

Energy Efficient Session Key Establishment in Wireless Sensor Networks

Yi Cheng and Dharma P. Agrawal

OBR Center for Distributed and Mobile Computing, Department of ECECS
University of Cincinnati, Cincinnati, OH 45221
{chengyg, dpa}@ececs.uc.edu

Abstract—Key distribution and management is the foundation of any secure services of wireless sensor networks. Due to the resource constraints, establishing cryptographic keys between communicating sensors in large-scale wireless sensor networks is a big challenge. Traditional asymmetric key cryptography is not suitable for wireless sensor environment because of its complexity. Based on the polynomial-key pre-distribution scheme, an improved session key establishment method is proposed in this paper. Compared with existing key distribution schemes, our proposed scheme has lower energy consumption, less key storage and lower communication overheads as well as better network resilience against node capture attacks.

Keywords: Communication security; Wireless sensor networks; Key establishment.

1. Introduction

With the recent technology improvements in tiny sensor devices, wireless sensor networks (WSNs) are expected to be applicable in various applications in the near future. A wireless sensor network is usually composed of a large number of small, low-cost wireless devices called sensor nodes. Each sensor has limited data processing capacity, memory storage and short radio transmission range. Depending on the applications, sensor nodes can be equipped with a variety of sensors to measure different physical characteristics such as sound, temperature, pressure, etc. [1][5][16]. Wireless sensors can be organized in clusters to tracking a particular object or monitoring an area [2][18].

In most cases, WSNs are deployed in a hostile environment and working on unattended mode, therefore, the critical data readings should be protected properly [4][10][13][16][17]. As we known, in wireless environment an adversary not only can eavesdrop the communication traffic in a network, but also can impersonate good nodes to spread misleading information intentionally. Furthermore, in many applications such as battlefield surveillance, wireless sensors may be physically destroyed or captured by the adversary. To ensure a sensor network work properly in such situations,

security mechanisms must be adopted to achieve the information security requirements.

In traditional wired networks or infrastructure-support wireless networks, information security is provided by encrypting the communication data and mutual authentication between communicating parties. Public-key cryptosystems and trusted third-party authentication protocols are used frequently to achieve the network security requirements. Since sensor nodes only have limited power, memory storage and data processing capacity, public-key based security protocols are not suitable for large-scale sensor networks because of their complicated hardware requirement and high energy consumption. Lack of infrastructure support is another limitation of WSNs, due to the unpredictable network topology, trusted infrastructure based protocols also can not be used for WSNs directly.

To achieve both security and efficiency, many key distribution and management mechanisms have been investigated [3][4][7][8][13][15][17][19][20]. In which, key pre-distribution protocols are considered to be a practical mechanism to solve the key distribution and management problems in wireless sensor environments [3]. The basic idea of key pre-distribution scheme is quite simple, just pre-loading a set of symmetric keys into sensor nodes before they are deployed. After deployment, each sensor exchanges its stored key information with its neighbors. If two neighboring nodes have a common key, they can use it to encrypt the communication between them.

Based on the distributed network architecture, several key pre-distribution schemes have been proposed in literature [3][4][13][19][20], tried to achieve security and scalability for the large-scale WSNs. We argue that for a large-scale pure distributed network, it is extremely hard for a key pre-distribution scheme to achieve both security and scalability requirements when the sensor nodes are resource-constrained. [5][6] showed that the hierarchical network architecture has better performance and scalability than flat structure for large-scale WSNs, since most data are destined to the sink node (base station), and which can be reached in a few hops in the hierarchical approach. Research shows that when using the same number of sensors in a given coverage area, the

hierarchical topology can significantly increase the network throughput and decrease the system delay [5][6].

To achieve security and network performance simultaneously, we proposed a new session key establishment mechanism for large-scale WSNs in this paper. Based on the hierarchical network architecture, an improved polynomial-key distribution and management mechanism is investigated to establish energy efficient session keys between communicating parties. Compared with existing schemes, our approach has better network performance, as well as the security.

The rest of this paper is organized as follows. Section 2 introduces some related work. We present our proposed scheme in detail in Section 3. Section 4 gives the security analysis and the performance evaluation of our scheme. Finally, we conclude our work in Section 5.

2. Related Work

Due to the resource constraints, the extremely large network size and the lack of the infrastructure support, traditional public-key based asymmetrical key distribution protocols and trusted infrastructure authentication security mechanisms are not suitable for WSNs. Pre-distributing secret keys into sensor nodes before they are deployed is an applicable solution for key management in wireless sensor environment [3]. Several key pre-distribution schemes have been proposed in the literature [3][4][13][19][20].

Eschenauer et al. [3] proposed a random key pre-distribution scheme in 2002. In their scheme, a subset of keys from a large size key pool is randomly selected and pre-loaded into sensors. If two nodes can find a common key, they can setup a secure link by it. Otherwise, they need to establish a path-key between them. According to the random graph theory, if the probability that any two nodes have at least one common key satisfies a critical value, the network is connected with high probability.

Based on [3], Chan et al. proposed a “ q -composite” scheme to improve the network resilience against node capture attacks [4]. Network resilience here is defined as how many secure links between non-captured nodes could be broken when a number of sensors are compromised. [4] requires two sensors share at least q ($q \geq 2$) common keys to establish a secure link. Chan et al. showed that when the number of the compromised nodes is less than a critical value, the network resilience can be improved.

Both [3] and [4] can not guarantee the full network connectivity. To achieve high network connectivity, an additional path-key establishment procedure is required, which not only degrades the security, but also produces additional communication overheads. In [3][4] a particular key may be reused for different pairs of nodes, some node’s capture could compromise the communication between non-captured nodes, which is the node capture attack problem in WSNs.

Cheng et al. proposed an efficient pairwise key establishment and management scheme (EPKEM) in [8]. In this scheme, a two-dimensional key matrix is

constructed to pre-distribute symmetric keys into sensors. Each sensor stores a row and a column from the key matrix. EPKEM guarantees every two nodes share at least two common keys after the deployment. Combined with the identities of the communicating parties, EPKEM can establish a distinct pairwise key for each pair of sensors. Although Cheng et al.’s scheme can provide better network resilience than previous schemes; it still has some limitations when used for large-scale WSNs. The communication overhead is still high, sensors need to store too many keys in the network initialization phase.

All above work are based on the pure distributed network architecture. [5][6] show that the hierarchical network architecture has the better network performance than the flat network, more investigations have been focused on this network architecture recently.

Jolly et al. proposed a low-energy key management protocol for hierarchical WSNs in [9]. In this work, some gateway nodes partition a sensor network into several distinct clusters. Each cluster has a cluster head and a set of sensors, sensor node only communicates with its cluster head. Each cluster head is pre-loaded a set of symmetric keys in its memory, and each of these symmetric keys is also pre-loaded into a particular sensor node. After deployment, each sensor needs to exchange its key information with its cluster head. If they share a common key directly, they can use it as their pairwise key. Otherwise, the cluster head needs to require the intended keys from other cluster heads.

Although [9] provides better network performance than previous key pre-distribution schemes, it is not secure for node capture attack. Each cluster head has lots of keys stored. Any cluster head’s capture could compromise a lot of keys in a network. [9] uses a group key to encrypt the communication among cluster heads, which is extremely dangerous for wireless sensor environment. Once the group key is compromised, the adversary can track all the communications among the cluster heads. In this case, the whole network would be broken.

To address the above problems in [9], we proposed an energy efficient session key establishment mechanism (EESK) for large-scale hierarchical WSNs. Based on the same network model in [9] and the polynomial-key establishment procedure, our scheme can improve the network security against the node capture attack.

3. Energy Efficient Session Key Establishment (EESK)

3.1. Network Model

In this paper, we use the same network model in [10][10]. Illustrated by Fig.1, there are three kinds of wireless devices are existed in our model: sink node/base station (BS), cluster head nodes (CH), and wireless sensor nodes (S).

Sensor nodes (S): Sensor nodes are resource constraints, each sensor only has limited power, memory

storage and short radio transmission range. Sensor nodes only communicate with their cluster head directly; no communication between sensor nodes exists. Sensor nodes are static after the deployment.

Cluster head nodes (CH): Compared with sensor nodes, cluster heads have considerably high energy resources. They are equipped with high power CPUs, large memory storages and radio transmission range. Cluster heads can communicate with each other directly, and relay information between sensor nodes and sink node.

Sink node/Base station (BS): Sink node is the most powerful node in a network. It has virtually unlimited computational and communication power, unlimited memory storage, and very large radio transmission range to reach all the sensors in the network.

In our network model, a large number of sensors are randomly distributed in an area. Sink node is deployed in a well-protected place. As shown in Fig.1, CHs partition the sensors into distinct clusters by some clustering algorithm, e.g., [12]. Each cluster has a cluster head and a set of sensors; cluster head aggregates the data from sensors, performs mission-related data processing, and sends the processed data to the sink node.

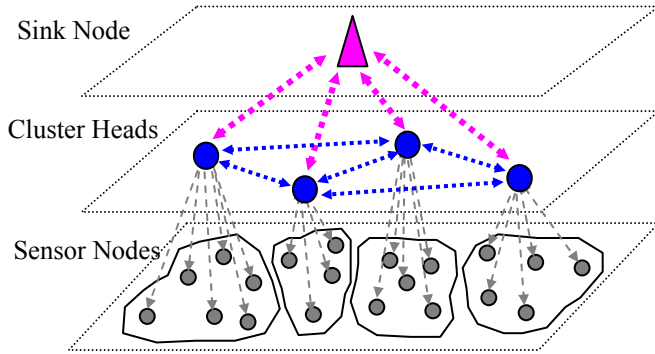


Fig.1. A hierarchical wireless sensor network architecture

3.2. Background of Polynomial Key Pre-distribution Scheme

To improve the network security against the node capture attack, a bivariate polynomial-key pre-distribution mechanism [13] is adopted in our scheme. Consider a bivariate polynomial $f(x, y)$ of degree t , defined as

$$f(x, y) = \sum_{i,j=0}^t a_{ij} x^i y^j \quad (1)$$

where the coefficients a_{ij} ($0 \leq i, j \leq t$) are randomly chosen from a finite field $GF(q)$.

The bivariate polynomial has a symmetric property such that

$$f(x, y) = f(y, x) \quad (2)$$

In the initialization phase, an offline Key Distribution Server (*KDS*) first initializes a set of sensors by giving each node p the polynomial $g_p(y) = f(p, y)$, which is the polynomial obtained by evaluating $f(x, y)$ at $x = p$. After deployment, node p knows that

$$g_j = \sum_{i=0}^t a_{ij} (p)^i \quad (0 \leq j \leq t) \quad (3)$$

where p is id of the node, g_j is coefficient of y^j in the polynomial $f(p, y)$.

In order to establish a session key between nodes p and q , p evaluates $f(p, y)$ at $y = q$, and q evaluates $f(q, y)$ at $y = p$. Since $f(p, q) = f(q, p)$, nodes p and q can obtain the same value from the two distinct evaluations. This value is only known by p and q , and can be worked as their pairwise communication key.

3.3 Our Proposed Approach

Our proposed key distribution scheme is a symmetric-key mechanism, which includes two phases: key pre-distribution phase and network initialization phase. The notations used in this paper are listed in Table 1.

Table 1: Notations used in our scheme

Notation	Description
BS	Sink node (Base station)
CH_i	Cluster head node i
S_i	Sensor node i
CH	Set of cluster heads in the network
S	Set of sensor nodes in the network
K_{A-B}	Symmetric key between A and B (A, B can be sink, cluster head or sensor node)
$E_K(data)$	Encrypted message by key K
$f_{CH}(x, y)$	t -degree bivariate symmetric polynomial (used for key calculation between cluster heads)
$f_{CH_i}(x, y)$ ($1 \leq i \leq m$)	t -degree bivariate symmetric polynomial (used for key calculation between cluster head i and sensors)

A) Key Pre-distribution Phase

Due to the resource constraints of wireless sensors, the best key distribution method is pre-loading symmetric keys into sensor nodes before they are deployed [3][4][13][19][20]. There are three kinds of wireless nodes in our network model, each kind of nodes have different keys stored inside, we describe them separately in follows. For convenience, we assume there are n sensor nodes and m cluster heads in the network, and each cluster has (n/m) members in its cluster.

Sink node: In our model, sink node needs to store $(n+m)$ keys inside, each key is shared with a particular sensor node or cluster head node.

Cluster heads: Each cluster head CH_i stores a symmetric key K_{CH_i-BS} and two polynomials $g_{CH}(y)$, $g_{CH_i}(y)$. $g_{CH}(y)$ is obtained by evaluating $f_{CH}(x, y)$ at $x = CH_i$, $g_{CH_i}(y)$ is the obtained by evaluating $f_{CH_i}(x, y)$ at $x = CH_i$.

Sensor nodes: Each sensor node S_i is pre-loaded two keys in its memory, one is K_{S_i-BS} , the other is K_{S_i-CH} . These two keys are generated in different ways to provide the required security. In our scheme, K_{S_i-BS} is randomly generated by *KDS*, which is the session key between S_i and the sink. K_{S_i-CH} is generated complicatedly to achieve a high level security. The procedure is illustrated as follows.

- 1) *KDS* randomly selects l ($l \geq 1$) polynomials from the m polynomials $f_{CH_i}(x, y)$, ($1 \leq CH_i \leq m$). Suppose $l = 2$ and polynomials $f_{CH_a}(x, y)$ and $f_{CH_b}(x, y)$ are selected by the *KDS*.
- 2) *KDS* evaluates $f_{CH_a}(x, y)$ at $(x = CH_a, y = S_i)$, $f_{CH_b}(x, y)$ at $(x = CH_b, y = S_i)$, respectively. Suppose $k_1 = f_{CH_a}(CH_a, S_i)$, $k_2 = f_{CH_b}(CH_b, S_i)$ after the evaluations.
- 3) *KDS* calculates the corresponding secret key K_{S_i-CH} by exclusive-or k_1 and k_2 under Equation (4):

$$K_{S_i-CH} = k_1 \oplus k_2 \quad (4)$$

- 4) K_{S_i-CH} will be pre-loaded into sensor S_i 's memory, and works as the session key between S_i and its intended cluster head. S_i also stores the two cluster heads' id CH_a and CH_b in its memory.

After key pre-distribution phase, each node in the network stores different keys in its memory. Those pre-loaded keys will be used to setup the eventual network after the deployment.

B) Network Initialization Phase

After the deployment, cluster heads partition the entire network into several distinct clusters and a network initialization phase is triggered. In network initialization phase, each cluster head needs to exchange the handshaking messages with its cluster members and setup a secure link with each of its members.

We still use the previous instance to illustrate this procedure. Suppose sensor S_i is a cluster member of cluster head CH_j after the deployment.

First, sensor S_i sends its id and stored cluster heads' ids CH_a and CH_b to its physical cluster head CH_j . CH_j then sends S_i to CH_a and CH_b to require the corresponding key shares. Once received the key share requirement message, CH_a and CH_b evaluate its stored polynomials with the corresponding sensor's id S_i and reply the calculated values k_1 and k_2 to CH_j . Once CH_j received k_1 and k_2 , it can calculate the intended session key K_{S_i-CH} by Equation (4). The procedure is illustrated as follows:

- 1) First, sensor S_i sends its id and CH_a and CH_b to its cluster head CH_j
- 2) CH_j sends S_i to CH_a and CH_b respectively
- 3) CH_a evaluates $g_{CH_a}(y) = f_{CH_a}(CH_a, y)$ at $(y = S_i)$, and $g_{CH}(y) = f_{CH}(CH_a, y)$ at $(y = CH_j)$
- 4) Suppose $k_1 = g_{CH_a}(S_i)$, $K_{CH_a-CH_j} = g_{CH}(CH_j)$. After calculation, CH_a sends back $E_{K_{CH_a-CH_j}}(k_1)$ to CH_j
- 5) CH_j decrypts $E_{K_{CH_a-CH_j}}(k_1)$ by $K_{CH_j-CH_a}$ to get k_1 , where $K_{CH_j-CH_a} = g_{CH}(CH_a) = f_{CH}(CH_j, CH_a)$
- 6) Similarly, CH_j can get k_2 from CH_b in the same way
- 7) CH_j uses k_1 and k_2 to calculate $K_{CH_j-S_i}$ by Equation (4)

Now, CH_j can communicate with its cluster member S_i securely by the session key $K_{CH_j-S_i}$. Once all the cluster heads calculated their session keys with its cluster members, the network initialization phase is finished.

After the network initialization phase, a secure hierarchical wireless sensor network has been established. Each sensor has session keys with its cluster head and sink node. Each cluster head has session keys with its cluster members, other cluster heads and the sink node. These session keys not only can be used to encrypt the exchanged information, but also can be used to mutually authenticate the intended communicating parties. In this case, malicious nodes can not join in the network by impersonating a good node. All the communications in the network is encrypted by a certain session key shared between the communicating parties, therefore, the information authenticity, confidentiality and integrity can be guaranteed in our proposed scheme.

4. Security Analysis and Performance Evaluation

In this section, we compare the performance of our scheme with the random key based pre-distribution schemes in [3][4], efficient pairwise key establishment and management scheme (EPKEM) in [8] and low-energy key management protocol (LEKM) in [9].

4.1 Security Analysis

Node capture attack is a serious threat in WSNs, an adversary may physically capture sensor nodes to obtain the secret information.

In [3][4], a same pairwise key may be used among different pairs of sensors, hence some sensors' capture could compromise the communication between other non-captured nodes. In LEKM [9] and our proposed EESK, no communication between sensor nodes exists, each sensor only needs to store two session key in its memory, which extremely reduce the key storage overhead for sensor nodes.

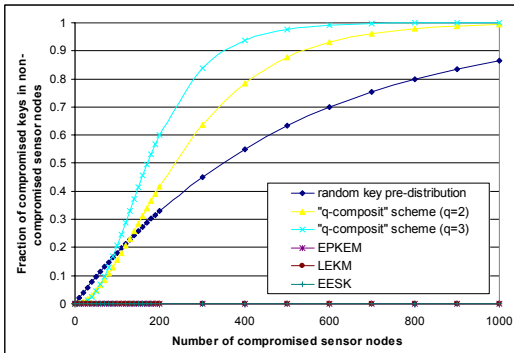


Fig.2. Fraction of compromised keys in non-captured sensor nodes vs. number of compromised sensor nodes

After the network initialization phase, any sensor's compromise does not affect the secure communication between non-compromised nodes. Fig.2 compares the resilience against sensor node capture attack for different schemes. We can see that EPKEM, LEKM and EESK can prevent the key compromise for non-captured sensor nodes no matter how many sensors are captured in a network.

In LEKM [9], all secret keys are pre-loaded in cluster heads (CHs) on the network initialization phase, and each CH stores (n/m) keys in its memory. Once a CH is captured in this phase, all its stored keys could be compromised. Furthermore, a group key is used in LEKM to secure the communication among CHs, which also could lead to the single-point failure attack in WSNs. Any single CH's capture could compromise all the communication between non-compromised CHs. If this case happened in the initialization phase, a malicious node can track all the exchanged key information between CHs, and break the entire network lately.

In EESK only polynomial shares are pre-loaded in CHs. Any two CHs need to setup a unique session key between them before they exchange the sensitive information. There is no group key exists in EESK, any communication between CHs need to be encrypted by the intended session key. Therefore, any CH's compromise does not affect the communication between non-compromised CHs. According to security property of t -degree bivariate polynomial, EESK can guarantee the network's security when there is no more than t CHs are compromised. Furthermore, in our network model, the CHs have considerably high energy and memory storage. By setting $t > m$ (where m is the number of cluster head nodes in a network), we can guarantee that even all the CHs are compromised, the coefficients of the polynomial are still keep secret to the adversary.

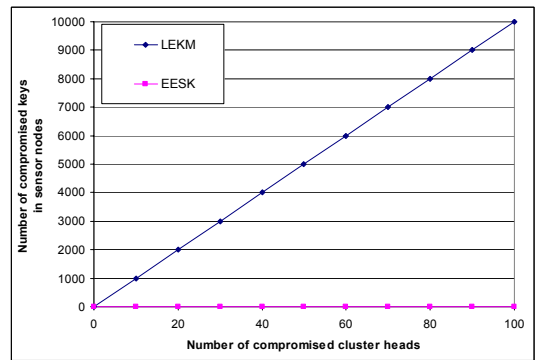


Fig.3. Number of compromised sensor keys vs. number of the compromised cluster heads in the network initialization phase

Fig.3 shows the resilience against cluster head node capture attack in the network initialization phase. Suppose there are 10,000 sensor nodes and 100 CHs in a network. In LEKM, each CH need store 100 sensor's symmetric key in its memory, hence each CH's compromise would compromise 100 sensor nodes' secret keys. The number of compromised key is linearly increasing with the number of compromised CH increases. In our EESK scheme, a 128-degree bivariate polynomial is used to calculate the session key between cluster heads and sensors. Since there is no symmetric-key used in EESK, and the degree of the polynomial is larger than the number of cluster heads, even all the 100 cluster heads are compromised, none of the keys of sensor nodes could be compromised in the network initialization phase.

4.2 Performance Evaluation

A. Maximum Supported Network Size

Wireless sensor networks are usually composed of a large number of sensors, therefore, proposed key distribution scheme should be scalable with the sensor nodes increases. [3][4][8] are based on the pure distributed network architecture, when the network size linearly increases, the number of keys stored in each sensor node also linearly increases. Due the limited memory storage in tiny sensors, the maximum supported

network size is limited for this kind of schemes. LEKM and our proposed EESK are based on the hierarchical network model, which has a better scalability than flat network, since each cluster head can take charge of a number of sensor nodes in its cluster. Theoretically, our proposed scheme can be suitable for any size of wireless sensor networks by the properly selected degree of polynomial shares.

B. Key Storage Overhead

In [3][4], to achieve the required network connectivity, each sensor needs to store a certain number of keys in its memory before it is deployed. Although [8] has a smaller key ring size than [3][4] for the same network size, its key storage overhead is still sub-linearly with the network size. In our proposed EESK scheme, each sensor only needs to store two keys in its memory no matter how many nodes in the network, which is extremely memory efficient for the large-scale wireless sensor networks.

C. Communication Overhead

In wireless sensor networks, radio communications consume much more energies than the code execution or calculations. To save the energy consumption, proposed security schemes should have low communication overhead. Compared with existing schemes, our proposed scheme is more energy efficient due to the lower communication overhead for sensor nodes.

In WSNs, the communication overhead mainly occurs in the network initialization phase, since each sensor needs to exchange key information with its neighbors. Unlike the previous schemes, each sensor only stores two keys in EESK, which means the handshaking message in EESK is much shorter than previous schemes. Hence the communication overhead of EESK is significantly lower than previous key pre-distribution schemes.

Additionally, in many applications, fresh sensors may need to be added into an existing network to replace the power exhausted nodes, which is another main energy-consuming procedure in WSNs. In [3][4], a fresh node needs to exchange its stored key information with the existing nodes when it is deployed in the network. This procedure is similar as the network initialization phase, and lots of communication overheads are involved. In [9], a complicated energy-consuming procedure also needs to add new sensors into an existing network. Sink node needs to re-assign symmetric keys to the new sensors and re-distribute the new keys to a particular cluster head. This procedure produces additional communication overhead in the network, especially for a large-scale network.

EESK is based on the polynomial share calculation, no additional key re-assignment and re-distribution operations are needed when new sensors join into an existing network. By pre-loading two keys into the new sensor under the same procedure in the key pre-distribution phase, a fresh node can be deployed in

anywhere to join a particular cluster. Sink node does not need to re-exchange key information with cluster heads, which extremely reduces the communication overhead in the network.

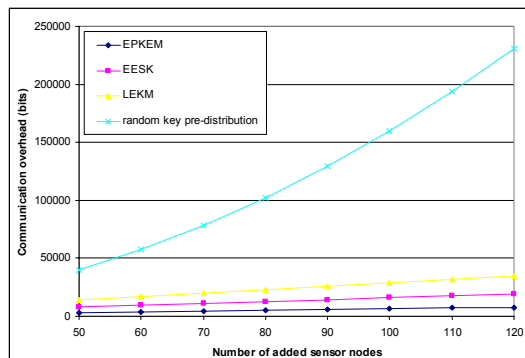


Fig.4. Communication overhead vs. Sensor node addition

To compare the communication overhead for different schemes when fresh sensors are added into a network, we assume all the ids are 16 bits, symmetric-keys and polynomial-shares are 128 bits. To compare with existing schemes, we assume there are 10,000 sensors and 100 cluster heads in our evaluation model, each cluster has 100 members inside, the average degree of sensor node is 60.

Fig.4 shows that that EPKEM has the lowest communication overhead since the new nodes only need to exchange key information with their one-hop neighbors. Random key pre-distribution schemes have the highest overhead, since the new nodes have to exchange key information with all the neighbors to establish a secure link. LEKM and EESK have lower communication overhead than random key pre-distribution schemes since the new nodes only need to exchange key information with their cluster heads. EESK can reduce 25% communication overhead than LEKM since there is no key re-broadcast procedure involved.

5. Conclusion

In this paper, we present an energy efficient session key establishment scheme for large-scale wireless sensor networks. Based on the hierarchical network structure and polynomial-key pre-distribution mechanism, our scheme is more suitable for large-scale wireless sensor networks with its better scalability and network performance. Compared with the existing key pre-distribution schemes, our scheme can achieve better network resilience against node capture attack, as well as better network performance in term of network size, key storage overhead and communication overhead.

Acknowledgement

This work has been supported by the Ohio Board of Regents Doctoral Enhancement Funds.

References

- [1] D. P. Agrawal and Q-A Zeng, *Introduction to Wireless and Mobile Systems*, Brooks/Cole Publishing, Aug. 2003.
- [2] Neha Jain and D. P. Agrawal, "Current trends in wireless sensor network design," *International Journal of Distributed Sensor Networks*, Vol.1, No.1, pp.101-122, 2005.
- [3] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," In *Proceedings of the 9th ACM conference on Computer and communications security*, November 2002.
- [4] H. Chan, A. Perrig, and D. Song, "Random key pre-distribution schemes for sensor networks," In *IEEE Symposium on Security and Privacy*, pages 197–213, Berkeley, California, May 11-14 2003.
- [5] P. Gupta and P. Kumar, "Internets in the sky: the capacity of three dimensional wireless networks," *Communications in Information Systems*, vol.1, no.1, pp. 33–50, 2001.
- [6] S. Zhao, K. Tepe, I. Seskar, and D. Raychaudhuri, "Routing protocols for self-organizing hierarchical ad-hoc wireless networks," in *IEEE Sarnoff 2003 Symposium*.
- [7] R. Blom, "An optimal class of symmetric key generation systems," *Advances in Cryptology: Proceedings of EUROCRYPT 84* (Thomas Beth, Norbert Cot, and Ingemar Ingemarsson, eds.), Lecture Notes in Computer Science, Springer-Verlag, 209:335–338, 1985.
- [8] Y. Cheng and D. P. Agrawal, "Efficient Pairwise Key Establishment and Management in Static Wireless Sensor Networks," in *Proceedings of the 2nd IEEE International Conference on Mobile Ad-Hoc and Sensor Systems*, Washington, DC, Nov. 7-10, 2005.
- [9] G. Jolly, M. C. Kuscü, P. Kokate, M. Younis, "A low-energy management protocol for wireless sensor networks," In *Proceeding of the Eighth IEEE International Symposium on Computers and Communication (ISCC'03)*, KEMER - ANTALYA, TURKEY. June 30 - July 3 2003.
- [10] M. Younis, M. Youssef, and K. Arisha, "Energy-Aware Routing in Cluster-Based Sensor Networks," in *Proceedings of the 10th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS2002)*, (Forth Worth, TX), October 2002.
- [11] K. Arisha, M. Youssef, and M. Younis, "Energy-Aware TDMA-Based MAC for Sensor Networks," in *Proceedings of the IEEE Workshop on Integrated Management of Power Aware Communications, Computing and Networking (IMPACCT 2002)*, May 2002.
- [12] G. Gupta, M. Younis, "Performance Evaluation of Load-Balanced Clustering of Wireless Sensor Networks," in *Proceedings of the 10th International Conference on Telecommunications (ICT'2003)*, Tahiti, Papeete – French Polynesia, February 2003.
- [13] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-secure key distribution for dynamic conferences," *Lecture Notes in Computer Science*, 740:471–486, 1993.
- [14] David W. Carman, Peter S. Kruus, and Brian J. Matt, "Constraints and approaches for distributed sensor network security". *NAI Labs Technical Report #00-010*, September 2000.
- [15] Sencun Zhu, Sanjeev Setia, and Sushil Jajodia, "LEAP: Efficient security mechanisms for large-scale distributed sensor networks," In *ACM CCS 2003*, pages 62–72, October 2003.
- [16] J. M. Kahn, R. H. Katz, and K. S. J. Pister, "Next century challenges: Mobile networking for smart dust," In *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 483–492, 1999.
- [17] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," In *First IEEE Int'l Workshop on Sensor Network Protocols and Applications*, May 2003.
- [18] Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communication Magazine*, vol. 40, no. 8, pp. 102–116, Aug. 2002.
- [19] Donggang Liu and Peng Ning "Location-Based Pairwise Key Establishments for Relatively Static Sensor Networks," *2003 ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN'03)*, October 31, 2003 George W. Johnson Center at George Mason University, Fairfax, VA, USA.
- [20] W. Du, J. Deng, Y. S. Han, and P. K. Varshney, "A pairwise key pre-distribution scheme for wireless sensor networks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS)*, Washington DC, USA, October 27-31 2003, pp. 42–51.