the initial parameters required by μ TESLA instead of unicast-based message transmission. In the simplest form, this extension distributes the μ TESLA parameters during the initialization of the sensor nodes. To provide more flexibility, especially to prolong the lifetime of μTESLA without requiring a very long key chain, they introduced a multi-level key chain scheme, in which the higher-level key chains are used to authenticate the commitments of lower-level ones. To further improve the survivability of the scheme against message loss and DoS attacks, they used redundant message transmissions and random selection strategies to deal with the messages that distribute key chain commitments. The resulting scheme, which is named multi-level μ TESLA, removes the requirement of unicast-based initial communication between base station and sensor nodes while keeping the nice properties of μTESLA. μTESLA is also extended by Liu *et al.* [60] to support multiuser scenario but the scheme assumes that each sensor node only interacts with a very limited number of users.

Schemes based on delayed key disclosure, like μTESLA, can suffer from DoS attack. During the subsequent interval when the message is in the buffer and the receiver waits on the disclosure time, an attacker can flood the network with arbitrary messages, claiming that they belong to the current time interval. Only in the next time interval can the nodes determine that these messages are not authentic. The use of public key cryptography could eliminate the need for such complicated protocols, increasing the security of the system and only requiring the public key of the base station to be embedded into all of nodes. Ren, *et al.* [61,62] presented several public-key-based schemes to achieve immediate broadcast authentication and thus avoid the security vulnerability intrinsic to μTESLA-like schemes. Those schemes are built upon the unique integration of several cryptographic techniques, including the Bloom filter, the partial message recovery signature scheme, and the Merkle hash tree.

Kondratieva and Seo [63] studied the problem of optimizing the authentication tree structure for sensor network environments. The procedure for finding the tree structure is formalized, in which the number of nodes with the longest authentication path length is made minimal. An algorithm for hash tree generation is introduced and it is proven that the proposed tree structure is optimal. The optimization of the authentication procedure is achieved by proposing an indexing scheme, supported by the least path protocol.

μ TESLA-family is interesting and can be applied to some WSN application areas. However, as public-key primitives become more and more feasible in WSNs, many WSNs would employ asymmetric approaches. The performance of current public-key-based broadcast authentication is far from satisfaction, and more researches are demanded.

4.2. Entity Authentication

Access control is a classical problem in many existing computer systems and applications. To achieve access control in wireless sensor networks, it is essential to authenticate the identities of users.

Benenson, Gedicke, and Raivio [64] proposed an entity authentication scheme of WSNs, based on elliptic curve cryptography. There is a certification authority (CA), which issues certificate for legitimate users' public keys. All nodes keep a copy of CA's public key; thus legitimate users can easily authenticate themselves to nodes via their certificates. This scheme details how to issue authenticated query, withstanding node capture using redundancy. Their main idea is that the nodes in the user's communication arrange interact with the user by public key cryptography, and then serve as gateways, on behalf of the user, to communicate with other targeted nodes by symmetric cryptography. They argued that the scheme is resilient to node capture since there are many sensor nodes in the vicinity of the user. However, this reasoning is rather questionable, and the benefit of gateways appears to be trivial while overall communication cost is definitely high.

Jiang and Xu [65] presented a distributed entity authentication scheme in wireless sensor networks. It is based on the self-certified keys cryptosystem, which is modified to use elliptic curve cryptography to establish pair-wise keys for use in the entity authentication scheme. The neighbor nodes of a user collaborate to determine whether the user is successfully authenticated or not, using threshold voting. However, the existence of this threshold mechanism lacks convincible reasons.

Wong *et al.* [66] proposed a dynamic strong-password-based entity authentication scheme for wireless sensor networks. It allows legitimate users to query data from any sensor nodes in an ad-hoc manner, and imposes lower computational overload than the two protocols above since it only requires symmetric primitives, such as one-way hash function and exclusive-or operations. Tseng, Jan, and Wang [67] enhanced Wong *et al.*'s scheme to thwart potential replay and forgery attacks. It also allows legitimate users to choose and change their passwords freely.

Tripathy and Nandi [68] used cellular automata based components to achieve entity authentication. Cellular automata is a dynamic system consisting of a grid of identical finite state machines, whose states are updated synchronously at discrete time steps according to a local update rule. The proposed security component is to achieve threshold authentication and group key establishment as a suitable alternate to countermeasure the node capture attacks. Information is distributed among several nodes and user can determine the correct answer only if at least some certain correct responses are obtained. All of these schemes above are based on conventional cryptography, symmetric or public-key. In fact, if this kind of cryptographic primitive is allowed, there are plenty of general entity authentication schemes that can be applied to sensor networks. The main issue left is how to distribute and manage secret-keys, passwords, public-key certificate effectively and efficiently in the environment of sensor networks. Some ultra-light entity authentication may be very useful in some wireless sensor networks.

V. Secure Data Aggregation

In many applications of wireless sensor networks, the base station is more interested in aggregated data than exact individual values from all sensors. By aggregating data, it is also greatly helpful to reduce the amount of data to be transmitted for conserving valuable energy. Indeed, current in-network aggregation schemes are beneficial to communication energy consumption but they are designed without considering possible security issues. Furthermore, wireless sensor networks are often designed with neighbor nodes sharing keys or with decryption at aggregator nodes. In either situation the potential for aggregator nodes to be physically compromised means that data confidentiality is at high risk. Therefore secure data aggregation is desirable where data can be aggregated without the need for decryption at aggregator nodes. Aggregation becomes especially challenging if end-to-end confidentiality between a source and a destination is required.

Hu and Evans [69] proposed a secure hop-by-hop data aggregation scheme. In their scheme, individual packets are aggregated in some pattern so that the base station can detect non-authorized inputs. On the other hand, their solution introduces a considerable communication overhead per packet. Moreover, they assumed that only leaf nodes under a tree-like network topology sense data, whereas the intermediate nodes do not have their own data readings, which is a little unrealistic, or at least too restricted. Jadia and Muthuria [70] extended the Hu-Evans scheme. Instead of relying on keys shared between the base station and sensor nodes for authentication, Jadia-Muthuria scheme makes use of one-hop as well as two-hop pairwise keys. It is intended to replace the data validation step of the Hu-Evans scheme with some other mechanism that does not require unnecessary key reception by all nodes. Those two schemes are resistent to only a single inside malicious node and outside intruder devices.

Yang, *et al.* [71] proposed SDAP, a secure hop-by-hop data aggregation protocol for sensor networks, using the principles of divide-and-conquer and commit-and-attest. In SDAP, a probabilistic grouping technique is utilized to dynamically partition the nodes in a tree topology into subtrees. A commitment-based hop-by-hop aggregation is conducted in each subtree to generate a group aggregate. The base station identifies the suspicious subtrees based on the set of group aggregates. Finally, each subtree under suspect participates in an attestation procedure to prove the correctness of its group aggregate. Feng *et al.* [72] proposed a family of secret perturbation-based schemes that protect sensed information confidentiality without disrupting the data aggregation.

Several secure aggregation algorithms have been proposed under the scenario that there is a certain class of node called aggregator. Przydatek, Song, and Perrig [73] proposed secure information aggregation (SIA) to identify forged aggregation values from all sensor nodes in a network. In SIA scheme, aggregators compute an aggregation result over the raw data together with a commitment to the data based on a Merkle-hash tree and send data to a trustable remote user, who later challenges the aggregators to verify the aggregation. They assumed that the bandwidth between a remote user and aggregators is a bottleneck in this scenario. Therefore the SIA scheme is intended to reduce this communication overhead while providing a mechanism to detect with high probability if aggregators are compromised.

Homomorphic encryption [74] is semantically-secure encryption which, in addition to standard guarantees, has additional properties, e.g. the sum of any two encrypted values is equal to the encrypted sum of the values. There are several efficient homomorphic cryptosystems, such as Unpadded RSA, El-Gamal, Goldwasser-Micali, Benaloh and Paillier [74]. Using homomorphic encryption, Kifayat *et al.* [75] presented the extended structure and density independent group based key management protocol (SADI-GKM) with the additional feature of secure data aggregation to provide better data confidentiality to every single node in a large scale wireless sensor network. Ren, Kim, and Park [76] also proposed a secure data aggregation scheme which supports end-to-end encryption using homomorphic encryption as well as hop-by-hop verification using ECC-based MAC.

Current proposed secure data aggregation schemes are rather elementary and more practical schemes are demanded. It is worthwhile to pay more attention how to apply homomorphic encryption to secure aggregation effectively. In the meantime, it is of great help to focus on specific popular aggregation protocols of WSNs to design realistic secure aggregation.

VI. Privacy

One challenge threatening the successful deployment of sensor networks is privacy. Although many privacy-related issues can be addressed by security mechanisms, one exception is source-location privacy. Adversaries may use RF localization techniques to perform hop-by-hop traceback to the source sensor's location.

Kamat *et al.* [77] provided a formal model for the source-location privacy problem in sensor networks and examined the privacy characteristics of different sensor routing protocols. They inspected two popular classes of routing protocols: flooding-based routing protocols, and the routing protocols involving only a single path from the source to the sink. They devised new techniques to enhance source-location privacy that augment these routing protocols. One of strategies, a technique called phantom routing, has been shown relatively flexible and capable of protecting the source's location, while not incurring a noticeable increase in energy overhead.

The model of k-anonymity is one of major mechanisms in protecting privacy. Gedik and Ling [78] described a personalized k-anonymity model for protecting location privacy against various privacy threats through location information sharing. First, they provided a unified privacy personalization framework to support location k-anonymity for a wide range of users with context-sensitive personalized privacy requirements. This framework enables each mobile node to specify the minimum level of anonymity it desires as well as the maximum temporal and spatial resolutions it is willing to tolerate when requesting for k-anonymity preserving location-based services (LBSs). Second, they devised an message perturbation engine which is run by the location protection broker on a trusted server and performs location anonymity on mobile users' LBS request messages, such as identity removal and spatio-temporal cloaking of location information. They developed a suite of C-temporal cloaking algorithms, called Clique Cloak algorithms, to provide personalized location k-anonymity, intending to avoid or reduce known location privacy threats before forwarding requests to LBS providers.

Jian *et al.* [79] proposed a location privacy routing protocol (LPR) that provides path diversity and protects receiver-location privacy in WSNs. Combining with fake packet injection, LPR is able to reduce the traffic direction information that an adversary can retrieve from eavesdropping. By making the directions of both incoming and outgoing traffic at a sensor node uniformly distributed, this system makes it hard for an adversary to perform analysis on locally gathered information and infer the direction to which the receiver locates.

VII. Conclusion

As WSNs grow in application area and are used more frequently, the need for security in them becomes inevitable and vital. However, the inherent characteristics of WSNs incur constraints to of sensor nodes, such as limited energy, processing capability, and storage capacity, etc. These constraints make WSNs very different from traditional wireless networks. Consequently, many innovative security protocols and techniques have been developed to meet this challenge. In this paper, we outline security and privacy issues in sensor networks, address the state of the art in sensor network security, and discuss some future directions for research.

Acknowledgments

The research is supported by NSERC Strategic Project grants.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[2] T. Arampatzis, J. Lygeros, and S. Manesis, “A Survey of Applications of Wireless Sensors and Wireless Sensor Networks,” in *Proceedings of the 13th Mediterranean Conference on Control and Automation*, 2005, pp. 719–724.

[3] Y.-M. Huang, M.-Y. Hsieh, and F. E. Sandnes, “Wireless Sensor Networks and Applications,” in *Sensors, Advancements in Modeling, Design Issues, Fabrication and Practical Applications*, 2008, pp. 199–219.

[4] W. Y. Chang, “Wireless Sensor Networks and Applications,” in *Network-Centric Service-Oriented Enterprise*, 2008, pp. 157–209.

[5] G.-Z. Yang and M. Yacoub, *Body Sensor Networks*.Springer, 2006.

[6] A. Perrig, R. Szewczyk, V. W. D. Culler, and J. D. Tygar, “SPINS: Security protocols for sensor networks,” in *Proceedings of the Annual International Conference on Mobile Computing and Networking (MOBICOM)*.Rome Italy: IEEE, 2001, pp. 189–199.

[7] S. Zhu, S. Setia, and S. Jajodia, “LEAP : efficient security mechanisms for large-scale distributed sensor networks,” in *Proceedings of the 10th ACM conference on Computer and Communication Security (CCS’ 03)*.Washington D.C.: ACM, 2003, pp. 62–72.

[8] B. Dutertre, S. Cheung, and J. Levy, “Lightweight Key Management in Wireless Sensor Networks by Leveraging Initial Trust,” SRI International, Tech. Rep. SRI-SDL-04-02, April 6 2004.

[9] R. Anderson, H. Chan, and A. Perrig, “Key Infection : Smart Trust for Smart Dust,” in *Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP’04)*, 2004, pp. 206–215.

[10] J.-B. Hwang, Y.-S. Hwang, and J.-W. Han, “Consideration of efficient nearest node discovering mechanisms for Key Infection,” in *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference*, vol. 3, 2006, pp. 1686–1689.

[11] L. Eschenauer and V. D. Gligor, “A key-management scheme for distributed sensor networks,” in *Proceedings of the 9th ACM conference on Computer and Communications Security*, Washington, DC, USA, 2002, pp. 41–47.

[12] J. Spencer, *The Strange Logic of Random Graphs*, ser. Algorithms and Combinatorics.Springer-Verlag, 2001, vol. 22.

[13] P. Erdös and A. Rényi, “On the evolution of random graphs,” *Bulletin of the Institute of International Statistics*, vol. 38, pp. 343–347, 1961.

[14] R. D. Pietro, L. V. Mancini, A. Mei, A. Panconesi, and J. Radhakrishnan, “Connectivity properties of secure wireless sensor networks,” in *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*.Washington DC, USA: ACM, 2004.

[15] J. Wu and D. R. Stinson, “Minimum node degree and k-connectivity for key predistribution schemes and distributed sensor networks,” in *Proceedings of the First ACM Conference on Wireless Network Security (WiSec’08)*, Alexandria, Virginia, USA, 2008.

[16] H. Chan, A. Perrig, and D. Song, “Random key predistribution schemes for sensor networks,” in *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, ser. 2003 IEEE Symposium on Security And Privacy.Berkeley, CA, United States: Institute of Electrical and Electronics Engineers Inc., 2003, pp. 197–213.

[17] D. Liu, P. Ning, and R. Li, “Establishing Pairwise Keys in Distributed Sensor Networks,” *ACM Transactions on Information and System Security (TISSEC)*, vol. 8, no. 1, pp. 41 – 77, 2005.

[18] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, “Perfectly-Secure Key Distribution for Dynamic Conferences,” *Information and Computation*, vol. 164, no. 1, pp. 1–23, 1998.

[19] W. Du, J. Deng, Y. S. Han, P. Varshney, J. Katz, and A. Khalili, “A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks,” *ACM Transactions on Information and System Security (TISSEC)*, 2005.

[20] D. Huang, M. Mehta, A. v. d. Liefvoort, and D. Medhi, “Modeling Pairwise Key Establishment for Random Key Predistribution in Large-Scale Sensor Networks,” *IEEE/ACM Transactions on Networking*, vol. 15, no. 5, pp. 1204 – 1215, 2007.

[21] D. Liu and P. Ning, “LocationBased Pairwise Key Establishments for Static Sensor Networks,” in *Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, ser. Conference on Computer and Communications Security, Fairfax, Virginia, 2003, pp. 72 – 82.

[22] W. Du, J. Deng, Y. S. Han, and P. K. Varshney, “A key predistribution scheme for sensor networks using deployment knowledge,” *IEEE Transactions on*

Dependable and Secure Computing, vol. 3, no. 1, pp. 62–77, 2006.

[23] J. Y. Chun, Y. H. Kim, J. Lim, and D. H. Lee, "Location-aware Random Pair-wise Keys Scheme forWireless Sensor Networks," in *Third International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing (SECPerU 2007)*, 2007, pp. 31–36.

[24] P. Traynor, R. Kumar, H. Choi, G. Cao, S. Zhu, and T. La Porta, "Efficient Hybrid Security Mechanisms for Heterogeneous Sensor Networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 6, pp. 663–677, 2007.

[25] S. S. Çamtepe and B. Yener, "Combinatorial Design of Key Distribution Mechanisms for Wireless Sensor Networks," *IEEE/ACM Transactions on Networking*, vol. 15, no. 2, pp. 346–358, 2007.

[26] D. Chakrabarti, S. Maitra, and B. Roy, "A Key Pre-distribution Scheme for Wireless Sensor Networks: Merging Blocks in Combinatorial Design," in *Information Security*.LNCS 3650, 2005, pp. 89–103.

[27] R. Wei and J. Wu, "Product Construction of Key Distribution Schemes for Sensor Networks," in *Selected Areas in Cryptography*.LNCS 3357, 2005, pp. 280–293.

[28] J. Lee and D. R. Stinson, "On the Construction of Practical Key Predistribution Schemes for Distributed Sensor Networks Using Combinatorial Designs," *ACM Transactions on Information and System Security (TISSEC)*, vol. 11, no. 2, pp. 1–35, 2008.

[29] M. Eltoweissy, M. H. Heydari, L. Morales, and I. H. Sudborough, "Combinatorial Optimization of Group Key Management," *Journal of Network and Systems Management*, vol. 12, no. 1, pp. 33–50, 2004.

[30] M. Eltoweissy, A. Wadaa, S. Olariu, and L. Wilson, "Group key management scheme for large-scale sensor networks," *Ad Hoc Networks*, vol. 3, no. 5, pp. 668–688, 2005.

[31] M. Eltoweissy, M. Moharrum, and R. Mukkamala, "Dynamic key management in sensor networks," *IEEE Communications*, vol. 44, no. 4, pp. 122–130, 2006.

[32] G. Gaubatz, J.-P. Kaps, and B. Sunar, "Public key cryptography in sensor networks-revisited," in *1st European Workshop on Security in Ad-Hoc and Sensor Networks (ESAS 2004)*, 2004.

[33] D. J. Malan, M. Welsh, and M. D. Smith, "A public-key infrastructure for key distribution in TinyOS based on elliptic curve cryptography," in *First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (IEEE SECON 2004)*, 2004, pp. 71–80.

[34] G. Bertoni, L. Breveglieri, and M. Venturi, "ECC Hardware Coprocessors for 8-bit Systems and Power Consumption Considerations," in *Third International Conference on Information Technology: New Generations (ITNG 2006)*, 2006, pp. 573–574.

[35] B. Doyle, S. Bell, A. F. Smeaton, K. McCusker, and N. E. O'Connor, "Security Considerations and Key Negotiation Techniques for Power Constrained Sensor Networks," *The Computer Journal*, vol. 49, no. 4, pp. 443–453, 2006.

[36] A. Liu and P. Ning, "TinyECC: A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks," in *International Conference on Information Processing in Sensor Networks (IPSN '08)*, 2008, pp. 245–256.

[37] P. Szczechowiak, L. Oliveira, M. Scott, M. Collier, and R. Dahab, "NanoECC: Testing the Limits of Elliptic Curve Cryptography in Sensor Networks," in *Wireless sensor networks*.LNCS 4913, 2008, pp. 305–320.

[38] L. B. Oliveira, M. Scott, J. Lopez, and R. Dahab, "TinyPBC: Pairings for authenticated identity-based non-interactive key distribution in sensor networks," in *Networked Sensing Systems, 2008. INSS 2008. 5th International Conference on*, 2008, pp. 173–180.

[39] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," in *Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications*, 2003, pp. 113–127.

[40] J. R. Douceur, "The Sybil Attack," in *First International Workshop on Peer-to-peer Systems (IPTPS' 02)*.LNCS 2429, 2002, pp. 251–260.

[41] J. Newsome, E. Shi, D. Song, and A. Perrig, "The Sybil attack in sensor networks: Analysis and defenses," in *Third International Symposium on Information Processing in Sensor Networks, IPSN 2004*, Monterey, CA, United States, 2004, pp. 259–268.

[42] Q. Zhang, P. Wang, D. S. Reeves, and P. Ning, "Defending against Sybil attacks in sensor networks," in *25th IEEE International Conference on Distributed Computing Systems Workshops (ICDCS 2005 Workshops)*, 2005, pp. 185–191.

[43] J. Yin and S. K. Madria, "Sybil attack detection in a hierarchical sensor network," in *Third International Conference on Security and Privacy in Communications Networks and the Workshops (SecureComm 2007)*, 2007, pp. 494–503.

[44] W. Wang and B. Bhargava, "Visualization of Wormholes in Sensor Networks," in *Proceedings of the 2004 ACM workshop on Wireless security*, Philadelphia, PA, USA, 2004, pp. 51 – 60.

[45] J.-H. Yun, I.-H. Kim, J.-H. Lim, and S.-W. Seo, "WODEM: Wormhole Attack Defense Mechanism in Wireless Sensor Networks," in *Ubiquitous Convergence Technology (ICUCT 2006)*.LNCS 4412, 2007, pp. 200–209.

[46] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," *Computer*, vol. 35, no. 10, pp. 54–62, 2002.

[47] M. Kim, I. Doh, and K. Chae, “Denial-of-Service(DoS) Detection through Practical Entropy Estimation on Hierarchical Sensor Networks,” in *The 8th International Conference Advanced Communication Technology (ICACT 2006)*, vol. 3, 2006, pp. 1562–1566.

[48] H. Chan and A. Perrig, “Security and privacy in sensor networks,” *Computer*, vol. 36, no. 10, pp. 103–105, 2003.

[49] B. Parno, A. Perrig, and V. Gligor, “Distributed Detection of Node Replication Attacks in Sensor Networks,” in *Proceedings of the IEEE Symposium on Security and Privacy*, 2005, pp. 49 – 63.

[50] H. Choi, S. Zhu, and T. F. La Porta, “SET: Detecting node clones in sensor networks,” in *Third International Conference on Security and Privacy in Communications Networks and the Workshops (SecureComm 2007)*, 2007, pp. 341–350.

[51] R. Brooks, P. Y. Govindaraju, M. Pirretti, N. Vijaykrishnan, and M. T. Kandemir, “On the Detection of Clones in Sensor Networks Using Random Key Predistribution,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1246–1258, 2007.

[52] C. Bekara and M. Laurent-Maknavicius, “A New Protocol for Securing Wireless Sensor Networks against Nodes Replication Attacks,” in *Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMOB 2007)*, 2007, pp. 59–59.

[53] A. Agah, S. K. Das, K. Basu, and M. Asadi, “Intrusion Detection in Sensor Networks: a Non-Cooperative Game Approach,” in *Proceedings of Third IEEE International Symposium on the Network Computing and Applications (NCA’04)*.IEEE Computer Society, 2004, pp. 343 – 346.

[54] P. Inverardi, L. Mostarda, and A. Navarra, “Distributed IDSs for enhancing Security in Mobile Wireless Sensor Networks,” in *20th International Conference on Advanced Information Networking and Applications (AINA 2006)*, vol. 2, 2006, pp. 116–120.

[55] J. Deng, R. Han, and S. Mishra, “INSENS: Intrusion-Tolerant Routing in Wireless Sensor Networks,” University of Colorado, Department of Computer Science, Tech. Rep. Technical Report CU-CS-939-02, 2003.

[56] Y. Zhang, W. Liu, W. Lou, and Y. Fang, “Location-based compromise-tolerant security mechanisms for wireless sensor networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 247–260, 2006.

[57] R. Poovendran, C. Wang, and S. Roy, *Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks* .Springer Verlag, 2007.

[58] A. Perrig, R. Canetti, J. D. Tygar, and D. Song, “Efficient authentication and signing of multicast streams over lossy channels,” in *Proceedings of IEEE Symposium on Security and Privacy*, Berkeley, CA, USA, 2000, pp. 56–73.

[59] D. Liu and P. NIing, “Multi-Level μ TESLA: Broadcast Authentication for Distributed Sensor Networks,” in *Proceedings of the 10th Annual Network and Distributed Systems Security Symposium*, 2003, pp. 263–276.

[60] D. Liu, P. Ning, S. Zhu, and S. Jajodia, “Practical broadcast authentication in sensor networks,” in *The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous 2005)*, 2005, pp. 118–129.

[61] K. Ren, W. Lou, and Y. Zhang, “Multi-user Broadcast Authentication in Wireless Sensor Networks,” in *4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON ’07)*, 2007, pp. 223–232.

[62] K. Ren, W. Lou, K. Zeng, and P. J. Moran, “On Broadcast Authentication in Wireless Sensor Networks,” *IEEE Transactions on Wireless Communications*, vol. 6, no. 11, pp. 4136–4144, 2007.

[63] V. Kondratieva and S.-W. Seo, “Optimized Hash Tree for Authentication in Sensor Networks,” *Communications Letters, IEEE*, vol. 11, no. 2, pp. 149–151, 2007.

[64] Z. Benenson, N. Gedicke, and O. Raivio, “Realizing robust user authentication in sensor networks,” in *Real-World Wireless Sensor Networks (REALWSN)*, 2005.

[65] C. Jiang, B. Li, and H. Xu, “An Efficient Scheme for User Authentication in Wireless Sensor Networks,” in *21st International Conference on Advanced Information Networking and Applications Workshops (AINAW ’07)*, 2007, pp. 438–442.

[66] K. H. Wong, Y. Zheng, J. Cao, and S. Wang, “A Dynamic User Authentication Scheme for Wireless Sensor Networks,” in *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC’06)*, 2006, pp. 244–251.

[67] H.-R. Tseng, R.-H. Jan, and W. Yang, “An Improved Dynamic User Authentication Scheme for Wireless Sensor Networks,” in *IEEE Global Telecommunications Conference (GLOBECOM ’07)*, 2007, pp. 986–990.

[68] S. Tripathy and S. Nandi, “Defense against outside attacks in wireless sensor networks,” *Computer Communications*, vol. 31, no. 4, pp. 818–826, 2008.

[69] L. Hu and D. Evans, “Secure aggregation for wireless networks,” in *Proceedings of the 2003 Symposium on Applications and the Internet Workshops (SAINT’03 Workshops)*, 2003, pp. 384 – 391.

[70] P. Jadia and A. Mathuria, “Efficient Secure Aggregation in Sensor Networks,” in *High Performance Computing (HiPC 2004)*.LNCS 3296, 2004, pp. 40–49.

[71] Y. Yang, X. Wang, S. Zhu, and G. Cao, “A Secure Hop-by-Hop Data Aggregation Protocol for Sensor

Networks," in *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, 2006, pp. 356 – 367.

[72] T. Feng, C. Wang, W. Zhang, and L. Ruan, "Confidentiality Protection for Distributed Sensor Data Aggregation," in *IEEE The 27th Conference on Computer Communications (INFOCOM 2008)*, 2008, pp. 56–60.

[73] B. Przydatek, D. Song, and A. Perrig, "SIA: Secure Information Aggregation in Sensor Networks," in *Proceedings of the first international conference on Embedded networked sensor systems*, Los Angeles, California, USA, 2003, pp. 255 – 265.

[74] C. Fontaine and F. Galand, "A survey of homomorphic encryption for nonspecialists," *EURASIP Journal on Information Security*, vol. 2007, no. 1, pp. 1–15, 2007.

[75] K. Kifayat, M. Merabti, Q. Shi, and D. Llewellyn-Jones, "Applying Secure Data Aggregation techniques for a Structure and Density Independent Group Based Key Management Protocol," in *Third International Symposium on Information Assurance and Security (IAS 2007)*, 2007, pp. 44–49.

[76] S. Q. Ren, D. S. Kim, and J. S. Park, "A Secure Data Aggregation Scheme for Wireless Sensor Networks," in *Frontiers of High Performance Computing and Networking ISPA 2007 Workshops*.LNCS 4743, 2007, pp. 32–40.

[77] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing Source-Location Privacy in Sensor Network Routing," in *Proceedings of 25th IEEE International Conference onDistributed Computing Systems. (ICDCS 2005)*, 2005, pp. 599–608.

[78] B. Gedik and L. Liu, "Location Privacy in Mobile Systems: A Personalized Anonymization Model," in *Proceedings of 25th IEEE International Conference onDistributed Computing Systems. (ICDCS 2005)*, 2005, pp. 620–629.

[79] Y. Jian, S. Chen, Z. Zhang, and L. Zhang, "Protecting Receiver-Location Privacy in Wireless Sensor Networks," in *26th IEEE International Conference on Computer Communications (INFOCOM 2007)*, 2007, pp. 1955–1963.

Authors

Zhijun Li received his B.E. degree in Electrical Engineering from Chongqing University, China and M.E. degree in Computer Engineering from University of Electronic Science and Technology of China, in 1997 and 2003 respectively. Currently, he is a Ph.D student in the department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests include authentication and key distribution in wireless sensor networks.

Guang Gong received a B.S. degree in mathematics in 1981 from Xichang Normal College, Xichang, an M.S. degree in applied mathematics in 1985 from Xidian University, Xian, and a Ph.D. degree in electrical engineering in 1990 from University of Electronic Science and Technology of China (UESTC), Chengdu. She received a Postdoctoral Fellowship from the Fondazione Ugo Bordoni, Rome, Italy, and spent the following year there. After return from Italy, she was promoted to an Associate Professor at the University of Electrical Science and Technology of China. During 1995-1998, she had worked with several internationally recognized outstanding coding experts and cryptographers including Dr. Solomon W. Golomb at the University of Southern California, Los Angeles. She joined, University of Waterloo, Ontario, Canada, in 1998, an Associate Professor at the Department of Electrical and Computer Engineering in September 2000. She is a full Professor since 2004. Her research interests are in the areas of sequence design, cryptography, and communications security. She has authored or co-authored more than 150 technical papers and one book, co-authored with Dr. Golomb, entitled as Signal Design for Good Correlation – for Wireless Communication, Cryptography and Radar, published by Cambridge Press in 2005. She serves as Associate Editors for several journals including an Associate Editor for Sequences for IEEE Transactions on Information Theory, and served on a number of technical program committees of conferences. Dr. Gong has received several awards including the Best Paper Award from the Chinese Institute of Electronics in 1984, Outstanding Doctorate Faculty Award of Sichuan Province, China, in 1991 and the Premier's Research Excellence Award, Ontario, Canada, in 2001.