Survey on Wireless Sensor Network for Message Authentication and Source Privacy

Vaishali Kisanrao Gulhane[1], **S. N. Shelke**[2]

Computer Engineering, Sinhgad Academy of Engineering, Kondhwa, Pune.

Professor, Computer Engineering, Sinhgad Academy of Engineering, Kondhwa, Pune.

Abstract: ***Message authentication is one of the most effective ways to defeat unauthorized and corrupted messages from being forwarded in wireless sensor networks (WSNs). Wireless Sensor Networks (WSN) are being very popular day by day, however one of the main concern in WSN is its limited resources. One have to look to the resources to generate Message Authentication Code (MAC) keeping in mind the feasibility of method used for the sensor network at hand.For this reason, many message authentication schemes have been developed, based on either symmetric-key cryptosystems or public-key cryptosystems. when the number of messages transmitted is larger than this threshold, the adversary can fully recover the polynomial. In this paper, we propose a scalable authentication scheme based on elliptic curve cryptography (ECC). While enabling intermediate nodes authentication, our proposed scheme allows any node to transmit an unlimited number of messages without suffering the threshold problem.***

Keywords: Hop-by-hop authentication, symmetric-key cryptosystem, public-key cryptosystem, source privacy, simulation, wireless sensor networks (WSNs), distributed algorithm, decentralized control.

1. Introduction

Wireless sensor networks simplify the collection and analysis of data from multiple locations. Target tracking and perimeter intrusion detection applications benefit from the ad-hoc deployment and self-organization capabilities of wireless sensor networks. However, sensor networks deployed in hostile environments must be fortified against attacks by adversaries. This thesis solves the security problem in wireless sensor networks deployed for surveillance and target tracking by applying appropriate security mechanisms to a target tracking method, Optimized Communication and Organization. Nodes of a WSN implement three main functionalities: sensing of the environment,aggregation and storage of recorded data and communication between the nodes. The communication between the nodes is particular important, because it is the only way for the sensing nodes to move recorded data to a node or machine which will store and analyze it.

Security requirements to prevent modification and insertion of false data into the network, which would otherwise alter the overall results. This can be achieved using Message Authentication Codes (MACs) or cryptographic signatures which are attached to network packets and validated by the receiver. Another approach,using classic Public Key Cryptography (PKC) with Public Key Infrastructure (PKI),involves a huge key distribution problem on a distributed network of wireless sensor nodes, since every node would need access to the senders' public keys. In this work we will give a general overview on possible authentication options for the particular constraints and characteristics of WSNs. This includes well established schemes like MACs, classical PKC, i.e. RSA signatures, but also more novel concepts like Cryptographically Generated Address (CGA), Identity-based Signature (IBS) and Attribute-based Signature (ABS). In addition, we do a brief analysis of their viability for use in authenticating nodes in WSNs. A recent publication showed that IBC is particular suitable for WSNs and compared various IBC signature algorithms for their application in WSNs. However their work was concentrated around pairing-based IBC algorithms

2. Related Work

Data source and base station are two important factors in the protection of location privacy in WSNs . Our discussions of related work focus on surveying the current techniques concerning the privacy protection of data source. Existing techniques of preserving the source–location privacy can be categorized into four typical classes: flooding, random walk, dummy injection, and fake sources.

WSNs target applications need a number of requirements which include range, antenna type, target technology, components, memory, storage, power, lifetime, security, computational capability, communication technology, size and programming interface. Lot of research has been done in the field of WSN, and nowadays with all kinds of survey made it was found that WSN is becoming too prone to attacks.

3. Terminology and Preliminary

This section briefly describes the terminology and the cryptographic tools.

3.1 Threat Model and Assumptions

The wireless sensor networks are implicit to consist of a huge number of sensor nodes. It is assumed that each sensor node recognizes its relative location in the sensor domain and is competent of communicating with its neighboring nodes directly using geographic routing. The entire network is fully connected through multi-hop communications. It is assumed that there is a security server (SS) that is liable for generation, storage and distribution of the security parameters among the network. This server will by no means be compromised. However, after deployment, the sensor nodes may be compromised and captured by attackers. Once compromised, all data stored in the sensor nodes can be obtained by the attackers. The compromised nodes can be reprogrammed and completely managed by the attackers. However, the compromised nodes will be unable to produce new public keys that can be accepted by the SS and other nodes. Two types of possible attacks launched by the adversaries are:

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

3.2 Design Goals

Our proposed authentication scheme aims at achieving thefollowing goals:

- **Node authentication:** The message receiver should be able to verify whether a received message is sent by the node that is claimed, or by a node in a particular group. In other words, the adversaries cannot pretend to be an innocent node and inject fake messages into the network without being detected.
- **Intermediate node authentication:** Every forwarder on the routing path should be able to verify the authenticity and integrity of the messages upon reception.
- **Message authentication:** The message receiver should be competent to authenticate whether a received message is sent by the node that is claimed or by a node in a exacting group. In other words, the adversaries cannot pretend to be a guiltless node and insert fake messages into the network without being captured.
- **Message integrity*:*** The message receiver should be clever to authenticate whether the message has been modified en-route by the adversaries. In other words, the adversaries cannot alter the message information without being detected.
- **Hop-by-hop message authentication:** Every forwarder on the routing path should be capable to validate the authenticity and integrity of the messages upon reception.
- **Identity and location privacy:** The adversaries cannot settle on the message sender's ID and location by analyzing the message data or the local traffic.
- **Node compromise resilience:** The scheme should be resilient to node compromise attacks. I does not matter how many nodes are compromised, the remaining nodes can still be safe.
- **Efficiency:** The scheme should be proficient in terms of both computational and communication overhead.

3.3. Terminology

Privacy is sometimes referred to as namelessness. It generally refers to the state of being unidentifiable within the ambiguity set (AS). Sender namelessness means that a particular message is not linkable to any sender, and no message is linkable to a particular sender.

4. Proposed Source Anonymous Message Authentication On Elliptic Curves-

In this section, we propose an unconditionally secure and efficient source anonymous message authentication scheme (SAMA). Our design enables the SAMA to be verified through a single equation without individually verifying the signatures.

A. Proposed MES Scheme on Elliptic Curves

Let p > 3 be an odd prime. An elliptic curve E is defined by an equation of the form: E : y 2 = x 3 + ax + b mod p, where a, b ∈Fp, and 4a 3 + 27b 2≠ 0 mod p. The set E(Fp) consists of all points (x, y) ∈Fp on the curve, together with a special point O, called the point at infinity. Let G = (xG, yG) be a base point on E(Fp) whose order is a very large value N. User A selects a random integer dA∈ [1, N − 1] as his private key. Then, he can compute his public key QA from QA = dA × G.

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

1) Select a random integer kA, 1 ≤ kA ≤ N − 1.
2) Calculate r = xA mod N, where (xA, yA) = kAG. If r = 0, go back to step 1.
3) Calculate hA l ←− h(m, r), where h is a cryptographic hash function, such as SHA-1, and l ←− denotes the l leftmost bits of the hash.
4) Calculate s = rdAhA + kA mod N. If s = 0, go back to step 2.
5) The signature is the pair (r, s).
6) When computing s, the string hA that results from h(m, r) shall be converted into an integer. Note that hA can be greater than N, but not longer.

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

1) Checks that QA≠ O, otherwise it is invalid
2) Checks that QA lies on the curve
3) Checks that nQA = O

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

1) Verify that r and s are integers in [1, N − 1]. If not, the signature is invalid.
2) Calculate hA l ←− h(m, r), where h is the same function used in the signature generation.
3) Calculate (x1, x2) = sG − rhAQA mod N.
4) The signature is valid if r = x1 mod N, it is invalid otherwise.

B. Proposed SAMA on Elliptic Curves-

Suppose that the message sender (say Alice) wishes to transmit a message m anonymously from her network node to any other nodes. The AS includes n members, A1, A2, · · · , An, e.g., S = {A1, A2, · · · , An}, where the actual message sender Alice is At, for some value t, 1 ≤ t ≤ n. In this paper, we will not distinguish between the node Ai and its public key Qi . Therefore, we also have S = {Q1, Q2, · · · ,Qn}.

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

1) Select a random and pairwise different ki for each 1 ≤ i ≤ n − 1, i≠ t, and compute ri from (ri ,yi) = kiG.
2) Choose a random ki∈Zp and compute rt from (rt, yt) = ktG − ∑ i≠t rihiQi such that rt ≠ 0 and rt≠ ri for any i≠ t, where hi l ←− h(m, ri).
3) Compute s = kt + ∑ i≠t ki + rtdtht mod N. The SAMA of the message m is defined as: S(m) = (m, S, r1, y1, · · · , rn, yn, s).

C. Verification of SAMA

Verification algorithm:

For Bob to verify an alleged SAMA (m, S, r1, y1, · · · ,rn, yn, s), he must have a copy of the public keys Q1, · · · , Qn. Then he:

1) Checks that Qi≠ O, i = 1, · · · , n, otherwise it is invalid
2) Checks that Qi , i = 1, · · · , n lies on the curve
3) Checks that nQi = O, i = 1, · · · , n After that, Bob follows these steps:

1) Verify that ri ,yi , i = 1, · · · , n, and s are integers in [1, N − 1]. If not, the signature is invalid.
2) Calculate hi l ←− h(m, ri), where h is the same function used in the signature generation.
3) Calculate (x0, y0) = sG − ∑n i=1 rihiQi .
4) The signature is valid if the first coordinate of ∑ i (ri ,yi) equals x0, invalid it is otherwise.

D. Security Analysis

Theorem1. The proposed source-anonymous message authentication scheme (SAMA) can provide unconditional message sender anonymity.

Theorem2. The proposed SAMA is secure against adaptive chosen-message attacks in the random oracle model.

5. AS Selection and Source Privacy

The appropriate selection of an AS plays a key role in message source privacy since the actual message source node will be hidden in the AS. In this section, we will discuss techniques that can prevent the attacker from tracking the message source through the AS analysis in combination with the local traffic analysis. Before a message is transmitted, the message source node selects an AS from the public key list in the SS as its choice. This set should include itself, together with some other nodes. When an attacker receives a message, he can possibly find the direction of the previous hop, or even the real node of the previous hop. However, if the attacker is unable to monitor the traffic of the previous hop, then he will be unable to analyze whether the previous node is the actual source node or simply a forwarder node. Therefore, the selection of the AS should create sufficient diversity so that it is infeasible for the attacker to find the message source based on the selection of the AS itself.

6. Key Management and Compromised Node Detection

6.1 Key Management Process

Various schemes have been proposed in the literature for key management techniques, these schemes have focused on many phases that are needed for this process to secure the WSN and to overcome the preceding obstacles in WSN. We illustrate here three key management schemes and explain the most important phases for each one of them. The three management techniques that we analysing are:

6.1.1. The First Technique that mentioned in [1]

As the name implies, this technique is designed for the Heterogeneous Sensor Networks (HSN) that is formed of many clusters. Each cluster is composed of one highly equipped sensor node that is called the cluster head or sink [1], and a number of less equipped sensor nodes, which are the typical sensor nodes. This key management scheme is having the following phases:

6.1.1.1. Pre-distribution Phase

This step is happened before the deployment of the sensors, there are many mechanisms to do this step, such as, the Pair-wise Key Pre-distribution, the Master Key Based Pre-distribution, the Base Station Participation, and the Probabilistic Key Pre-distribution. The Base Station (BS) mechanism is used in the Key-chain approach. So that, some calculations need to take place prior to the nodes deployment process. These calculations start with the generation of two key chains. These chains generation is done by the Base Station (BS) using the two one-way functions F1 and F2: {n10k10, n11k11, n12k12, n13k13..., n1nk1n} {n20k20, n21k21, n22k22, n23k23..., n2nk2n} Where k1 (n+1) = F1 (k1n), k2 (n+1) = F2 (k2n); n1n is the ID of key k1n on the first key-chain, and n2n is the ID of key k2n on the second key-chain. Each sensor, i.e., typical node, is pre-loaded with n1, n2, and an initial key Kinit. n1 and n2 are the IDs of each key on the two key chains, that have been generated by BS for each sensor, and Kinit = K1n1 ⊕ K 2n2 . The Cluster Head, i.e. the highly equipped node will be pre-loaded with F1, F2, K10, and K20, where F1 and F2 are the two one-way functions, K10 and K20 are the first keys on each key-chain.

6.1.1.2. Pair-wise Key Establishment-

This step may happen in different forms depending on who are these pairs. We have pair-wise key establishment and authentication that happens between nodes of the same type, cluster key establishment and authentication that happens between two different types of sensor nodes, and the global key establishment and authentication that happens in what called the distributed WSNs that has only a manager node without the existence of the cluster head. The Key-chain technique uses two types of key establishment, as a first step, it makes a pair-wise key between the cluster head and sensor, but before doing it, the node that has the ability to play the cluster head role will generate two random numbers N1, N2 and calculate the cluster key Kbrod, as Kbrod = K1N1 ⊕K2N2. The cluster head could do this calculation because it is pre-loaded with F1 and F2, the two one-way functions.

International Journal of Network Security & Its Applications (IJNSA) Vol.6, No.6, November 2014 53 The scenario of forming the cluster head starts with broadcasting a Hello message by the cluster head to the nearby sensors, this message includes the ID of the cluster head and will be called message 1. Each sensor receive a Hello message will join the cluster of the cluster head that sends a Hello message with the best signal noise ratio (SNR). After choosing the cluster head, sensor sends message to the cluster head, this message contains the sensor ID, and the two random numbers n1 and n2, and this message is message 2 in this scenario. When the cluster head receives message 2, it calculates the Kinit for each sensor joined its cluster and find a new key called the pair-wise key Kpair, where Kpair = K|N1-n1 |⊕ K|N2-n2|. This step comes to authenticate the sensor's legitimacy. Then the cluster head generates a random number R1 for the next time communication. After this the cluster head will send message 3 for the sensor, which contains an encryption for the R1 value,Kpair, and Kbrod, this encryption will be done using the Kinit. So only the legal sensor has the right Kinit, and could decrypt message 3 to obtain the information with it. If the sensor is a legal one, it will get the values of R1, Kpair, and the cluster key Kbrod, and stores those values. Then it will reply the cluster head with message 4, which has the encryption of R1 with Kpair. The cluster head must decrypt message 4, and checks R1, in case it matches the original value, the cluster head would store the sensor ID and Kpair. Till now the cluster head and each sensor belong to its cluster establish pair-wise key for future communication. As a second step in the Pair-wise key establishment phase, the Key-chain approach uses another type of pair-wise key establishment, the type that happens between two nodes of the same category, i.e. between the clusters heads. Communications between the BS and the cluster heads could be achieved by using the relaying strategy. All cluster heads send data to the BS via multihops of other cluster heads. At beginning, the distant or the far away cluster head tries to join into the close cluster, and that means it will be a child node for the cluster head of this cluster. After the successfully joining, both cluster heads broadcast the random number N2, and this is message 1 for this scenario. Then both cluster heads can calculate the pair-wise keys use the following equation, where N2 ' is the other node random number. Kpair = K2N2 ; N2 = N2 ' K2N2 ⊕ K2N2; N2 ≠ N2 ' At last, cluster head distribute the cluster key to child cluster head in message 2, after encryption this information using the calculated Kpair in the previous step.

6.1.1.3. Key Renovation

This step means having the ability to re-keying the sensors with new keys as a way to have an intrusion detection mechanism to detect compromised nodes. So the key renovation and revocation phase is an essential component in key management techniques. Using the Key-chain technique that mentioned in [3], each sensor in the cluster has a unique pairwise key, if we don't consider the probability of more than two sensors pre-loaded with the same n1, n2. If a node is compromised, we only need to delete the related pair-wise key in its cluster head. This will not affect the other nodes and links. As soon as a node is compromised or the key period expired, the cluster head will renew the cluster key. Then cluster head generate another two random numbers N1’, N2’, calculate the new cluster key Kbrod’, and distribute it to the cluster numbers encrypting with each pair-wise key. To reduce the communication costs, a piece of message can include several nodes key renovation information.

6.2. Compromised Node Detection-

Many techniques have been proposed till now for detection and recovery of compromised node. This paper gives some idea regarding various compromised node detection and

6.2.1Weighted Trust Evaluation Scheme

The author introduced weighted trust evaluation scheme in hierarchical network architecture, which consists of three different sensors at three different layers. In the trailing position of the architecture contains low power Sensor Nodes (SN), which gathers the information about various sensors at this lower layer level. The middle layer contains the Forwarding Node (FN), assume that who is trustful and won’t be compromised. The FN is responsible for collect information from the lower layer and compute aggregation result and commit the information to Access point (AP).The FN is also responsible for verifying correctness of the information gathered from SN. The Access point or Base station is placed at the leading position of the architecture and assume who is also trustful, who is responsible to transfer the output to the outside world. This scheme is based on the assumption FN and Base station, both are trusted. In fact the adversary can gain control over the BS then it leads to create any possible attacks in the network. Another critical assumption is that most of the sensor nodes are work in proper condition .If number of compromised nodes are more than number of normal nodes then there may be a chance to choose normal node as compromised node and it will create number of false positive. Through simulation result the author verified that correctness and effectiveness of the compromised node detection scheme.

6.2.2. STL Approach

Generally WSN consists of hundreds or thousands of sensor nodes and to create effective topology as well as to protect all nodes from accessible attacks are impractical. To overcome this situation the author introduced Stop Transmission and Listen approach, which is the one of the simple and effective technique for detecting a malicious node. In this number of sensor nodes are deployed in an environment and each sensor nodes having a built in time limit to stop their transmission. Each node starts their sensing process with in their sensing region and each node has the capability to detect the malicious node. After sensing the sensed data is forwarded to sink node and each node has stop their transmission in every few seconds and listen malicious behavior.Simulation result shows the effectiveness of the approach.

6.2.3. Auto regression technique

In this paper, the author considered the following assumption for detecting maliciousness of the different sensor nodes in the same network. The sensor network is static as well as each sensor node passed a onetime authentication procedure. Every sensor node has the capability to store up to hundreds of bytes of keying material in order to secure the transfer of information through symmetric cryptography. Base station will not be compromised at any cost. Due to this assumption the networks avoid various attacks such as eavesdropping, traffic analysis, spoofing, sinkhole, selective forward attack, wormhole attack, Sybil attack and Hello flood attack. The biggest threat for wireless sensor network is the node capturing attack, where an adversary gains full control over sensor nodes through direct physical access. To avoid these kind of attack the author introduced Auto Regression model (AR model).In this the time series of measured data provided by each sensor node and relies on an autoregressive predictor placed in base station. The basic principle followed is: For each sensor nodes collect past and present values and it will be compared with the threshold and detect whether that sensor node behave normally or abnormally.

6.2.4. Dual Threshold

In this work the author considered the following assumptions: Then numbers of sensors are deployed in the monitored area and having the transmission range rc. Each node knows its neighbors and their transmission range. If two nodes are neighbors of each other if their distance is less than or equal to rc. The trust values of the neighbor is calculated based on Weighted directed graph and its lies between 0 and 1.

6.2.5. SWATT

Software based authentication for Embedded Devices Our environment is surrounded by number of embedded devices ranging from java enabled cell phones to sensor networks and smart appliances. If an adversary can compromised one of our devices and modifying the memory contents. To avoid this kind of maliciousness the author introduced Software based Authentication (SWATT) to verify the memory contents of the embedded devices. SWATT can be applied in varies field such as network printers, smart cell phones, Electronic voting machines, smart cards etc.

7. Performance Analysis

In this section, we will evaluate our proposed authentication scheme through both theoretical analysis and simulation demonstrations. We will compare our proposed scheme with the bivariate polynomial-based symmetric-key scheme described in [2]. A fair comparison of our proposed scheme and the scheme proposed in [2] should be performed with n = 1.

A. Theoretical Analysis

The secret overate polynomial is defined as [1]: f(x, y) = ∑ dx i=0 ∑ dy j=0 Ai,jx i y j , where each coefficient Ax,y is an element of a finite field Fp, and dx and dy are the degrees of this polynomial. dx and dy are also related to the message length and the computational complexity of this scheme. From the performance aspect, dx and dy should be as short as possible. On the other hand, it is easy to see that the intruders can recover the polynomial f(x, y) via Lagrange interpolation when either more than dy + 1 messages transmitted from the base station are received and recorded by the intruders, or more than dx + 1 sensor nodes have been compromised, In this case, the security of the system is totally broken and cannot be used anymore. This property requires both dx and dy to be very large for the scheme to be resilient to node compromising attack. An alternative approach based on perturbation of the polynomial was also explored. The main idea is to add a small amount of random noise to the polynomial in the original scheme so that the adversaries will no longer be able to solve the coefficients using Lagrange interpolation.

While hop-by-hop authentication can be achieved through a public-key encryption system, the public-key-based schemes were generally considered as not preferred, mainly due to their high computational overhead. However, our research demonstrates that this is not always true, especially for elliptic curve public-key cryptosystems. In our scheme, each SAMA contains an AS of n randomly selected nodes that dynamically changes for each message. For n = 1, our scheme can provide at least the same security as the bivariate polynomial-based scheme. For n > 1, we can provide extra source privacy benefits. Even if one message is corrupted, other messages in the network can still be secure. Therefore, n can be much smaller than the parameters dx and dy. In fact, even a small n may provide adequate source privacy while ensuring high system performance.

B. Experimental Result

In this section, we compare the bivariate polynomial-based scheme and our scheme basedon comparable security levels.

Simulation parameter setup: The bivariate polynomial based scheme is a symmetric-key-based implementation, while our scheme is based on ECC. This requires us to determine the comparable key sizes. If we choose the key size to be l for the symmetric-key cryptosystem, then the key size for our proposed ECC will be 2l according to [5], which is much shorter than the traditional public-key cryptosystem. This progress facilitates the implementation of the authentication scheme using ECC. In our simulation setting, we choose five security levels, which are indicated by the symmetric-key sizes l: 24bit, 32bit, 40bit, 64bit, and 80bit, respectively. The comparable key sizes of our scheme are 48bit, 64bit, 80bit, 128bit, and 160bit, respectively. We also need to determine dx and dy for the bivariate polynomial-based scheme, and the n for our scheme. In our simulation, we select dx equal dy and choose three values for them: 80, 100, and 150. We assume that WSNs do not contain more than 2 16 nodes in our simulation, which is reasonably large. For size n of the AS, we choose three values in the simulation: 10, 15, and 20.

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

3) Communication overhead and message transmission delay: The communication overhead is determined by the message length. For the bivariate polynomial-based scheme, each message is transmitted in the form of < m, MAFm(y) >, where ∑ MAFm(y) is defined as: MAFm(y) = f(h(m), y) = dy j=0 Mjy j . MAFm(y) is represented by its dy + 1 coefficients, Mi ,∈Zp, 0 ≤ i ≤ dy, where p ∈ (2l−1 , 2 l) is a large prime number. The total length of the message is l(dy + 1). For our scheme, assuming that the network is composed of λ nodes in total, each ID will be of the length: ⌈log2 λ⌉. When n nodes are included in the AS, the length of S is n⌈log2 λ⌉. Therefore, the total length of one message for our scheme is: 4l(n + 1) + n⌈log2 λ⌉.

4) Simulation results: The simulation results, carried out in ns-2 on a RedHat Linux system, demonstrate that our proposed scheme has a much lower energy consumption and message transmission delay; see Fig. 1(a)&(b). The security levels 1, 2, 3, 4 correspond to symmetric key sizes 24bit, 32bit, 40bit, 64bit, and elliptic curves key sizes 48bit, 64bit, 80bit, 128bit, respectively. Our simulations also show that the delivery ratio of our scheme is slightly better than the bivariate polynomial-based scheme. Our simulation on memory consumption derived in TelosB.

8. Conclusion

Wireless Sensor Networks are one of the emerging fields in research area. Wireless sensor networks has a remarkable feature to monitor environmental and physical phenomenon such are temperature, pressure, humidity etc.. In this paper we discussed various aspects of wireless sensor networks and also discussed various types of WSNs and their applications and classify various categories of routing protocols. The routing protocols in WSN has become one of the most important research areas and introduced unique challenges compared to traditional data routing in wired networks. The main aim behind the routing protocol design is to keep the sensors operating for a long time, thus extending the network life time. Although many routing protocols have been proposed for sensor networks, many issues still remain to be addressed.

References

[1] Guohua Ou1,Jie Huang, and Juan Li, (2010), "A Key-Chain Based Key Management Scheme for Heterogeneous Sensor Network", pp. 358-361.

[2] W.Zhang, N. Subramanian, and G. Wang, “Lightweight and compromiseresilient message authentication in sensor networks,” in IEEE INFOCOM, (Phoenix, AZ.), April 15-17 2008.

[3] Guohua Ou1, Jie Huang, and Juan Li, (2010), "A Key-Chain Based Key Management Scheme for Heterogeneous Sensor Network", pp. 358-361.4.

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

[5] “Cryptographic Key Length Recommendation,” http://www.keylength.com/en/3/, 2013.

Author Profile

Vaishali Golhane received the B.E. degrees in Computer Engineering in 2013.