Security Mechanisms in Wireless Sensor Networks

Pejman Niksaz
Science &Research Branch, Islamic Azad University, Yazd, Iran

Pejmanniksaz@yahoo.com

Abstract__ Wireless Sensor Networks (WSNs) have been recognized for their utility in a variety of different fields including military sensing and tracking, environmental monitoring, patient monitoring and tracking smart environments. The more scientists try to develop further cost and energy efficient computing devices and algorithms for WSNs, the more challenging it becomes to fit the security of WSNs into such a constrained environment. Thus, familiarity with the security aspects of WSNs is essential before designing WSN systems. In order to provide effective integrity, confidentiality, and authentication during communication, the need for additional security measures in WSNs emerges. In this paper, we review some security mechanisms used to overcome these attacks.

Keywords: sensor, security, attack, WSN, challenge.

1- INTRODUCTION

Sensor networks refer to a heterogeneous system combining tiny sensors and actuators with general-purpose computing elements. These tiny sensors have some limitations in power supplies, bandwidth, memory size and energy [1], [2], [3]. Thus, the resource-limited nature of sensor networks poses great challenges for security [4]. Furthermore, sensor networks can be used in a wide range of applications. For example, in the military, wireless sensor networks have been used for some applications such as sensing techniques for military commands, control, communications, computing, intelligence, surveillance, reconnaissance and targeting systems. In healthcare, sensor nodes can also be used for monitoring patients and assisting disabled patients. In addition, there are lots of applications for wireless sensor networks including commercial applications for managing inventory, monitoring product quality and monitoring disaster areas [5], [6].

Because of the resource-constrained nature of wireless sensor networks, we should consider the best and the most suitable security mechanism against adversaries in wireless sensor networks [7]. Generally, there are some serious limitations with current security mechanisms. For understanding these limitations, it is essential to realize differences between WSN and general ad hoc networks [8], [9]. The most important differences between sensor networks and ad hoc networks are:

- The number of sensor nodes in a sensor network can be significantly higher than the nodes in an ad hoc network.

- Sensor nodes are densely deployed.
- Sensor nodes are prone to failures.
- The topology of a sensor network becomes varies constantly.
- Sensor nodes basically use a broadcast communication. In contrast, most ad hoc networks are based on point-to-point communications.
- Sensor nodes are limited in power, computational capacities, and memory.
- Sensor nodes may not have global identification (ID) because of the large amount of overhead and large number of sensors.

Rest of the paper discusses different kinds of security mechanisms in WSN.

Rest of the paper discusses different security mechanisms and solutions in wireless sensor networks.

2- SECURITY MECHANISMS

A. TinySec: A Link Layer Security Architecture for Wireless Sensor Networks

In [10], the researchers describe why Link Layer security is suitable for sensor networks. The network may route packets injected by an attacker to many hops before they are detected. This event just happens when message integrity is only checked at the final destination. This kind of attack can be energy-consuming and in addition will waste the bandwidth. Link layer security architecture can detect unauthorized packets when they are first injected into the network. TinySec provides the basic security properties of message authentication and integrity (using MAC), message confidentiality (through encryption), semantic security (through an Initialization Vector) and replay protection. TinySec supports two kinds of security options: authentication only mode (TinySec-Auth) and authenticated encryption mode (TinySec-AE). In authentication only mode, TinySec authenticates whole the packet with a MAC, but the data payload is not encrypted. On the other hand is authenticated encryption. In this mode, TinySec encrypts the data payload and authenticates the packet with a MAC. The MAC is computed over the encrypted data and the packet header. For further information about the implementation and performance results of TinySec, refer to [10].

Secure Routing

This section consists of two subsections: 1-SPINS-2-SEER

A. SPINS:Security Protocols for Sensor Networks

In [11] Perrig et al. proposed SPINS protocol which includes two optimized security building blocks that are SNEP and µTESLA. A summary of SNEP and µTESLA is presented as follows:
SNEP
Some methods can be used by SNEP to achieve confidentiality such as using encryption and also message authentication code (MAC) to achieve two-party authentication and data integrity. Before message encryption, sender attaches a random bit string with message and this characteristic can cause to semantic security, replay protection and weak freshness. One of the methods SNEP using for preventing of additional communication overhead of sending this extra random bit with each message is sharing a counter between the communicating nodes for the block cipher in counter mode (CTR). The communicating parties increase the shared counter after each block.

B.µTESLA

µTESLA protocol can provides effective authenticated broadcast [8], but it is not designed for limited computing environments. µTESLA proves that an initial packet with a digital signature is authentic and not a forgery. The only method µTESLA is using can be symmetric mechanisms. Disclosing a key in each packet requires too much energy for sending and receiving. In a period of time, µTESLA discloses the key for once. It is expensive to store a one-way key chain in a sensor node. The number of authenticated senders has been restricted by µTESLA. For the base station to broadcast authenticated information to the nodes, µTESLA requires that the base station and nodes are loosely time synchronized, and each node knows an upper bound on the maximum synchronization error. For sending an authenticated packet, the base station easily calculates a MAC on the packet with a key that is secret. When a node gets a packet, it can verify that the corresponding MAC key was not yet disclosed by the base station (based on its loosely synchronized clock, its maximum synchronization error, and the time schedule at which keys are disclosed). Since a receiving node is assured that the MAC key is known only by the base station, the receiving node is assured that no adversary could have changed the packet in transit. The node stores the packets in a buffer. At the time of key exposure, the confirmation key has been broadcast to all receivers. When a node receives the exposed key, it can simply determine the validity of the key. The correct key can be used by a node to authenticate the packet stored in its buffer.

B. SEER: SECURE AND ENERGY EFFICIENT MULTIPLE ROUTING PROTOCOL

SEER has been proposed by Nasser et al in [12]. This is a Secure and Energy-Efficient multipath Routing protocol in which base station accomplishes the route discovery, maintenance and route selection. Instead of using a single path, base station periodically select a new path from multipath based on current energy level of nodes along each path. Such attacks on routing protocols such as Wormhole and Sinkhole can be defended by SEER. SEER can also defend selective forwarding attack as the attacker cannot include itself in the routing path to launch the selective forwarding attack. If any compromised node selectively drops packet it can be detected by the next hop as SEER. I this cases SEER uses a sequence number that identify each packet.

ZIGBEE

Zigbee [13] allows other devices to join the network and also distributes the keys. This protocol plays the three roles as follows:

1) Trust manager has been created in accordance with the authentication of devices that request to join the network.

2) Network manager for maintaining and distributing network keys, and

3) Configuration manager for enabling end-to-end security between devices. This can work in both Residential Mode and Commercial Mode. For low security residential applications, Trust Center Residential Mode has been used. In addition, for high-security commercial applications, a Commercial Mode has been designed.

Three sorts of keys have been employed as follows:

1) Master Key, 2) Link Key, and 3) Network Key.

Master keys are installed first, either in the factory or out of band. They are sent from the Trust Center and are the foundation for long-term security between two devices.

The basis of security between two devices is the Link key, while Network keys are the basis of security across the entire network. Link and Network keys, which have either been installed in the factory or out of band, employ a symmetrical key-key exchange (SKKE) handshake between devices. The key is transported from the Trust Center for both types of keys. This operation takes place in commercial mode when residential mode does not allow for authentication.

802.15.4

Link layer security services and three modes of operation can be provided by 802.15.4 standard [14]. These modes consist of an unsecured mode, an Access Control List (ACL) mode and a secured mode. In unsecured mode, there are no security services. In ACL mode, a list of devices permitted to communicate with them is maintained. Any communication from devices not on the list is ignored. This mode does not offer cryptographic security so it is unimportant for the message source address to be spoofed. Secured mode offers seven security suites and, depending on which is used, any of four security services are offered. These include access control, data encryption, frame integrity and sequential freshness. In order to decrease energy consumption, 802.15.4 security suites should be implemented on the radio chips and all the necessary cryptographic computations should be done in the hardware.

Hence, 802.15.4 standard, if implemented correctly, can be used as a good base for building higher level, fully featured security suites.

Cryptography

One of the ways for ensuring security services is Cryptography. These encryption-decryption techniques are used for the traditional wired networks and so are not appropriate for direct application to WSNs. WSNs consist of tiny sensors which have some limitations such as lack of processing, memory and battery power [15]. For any encryption operations, transmission of extra bits, extra processing, memory and battery power are required. Moreover, applying security mechanisms such as encryption could also increase delay, jitter and packet loss in WSNs [16]. Two different kinds of cryptography have been proposed as follows.

Public key cryptography in WSNs

Some factors such as the code size, data size, processing time, and power consumption make

certain approaches unsuitable to be employed in WSNs as public key algorithm techniques. These include the Diffie-Hellman key agreement protocol [17] or RSA signatures [18].

From a computational point of view, public key algorithms such as RSA are fast and usually accomplish millions of multiplication instructions for carrying out a single-security operation. Further, the number of required clock cycles to perform a multiplication instruction can establish the efficiency of microprocessor's public key algorithm [19].

Public key algorithms such as RSA usually take up to minutes to perform cryptographic operations in resource-constrained wireless devices, and thus are susceptible to DoS attacks. On the other hand, Carman et al. found that it usually takes a microprocessor thousands of nano-joules to do a simple multiplication function with a 128-bit result [20]. The energy consumption of symmetric key cryptographic algorithms and hash functions is much less than that required for computational public key algorithms. For instance, the encryption of a 1024-bit block consumes approximately 42mJ on MC68328 DragonBall processor using RSA, and the estimated energy consumption for a 128- bit AES block is much lower at 0.104 mJ [19].

By using the right selection of algorithms and associated parameters, recent studies have proven that through optimization and low power techniques, public key cryptography can be applied to sensor networks [21], [22], [23]. The investigated public key algorithms include Rabin's Scheme [24], Ntru-Encrypt [25], RSA [19], and Elliptic Curve Cryptography (ECC) [26], [27]. Most studies in the literature consider RSA and ECC algorithms. One of the advantages of ECC is that it offers equal security for a far smaller key size, so in this way it can reduce processing and communication overhead. As an example, RSA with 1024-bit keys (RSA-1024) provides a currently accepted level of security for many applications and is equivalent in strength to ECC with 160-bit keys (ECC-160) [28]. In the year 2010, in order to protect data, RSA Security presented RSA-2048 as the new minimum key size, which is equivalent to ECC with 224-bit keys (ECC-224) [29].

Wander et al. investigated the energy cost of authentication and key exchange based on RSA and ECC cryptography on an Atmel ATmega128 processor [30]. The ECC-based signature has been produced and considered with the Elliptic Curve Digital Signature Algorithm (ECDSA) [27]. A simplified version of the SSL handshake is the key exchange protocol, which contains two parties: a client starting the communication and a server responding to the client [28].

In the handshake process, the two parties verify each other's certificate and negotiate the session key to be used in the communication. The results have proven that ECDSA signatures are meaningfully cheaper than RSA signatures.

Further, the efficiency of ECC-based key exchange protocol is better than the RSA-based key exchange protocol at the server side. However, there is not a significant difference in the energy cost between the two key exchange protocols at the client side.

In [28], a system called TinyPK has is described where the RSA system has been implemented on Mica2 motes using TinyOS development environment. The researchers have shown that, by using this scheme in resource-constrained sensor nodes, authentication and key agreement protocol can be effectively realized.

While public key cryptography can be feasible in sensor nodes, the private key operations are still expensive. For example, the investigation in [31] focuses on the public key operations and assumes the private key operations have been performed by a base station or a third party. By using the small integer e = 216 + 1 as the public key, the public key operation time can be fast enough, whereas the private key operation time does not modify. Many security services use public key algorithms because of the limitation of private key operation occurring only at a base station.

Symmetric key cryptography in WSNs

Since most public key cryptographic mechanisms are computationally intensive, research studies of WSNs tend to focus on use of symmetric key cryptographic techniques. Symmetric key cryptographic mechanisms use a single shared key between the two communicating hosts that is used both for encryption and decryption. However, one major challenge for deployment of symmetric keys is how to securely distribute the shared key between

the two communicating hosts. This is a non-trivial problem since pre-distributing the key may not always be feasible.

Five popular encryption schemes, RC4 [32], RC5 [33], IDEA [34], SHA-1 [35], and MD5 [36], were evaluated on six different microprocessors ranging in word size from 8-bit (Atmet AVR) to 16-bit (Mitsubishi M16C) to 32-bit widths (StrongARM, XScale). The execution time and code memory size were measured for each algorithm and platform. The experiments indicated uniform cryptographic cost for each encryption class and each architecture class. The impact of caches was negligible, while Instruction Set Architecture (ISA) support is limited to specific effects on certain algorithms. Moreover, hashing algorithms (MD5, SHA-1) incur higher overhead than encryption algorithms (RC4, RC5, and IDEA).

Still, the decision depends on the computation and communication capability of the sensor nodes. Open research issues range from cryptographic algorithms to hardware design as described below.

Recent studies on public key cryptography have demonstrated that public key operations may be practical in sensor networks. However, private key operations are still too expensive in terms of computation and energy cost to accomplish in a sensor node. The application of private key operations to sensor nodes needs to be studied further.

Symmetric key cryptography is superior to public key cryptography in terms of speed and low energy cost. However, the key distribution schemes based on symmetric key cryptography are not perfect. Efficient and flexible key distribution schemes need to be designed. It is also likely that more powerful motes will need to be designed to support the increasing requirements on computation and communication in sensor nodes.

Defense against DoS attacks

In this section, defense mechanisms against DOS attacks will be presented.

Defense in the physical layer

Jamming attacks may be prevented by employing various spread-spectrum communications such as frequency hopping and code spreading [37]. Frequency-hopping spread spectrum (FHSS) is an approach where signals are transmitted by rapidly switching a carrier between different frequency channels using a pseudo-random sequence known to both the transmitter and the receiver. When a potential attacker is unable to predict the frequency selection sequence, it is impractical for him to jam the frequency being used at a given time. Code spreading is another technique for defending against jamming. However, it needs greater design complexity and energy and is not suitable for use with WSNs. Generally, to maintain low cost and low power requirements, sensor devices are limited to single-frequency use and are therefore highly susceptible to jamming attacks. One approach to tolerating jamming attacks in WSNs is to identify the jammed part of the network and effectively avoid it by routing around. Wood and Stankovic [38] have proposed an approach where the nodes along the perimeter of a jammed region report their status to the neighbors and collectively the affected region is identified and packets are routed around it.

Defense in the link layer

A common defense against collision attacks is the use of error correcting codes [38]. Most codes work best with low levels of collisions such as those caused by environmental or probabilistic errors. However, these codes also increase extra processing and communication overhead. It is rational to assume that an attacker will always be able to corrupt more than what can be corrected. Although it is possible to detect these cruel collisions, no complete defense mechanism against them is known today.

A possible defense against energy exhaustion attacks is to utilize a rate limiting MAC admission control. This would allow the network to pay no attention to those requests that deliberately exhaust the energy reserves of a node. A second technique is to use time division multiplexing where each node is allocated a time slot in which it can transmit [38]. This removes the necessity of arbitration for each frame and can solve the indefinite postponement problem in a back-off algorithm. However, it is still susceptible to collisions.

The effect of an attack launched against a link layer attack can be lessened by use of small frames, since they decrease the amount of time an attacker has to capture the communication channel [38]. However, this technique often reduces efficiency and is susceptible to further unfairness as an attacker may attempt to retransmit rapidly instead of waiting for a random time interval.

Defense in the network layer

A countermeasure against spoofing and alteration is to append a message authentication code (MAC) after the message. By adding a MAC to the message, the receivers can confirm whether the messages has been spoofed or altered. A possible defense against selective forwarding attacks is using digital watermarking technology [39]. A second defense is to discover the malicious node or to consider it as failed and seek an alternative route.

Defense in the transport layer

To defend against flooding DoS attacks at the transport layer, Aura et al. have proposed a mechanism using client puzzles [40]. The idea is that each connecting client should express its commitment to the connection by solving a puzzle. As an attacker will not have infinite resources, it will be impossible for him to create new connections fast enough to cause resource starvation on the serving node.

A possible defense against de-synchronization attacks on the transport layer is to enforce an obligatory authentication of all packets communicated between nodes [38]. If the authentication mechanism is safe, an attacker will not have the ability to send any spoofed messages to any destination node.

Defense against Sybil attacks

Any defense mechanism against the Sybil attack must ensure that a framework is in place in the network to corroborate that a specific identity is the only identity held by a given physical node [41]. Newsome et al. have described three orthogonal dimensions of the Sybil attack taxonomy [41]. The three dimensions are: 1) direct vs. indirect communication, 2) fabricated vs. stolen identities, and 3) simultaneity. In direct communication, the Sybil nodes communicate directly with legitimate nodes. In this attack, when a legitimate node sends a radio message to a Sybil node, one of the malicious devices listens to the message. In indirect communication, no legitimate nodes are able to communicate directly with the Sybil nodes. Messages sent to a Sybil node are routed through one or more malicious nodes, which pass the message on to the Sybil node. In case of fabricated identities, the attacker creates arbitrary new Sybil identities. However, if a mechanism is in place to detect false identities, an attacker cannot fabricate new identities. In this case, the attacker needs to assign other legal identities to Sybil nodes. This identity theft may go undetected if the attacker destroys or, for a limited period of time, disables the impersonated nodes. In case of simultaneous attacks, the attacker tries to have all the Sybil identities participate in the network simultaneously. Repeatedly, the attacker may present a large number of identities over a period of time, while deploying a small number of identities at any given point of time.

Newsome et al. primarily describe direct validation techniques, including a radio resource test. In the radio test, a node assigns each of its neighbors a channel and listens to each of them. If the node detects a transmission on the channel, it is assumed that the node transmitting on the channel is a physical node. Likely, if the node does not discover a transmission on the specified channel, it assumes that the identity assigned to the channel is not a physical identity. Another technique to defend against the Sybil attack is to use random key pre-distribution techniques [41]. In random key pre-distribution, a random set of keys or key-related information are/is assigned to each sensor node. Thus, in the key set-up phase, each node can detect or compute the usual keys it shares with its neighbors. The common keys are used as shared secret session keys to ensure node-to-node secrecy. Newsome et al. have proposed that the identity of each node is associated with the keys assigned to the node [42]. With a particular set of captured keys, there is little probability that a randomly created identity will work.

Defense against wormhole attack

Hu et al. have proposed a new and general mechanism called packet leashes for discovering and defending against wormhole attacks [43]. In a wormhole attack, a malicious node eavesdrops on a series of packets, then tunnels them through a path in the network and sends them again. This is done to create a false representation of the distance between the two colluding nodes. It is also utilized, mostly, to prevent the routing protocol from continuing by misleading the neighbor discovery process. Guo et al. have presented a strategy based on neighbor node verification [44]. Parasannajit et al. have also used an approach to detect wormholes in a WSN [45].

In the mechanism proposed by the authors, a distance estimation is made between all the sensor nodes in a neighborhood. Using multi-dimensional scaling, a virtual layout of the network is then computed, and a surface smoothing strategy is applied to revise the round-off errors.
Finally, the form of the resulting virtual network is analyzed. If any wormhole exists, the form of the network will bend and curve towards the wormhole. Otherwise, the network will appear flat.

Detection of node replication attack

Parno, Perrig and Gligor have proposed a mechanism for distributed detection of node replication attacks in WSNs [46]. To address the most basic and important limitations of the existing mechanisms (e.g. single point of failure for centralized schemes, or neighborhood voting protocols which are not successful to discover distributed replications), the researchers have proposed two algorithms that work through the common actions of multiple nodes. The algorithms are: 1) randomized multicast, and 2) line-selected multicast. The randomized multicast algorithm distributes the location information of a node to accidentally-selected witnesses, cashing in on a birthday paradox to detect replicated nodes. The line-selected multicast uses the network topology to detect replication as follows. The randomized broadcast has developed from conventional node-to-node broadcasting. In traditional node-to-node broadcasting, each node in the network uses an authenticated broadcast message to flood the network with its location information. Each node keeps the location information of its neighbors and if it gets a contradictory claim, it prevents the insulting node. This protocol can achieve 100% detection of all duplicate location claims if the broadcasts extend to all the nodes. Altogether, the total communication cost for the protocol is $O(n2)$, which is too high for a large WSN. To decrease the communication cost of node-to-node-broadcast, deterministic multicast mechanisms may be utilized where a node's location claim is shared with a limited subset of deterministically selected witness nodes. The witnesses are selected as a function of the node's ID. If the adversary replicates a node, the witnesses will receive two different location claims for the same node ID. The contradictory location claims trigger the revocation of the replicated node.

The randomized multicast approach discussed by Parno et al. makes improves the robustness of the deterministic multicast. It randomizes the witnesses for a given node's location claim so that the adversary cannot anticipate their identities. When a node reports its location, each of its neighbors transmits a copy of the location claim to a set of accidentally chosen witnesses. If the adversary replicates a node, then two sets of witnesses will be chosen. In a network of n nodes, if each location creates $\sqrt{n}$ witnesses, then the birthday paradox predicts at least one collision with high probability (i.e. at least one witness will get a pair of conflicting location claims). The two conflicting location claims form enough proof to revoke the node, so the witness can flood the pair of location claims through the network and each node can independently verify the revocation decision.

Unfortunately, the communication and storage overheads for randomized multicast is too high--$O(n2)$ and $O(\sqrt{n})$ respectively. The researchers have recommended some methods for improving the communication and storage overhead. To decrease the communication cost of the randomized multicast method, Parno et al. have discussed an alternative algorithm--the line selected multicast. This algorithm is based on the rumor routing protocol [46]. The idea is that a location claim moving from source s to destination d will also move through several intermediate nodes. If each of these nodes records the location claims, then the path of the location claim through the network can be thought

of as a line segment. The destination of the location claim is one of the accidentally chosen witnesses. As the location claim routes through the network towards a witness node, the intermediate sensors check the claim. If a conflicting location claim ever crosses a line, then the node at the intersection finds the conflict and starts a revocation broadcast. The line selected multicast algorithm has a communication overhead of $O(n\sqrt{n})$ as long as each line segment is of length $O(\sqrt{n})$ nodes. The storage overhead of algorithm is $O(\sqrt{n})$.

Defense against traffic analysis attack

Deng, Han and Mishra have suggested a mechanism for holding off against traffic analysis attack in a WSN [47]. The authors suggest that, since the base station is a central point of failure, once the location of the base station is found an adversary can break or destroy it and make the data-gathering functionalities of the entire WSN inefficient. Two classes of traffic analysis attacks in WSNs are introduced: 1) rate monitoring attack, and 2) time correlation attack. In a time correlation attack, an adversary checks the packet-sending rate of nodes near the adversary and moves closer to the nodes that have a higher packet-sending rate. In a time correlation attack, an adversary observes the correlation in sending time between a node and its neighbor node (one that is assumed to be forwarding the same packet) and deduces the path by following the sound of each forwarding operation as the packet propagates towards the base station. The methods presented by the researchers include stop rate monitoring attacks and time correlation attacks. The method consists of four techniques. First, a multiple parent routing scheme is identified that permits a sensor node to forward a packet to one of its multiple parents. This makes the patterns less important in terms of routing packets towards the base station. Second, a controlled random walk is considered for a multi-hop path moved by a packet through the WSN towards the base station. This gives out packet traffic, thereby making the rate monitoring attack useless. Third, random forgery paths are introduced to mix up an adversary from tracking a packet as it traverses towards a base station. This alleviates the effectiveness of time correlation attacks. At the end, multiple random areas of high communication activities are made to mislead an adversary as to the true location of the base station, which further increases the difficulty of launching rate monitoring attacks. The mixture of these four strategies makes the introduced method incredibly sturdy to any traffic analysis attack.

Defense against attacks on sensor privacy

Several mechanisms have been introduced for protecting information privacy in WSNs. Some of them are discussed below.

Anonymity mechanisms

Exact location information makes accurate identification of a user possible. This is a serious risk to privacy. One solution for dealing with this problem is to make the data source anonymous. An anonymity mechanism depersonalizes the data before it is freed from the source. Sisheng Chen et al. have discussed a new secure anonymous routing scheme for clustered wireless ad hoc networks [48]. Since ensuring all anonymity is an impractical suggestion, a balance is necessary between anonymity and disclosure of public information in most privacy protection methods. Four different methods have been presented to aid with this [49], [50], [51], [52]. These methods including: 1) decentralization of storage of sensitive data, 2) establishment of secure channel for communication, 3) changing the pattern of data traffic, and 4) exploiting mobility of the nodes. The confidential location data is to be kept in a spanning tree of nodes so that no single node holds a complete view of the location information. Communication employing secure protocols like SPINS will make eavesdropping and active attacks on a WSN incredibly challenging. The data traffic pattern may be altered by selectively putting some false data in the network traffic so that traffic analysis by an external entity will not be effective. Mobile sensor nodes make attacks on location privacy very hard. The *Cricket* system [53] is a location support system for in-building, mobile, location-dependent applications. It permits applications running on mobile and static nodes to discover their physical location from a set of listeners. The listeners hear

and examine information carefully from beacons in a building.

Policy-based approaches

In policy-based defense mechanisms, the access control decisions and authentication methods are considered according to a specified set of privacy policies. Kim, Dong Seong et al. have discussed the theory of private authentication and shown its application in a radio frequency identification (RFID) domain [54]. Hyo-Sang Lim et al. have presented a policy-based approach by which one can specify data integrity policies on the basis of the needs of collaborations [55]. Snekkenes has proposed different parameters for access control that allow specifying policies in the context of a mobile network [56]. Some of the parameters including speed, identity, time of request, location and the located object. Myles et al. have discussed the architecture of a centralized location server that controls access requests from client applications through a set of validator modules according to a set of XML-coded privacy policies [57]. Hengarter and Steenkiste have described different challenges that arise for the specification and execution of policies controlling access to location information [58]. The researchers have also presented a design framework of an access control mechanism that is adaptable enough to be deployed in various environments.

Information flooding

Ozturk et al. have presented different changes to WSN routing protocols for protecting the location information of a source node [59]; namely, a set of flooding protocols. The randomized data routing and phantom traffic generation mechanisms are utilized so that it is hard for an adversary to track data sources. For ensuring privacy of source location, the researchers have considered four kinds of flooding-based routing protocols: 1) baseline flooding, 2) probabilistic flooding, 3) flooding with fake messages, and 4) phantom flooding.

Baseline flooding: In baseline flooding, each node in the network sends a message only once and no node passes a message again that it has formerly transmitted. Whenever a message reaches an intermediate node, the node first checks whether it has received and forwarded the message before. If it happens for the first time, the node sends out the message to all its neighbors. Otherwise, it just discards the message.

Probabilistic flooding: In probabilistic flooding, only a subset of nodes in the whole network take part in data forwarding, while the others easily throw the messages they receive away. One feasible weakness of this method is that some messages may get lost in the network and so impact on the total network connectivity. However, the researchers have validated that this is not an important problem.

Flooding with fake messages: Flooding does not provide privacy protection because an adversary can simply recognize the shortest path between the source and the sink and then back trace to the source location. The most important reason for the lack of location privacy is that there is only one source node. One method for reducing the hazard of source location privacy breaking is to increase the effectiveness of flooding protocols. This allows for more sources to be considered that inject fake messages into the network. If the fake messages have the same length as the real messages and they are also encrypted, it will be impractical for an adversary to differentiate among them.

Phantom flooding: Phantom flooding has the same principle as that of probabilistic flooding. It tries to direct messages to different locations of the network so that the adversary cannot receive an unchanging stream of messages to track the source. Nevertheless, probabilistic flooding is ineffective since shorter paths are likely to deliver more messages. Phantom flooding entices an attacker away from the genuine source and towards a fake source, called the phantom source. In phantom flooding, each message follows two steps: 1) walking phase, which may be a random walk or a directed walk, and 2) a subsequent flooding meant to deliver the message to the sink. Whenever the source broadcasts a message, the message is unicast in a random fashion within the first *hwalk* hops. This is named the *random walk* phase. After the *hwalk* hops, the message has been flooded using the baseline flooding technique. This is the *flooding* phase. Phantom flooding enormously improves the privacy and network security period because every message may take a different (shortest) path to get to any node in the network. Deng, Han and Mishra have tried to solve the problem of defending a base

station against physical attacks by hiding the geographic location of the base station [60]. The researchers have examined lots of countermeasures against traffic analysis techniques aimed at differentiating the location of a base station. In the discussed method, a degree of randomness is considered while choosing the multi-hop route to the base station. Then, random fake packets are introduced as a packet is forwarded towards a base station. Metrics such as entire entropy of the network, entire energy consumed, and the ability to defend against heuristic based techniques to locate the base station are assessed analytically as well as by extensive simulations. Xi, Schwiebert and Shi [61] have investigated an effective attack on the flooding-based phantom routing proposed by Ozturk et al. [59]. The researchers have also presented *greedy random walk* (GROW) protocol, a two-way random walk (i.e., from both source and sink) to decrease the opportunity for an eavesdropper to collect location information. In the presented method, the sink first starts an *N*-hop random walk, and the source then initiates an *M*-hop random walk. Once the source packet gets to an intersection of these two paths, it is sent through the path created by the sink. Local broadcasting is utilized to discover when the two paths intersect. In order to minimize the chance of backtracking along the random walk, the nodes are held in a bloom filter as the walk progresses. At each step, the intermediate nodes are checked against the bloom filter to ensure that backtracking is minimized.

Intrusion detection

The security mechanisms applied in secure routing protocols and secure data aggregation protocols are configured in advanced to hold an attacker back from breaking the security of the network. However, these security mechanisms cannot ensure security of a WSN by themselves. Since it is practical for an attacker to compromise a sensor node, it is not difficult for him to inject false data into a WSN. Authentication and data encryption are not enough for ensuring data security. One other method has been considered for solving these problems and this involves mechanisms for discovering and responding to intrusions. An intrusion detection system (IDS) watches and checks a host or network carefully in order to detect questionable activity patterns outside normal and expected behavior [62]. It is carried out according to the assumption that there exists an obvious difference in the behavior of an intruder and legal user in the network such that an IDS can match those pre-programmed or possible learned rules. According to the analysis model utilized for analyzing the audit data to detect intrusions, intrusion detection systems are usually organized into two sorts: 1) rule-based intrusion detection systems and 2) anomaly-based intrusion detection systems [63]. Rule-based intrusion detection systems have been utilized to detect known patterns of intrusions as in [64] and [65]. The anomaly-based systems are used to detect new or unknown intrusions as in [66] and [67]. Rule-based IDS has a low false-alarm rate compared to an anomaly-based system, and an anomaly-based IDS has a high intrusion detection rate in comparison to a rule-based system. However, WSNs are totally application-specific and lack basic information on topology, normal usage, expected communication patterns, etc. It is impossible to pre-install some unchanging patterns in sensors before they are deployed. Additionally, because of sensors limitations, to learn and discover these parameters after deployment is both time- and energy-consuming. As a result, current intrusion detection schemes in ad hoc networks may not be altered to WSNs. Existing investigations focus on how to discover and get rid of injected false information. Thus, cooperation between sensors, especially neighboring nodes, is essential to decide the report correctness.

Brutch and Ko have investigated different kinds of existing attacks against WSNs and suggested three various architectures for intrusion detection [68]. The first is a stand-alone architecture. In this case, each node functions as an independent intrusion detection system and is in charge of discovering attacks sent towards it. The nodes do not give and receive, and intrusion data and no helpful detection mechanisms are deployed. The second method is a distributed and cooperative architecture. In this architecture, an intrusion detection agent is deployed on each node. While the local agents are in charge of finding local attacks on the nodes, they also cooperate between themselves

by exchanging intrusion-related data to discover global intrusion attempts. The last proposed architecture is a hierarchical architecture. This is appropriate for a multi-layered WSN where the network is separated into clusters. In this case, the cluster-head node is in charge of routing within a cluster. The multi-layered networks are mainly utilized for event correlation.

Zhu et al. have suggested an interleaved hop-by-hop authentication (IHOP) scheme in [69]. IHOP assures that the base station will find any injected false data packets when no more than a certain number *t* of nodes are compromised. The sensor network is organized in a cluster-based hierarchy. Each cluster-head builds a route to the base station and each intermediate node has an upper associate node and a lower associate node that is *t* + 1 hops away. IHOP utilizes a number of shared keys: 1) every node shares a master key with the base station, 2) each node recognizes its one-hop neighbors and has founded a pair-wise key with each of them, and 3) a node can found a pair-wise key with another node that is multiple hops away if required. Further, IHOP also presumes that the base station has a mechanism to authenticate broadcast messages, e.g., μ*TESLA*. A cluster-head brings the information from its members together and forwards a description to the base station only when at least *t* + 1 sensors observe the same result. Meanwhile, a cluster-head also collects the MACs from detecting nodes. Each detecting node forwards two MACs to the cluster-head: a MAC using the key shared with the base station, referred to as the individual MAC, and a MAC utilizing the key shared with its upper associate nodes, referred to as the pair-wise MAC. The cluster-head then compresses the *t* + 1 individual MACs by XORing them to decrease the size of the description. However, the pair-wise MACs are not compressed for broadcasting. If they were, a node replaying the message does not have the ability to extract the pair-wise MACs and a compressed MAC for the base station. When an intermediate node gets a description, it confirms the MAC of its lower associate node. If it does not succeed, the description is removed. In a different way, it eliminates the MAC, creates a new MAC using its upper associate node pair-wise key, and appends a description to it. However, the pair-wise MACs are not compromised for broadcasting. If they were, a node passing on the message would not be able to extract the pair-wise MACs of interest to it. As a result, a legal description includes *t* + 1 pair-wise MACs and a compressed MAC for the base station.

IHOP ensures that the base station can find out false data packets when no more than *t* nodes are compromised. However, the investigators have not illustrated how to choose the parameter *t* for a sensor network. Wang et al. have suggested a scheme to discover whether a node is faulty or malicious with the collaboration of neighbor nodes [70]. In the suggested scheme, when a node suspects that one of its neighbors is faulty, it broadcasts messages to claim the opinions on the behavior of this suspected node from other neighbors. After collecting the results together, the node examines the effects to determine whether the suspect has a problem. Researchers have formalized the problem as an issue of creating a dominating tree to protect all the neighbors of the suspect. They suggest two tree-based propagation collection protocols to form a dominating tree and collect the information together via the tree structure. Albers et al. have described an intrusion detection architecture based on a local intrusion detection system (LIDS) on each node in a wireless ad hoc network [71]. For detecting a network-wide intrusion, the LIDS on the nodes collaborate with each other and give or receive two kinds of data--security data and intrusion alerts. The security data is utilized to exchange information with other network hosts. Intrusion alerts are employed to announce LIDS in the neighboring nodes to exchange intrusion-related information. Although the framework is for an ad hoc network, its method of local anomaly detection and ability to detect any network-wide intrusion can be altered to advance an IDS for a WSN [72]. Intrusion detection in WSNs is still significantly open to investigation.

Two key research issues are:

1) Because of WSN limitations, intrusion detection has many aspects of concern that are not present in other network types. 2) The challenge of intrusion detection in WSNs requires better definition. The suggested IDS protocols in the literature focus on filtering injected false information [73]. These protocols need to be better to address scalability problems.

Defense against physical attacks

Sensor nodes should be equipped with particular hardware in order to keep them safe from a possible physical attack. The sensor nodes in a WSN may be kept safe from tampering by tamper proofing the physical packages of the sensors [74]. Authors have also suggested mechanisms that concentrate on building tamper-resistant hardware to make the memory contents on the sensor chip inaccessible to a potential external attacker [75], [76], [77].

Special-purpose software and hardware may also be deployed outside the sensor nodes to detect physical tampering. Self-termination of sensor nodes is a successful mechanism to protect against feasible data theft in the event of a physical attack. The main idea in this case is that whenever a sensor senses an attack, it causes itself to fail and ruins all data and keys stored in its memory. This is possible in a large-scale WSN where there is enough redundancy of information and connectivity between the nodes. However, the basic problem is to correctly recognize a physical attack. An easy solution is to occasionally discover if the neighborhood information for each node is correct. In case of a mobile sensor network, this is an open issue.

In [78], [79], the researchers explain techniques for pulling out protected software and data from smart card processors. This includes manual micro-probing, laser cutting, focused ion-beam manipulation, glitch attacks, and power analysis, most of which are also feasible physical attacks on the sensor. Andersen et al. present examples of low-cost protection countermeasures that make such attacks significantly more difficult [120]. Deng et al. have suggested different methods for protecting sensors by deploying components outside them [78]. Sastry et al. have discussed ECHO protocol for secure and reliable location verification of sensor nodes in a WSN [79]. The program functions according to the physical properties of sound and RF signal propagation from the sensor nodes. It is not feasible for an adversary to behave in a dishonest way in order to get advantages and falsely claim a shorter distance from the base station by giving and receiving its ultrasonic sound response early because it will not have the ability to produce the required nonce for verification. In [80], the researchers suggest defense mechanisms against search-based physical attacks. The authors have also presented a systematic modeling framework for blind physical attacks [81]. The defense mechanism against physical attacks as proposed by the researchers entails two steps. In the first step, the sensors discover the attacker and broadcast attack notification messages in the network. In the second step, the sensors that get the notification messages schedule their states to switch off mode. Seshadri et al. have been offered a mechanism called SWATT to discover a sudden and abrupt modification in the memory contents of a sensor node [82]. An abrupt change in the memory content of a sensor shows possibility of a physical attack.

Trust management

Another method for enforcing high-level of security in WSNs is application of trust- and reputation-based frameworks. Trust-based schemes certainly can protect against attacks which are beyond the capabilities of cryptographic security. For instance, problems like judging the quality and reliability of sensor nodes and wireless links, data aggregation, reliability and correctness of aggregator nodes, timeliness in packet forwarding of the sensors, etc. These can all be solved using a systematic approach with the help of a trust-based framework. However, trust-based models usually entail high computational overhead, and building an effective scheme for resource-limited WSNs is a very difficult task. Pirzada and Mcdonald [83] have suggested a method for building trust relationships among nodes in an ad hoc network. It is presumed that the nodes in the network passively watch and check the packets received and forwarded by the other nodes. The receiving and forwarding activities by the nodes are called events. Events are detected and given a weight depending on the kind of application requiring a trust relationship with other nodes. The weights show the importance of the detected events for the corresponding application. The trust values for all events from a node are combined utilizing weights to compute an aggregate trust level for the node. The computed trust values are utilized as link weights for the computation of routes. Links that join more trust-worthy nodes

together will have smaller weights. A shortest-path routing algorithm would compute the most trustworthy paths in a network.

In [84], the researcher suggests methods of discovering paths from a source node to a chosen target node in a peer-to-peer computing paradigm. Extending this method, Zhu et al. [85] offered a feasible manner to compute trust in wireless networks by treating unique mobile devices as nodes of a delegation graph *G* and mapping a delegation path from a source node *S* to a target node *T* into an edge in the corresponding transitive closure of the graph *G*. From the edges of the transitive closure of the graph *G*, the trust values of the wireless links are computed. In one suggested trust-based framework, an undirected transitive signature scheme is utilized within the authenticated transitive graphs. Yan et al. have recommended a security solution according to trust framework to make sure data protection, secure routing, and other security characteristics are enabled in an ad hoc network [86]. Some methods of logical and computational trust analysis and evaluation are used for nodes. Each node assesses the trust of its peers according to some factors such as experience statistics, data value, intrusion detection results, recommendations from its other neighbors. Ren et al. have presented a method to found trust relationships between nodes in an ad hoc network [87]. The proposed framework is a probabilistic solution according to a distributed trust model. A secret dealer is considered only in the system bootstrapping phase to start the trust propagation. Shorter and more robust trust chains are subsequently organized between the nodes. A fully self-organized trust establishment manner is then adopted to obey to the dynamic membership modifications.

Ganeriwal and Srivastava have offered a reputation-based framework for high integrity sensor networks [88]. The framework applies a beta distribution for reputation representation, updates, and integration. Tanachaiwiwat et al. [89] have discussed a mechanism of location-centric isolation of nodes exhibiting misbehavior and trust-based routing between nodes in sensor networks. The trust value of a node is computed according to the cryptographic suite being employed, availability statistics and the packet forwarding information of the node. If the computed trust associated with a node decreases below a threshold, the node's location is considered insecure and it is avoided in routing process. Linag and Shi have accomplished extensive work on growth of models and evaluating robustness and security of different aggregation algorithms in open and untrusted environments [90], [91]. These models will likely need to be altered for the deployment of trust-frameworks in WSNs. In [92], the researchers have described a personalized trust model called *PET* for nodes in a WSN. In [92], a thorough examination of the inference model of trust is presented along with a description of approaches to aggregation of different ratings received from peer sensor nodes. The researchers have concluded that the memory constraint is a challenging limitation for sensor nodes for keeping knowledge related to computation of a trust-based framework. The simulation results indicate that it is better to treat ratings received from various evaluators (i.e., nodes) with equal weight and easily compute the average to arrive at the final trust value. This method not only has a very low computational overhead, it also produces very reasonable results in practice. The researchers also observe that for a trust model, the most significant and critical problem is how to adaptively adjust the parameters of the model according to the modifications in environment.

Steganography

While cryptography focuses on hiding the content of a message, steganography hides the existence of the message. Steganography is the art of confidential communication by embedding a message into multimedia data (image, sound, video, etc.). The major goal of steganography is to change the carrier in a way that is not noticeable so that it looks just like ordinary. It hides the existence of a confidential channel. In the case that we want to send a covert data without sender information or when we want to distribute covert data publicly, it is very helpful [93].

Physical Layer Secure Access

Physical layer secure access in WSNs could be provided by employing frequency hopping. A dynamic combination of the parameters like hopping set (accessible frequencies for hopping), dwell time (time interval per hop) and hopping pattern (the sequence in which the frequencies from the accessible hopping set is employed) could be employed with a little expense of memory, processing and energy resources. Essential points in physical layer secure access are the efficient design so that the hopping sequence is changed in less time than is needed to detect it and for using this both the sender and receiver should maintain a synchronized clock.

TRANS: Trust Routing for Location Aware Sensor Networks

Tanachaiwiwat et al. have proposed a new method called TRANS (Trust Routing for Location Aware Sensor Networks) [94]. The TRANS routing protocol is designed to be employed in data centric networks. It also exploits a loose time synchronization asymmetric cryptographic scheme to make sure message confidentiality.

Localized Encryption and Authentication Protocol (LEAP) (Hierarchical Key Management)

Zhu et al. have described a key management protocol named a localized encryption and authentication protocol (LEAP) for large-scale distributed sensor networks, where each sensor node can found pair-wise keys with its one-hop neighbor [95]. LEAP can reduce the effect of selective forwarding attack as it makes use of local broadcast. Thus, the impact of this attack cannot be transferred more than 2 hops away. LEAP can stop HELLO Flood attack as the node takes packets only from its authenticated neighbor. LEAP can also prevent Sybil attack by providing individual ID authentication for each node. The LEAP protocol presented by Zhu et al. [95] utilizes multiple keying mechanisms. Their conclusion is that no single security requirement accurately supports all sorts of communication in a wireless sensor network. Therefore, four various keys are employed relying on whom the sensor node is communicating with. Sensors are preloaded with the first key from which further keys can be founded. As a security precaution, the first key can be deleted after its use in order to make sure that a compromised sensor cannot add extra compromised nodes to the network.

Individual Key

Every node has an individual key that it shares pairwise with the base station. This key is employed for secure communication between a node and the base station. For instance, a node may send an alert to the base station if it detects any unnatural or unexpected behavior by a neighboring node. Likely, the base station can utilize this key to encrypt any sensitive information, e.g. keying material or specific instruction it sends to a unique node.

Group Key

This is a shared key that is employed by the base station for encrypting messages that are sent out to the whole group. However, since the group key is shared between all the nodes in the network, an effective re-keying mechanism is essential for updating this key after a compromised node is revoked.

Cluster Key

A cluster key is a key shared by a node and all its neighbors, and it is significantly employed for securing locally broadcast messages (e.g. routing control information) or securing sensor messages which can benefit from passive participation. Scientists have proved that in-network processing techniques including data aggregation and passive participation are very essential for saving energy consumption in sensor networks [95]. Therefore, in LEAP, each node possesses an individual cluster key that it employs for securing its messages, while its immediate neighbors apply the same key for decryption or authentication of its messages.

Pairwise Shared Key

Every node shares a pairwise key with each of its immediate neighbors. In LEAP, pairwise keys are utilized for securing communications that need privacy or source authentication. For instance, a node can employ its pairwise keys to secure the distribution of its cluster key to its neighbors, or to secure the transmissions of its sensor readings to an aggregation node. Pay attention that the use of pairwise keys prevents passive participation.

Secure broadcasting and multicasting protocols

Multicasting and broadcasting methods are employed mainly to decrease the communication and management overhead of forwarding a single message to multiple receivers. For the purpose of ensuring that only legal group members receive the multicast and broadcast communication, suitable authentication and encryption mechanisms must be established.

To deal with this matter, several key management schemes have been created: centralized group key management protocols, decentralized key management protocols, and distributed key management protocols [96]. Initially, we will describe some generic security mechanisms for multicast and broadcast communication in wireless networks. Then we will present some of the famous suggestions specific to WSNs.

In the case of the centralized group key management protocols, a central authority is employed to maintain the group. Decentralized management protocols, however, separate the task of group management between multiple nodes. In distributed key management protocols, the key management activity is distributed between a set of nodes rather than on a single node. In some cases, the whole group of nodes is in charge of key management [96]. An effective approach to distribute keys in a network is to employ a logical key tree. Such methods fall under the category of centralized key management protocols. Some schemes have been developed for WSNs according to logical key tree techniques [97], [98], [99]. While centralized solutions are not always the most effective ones, these methods may sometimes be very efficient for WSNs, as relatively heavier computations can be usually done in powerful base stations. Di Pietro et al. have suggested a directed diffusion-based multicast method for WSNs that makes use of a logical key hierarchy [100]. In the logical hierarchy, a central key distributor is at the transformed into an interest and then diffused throughout the network. The source node then initiates data collection from the network according to the propagated interest. Root of a tree, and the nodes in the network are the leaf level. The internal nodes of tree include keys [101].

The dissemination method also sets up definite gradients designed to draw events toward the interest. The collected data is then sent back to the source along the reverse path of the interest propagation. The directed diffusion-based logical key hierarchy scheme permits nodes to connect and leave groups. The key hierarchy is utilized to efficiently re-establish keys for the nodes below the node that has left the group. When a node joins a group, a key set is created for the new node according to the keys within the existing key hierarchy. Kaya et al. explain the problem of multicast group management [102]. In their discussion, the nodes in a network are categorized based on their locality and a security tree is built on the groups.

Lazos and Poovendran have presented a tree-based key distribution scheme that is identical to the directed diffusion-based logical key hierarchy discussed by Di Pietro et al. [103]. In their suggested scheme, a routing-aware tree is built in which the leaf nodes are assigned keys according to all relay nodes above them. As the scheme takes advantage of routing information for construction the key hierarchy, it is more energy-efficient than routing schemes that arbitrarily arrange nodes into a routing tree.

The researchers have also proposed a greedy routing-aware key distribution algorithm. In [104], the researchers have suggested a method that utilizes geographic location information (e.g. GPS data) for construction of a logical key hierarchy for secure multicast communication. The nodes, based on the geographical location information, are categorized into various clusters. The nodes within a cluster have the ability to reach each other with a single hop communication. Table 2 presents some

important security schemes and major features of them. are utilized in the re-keying process. The directed diffusion is an energy-efficient data dissemination method for WSNs. In directed diffusion, a query is

Table 2- Security schemes and attacks deterred in WSN

Security Schemes	**Attacks Deterred**	**Network Architecture**	**Major Feature**
TIK	Wormhole Attack Information or Data spoofing	Traditional wireless sensor Network	Based on symmetric cryptography, Requires accurate time synchronization between all communicating parties, implements temporal leashes
Random key pre-distribution	Data and Information spoofing, Attacks in Information in transit	Traditional wireless sensor Network	Provide resilience of the network, protect the network even if part of the network is compromised, provide authentication measures for sensor nodes.
REWARD	Blackhole Attacks	Traditional wireless sensor Network	Uses geographic routing , Takes advantage of the broadcast inter-radio behavior to watch neighbor transmission and detect Blackhole attacks
Tiny sec	Data and Information spoofing, Message Replay Attack	Traditional wireless sensor Network	Focus on providing message Authenticity, integrity and confidentiality
SNEP & µTESLA	Data and information spoofing, Message Replay Attack	Traditional wireless sensor Network	Semantic Security , Data authentication Replay Protection , Weak Freshness, Low Communication Overhead.
LEAP	Hello Attack, Sybil Attack, Sinkhole Attack, Wormhole Attack	Traditional wireless sensor Network	Using MAC , Variable Overhead , Pre-deployed key Agreement , Privacy, Source Authentication Encryption – Decryption
ZiGBEE	Sybil Attack, Sinkhole Attack, Wormhole Attack	Traditional wireless sensor Network	Encryption , Authentication, Data Freshness, Data Integrity, 4.8 or 16 Bytes Overhead , Using MAC, Trust center key agreement
802.15.4	Link Layer Security Service	Traditional wireless sensor Network	Encryption , Data Freshness, 4.8 or 16 Bytes Overhead , Using MAC Decreases Energy Consumption, Frame Integrity
Minisec	Massage Replay Attack, Data and Information Spoofing	Traditional wireless sensor Network	Encryption , Data Integrity, Data Freshness (CTR), 4 or 3 Bytes Overhead , Using MAC it can uses any Key Agreement
Secure	Eaves dropping ,	Traditional	Complex to design, Operation with

Multicasting Protocols	Interference , Data and Information Spoofing	wireless sensor Network	Limited Resources

Key Distribution / Management

One of the most important problems of symmetric cryptography is how to distribute shared keys to communicating nodes. Another challenge is to keep shared keys confidential only among the communicating hosts so that adversaries cannot reach them. Moreover, lightweight ciphers, and efficient key distribution and management are necessary security requirements for WSNs [105]. Key pre-distribution is a key management scheme where each sensor node is provided with some keys before deployment and, after receiving the target position, the sensor nodes build up a secure network between them according to those keys. The other significant aspect of WSNs is in-network processing, as it provides energy efficiency. In such cases, hierarchical key management (LEAP) is needed to provide security to various levels of communication [96]. The following discussion focuses on some related works according to these two kinds of key management protocols. In PIKE [106], Chan and Perrig discuss a method for founding a key between two sensor nodes that is based on the common trust of a third node somewhere within the sensor network. Huang et al. [107] explain a hybrid key establishment scheme that utilizes the difference in computational and energy limitations between a sensor node and the base station.

When organizing a sensor network, one of the first security requirements is to found cryptographic keys for later secure communication. The founded keys should be strong to attacks and flexible for dynamic update. Key management refers to the task of supporting the establishment and maintenance of key relationships among valid parties based on a security policy.

Static Key Management Schemes

These schemes presume that administrative keys have been used previously in the nodes and that they will not be altered. Administrative keys are created prior to deployment, assigned to nodes either accidentally or according to some deployment information, and then given out to nodes. For communication key management, static schemes employ the overlapping of compromised sensors do not lead to the compromise of pairwise keys shared among non-compromised sensors.

However, these methods still have some constraints. For the basic probabilistic and the *q*-composite key pre-distribution, a small number of compromised sensors may disclose a large fraction of pairwise keys shared among non-compromised sensors. administrative keys to establish the eligibility of neighboring nodes to create a direct pairwise communication key [108].

Most existing schemes in this case are constructed on the seminal random key pre-distribution scheme discussed by Eschenauer and Gligor [109]. Subsequent extensions to that scheme include employing key polynomials [110] and deployment knowledge [111] to improve scalability and resilience to attacks. Broadly, these static key pre-distribution schemes are made up of three steps [112]: 1) key setup prior to deployment, 2) shared-key discovery after deployment, and 3) path-key establishment if two sensor nodes do not share a key.

Random pairwise key pre-distribution scheme

The random key pre-distribution scheme was described first by Du, W. Deng [113]. Given an *n*-sensor WSN, the basic random key pre-deployment strategy described consists of the following steps in the key pre-distribution phase [114]:
1) A large pool of *P* keys (2^{17}-2^{20}keys) is generated, and 2) Each sensor is given *k* random discriminated keys from the pool.

At the shared-key discovery phase, each node sends out its set of key identifiers and gets one message from each node within its radio range. Nodes which detect that they include a shared key

can then discover that their neighbor actually keeps the key through a challenge–response protocol.
The shared key then becomes the key for that pair. The probability of key share between two sensor nodes is

$$\big((p-k)!\big)^2/(p-k)!\,p! \qquad (1)$$

After the shared-key discovery phase is done, there will be a number of unused keys left in each sensor's key ring and these keys can be put to work by each sensor node for path-key establishment. If two sensor nodes do not share a common key, they can discover an intermediate node that has shared pairwise keys with both of them. The intermediate node can act as a key distribution center to setup a pairwise key among them.

Chan et al. [114] evolved the *q*-composite key pre-distribution and the random pairwise keys schemes. The *q*-composite key pre-distribution needs two sensors to share at least *q* pre-distributed keys to establish a pairwise key. One of the characteristics of the random pairwise keys scheme is that

The security proof ensures that this scheme is secure and *t*-collusion resistant [94].
It means, the collusion of no more than *t* compromised sensor nodes, doesn't know anything about the pairwise key among any two non-compromised nodes. The most important polynomial pool-based key pre-distribution has a constraint. It can stand no more than *t* compromised nodes, where the value of *t* is restricted by the memory accessible in sensor nodes.

In a large sensor network it is more probable that there are more than *t* sensor nodes are compromised.

A really good approach is to employ a strategy for subset assignment accidentally during the setup phase [115]. That is, for each sensor the setup server chooses a random subset of created polynomials and gives the polynomial shares of these polynomials to the sensor.
If no more than *t* shares on the same polynomial are revealed, no pairwise keys constructed utilizing this polynomial among any two non-compromised sensor nodes will be revealed. If two sensors fail to found a pairwise key precisely, they must begin the

$$f(x,y)=\sum_{i,j=0}^{t} a_{ij}x^i y^j \qquad (2)$$

Polynomial pool-based pairwise key pre-distribution

Polynomial pool-based pairwise key pre-distribution is presented in [94]. In this paper, the following bivariate *t*-degree polynomial is applied. The polynomial is over a finite field Fq where q is a main number that is large enough to accommodate a cryptographic key such that it has the property of $f(x,y) = f(y,x)$. It is presumed that each sensor has an individual ID. For any two sensor nodes *u* and *v*, they use a common key $f(u,v)$ together.

To pre-distribute pairwise keys, the setup server computes a random polynomial share of $f(u,y)$ for each sensor *u*. To found a pairwise key for sensors *u* and *v*, node *u* requires computation of the common key $f(u,v)$ by evaluating $f(u,y)$ at point *v*, and node v requires computation of the same key $f(v,u) = f(u,v)$ by evaluating $f(v,y)$ at point *u*.

path-key establishment step. During this step, a source sensor node attempts to discover another node that has direct pairwise keys with both nodes. The common node acts as a KDC. It creates a random key and forwards this to the pair nodes in a secure channel. In practice, a sensor may be limited to only contacting its neighbors within a specified range for generating a path key.

Location based pairwise key pre-distribution scheme

The location-based pairwise key pre-distribution scheme employs the sensor location information to pre-distribute pairwise keys [116]. The basic idea of this method is to have each sensor share pairwise keys with its *c* closest neighbors. For each sensor u, the setup server first finds a set *S* of *c* sensors whose expected locations are closest to the expected location of u. For each sensor v in S, the setup server randomly creates an individual pairwise key $K_{u,v}$. $(u, K_{u,v})$ and $(v, K_{u,v})$ are forwarded to nodes v and u, respectively.

If two sensors u and v want to set up a pairwise key to secure the communication among them, they only need to check whether they have a pre-deployed pairwise key with the other party. The algorithm to recognize such a common key is insignificant, because each pairwise key in a specific sensor was associated with a sensor ID.
To add a new sensor after organizing the sensor network, the setup server has to notify a number of existing sensors in the network about the addition of the new sensor. It may cause a lot of communication overhead. This can be improved by an approach according to a pseudorandom function (PRF) [117] and a master key shared among each sensor and the setup server. For each pair of neighboring sensor nodes u and v, node u saves a master key K_u and node v saves a pairwise key $K_{u,v}$ where $k_{u,v}$ = PRF(K_u, v). The direct key establishment stage seems to be the major scheme. The only distinction is that one of two sensors has a pre-distributed pairwise key andthe other only needs to compute the key utilizing its master key and the ID of the other party.

Group-wise key distribution

The previously described key management schemes are focused on pairwise key distribution. For group-wise key distribution, an easy method is to employ existing pairwise keys to found group-wise keys. As an example, lightweight key management system considers a WSN where groups of sensor nodes are deployed in different phases [118]. It suggests distribution of group-wise keys through the links which are secured with pairwise keys. Yet another way is to pre-distribute polynomial shares to sensor nodes by determining which group members can create a common group key [119].

Dynamic Key Management Schemes

Dynamic key management schemes may modify administrative keys periodically it is requested or when node capture is detected. The most significant advantage of dynamic keying is the improved network survivability because any captured key(s) is replaced in a timely approach in a process known as rekeying. Another advantage of dynamic keying is that it provides better support for network scalability.

The basic problem in dynamic keying is to design a secure yet effective rekeying mechanism. A suggested solution to this challenge is to employ an exclusion-based system (EBS), a combinatorial formulation of the group key management problem discussed in [120]. The EBS assigns each node k keys from a key pool of size $k + m$. If a node capture is discovered, rekeying happens. In the rekeying process, replacement keys are created, and then encrypted with all the m keys unknown to the captured node. These are eventually distributed to other nodes that collectively know the m keys. A disadvantage to this EBS scheme is that a small number of nodes may collude and collectively disclose all the confidential keys.

LOCK is an EBS-based dynamic key management scheme for clustered sensor networks. The physical network model is a three-tier WSN with the base station at the top, followed by cluster heads, then sensor nodes. There is no presumption about location knowledge in LOCK. When the nodes are basically released into the environment, they generate a set of backup keys. These sets of backup keys are only shared with the base station, not the local cluster heads. If a node is captured, other nodes are rekeyed locally so that the compromised node does not have the ability to communicate with others.

If a cluster head is compromised, the base station starts a rekeying at the cluster head level. Also, nodes within the cluster ruled by a compromised cluster head directly rekey with the base station.

Key Management Schemes Supporting In-Network Processing

There is some key management schemes designed for supporting in-network processing, which is one of the outstanding methods in sensor networks. LEAP is a key management protocol for sensor networks that is designed to support in-network processing, while at the same time limiting the security impact of a compromised node to its immediate network neighborhood. LEAP contains efficient protocols for supporting four kinds of key schemes for various kinds of messages sent out in WSNs and includes an efficient scheme for local

broadcast authentication. LEAP is an effective scheme for key establishment that resists many types of attacks in the network, including the Sybil, sinkhole, wormhole, and so on. LEAP also provides efficient schemes for node revocation and key updating in WSNs.

CONCLUSION

One of the basic goals for wireless sensor networks (WSNs) is to collect information from the physical world. Using WSNs has lots of advantages including: 1) avoiding unnecessary wiring; 2) accommodating new devices at any time; 3) providing flexibility to go through partitions, and 4) Having enhanced mobility and reducing the cost of implementation compared to wired networks. In contrast, disadvantages can include: 1) possible loss of signal; 2) signals not readily accessing the networks; 3) the nodes in WSNs being battery-powered.

Security in sensor networks has been an increasingly important issue for academia, industry individuals and groups working in this fast growing research area. This article includes many security solutions or mechanisms that have been proposed for WSNs. However, there is no security mechanism that can provide complete security. Designing a secure WSN requires proper mapping of security solutions or mechanisms with different security parameters.

REFERENCES

[1] Culler, D. E and Hong, W., "Wireless Sensor Networks", Communication of the ACM, Vol. 47, No. 6, June 2004, pp. 30-33.

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

[3] S. Loveric, M. Sieffert ,"Survey on Security Challenges as Related to Wireless Sensor Networks", Technical Report, University of Binghamton,2007.

[4] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y, and Cayirci, E., "Wireless Sensor Networks: A Survey", Computer Networks, 38, 2002,pp. 393-422.

[5] Undercoffer, J., Avancha, S., Joshi, A., and Pinkston, J., "Security for Sensor Networks", CADIP Research Symposium, 2002, available at, http://www.cs.sfu.ca/~angiez/personal/paper/sensor-ids.pdf

[6] S. Datema. A Case Study of Wireless Sensor Network Attacks. Master's thesis, Delft University of Technology, September 2005.

[7] John PaulWalters, Zhengqiang Liang,Weisong Shi and Vipin Chaudhary. Wireless sensor network security: A survey. *Security in Distributed, Grid, and Pervasive Computing*, 2006.

[8] Zhou, L. and Haas, Z. J., "Securing ad hoc networks", IEEE Network, Volume 13, Issue 6, Nov.-Dec. 1999, pp. 24 – 30.

[9] P. Sakerindr and N. Ansari, "Security Services in Group Communications over Wireless infrastructure, Mobile Ad Hoc and Sensor Networks", *IEEE Wireless Communications*, vol. 14, no. 5, Oct. 2007, pp. 8-20.

[10] Karlof, C., Sastry, N., and Wagner, D., "TinySec: a link layer security architecture for wireless sensor networks", Proc. of the 2nd international conference on Embedded networked sensor systems, Baltimore, MD, USA, 2004, pp. 162 – 175.

[11] Perrig, A., Szewczyk, R., Wen, V., Culler, D., and Tygar, J. D., "SPINS: Security Protocols for Sensor Networks", Wireless Networks, vol. 8, no. 5, 2002, pp. 521-534.

[12] Nasser, N.; Yunfeng Chen; "Secure Multipath Routing Protocol for Wireless Sensor Networks", ICDCSW '07 Proceedings of the 27th International Conference on Distributed Computing Systems Workshops, ISBN:0-7695-2838-4, 2007.

[13] Min Zhou; Zhang-long Nie; "Analysis and design of ZigBee MAC layers protocol", International Conference on Future Information Technology and Management Engineering (FITME), ISBN: 978-1-4244-9087-5, Oct 2010.

[14] Linnarsson, F.; Peng Cheng; Oelmann, B.; "Analysis of the IEEE 802.15.4 Standard for a Wireless Closed Loop Control System for Heavy Duty Cranes", International Symposium on Industrial Embedded Systems, ISBN: 1-4244-0840-7, July 2007.

[15] Iftekhar Salam, M.; Kumar, P.; HoonJae Lee; "An efficient key pre-distribution scheme for wireless sensor network using public key cryptography", sixth International Conference on networked Computing and Advanced Information Management, ISBN: 978-89-88678-26-8, Aug 2010.

[16] D. Hankerson, A.Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*, New York, Springer-Verlag, 2004.

[17] D.J. Malan, M. Welsh, and M.D. Smith, "A public-key infrastructure for key distribution in TinyOS based on elliptic curve cryptography", In *Proceedings of the 1st IEEE International Conference on Sensor and Ad Hoc Communications and Networks*, Santa Clara, California, October, 2004.

[18] R.L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", *Communications of the ACM*, Vol. 26, No. 1, pp. 96-99, 1983.

[19] D.W. Carman, P.S. Krus, and B.J. Matt, "Constraints and approaches for distributed sensor network security", Technical Report 00-010, NAI Labs, Network Associates Inc., Glenwood, MD, 2000.

[20] N. Gura, A. Patel, A. Wander, H. Eberle, and S. Shantz, "Comparing elliptic curve cryptography and RSA on 8-bit

CPUs", In *Proceedings of Workshop on Cryptographic Hardware and Embedded Systems (CHES '04)*, August 2004.
[21] G. Gaubatz, J.P. Kaps, and B. Sunar, "Public key cryptography in sensor networks-Revisited", In *Proceedings 1st European Workshop on Security in Ad-Hoc and Sensor Networks (ESAS '04)*, 2004.
[22] A.S. Wander, N. Gura, H. Eberle, V. Gupta, and S.C. Shantz, "Energy analysis of public-key cryptography for wireless sensor networks", In *Proceedings of the 3rd IEEE International Conference on Pervasive Computing and Communication*, March 2005.
[23] M.O. Rabin, "Digitalized Signatures and Public Key Functions as Intractable as Factorization", Cambridge, MA, Technical Report, 1979.
[24] J. Hoffstein, J. Pipher, and J.H. Silverman, "Ntru: A ringbased public key cryptosystem", In *Proceedings of the 3rd International Symposium on Algorithmic Number Theory*, London, Springer-Verlag, 1998, pp. 267-288.
[25] V.S. Miller, "Use of elliptic curves in cryptography", In *Lecture Notes in Computer Sciences: 218 on Advances in Cryptology- CRYPTO 85*, New York, Springer-Verlag, 1986, pp. 417-426.
[26] N. Kobiltz, "Elliptic curve cryptosystems", *Mathematics of Computation*, Vol. 48, pp. 203-209, 1987.
[27] Elliptic Curve Cryptography, SECG Std. SEC1, 2000, available at http://www.secg.org/collateral/sec1.pdf.
[28] B. Kaliski, TWIRL and RSA Key Size, RSA Laboratories, Technical Note, May 2003.
[29] Recommended Elliptic Curve Domain Parameters, SECG Std. SEC2, 2000, available at http://www.secg.org/collateral/sec2.pdf.
[30] Song Wen; Ruiying Du; "The Conformation of Trusted Sensor Network", International Conference on Wireless Communication, Networking and Mobile Computing; ISBN:978-1-4244-1311-9, Sep 2007 .
[31] Gottesman, D.; "Private key and public key quantum cryptography", Summaries of presented at the Quantum Elecronics and Laser Science Conference. ISBN: 1-55752-708-3, 2002.
[32] Qian Yu; Zhang, C.N.; "RC4 state and its applications", Ninth Annual International Conference on Privacy, Security and Trust(PST), ISBN:978-1-4577-0582-3, July 2011.
[33] Elkeelany, O.; Olabisi, A.; "Case study: integrated design of RC5 encryption", In IEEE Proceeding, ISBN:1-4244-1029-0, March 2007.
[34] Ceri, S.; Fraternali, P.; Paraboschi, S.; Baralis, E.; Fravega, A.; "The IDEA tool set", 13th International Conference on Data Engineering, ISBN:0-8186-7807-0, April 1997.
[35] Cheng Xiao-hui; Deng Jian-zhi; "Design of SHA-1 Algorithm Based on FPGA", Second International Conference on Neworks Security Wireless Communications and Trusted Computing(NSWCTS), ISBN:978-1-4244-6598-9, April 2010.
[36] Chahar, R.K.; Datta, G.; Rajpal, N.; "Design of a New Security Protocol", Internaional Conference on Computational Intelligence and Multimedia Application, ISBN:0-7695-3050-8, Dec 2007.
[37] Xue Li; Chakravarthy, V.D.; Bin Wang; Zhiqiang Wu; "Spreading Code Design of Adaptive Non-Contiguous SOFDM for Dynamic Spectrum Access", IEEE Journal of Selected Topics in Signal Processing, ISSN:1932-4553, Feb 2011.
[38] D.Wood, A.Stankovic; "Danial of Service in Sensor Networks", IEEE Journal, ISSN: 0018-9162, 2002.
[39] Zhang, Deng-yin; Xu, Chao; Siyuan, Lin; "Detecting selective forwarding attacks in WSNs using watermark", International Conference on Wireless Communications and Signal Processing (WCSP), ISBN:978-1-4577-1008-7, Nov 2011.
[40] Zhang Laishun; Zhang Minglei; Guo Yuanbo; "A Client Puzzle Based Defense Mechanism to Resist DoS Attacks in WLAN", International Conference Forum on Information Technology (IFITA), ISBN:978-1-4244-7622-0, July 2010.
[41] J. Newsome, R. Shi, D. Song, and A. Perrig. The sybil attack in sensor networks: analysis & defenses. In *Proceedings of 3rd International Symposium on Information Processing in Sensor Networks (IPSN ?4)*, April 2004.
[42] Jinchao, Zhao; "Research on Key Predistribution Scheme of Wireless Sensor Networks", Fifth International Conference on Intelligent Computation Technology and Automation (ICICTA), ISBN:978-1-4673-0470-2, Jan 2012.
[43] Hu, Y.-C.; Perrig, A.; Johnson, D.B.; "Packet leashes: a defense against wormhole attacks in wireless networks", Twenty Second Annual Joint Conference of the IEEE Computer and Communications, ISBN:0-7803-7752-4, April 2003.
[44] Jin Guo; Zhi-yong Lei; "A kind of wormhole attack defense strategy of WSN based on neighbor nodes verification", IEEE 3rd International Conference on Communication Software and Networks (ICCSN), ISBN: 978-1-61284-485-5, May 2011.
[45] Prasannajit, B.; Venkatesh; Anupama, S.; Vindhykumari, K.; Subhashini, S.R.; Vinitha, G.; "An Approach Towards Detection of Wormhole Attack in Sensor Networks", First International Conference on Integrated Intelligent Computing (ICIIC), ISBN:978-0-7695-4152-5, Aug 2010.
[46] Parno, B.; Perrig, A.; Gligor, V.; "Distributed detection of node replication attacks in sensor networks", IEEE Symposium on Security and Privacy, ISSN:1081-6011, May 2005.
[47] Deng,j.; Han, R.; Mishra, S.; "Countermeasure Against Traffic Analysis Attacks in Wireless Sensor", SECURECOMM 2005.
[48] Sisheng Chen; Li Xu; Zhide Chen; "Secure Anonymous Routing in Trust and Clustered Wireless Ad Hoc Networks", Second International Conference on Communications and Networking, ISBN:978-1-4244-1009-5, Aug 2007.
[49] M. Gruteser and D. Grunwald, "A methodological assessment of location privacy risks in wireless hotspot networks, In *Proceedings of the 1st International Conference on Security in Pervasive Computing*, 2003.
[50] M. Gruteser, G. Schelle, A. Jain, R. Han, and D. Grunwald, "Privacy-aware location sensor networks", In *Proceedings of the 9th USENIX Workshop on Hot Topics in Operating Systems, (HotOS IX)*, 2003.

[51] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location support system", In *Proceedings of the 6th Annual ACM International Conference on Mobile Computing and Networking (MOBICM)*, August 2000.
[52] A. Smailagic, D.P. Siewiorek, J. Anhalt, and Y. Wang, D. Kogan, "Location sensing and privacy in a context aware computing environment", *Pervasive Computing* 2001.
[53] Popa, M.; Ansari, J.; Riihijarvi, J.; Mahonen, P.; "Combining Cricket System and Inertial Navigation for Indoor Human Tracking", IEEE Wireless Communications and Networking Conference, ISSN: 1525-3511, April 2008.
[54] Kim, Dong Seong; Shin, Taek-Hyun; Park, Jong Sou; "A Security Framework in RFID Multi-domain System", The Second International Conference on Availability, Reliability and Security, ISBN: 0-7695-2775-2, April 2007.
[55] Hyo-Sang Lim; Chenyun Dai; Bertino, E.; "A policy-based approach for assuring data integrity in DBMSs", 2010 6th International Conference on Collaborative Computing, Networking, Application and worksharing, ISBN:978-963-9995-24-6, Oct 2010.
[56] E. Snekkenes, "Concepts for personal location privacy policies", In *Proceedings of the 3rd ACM Conference on Electronic Commerce*, pp. 48-57, ACM Press, 2001.
[57] G. Myles, A. Friday, and N. Davies, "Preserving privacy in environments with location-based applications", *IEEE Pervasive Computing*, Vol. 2, No. 1, pp. 56-64, 2003.
[58] U. Hengartner and P. Steenkiste, "Protecting access to people location information", In *Proceedings of the 1st International Conference on Security in Pervasive Computing* , LNCS, Springer, March 2003.
[59] C. Ozturk, Y. Zhang, and W. Trappe, "Source-location privacy in energy-constrained sensor network routing ", In *Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks*, 2004.
[60] J. Deng, R. Han, and S. Mishra, "Countermeasures against traffic analysis in wireless sensor networks", Technical Report CU-CS-987-04, University of Colorado at Boulder, 2004.
[61] Y. Xi, L. Schwiebert, and W. Shi, "Preserving privacy in monitoring-based wireless sensor networks", In *Proceeding of the 2nd International Workshop on Security in Systems and Networks (SSN '06)*, IEEE Computer society, 2006.
[62] A.D. Wood and J.A. Stankovic, "Denial of service in sensor networks", *IEEE Computer*, Vol. 35, No. 10, pp. 54-62, 2002.
[63] I. Sato, Y. Okazaki, and S. Goto, "An improved intrusion detection method based on process profiling", *IPSJ Journal*, Vol. 43, No. 11, pp. 3316-3326, 2002.
[64] S. Marti, T.J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks", In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, New York, ACM Press, 2000, pp. 255-265.
[65] Y. Zhang, W. Lee, and Y.-A. Huang, "Intrusion detection techniques for mobile wireless networks, *Wireless Networks*, Vol. 9, No. 5, pp. 545- 556, 2003.
[66] Y. Huang, W. Fan, W. Lee, and P.S. Yu, "Cross-feature analysis for detecting ad-hoc routing anomalies", In *Proceedings of the 23rd International Conference on Distributed Computing Systems*, Providence, RI, May 2003.
[67] Y. Huang and W. Lee, "Attack analysis and detection for ad hoc routing protocols", In *Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection*, Sophia Antipolis, France, September 2004.
[68] P. Brutch and C. Ko, "Challenges in intrusion detection for wireless ad-hoc networks", In *Proceedings of the Symposium on Applications and the Internet Workshops (SAINT'03 Workshops)* 2003.
[69] S. Zhu, S. Setia, S. Jajodia, and P. Ning, "An interleaved hop by hop authentication scheme for filtering of injected false data in sensor networks", In *Proceedings of IEEE Symposium on Security and Privacy*, Oakland, CA, May 2004, pp. 259-271.
[70] G. Wang, W. Zhang, C. Cao, and T.L. Porta, "On supporting distributed collaboration in sensor networks", In *Proceedings of MILCOM*, 2003.
[71] P. Albers and O. Camp, "Security in ad hoc networks: A general intrusion detection architecture enhancing trust-based approaches", In *Proceedings of the 1st International Workshop on Wireless Information Systems, 4th International Conference on Enterprise Information Systems*, 2002.
[72] D. Estrin, R. Govindan, J.S. Heidemann, and S. Kumar, "Next century challenges: Scalable coordination in sensor networks", *Mobile Computing and Networking*, pp. 263-270, 1999.
[73] F. Ye, L.H. Luo, and S. Lu, "Statistical en-route detection and filtering of injected false data in sensor networks", In *Proceedings of IEEE INFOCOM*, Hong Kong, 2004.
[74] A.D. Wood and J.A. Stankovic, "Denial of service in sensor networks", *IEEE Computer*, Vol. 35, No. 10, pp. 54-62, 2002.
[75] R. Anderson and M. Kuhn, "Low cost attacks on tamper resistant devices", In *Proceedings of the International Workshop on Security Protocols (IWSP), Lecture Notes in Computer Science(LNCS)*, 1997.
[76] R. Anderson and M. Kuhn, "Tamper resistance- a cautionary note", In *Proceedings of the 2nd USENIX Workshop on Electronic Commerce*, Oakland, California, 1996.
[77] O. Komerling and M.G. Kuhn, "Design principles for tamper-resistant smart card processors", In *Proceedings of USENIX Workshop on Smartcard Technology*, Chicago, Illinois, USA, May 1999.
[78] J. Deng, R. Han, and S. Mishra, *"Security, privacy, and fault tolerance in wireless sensor networks"*, Artech House, August 2005.
[79] N. Sastry, U. Shankar, and D. Wagner, "Secure verification of location claims", In *Proceedings of ACM Workshop on Wireless Security*, September 2003.
[80] X. Wang, W. Gu, S. Chellappan, Dong Xuan, and Ten H. Laii, "Search-based physical attacks in sensor networks: Modeling and defense, Technical report, Department of Computer Science and Engineering, Ohio State University, February 2005.
[81] X. Wang , W. Gu, S. Chellappan, K.Schoseck, and D. Xuan, "Lifetime optimization of sensor networks under

physical attacks", *In Proceedings of IEEE International Conference on Communications*, May 2005.
[82] A. Seshadri, A. Perrig, L. Van Doorn, and P.Khosla,"SWATT: Software-based attestation for embedded devices", In *Proceedings of the IEEE Symposium on Security and Privacy*, May 2004.
[83] A. Pirzada and C. McDonald, "Establishing trust in pure ad hoc networks", In *Proceedings of the 27th Australian Conference on Computer Science*, Dunedin, New Zealand, 2004, pp. 47-54.
[84] A. Oram, "Peer-to-Peer: Harnessing the power of disruptive technologies", *O'Reilly & Associates*, March 2001.
[85] H. Zhu, F. Bao, R.H. Deng, and K. Kim, "Computing of trust in wireless networks", In *Proceedings of 60th IEEE Vehicular Technology Conference*, Los Angeles, California, September, September 2004.
[86] Z. Yan, P. Zhang, and T. Virtanen, "Trust evaluation based security solution in ad hoc networks", In *Proceedings of the 7th Nordic Workshop on Secure IT Systems*, 2003.
[87] K. Ren, T. Li, Z. Wan, F. Bao, R.H. Deng, and K. Kim, "Highly reliable trust establishment scheme in ad hoc networks", *Computer Networks: The International Journal of Computer and telecommunications Networking*, Vol 45, pp. 687-699, August 2004.
[88] S. Ganeriwal and M. Srivastava, "Reputation-based framework for high integrity sensor networks", In *Proceedings of the 2nd ACM Workshop on Security on Ad Hoc and Sensor Networks*, Washington DC, USA, 2004.
[89] S. Tanachaiwiwat, P. Dave, R. Bhindwale, and A. Helmy, "Location-centric isolation of misbehavior and trust routing in energy-constrained sensor networks", In *Proceedings of IEEE International Conference on Performance, Computing, and Communications*, pp. 463-469, April 2004.
[90] Z. Liang and W. Shi, "Analysis of ratings on trust inference in the open environment", Technical report MIST-TR-2005-002, Department of computer Science, Wayne State University, February 2005.
[91] Z. Liang and W. Shi, "Enforcing cooperative resource sharing in untrusted peer-to-peer environment", *ACM Journal of Mobile Networks and Applications (MONET)*, Vol 10, No. 6, pp. 771-783, 2005.
[92] Z. Liang and W. Shi, "PET: A Personalized Trust model with reputation and risk evaluation for P2P resource sharing", In *Proceedings of the HICSS-38*, Hilton Waikoloa Village Big Island, Hawaii, January 2005.
[93] Khademi, M.; Tinati, M.A.; "Audio steganography by using of linear predictive coding analysis in the safe places of discrete wavelet transform domain", 19th Iranian Conference on Electrical Engineering (ICEE), ISBN: 978-964-463-428-4, May 2011.
[94] S. Tanachaiwiwat, P. Dave, R. Bhindwale, and A. Helmy. Poster abstract secure locations: routing on trust and isolating compromised sensors in locationaware sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 324–325. ACM Press, 2003.
[95] S. Zhu and S. Setia and S. Jajodia. Leap: Efficient security mechanisms for large-scale distributed sensor networks. In Proc. of the 10th ACM Conference on Computer and Communications Security (CCS '03). 62–72. 2003.
[96] S. Rafaeli and D. Hutchison, "A survey of key management for secure group communications", *ACM Computing Survey*, Vol 35, No. 3, pp. 309-329, 2003.
[97] L. Lazos and R. Poovendran, "Secure broadcast in energy aware wireless sensor networks", In *IEEE International Symposium on Advances in Wireless Communications (ISWC'02)*, 2002.
[98] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks", *Mobile Computing and Networking*, pp.56-67, 2000.
[99] R. Di Pietro, L.V. Mancini, Y.W. Law, S. Etalle, and P. Havinga, "LKHW: A directed diffusion-based secure multicast scheme for wireless sensor networks", In *Proceedings of the 32nd International Conference on Parallel Processing Workshops (ICPPW'03)*, IEEE Computer Society Press, 2003, pp. 397- 406.
[100] L. Lazos and R. Poovendran, "Energy-aware secure multicast communication in ad-hoc networks using geographic location information", In *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, 2003.
[101] T. Kaya, G. Lin, G. Noubir, and A. Yilmaz, "Secure multicast groups on ad hoc networks", In *Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '03)*, pp. 94-102, ACM Press 2003.
[102] L. Lazos and R. Poovendran, "Secure broadcast in energy aware wireless sensor networks", In *IEEE International Symposium on Advances in Wireless Communications (ISWC'02)*, 2002.
[103] Bala Krishna, M.; Doja, M.N.; "Symmetric key management and distribution techniques in wireless ad hoc networks ", International Conference on Computational Intelligence and Communication Networks, ISBN: 978-1-4577-2033-8, Oct 2011.
[104] S. Zhu, S. Setia, and S. Jajodia, "LEAP: Efficient security mechanism for large –scale distributed sensor networks", In *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pp. 62-72, New York, NY, USA, 2003, ACM Press.
[105] Haowen Chan and Adrian Perrig. "PIKE: Peer Intermediaries for Key Establishment in Sensor Networks". In Proceedings of IEEE Infocom, March 2005.
[106] Q. Huang, J. Cukier, H. Kobayashi, B. Liu, and J. Zhang. Fast authenticated key establishment protocols for self-organizing sensor networks. Proc. of 2nd ACM WSNA Conference, pp.141-150, 2003.
[107] M. Eltoweissy, M. Moharrum, and R. Mukkamala, Dynamic key management in sensor networks, *IEEE* AQ3 *Communications Magazine*, April 2006.
[108] L. Eschenauer and V.D. Gligor, A key-management scheme for distributed sensor networks, in *Proc. 9th ACM Conference on Computer and Communications Security*, November 2002, pp. 41–47.
[109] C.Blundo, A. DeSantis,A.Herzberg, S.Kutten, U.Vaccaro, andM.Yung, Perfectly-secure key distribution for dynamic conferences, in *Advances in Cryptology—CRYPTO '92*, LNCS 740, 1993, pp. 471–486.

[110] D. Liu and P. Ning, Location based pairwise key establishments for static sensor networks, in *1st ACM Workshop on Security of Ad Hoc and Sensor Networks*, 2003.
[111] L. Xing and H.E.Michel, Integrated modeling for wireless sensor networks reliability and security, *Annual Symposium on Reliability and Maintainability (RAMS'06)*, January 2006, pp. 594–600.
[112] I.Onat andA.Miri, An intrusion detection system for wireless sensor networks, in *Proc. IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'2005)*, Vol. 3, August 2005, pp. 253–259.
[113] H. Chan, A. Perrig, and D. Song, Random key predistribution schemes for sensor networks, in *IEEE Symposium on Research in Security and Privacy*, May 2003, pp. 197–213.
[114] D. Liu and P. Ning, Establishing pair-wise keys in distributed sensor networks, in *Proc. 10th ACM Conference on Computer and Communications Security*, October 2003, pp. 52–61.
[115] O. Goldreich, S. Goldwasser, and S. Micali, How to construct random functions, *Journal of the ACM*, 33(4), October 1986, pp. 792–807.
[116] B. Dutertre, S. Cheung, and J. Levy, Lightweight key management in wireless sensor networks by leveraging initial trust, Technical Report, SRI-SDL-04-02, System Design Laboratory, April 2004.
[117] L. Morales, I.H. Sudborough, M. Eltoweissy, and M.H. Heydari, Combinatorial optimization of multicast key management, in *Proc. 36th Hawaii International Conference on System Sciences*, 2002.
[118] M. Eltoweissy, M. Moharrum, and R. Mukkamala, Dynamic key management in sensor networks, *IEEE* AQ3 *Communications Magazine*, April 2006.
[119] A. Freier, P. Karlton, and P. Kocher, „The SSL Protocol", version 3.0., available at http://home.netscape.com/eng/ssl3/
[120] S. Zhu, S. Setia, and S. Jajodia, LEAP: Efficient security mechanisms for large-scale distributed sensor networks, in *Proc. 10th ACM Conference on Computer and Communications Security (CCS'03)*, October 2003.