An Efficient ID-based Bilinear Key Predistribution Scheme for Distributed Sensor Networks*

Tran Thanh Dai, Cao Trong Hieu and Choong Seon Hong**

Networking Lab, Department of Computer Engineering, Kyung Hee University
Giheung, Yongin, Gyeonggi, 449-701 Korea
daitt@networking.khu.ac.kr, hieuct@networking.khu.ac.kr and cshong@khu.ac.kr

Abstract. Security requirements are very pressing in distributed sensor networks due to exploitation purposes of these networks in human life, especially in military tasks. To obtain security in these sorts of networks, it is crucial to enable message encryption and authentication features among sensor nodes. This thing could be performed using keys agreed upon by communicating nodes. Nonetheless, acquiring such key agreement in distributed sensor networks becomes extremely intricate due to resource constraints. Up to now, there are many key agreement schemes proposed wired and wireless networks of which key predistribution schemes are considered to be the fittest solutions. Based on this observation, in this paper, we propose a key predistribution scheme relying on sensor nodes' unique identifiers. Our scheme exhibits several noteworthy properties: direct pairwise key establishment permission with explicit key authentication, high resiliency against information-theoretic security attack (node capture attack). We also present a detailed security and performance analysis of our scheme in terms of node capture attack, memory usage, communication overhead, and computational overhead.

1 Introduction

Advances in wireless communications and electronics over the last few years have sped up the development of networks of low-cost and multifunctional sensors. These sensors are tiny in size and able to sense, process data, and communicate with each other, typically over a radio frequency channel. They are usually deployed in a immense number and in the form of distributed networks to detect events or phenomena, collect and process data, and transmit sensed and processed information to interested users. Those distributed sensor networks are anticipated to be widely applied to many fields of human life ranging from civil applications to military applications.

In most of the applications, we truly need security measures to protect each sensor node in particular and the entire distributed sensor networks in general from malicious adversaries. According to typical approaches, security measures could be fulfilled based on efficient key agreement schemes. Nonetheless, sensor nodes typically

* This work was supported by MIC and ITRC projects
** Dr. C. S. Hong is the Corresponding Author

operate in unattended conditions; have limited computational capabilities and memory, and battery-power capacity. Due to these resource limitations, the materialization of the efficient key agreement schemes in distributed sensor networks becomes a deeply intricate task. In fact, there are many key agreement schemes proposed for wired and wireless network environments which have been proved to be efficient and secure like trusted server schemes, public key based schemes, and key predistribution schemes. Nevertheless, constrained computation and energy resources of sensor nodes often make the first two schemes infeasible or too expensive for distributed sensor networks [8], [9], [10]. Recently, there are some attempts to solve the key agreement problem for sensor networks using elliptic curve cryptography (ECC) [11], [12]. However, the energy consumption of ECC is still expensive, especially compared to symmetric key based algorithms. Based on these analyses, it is straightforward to realize that key predistribution schemes seem to be the most feasible solution for the key agreement problem in distributed sensor networks.

A key predistribution scheme is a method to distribute off-line initial private pieces of information (keying materials) among a set of users, such that each group of a given size (in our scheme it is equals to two for the pairwise key generation purpose) can compute a common key for secure communication [13]. One branch of the key predistribution schemes is the ID-based key predistribution scheme. In that scheme, no previous communication is required and its key predistribution procedure consists of simple computations. Furthermore, in order to establish the key, each party should only input its partner's identifier to its secret key sharing function [14].

Due to those sorts of noteworthy properties, in this paper, we propose a highly resilient, resource-efficient and ID-based key predistribution scheme. Main contributions of our scheme are as follows:

1. Direct pairwise key establishment permission with explicit key authentication.
2. Substantially improved network resiliency against information-theoretic security attack (node capture attack).
3. Detailed theoretical analysis of security, memory usage, and communication and computation overhead.

The rest of the paper is organized as follows: section 2 mentions the related work; section 3 gives an overview of our building block; section 4 presents our proposed scheme; section 5 deals with the detailed security analysis; section 6 discusses performance analysis; section 7 concludes the paper.

2 Related Work

Recently, symmetric key cryptography has been received extensive studies to obtain various aspects of security in sensor networks. Perrig et al. [15] developed a security architecture for sensor networks which is comprised of two link layer protocols: SNEP and μTELSA. SNEP (Secure Network Encryption Protocol) provides data confidentiality, two-party authentication, and data freshness. μTELSA, the second part of SPINS, provides authenticated broadcast for sensor networks. Liu and Ning [16] proposed a multi-level key chain method for the initial commitment distribution

in μTESLA. Karlof, Sastry and Wagner [17] developed TinySec, the first fully implemented link layer security architecture for sensor networks. Eschenauer and Gligor [18] proposed a probabilistic key predistribution scheme recently for pairwise key establishment. The main idea is to let each sensor node randomly pick a set of keys from a key pool before deployment so any two sensor nodes have a certain probability of sharing at least one common key. Chan et al. [8] further extended this idea and developed three mechanisms for key establishment using the framework of predistributing a random set of keys to each node. The first one is q-composite keys scheme. This scheme is mainly based on [18]. The difference between this scheme and [18] is that q common keys, instead of just a single one, are needed to establish secure communication between a pair of nodes. By increasing the amount of key overlap required for key setup, the resiliency of the network is increased against node capture. The second one is multipath key reinforcement scheme applied in conjunction with [18] to yield greatly improved resilience against node capture attacks by trading off some network communication overhead. The main attractive feature of this scheme is that it can strengthen the security of an established link key by establishing the link key through multiple paths. The third one is random pairwise keys scheme. The purpose of this scheme is to allow node-to-node authentication between communicating nodes. Du et al. [19] proposed a method to improve [18] by exploiting a priori deployment knowledge. Specifically, by using node deployment knowledge and a wise key ring setup, the sensor networks get much higher probability of establishing a secure link between any pairwise of nodes. Zhu et al. [22] proposed a protocol suite named LEAP to help establish individual keys between sensors and a base station, pairwise keys between sensors, cluster keys within a local area, and a group key shared by all nodes.

3 Overview of Matsumoto-Imai'sKey Predistribution Scheme

Matsumoto-Imai (MI) proposed a linear key predistribution scheme in [1] that allows distributing a common key to an arbitrary group of entities in a network without previous communications among the group nor accesses to any public key directory or whatsoever. In this section, we briefly describe how Matsumoto-Imai's key predistribution scheme works (MI scheme for short).

Let q be a prime power and m, l be positive integers. Let $\Psi = GF(q)$ and $\Psi^m = \left\{ x \mid x = \left[x_1\ x_2\ \ldots\ x_m \right], x_i \in \Psi, i = \overline{1,m} \right\}$.

Suppose that each entity's (say, *entity i*'s) identity y_i is a member of a set Υ and that $y_i \neq y_j, \forall i \neq j$.

And let Γ denote an one-way algorithm implementing an injection from Υ to Ψ^m.

The key setup server selects l (m, m) symmetric matrices $M^\tau s\ (\tau = \overline{1,l})$ over Ψ randomly and independently from other entities.

The key setup server generates the *secret key sharing functions* Φ_i s:

$$\Phi_i(\omega) = \phi_i \Gamma(\omega)^T, \omega \in \Upsilon$$

for each $y_i \in \Upsilon$. Here, $\Gamma(\omega)^T$ is the transpose of $\Gamma(\omega)$ and ϕ_i is an (*l*, *m*) matrix defined by

$$\phi_i^T = \left[M_1 \Gamma(y_i)^T, ..., M_l \Gamma(y_i)^T \right]$$

Each entity *I* receives its own Φ_i from the center.

If entity A and entity B want to establish a pairwise cryptographic key, entity A computes $\Phi_A(y_B)$ and entity B computes $\Phi_B(y_A)$ independently. They are *l*-vectors over Ψ. It is easy to realize that both vectors are the same. This scheme could be used for key sharing among *n* entities by using symmetric *n*-linear mappings instead of the aforementioned symmetric bilinear mappings.

4 ID-Based Bilinear Key Predistribution Scheme

Since the purpose of MI scheme is to apply to the smart-card-based systems, not for distributed sensor networks, so we propose an ID-based bilinear key predistribution scheme inspired by MI scheme. We will later show that our scheme exhibit the fascinating properties satisfying security requirements due to specific characteristics of distributed sensor networks which have not been mentioned in MI scheme. Accordingly, our scheme consists of three phases, namely keying material predistribution, pairwise key establishment, and pairwise key reinforcement. The following are detailed description of these phases.

Keying material predistribution: Assume that each sensor node has a unique identification whose range is from 1 to *N* where *N* is the maximum number of sensor nodes that could be deployed during the entire lifespan of the sensor network. Each of the unique identifications is represented by $m = \log_2(N)$ bit effective ID in sensor nodes' memory. The keying material predistribution phase is to predistribute secret key sharing functions to each sensor nodes before deployment such that after deployment, neighboring sensor nodes can find a secret common key between them using these functions. It consists of the following steps:

1. Key setup server generates l $(m \times m)$ symmetric matrices $M^\tau s\ (\tau = \overline{1,l})$ over finite field *GF(2)*. The M^τ s are private information and kept secret from both sensor nodes and adversaries. M^τ is used to generate the τth bit of a pairwise key between two neighboring sensor nodes, so *l* is the length of this key.
2. Key setup server computes secret key sharing function Φ_i for each sensor node S_i by first computing $\Phi_i^\tau = y_i M^\tau\ (\tau = \overline{1,l})$ (1) and then generating

Φ_i as $\Phi_i = \begin{bmatrix} \Phi_i^1 \\ \Phi_i^2 \\ .... \\ \Phi_i^l \end{bmatrix}$ where $y_i (i = \overline{1, N})$ is the *m*-dimensional vector, the effective ID of sensor node S_i. This function is then distributed to each sensor node before node deployment.

Pairwise key establishment: After completing the keying material predistribution phase, each sensor node possesses a secret key sharing function. The object of this phase is to establish pairwise keys among neighboring sensor nodes using those functions. The procedure for establishing two neighboring sensor nodes S_i and S_j is described as follows with and added step to allow explicit key authentication.

1. After being deployed, S_i and S_j instantly broadcast their effective IDs y_i and y_j to their neighboring nodes. Since S_i and S_j are neighbors, S_i will get S_j's effective ID y_j and vice versa.
2. S_i computes the possible pairwise key $K_{ij} : K_{ij}^{\tau} = \Phi_i^{\tau} y_j^T (\tau = \overline{1, l})$ (2), where K_{ij}^{τ} indicates the τ th bit of the possible pairwise key K_{ij} between S_i and S_j. S_j carries out in the same way to get the possible pairwise key K_{ji}.
3. Up to this step, S_i / S_j needs to certify that the other has the same key as the one it computed. To do this, S_i / S_j has to show the other that it has the other's computed key by revealing secret information without revealing the computed key. As in [2], S_i / S_j generates a message M_i/M_j containing y_j / y_i, calculates the *message authentication code* (MAC) of M_i/M_j as a function of M_i/M_j and its computed key: $MAC = C_{K_{ij}}(M_i)$ / $MAC = C_{K_{ji}}(M_j)$ and then send M_i/M_j plus MAC to the other (MAC can be calculate using a key-dependent one-way hash function such as HMAC [3]).

$$S_i \xrightarrow{M_i \| C_{K_{ij}}(M_i)} S_j \qquad S_i \xleftarrow{M_j \| C_{K_{ji}}(M_j)} S_j$$

4. The recipient performs the same calculation on the received message, using its computed key, to generate a new *MAC*. The received *MAC* is compared to the calculated *MAC*. If the received *MAC* matches the calculated *MAC* then the receiver is assured that the message is from the alleged sender and its computed key is exactly the same as that of the alleged sender. Since no one

else knows the secret key, no one else could prepare a message with a proper *MAC*.

Up to this point, any two neighboring sensor nodes can establish a pairwise key to secure their communication link. However, as shown in [1], our proposed scheme is vulnerable to the *information-theoretic security attack* discussed later against the network resiliency. To prevent this sort of attack, there are two approaches. The first one is to allow two neighboring sensor nodes to take part in the *pairwise key reinforcement* phase. The second one will be discussed later on in security analysis section.

Pairwise key reinforcement (optional): This phase is aimed to reinforce a pairwise key between two neighboring sensor nodes S_i and S_j. It happens as follows: S_i and S_j randomly generate k_i and k_j respectively such that their lengths are equal to K_{ij} / K_{ji}. These keys are encrypted by K_{ij}, K_{ji} and transmitted to each end. Then, S_i / S_j computes a new pairwise key with S_j / S_i using the formula: $K = K_{ij} \oplus k_i \oplus k_j$ ($K = K_{ji} \oplus k_j \oplus k_i$) (3). In addition to the avoidance of information-theoretic attack, these formulas show that each node has the equal right to decide the value of the potential key *K*. It ensures that no node can get an advantage over the other from *K* selection.

This scheme substantially improves the security, resiliency and enable node to node authentication in the network. These features as well as other parameters will be thoroughly analyzed in the following sections of this paper.

5 Security Analysis

As already mentioned above, our scheme is vulnerable to information-theoretic security attack against network resiliency. Indeed, our scheme has a certain collusion threshold. As mentioned, $M^{\tau}(\tau = \overline{1,l})$ is a $(m \times m)$ matrix. By using *m* linearly independent secret Φ_i^{τ} s, M^{τ} can be easily revealed. Therefore, *m* is the value of the collusion threshold. In other words, an adversary only needs to compromise *m* sensor nodes to be able to compute any pairwise key of any two uncompromised neighboring sensor nodes using their effective IDs. It implies that with only *m* compromised sensor nodes, the adversary can compromise the entire network.

A straightforward solution to the attack is to increase the value of *m*. However, the increase in the value of *m* leads to the increase of memory size of sensor nodes needed to store Φ_i. The figure 1 show the relationship between *m* (number of compromised nodes), pairwise key length *l* and memory usage.

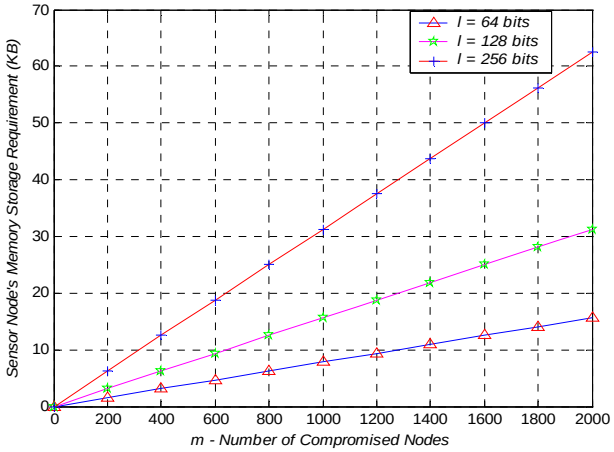

Fig. 1. Memory storage requirement against information-theoretic security attack

Assume that the key length of the pairwire key l = 128 bits, from the fig. 1, it is easy to realize that the solution can offer resistance to a collusion attack of up to 2000 compromised sensor nodes while using only about 32 KB of each sensor node's memory storage. This number of memory storage consumption is considered to be suitable for most sensor hardware platforms such as Berkeley Mica Motes with 128KB program memory [8]. Therefore, by increasing little amount of memory storage, the resiliency is significantly improved against *information-theoretic security attack*. This solution is considered to be acceptable in the sense that to successfully compromise the network, the adversary has to perform large scale attacks which are very expensive and more easily detectable.

The other solution has been briefly mentioned in section 3. This solution is partly inspired by an assumption in [4]. Accordingly, in this solution we assume that there exists a lower bound on the time interval T_{min} that is necessary for an adversary to compromise enough m sensor nodes, and that the time T_{est} for newly deployed sensor node to discover its immediate neighbors and establish initial pairwise keys with them is smaller than T_{min}. Taking advantage of the time interval T_{min}, two neighboring sensor nodes need to quickly exchange k_i and k_j to each other and then use (3) to change their initial pairwise key K_{ij} (K_{ji}) to the permanent pairwise key K. By doing in this way, we can eliminate the *information-theoretic security attack* from the entire network since the adversary could not compute the pairwise key K using (2).

In addition to information-theoretic security attack (node capture attack), our scheme also enable node to node authentication feature as already discussed. This feature, together with encryption techniques, is considered as a powerful tool to prevent some specific attacks carried out only in sensor networks such as sybil attack, sinkhole attack, hello flood attack, acknowledgement spoofing attack, etc [5], [6].

6 Performance Analysis

In this section, we analyze the performance of our scheme in term of memory usage, communication overhead and computational overhead.

As already analyzed in the aforementioned section, in our scheme, memory usage in each sensor node is in proportion to *m* given pairwise key length *l* and *l* given *m*. Increasing *l*, *m* or both result in the increase in security level (collusion threshold) but it implies more memory consumption to store that keying material in sensor nodes. The other approach to obtain higher security level (by eliminating collusion threshold attack) while the consumption of sensor nodes' memory storage could be significantly reduced is to include pairwise key reinforcement phase in the scheme. However, in this case, communication and computational overhead will be slightly increased. Thus, there must be trade-offs among security achievement, memory usage, communication overhead and computational overhead.

In our scheme, to establish the pairwise key, S_i and S_j need only to transmit three packets in case the pairwise key reinforcement phase is included. One packet is transmitted in the broadcast form. The other two packets are transmitted in the unicast form. These packets essentially contain the effective IDs of two nodes. Thus, the size of these packets is rather small. Therefore communication overhead of our scheme is rather low and can be acceptable in the distributed sensor network environment.

Considering computational overhead, it is easy to realize that our scheme is mainly based on multiplications of matrices $M^{\tau}(\tau = \overline{1,l})$ and sensor nodes' effective IDs over *GF(2)*. These multiplications essentially are exclusive-OR and AND bit operations. These multiplications consume much less computational time and require much less energy as well. The remaining computation constituting the overall computational overhead is *MAC* generation operations. These operations are considered as the least complex of the cryptographic algorithms and should intuitively incur the least energy cost [8]. For these reasons, the overall computational overhead of our scheme is not worth considering.

7 Conclusion

In this paper, we proposed a key predistribution scheme for distributed sensor networks inspired by the ID-based key predistribution scheme. Consequently, our scheme obviously inherits the noteworthy properties from that sort of scheme. First, the number of packets exchanged to establish a pairwise key between two sensor nodes which want to establish a secure communication channel is substantially minimized. Second, the key distribution procedure is composed of simple calculations so that computational costs are quite small and suitable for such computation limited devices as sensor nodes. Lastly, each sensor node has only to input its partner's identifier to its secret key sharing function to generate the desired key. Moreover, our schemes present two approaches which have been analyzed to cost sensor nodes much less their resource to tackle *information-theoretic security attack* inherited from

ID-based key predistribution schemes. Our schemes also expose a technique that enable explicit key authentication which is expected to be the most effective solution to some sorts of attacks in distributed sensor networks. For all those reasons, there is no doubt that our scheme is an appropriate solution to the key agreement problem in distributed sensor networks.

References

1. Matsumoto, T., and Imai, H., "On the KEY PREDISTRIBUTION SYSTEM: A Practical Solution to the Key Distribution Problem", Advances in Cryptology - Crypto'87, Lecture Note in Computer Science, Vol. 293, 1988, pp. 185-193.
2. Stallings, W., "Cryptography and Network Security: Principles and Practice", Prentice Hall, 1998.
3. Rhee, M. Y., "Internet Security: Cryptographic Principles, Algorithms, and Protocols", Wiley, 2003.
4. Zhu, S., Setia, S., and Jajodia, S., "LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks", CCS'03, Washington, DC, USA, October 2003.
5. Wood, A. D., and Stankovic, J. A., "Denial of Service in Sensor Networks", IEEE Computer, Vol. 35, No. 10, October 2002, pp. 54-62.
6. Karlof, C., and Wagner, D., "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures", Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications, May 2003, pp. 113 - 127.
7. Potlapally, N. R., Ravi, S., Raghunathan, A., and Jha, N. K., "A Study of the Energy Consumption Characteristics of Cryptographic Algorithms and Security Protocols", IEEE Transactions on Mobile Computing, Vol. 5, No. 2, February 2006.
8. Chan, H., Perrig, A., and Song, D., "Random key predistribution schemes for sensor networks", Proceedings 2003 IEEE Symposium on Security and Privacy, May 2003, pp. 197-213.
9. Du, W., Deng, J., Han, Y. S., Varshney, P. K., Katz, J., and Khalili, A., "A Pairwise Key Predistribution Scheme for Wireless Sensor Networks", ACM Transactions on Information and System Security, Vol. 8, No. 2, May 2005, pp. 228-258.
10. Liu, D., Ning, P., and Li, R., "Establishing Pairwise Keys in Distributed Sensor Networks", ACM Transactions on Information and System Security, Vol. 8. No. 1, February 2005, pp. 41-77.
11. Blaß, E.-O., and Zitterbart, M., "Towards Acceptable Public-Key Encryption in Sensor Networks", Proceedings of the 2nd International Workshop on Ubiquitous Computing, ACM SIGMIS, May 2005.
12. Wander, A. S., Gura, N., Eberle, H., Gupta, V., and Shantz, S. C., "Energy analysis of public-key cryptography for wireless sensor networks", Proceedings of the 3rd IEEE International Conference on Pervasive Computing and Communications, March 2005, pp. 324 - 328.
13. Blundo, C., Santis, A. D., Herzberg, A., Kutten, S., Vaccaro, U., and Yung, M. "Perfectly-secure key distribution for dynamic conferences", Advances in Cryptology - CRYPTO '92, Lecture Notes in Computer Science, Vol. 740, 1993, pp. 471-486.
14. Hanaoka, G., Nishioka, T., Zheng, Y., and Imai, H., "A Hierarchical Non-interactive Key-Sharing Scheme with Low Memory Size and High Resistance against Collusion Attacks", The Computer Journal, Vol. 45, No. 3, 2002.

15. Perrig, A., Szewczyk, R., Wen, V., Culler, D., and Tygar, D., "SPINS: Security protocols for sensor networks", Proceedings of 7th Annual International Conference on Mobile Computing and Networks, July 2001.
16. Liu, D., and Ning, P., "Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks", Proceedings of the 10th Annual Network and Distributed System Security Symposium, February 2003, pp. 263-276.
17. Karlof, C., Sastry, N., and Wagner, D. "TinySec: a link layer security architecture for wireless sensor networks", Proceedings of the 2nd international conference on Embedded networked sensor systems, November 2004.
18. Eschenauer, L., and Gligor, V. D., "A key-management scheme for distributed sensor networks", Proceedings of the 9th ACM Conference on Computer and Communications Security, November 2002, pp. 41-47.
19. Du, W., Deng, J., Han, Y.S., Chen, S., and Varshney, P.K., "A key management scheme for wireless sensor networks using deployment knowledge", INFOCOM 2004, Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies, Vol. 1, March 2004.
20. Blom, R., "An optimal class of symmetric key generation systems", Advances in Cryptology, Lecture Notes in Computer Science, Vol. 209, 1985, pp. 335-338.
21. Du, W., Ding, J., Han, Y., and Varshney, P., "A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks", Proceedings of the ACM Conference on Computer and Communication Security (CCS'03), Washington, D.C., October 2003.
22. Zhu, S., Setia, S., and Jajodia, S., "LEAP: Efficient security mechanisms for large-scale distributed sensor networks", Proceedings of 10th ACM Conference on Computer and Communications Security (CCS'03), October 2003, pp. 62-72.