Research Journal of Applied Sciences, Engineering and Technology 4(14): 2211-2215, 2012
ISSN: 2040-7467

Submitted: March 10, 2012 Accepted: March 22, 2012 Published: July 15, 2012

A Scalable and DoS-Resistant Key Management Scheme in Wireless Sensor Network

[1,2,3]Lingyu Lee, [3]Yunsheng Zhang, [1,2]Ying Jiang and [2]Lei Zhao
[1]Yunnan Key Lab of Computer Technology Application, Yunnan, China
[2]Faculty of Information Engineering and Automation,
[3]Faculty of Mechanical and Electrical Engineering, KunMing University of Science and Technology, Yunnan, China

Abstract: Pair-wise key establishment with authentication is basic and necessary to configure secure sensor networks. In addition, nodes in a sensor network may be lost because of the power exhaustion problem or malicious attacks. Thereby pair-wise key establishment and new node deployment are necessary in the sensor network. Recently, a study was proposed a practical access control protocol for Secure Sensor Networks based on the elliptic curve cryptography. We point out its weakness and propose more efficient scheme as satisfying better security strength.

Key words: Authentication, DoS, key establishment, sensor network security

INTRODUCTION

A WSN is large number of sensors distributed over a sensor field, which may be deployed in hostile areas where communication is monitored and nodes are subject to capture and surreptitious use by an adversary. To enable secure communication between sensor nodes, it is critical for a WSN to provide security services as authentication and key management. Key management consists of key establishment, key revocation and key update. It is a big challenge in sensor networks because the nodes may not know anything about their neighbors before deployment. There are four types of general key agreement schemes: The trusted-server scheme, the self-enforcing scheme, the key pre-distribution scheme and no key pre-distribution scheme (Youssou Faye and Thomas, 2011. The practical option for distribution keys to the sensor nodes of wireless sensor networks rely on pre-distribution, that is, key information is distributed among all sensor nodes prior to deployment. Cryptographic algorithms require keys to be shared between entities (sensor nodes and base station and user) (Chan *et al*., 2003).

Existing key pre-distribution schemes can be divided into two approaches in WSNs, namely, global key based and unique key based. In the schemes base on global key, one may set up a global key among the network so that two sensor nodes can establish a key based on this global key. In unique key based approach, it assigns each sensor node a unique random key with each of the other nodes. However, the former is vulnerable to the compromise of a single node and the latter introduces huge storage overhead on sensor nodes (Xiao *et al*., 2007).

Huang proposed a Novel Access Control Protocol (NACP) based on ECC and authentication hash chain to support the practical implementations for sensor networks (Huang, 2009). NACP is quite adequate for power and resource constrained sensor nodes and could be easily implemented as a dynamic access control. However, in (Kim and Lee, 2009) point out that NACP is insecure to the replay attack, against new node masquerading attack in the presence of an active adversary and has the lack of hash chain renewability which is one of important aspects in resource constrained sensor networks. Then they proposed an Enhanced Novel Access Control Protocol (ENACP) using the elliptic curve cryptography and the hash chain. Unfortunately, in (Zeng *et al*., 2010). Identified an inherent flaw in their design and demonstrated that ENACP is vulnerable to a new node masquerading attack and a legal node masquerading attack, in violation of their security claims.

Additionally, Lee *et al*. in (Srisooksai *et al*., 2012) also prove that NACP is not dynamic access control in a broad view and authentication by hash chain require huge communication overhead and memory cost. They propose a Practical Access Control Protocol (PACP) to conquer the insecurities in NACP. Compared with other schemes, the PACP is simple to implement for pair-wise key establishment. However, this study points out that their protocol is insecure in DoS attack and inherent design

Corresponding Author: Lingyu Lee, Yunnan Key Lab of Computer Technology Application, Yunnan, China

flaws in scalable. To solve these problems, this study proposes an efficient and practice access control scheme with better security strength.

REVIEW OF LEE'S SCHEME

We review Lee's scheme. It consists of the three phases: initialization phase, key establishment phase and a new node addition phase.

Initialization phase: Suppose there are a number of r neighborhood nodes with identities $\{N_1, N_2, \ldots\ldots N_r\}$ in a designated area. The base station first generates a number of m (m>r) secret keys $\{k_1, k_2, \ldots, k_r, k_{r+1}, \ldots, k_m\}$ and preloads a set of hash value HV_i, which based on the secret key k_i, to corresponding node N_i. For example, $HV_i = \{h(k_1, N_i), h(k_2, N_i), \ldots, h(k_r, N_i), h(k_{r+1}, N_i), \ldots, h(k_m, N_i)\}$is for node N_i.

Authentication and key establishment phase: Suppose a node N_i and a node N_j is neighborhood. Note that each node already recognized its neighbor nodes with broadcasting their identities. The processes of authentication and key establishment between two nodes, N_i and N_j, are as follows:

Step 1: N_i generates a random number t_i and computes point $A_i = t_iP = (A_{xi}, A_{yi})$ over the elliptic curve E_q and $s_i = h(N_i \| A_{xi} \| h(k_i \| N_j))$. N_i sends $\{A_i, s_i, N_i\}$ to N_j. N_j also generates a random number t_j and computers the point $A_j = tjP (A_{xj}, A_{yj})$ and $s_j = h(N_j \| A_{xj} \| h(kj \| N_i))$. N_j sends $\{A_j, s_i, N_j\}$ to N_i. both N_i and N_j check A_j and A_i, respectively. If not fresh, they withdraw.

Step 2: N_i computes $K_{ij} = t_iA_j = (K_{xij}, K_{yij})$ and $z_i = h(N_i \| K_{xij} \| h(ki \| N_j))$ and delivers z_i to N_j. After receiving zi , N_i computes $K_{ij} = tj\ A_i = (K_{xij}, K_{yij})$ and checks whether $h(N_i \| A_{xi} \| h(_{ki} \| N_j)) = s_i$ and $h(N_i \| K_{xij} \| h(k_i \| N_j)) = z_i$ are hold, where $h(k_i \| N_j)$ is a preload value in set HV_j. If these conditions are satisfied, the N_j can make sure that the node N_i is a legitimate node.

Step 3: N_j also computes $z_j = h(N_j \| K_{xij} \| h(k_j \| N_i))$ and sends z_j to the node N_i.

Step 4: The node N_i checks whether $s_j = h(N_j \| A_{xj} \| h(k_j \| N_i))$ and $z_j = h(N_j \| K_{xij} \| h(k_j \| N_i))$ with preloaded value $h(k_j \| N_i)$ in set HV_i. If it holds, N_i also authenticates N_j as a legal node and confirms the shared session key.

Node addition: When a new node with identity $N_r = 1$ is added, base station generates k_{r+1} and $HV_{r+1} = \{h(k_1, N_{r+1}), h(k_2, N\ r+1), .., h(k_r, N_{r+1}), h(k_{r+1}, N_{r+2}), .., h(k_m, N_{r+1})\}$. Like other nodes, base station preloads HV_{r+1} to the node. Authentication and key establishment is performed as the above.

Weakness of lee's scheme: It is claimed in Lee's paper that his protocol can resist against both passive and active attacks. Unfortunately, we find out that his protocol is unsecure against the DoS attack. Moreover, we point out an inherent design flaw in scalable.

DoS attack: To facilitate discussion, we assume in the following that N_i and N_j want to establish a new pair-wise key between them and A_{i1} is the point over the elliptic curve E_q, which N_j used in the last time. Recall that in authentication and key establishment phase, N_i firstly check the A_{i2}, the point over elliptic curve N_j uses this time. If A_{j2} is different to A_{j1}, N_i computes $\{A_j, s_j, N_j\}$ and sends them to N_j. And then verifies z_i and s_i after receiving z_i from N_j. Hence, N_i can authenticate whether a node is legal or not only after N_i works out $\{A_j, s_j, s_i, z_i\}$ with three times communication. It's a large overhead for a resource-constrained sensor node. When an attacker performs the DoS attack and the node has passed through authentication u times, her/she has been collected a group of points over elliptic curve E_q $\{A_{i1}, A_{i2}, \ldots A_{jr}, \ldots, A_{iu}\}$, by intercepting the secret change messages between two nodes (i.e., N_i and N_j). What he/she needs to do is just to broadcast A_{jr}(a point of elliptic curve E_q different from the last time) to all nodes except to the node N_i, repeatedly. All neighbor nodes exhaust their resources and render them less capable of providing network services.

Lack of scalability: The number of new nodes that can be added later is only dependent on the number m that the base station chose in initialization phase. A naïve solution is just by using a large enough number, the base station generates corresponding secret keys and preloads the set of hash value $HV_i = \{h(k_1, N_i), h(k_2, N_i), \ldots, h(k_r, N_i), h(k_{r+1}, N_i), \ldots, h(k_m, N_i)\}$ for each node. However, the node over sensor networks has a severe memory restrictions and the system does not guarantee that how many new nodes requires in a node lifecycle. Thereby, more efficient solution should be devised.

NETWORK MODEL AND THREAT MODEL

Network model: In our network model, the WSN consists of a base station and many sensor nodes. Sensor nodes can communicate each other in the transmission range, which is the communication region of wireless sensor nodes as circle area of radius y. We assume that sensor nodes form a number of clusters after deployment, each containing r nodes. And one node can only belong to

one cluster. In each cluster, one node is randomly selected as the cluster head. All nodes take turns to serve as the cluster head to balance energy consumption. To facilitate discussion, we only consider the bidirectional links between neighbor nodes and assume that sensor nodes simply discard or ignore those links that are not bidirectional.

Threat model: We assume that an adversary can launch both passive and active attacks. In passive attacks, the adversary may eavesdrop or copy the transmitted messages in the WSN or. With active attacks, the adversary can modify or replay transmitted messages. Additionally, the adversary may launch DoS attacks. The adversary may jam the communication channel. We think over the secure connectivity for the sensor networks.

Proposed scheme:

Initialization phase:

Step 1: BS (base station) randomly chooses two large safe primes p and q and computes a public modulus n = pq. Then BS chooses a pair of integers e and d satisfying the properties ed = 1(mod ϕ(n)) and d is large enough, where ϕ(n) is the Euler-Totient function. BS chooses a public one-way hash function h(·).The private key (p, q, d) is kept secret by BS, while the public key of BS is (n, e), which preloads to each node.

Step 2: Suppose there are r neighborhood nodes in a cluster with identities $\{N_1. N_2, \ldots, N_r\}$ in a designated area. BS generates r secret keys $\{k_1, k_2, \ldots, k_r\}$. Then BS computes a set of hash value HV_i for each node. For example, $HV_i = \{h(k_1, N_i), h(k_2, N_i), \ldots, h(k_r, N_i)\}$ is for N_i, as the same as Lee's scheme. Finally BS preloads HV_i to its corresponding node.

Step 3: BS performs all necessary processes to configure an elliptic curve E_q and publishes the elliptic parameters {Eq, P, m}.

So each node has been preload the public key (n, e), its corresponding HV_i and elliptic parameters {Eq, P, m} before deployment.

Authentication and key establishment phase: Every node need store the number of the last time that has authenticated and established successfully with another node in the cluster. For example both N_i and N_j store $(S_c)_{ij}$, which means the cth time has authenticated and established pair-wise key successfully between N_i and N_j. The processes of authentication and key establishment between two nodes N_i and N_j are as follows:

Step 1: The node N_i generates a random number t_i and computes the point $A_i = t_iP = (A_{xi}, A_{yi})$ over the elliptic curve E_q and $s_i = h(N_i \| A_{xi} \| h(k_j \| N_i) \| (S_{c+1})_{ij})$. N_i sends $\{A_i, s_i, N_i, h((Sc+1)_{ij} \| N_i)\}$ to N_j. Similarly, the node N_j generates a random number t_j and computes the point $A_j = t_jP = (A_{xj}, A_{yj})$ over the elliptic curve E_q and $s_j = h(N_j \| A_{xj} \| h(k_i \| N_j) \| (Sc+1)_{ij})$. N_j sends $\{A_j, s_i, N_j, h((S_{c+1})_{ij} \| N_j)\}$ to N_i.

Step 2: After receiving $\{A_j, s_i, N_i, (S_{c+1})_{ij})\}$, N_j read $(S_c)_{ij}$ from its memory, check whether $h((S_c)_{ij} +1 \| N_i) = h(((S_{c+1})_{ij}) \| N_j)$. If the equation holds, N_i check $s_j = h(N_j \| A_{xj} \| h(k_i \| N_j) \| (S_{c+1})_{ij}))$, otherwise. Only if it holds, N_i authenticates N_j as a legal node. Similarly, the node N_j check $(S_{c+1})_{ij})$ firstly, then check s_i, If both two of them are correct, N_j authenticates N_i as a legal node. Both N_i and N_j refresh the $(S_c)_{ij}$ to $(S_{c+1})_{ij})$ in its memory, respectively.

Step 3: The node N_i computes the session key $K_{ij} = t_iA_j = (K_{xij}, K_{yij})$. And N_j computes the session key $K_{ij} = t_jA_i = (K_{xij}, K_{yij})$.

New node injection phase:

Step 1: When a new node with identity N_{r+1} is added, BS also generate k_{r+1} and preloads it with $HV_{r+1} = \{h(k_1 \| N_{r+1}), h(k_2 \| N_{r+2}), \ldots, h(k_r \| N_{r+1})\}$ to the new node N_{r+1}. Then BS computes $V_{r+1} = h(k_{r+1} \| N_i)^d \pmod n$, $i \in (0 \ldots n)$ and broadcasts V_{r+1} to the cluster which the node $N_{r+1}$is added.

Step 2: All nodes in the cluster, say N_i, compute $V_{r+1}^e = h(k_{r+1} \| N_i)$ and store it into its own HV_i. Then the new node can establish pair-wise key with other nodes in cluster.

Security analysis: This section gives security analyses of the proposed scheme. In our propose scheme the security of authentication and key establishment phase is based on one-way hash function and the elliptic curve discrete

logarithm problem. And the security of new node addition phase is based on factoring problem. For the security analyses, we could consider some attacks including DoS attack, masquerading attack.

DoS attack: we assume in the following that N_i and N_j want to establish a new pair-wise key between them and the adversary has gotten $\{A_{i1}, A_{j1}, A_{i2}, A_{j2}, \ldots, A_{iu}, A_{ju}\}$, which is a set of the points over the elliptic curve E_q that N_i and N_j used to establish keys before. Recall that in authentication and key establishment phase, N_i (or N_j) reads $(S_c)_{ij}$ from its memory, check whether h((Sc)ij+1 || N_i)=h((Sc+1)ij || N_j) firstly. And the node check $s_j = h(N_j \| A_{xj} \| h(k_i \| N_j) \| (S_{c+1})_{ij})$. Because the $(S_c)_{ij}$ does not

include in the transmitted messages at any time, the adversary cannot eavesdrop the $(S_c)_{ij}$. Only when the former equation holds, the N_j can make sure that the node N_i is a legitimate node. Thus, in order to create a right $(S_c)_{ij}$, an adversary must have abundant resources to be able to promptly compute many times. Note that BS assigns each sensor node a unique pair-wise key between two nodes. In contrast, although checking the equation slightly increase legitimate nodes computational load, they are still able to provide services regardless the existence of the attack.

Masquerading attack: There are two kinds of masquerading attacks, legal node masquerading attack and new node masquerading attack.

- **Legal node masquerading attack:** Even if an attacker could eavesdrop the overall communication sessions between the node N_i and the node N_j and obtains the information of Ai, Aj, si, sj, $h((S_{c+1})_{ij}) \| Nj)$ and $h((S_{c+1})_{ij}) \| N_i$) from step1 in the authentication and key establishment phase, it is very difficult for him/her to derive the correct $(S_{c+1})_{ij})$ and k_i from $h((S_{c+1})_{ij}) \| N_j)$ and s_i, respectively. Moreover, he/she also cannot derive t_i and t_j from A_i and A_j, respectively. So it is difficult for the adversary to work out the legalized session key K_{ij}, because of the elliptic curve discrete logarithm problem. Thereby, our propose scheme can withstand to the legal node masquerade attack.
- **New node masquerading attack:** When some sensor nodes need to be add into WSN, say the node Nr+1. For this, the BS preloads a secret key kr+1, h(·), a set of hash value HVr+1, public key of BS is (n, e) and ECC parameters {Eq, P, m} to Nr+1's memory and broadcasts $Vr+1 = h(kr+1 \| N_i)^d$ (mod n), I∈(0,..., n) for I∈(0,... ,n) to inform the new addition to all existing nodes in the cluster. All nodes in the cluster, say N_i compute $Vr+1^e = h(kr+1 \| N_i)$

  and store it into its own HVi. And N_i get ready to establish pair-wise with the new node. Since the adversary has no way to know the private key of BS, he/she cannot masquerade a new legal node with the same as from the legalized BS.

Scalable: In step 1 of initialization phase, the BS computes the private key (p, q, d) and the public key (n, e). The private key is kept secret by BS, while the public key of BS is preloads to each node. When some sensor nodes were lost or exhausted, a new sensor node needs to be deployed (say N_{r+1}). The BS calculates $V_{r+1} = h(k_{r+1} \| N_i)^d$ (mod n), i∈(0...n) for i∈(0...n) and broadcasts V_{r+1}. Hence, the BS need not to decide how many new can add after the WSN is deployed in initialization phase. Addition, each node's HV, the set of hash value, only contains the value of the new node and the node deployed before in the cluster.

Supporting secure connectivity: When the attacker captures a node physically, he/she can obtain all the secrets loaded in the node. He/she can establish pair-wise keys with other nodes. However, there is no way to affect the security connectivity between any other node pair. Hence the secure connectivity is provided when there are only small effects on the security of the overall networks.

CONCLUSION

This study has shown that lee's Practice Access Control Protocol (PACP) is insecure against the Dos attack and inherent design flaws in scalable. To solve these problems, this study proposes an efficient and practice access control scheme with better security.

Notations:

Symbol	Description
N_i	An identity of the node Ni
t_i	A random number of the node N_i
k_i	A secret key of the node N_i
h(·)	A secure one-way hash function
‖	A concatenation operation
p, q	Two large primes
e, d	A pair of integers
n	A public modulus n = pq
ϕ(n)	The Euler-Totient function
E_q	An elliptic curve
$A_i = (x_i, y_i)$	A point over E_q
s	A generator of a group G
O	An order of E_q at least 160 bitgs
K_{ij}	A session key established between N_i and N_j

REFERENCES

Chan, H., A. Perrig and D. Song, 2003. Random key predistribution schemes for sensor networks. Carnegie Mellon University, Pittsburgh, PA 15213.

Huang, H.F., 2009. A novel access control protocol for secure sensor networks. Comp. Stand. Inter., 31(2): 272-276.

Kim, H.S. and S.W. Lee, 2009. Enhanced novel access control protocol over wireless sensor networks. IEEE T. Consum. Electr., 55(2): 492-498.

Srisooksai, T., K. Keamarungsi, P. Lamsrichan and K. Araki, 2012. Practical data compression in wireless sensor networks: A survey. J. Netw. Comput. Appl., 35(1): 37-59.

Xiao, Y., V.K. Rayi, B. Sun, X. Du, F. Hu and M. Galloway, 2007. A survey of key management schemes in wireless sensor networks. Comput. Commun., 30(11-12): 2314-2341.

Youssou Faye, I.N. and N. Thomas, 2011. A survey of access control schemes in wireless sensor network.pdf. World Acad. Sci. Eng. Technol., 59: 814-823.

Zeng, P., K.K. Choo and D.Z. Sun, 2010. On the security of an enhanced novel access control protocol for wireless sensor networks. IEEE T. Consum. Electr., 56(2), 566-569.