ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 7, July 2016

Peer-To-Peer Key Exchange Mechanism in Dynamic Wireless Sensor Networks

Shwetha Rokhade[1], Prof. Hanumathappa S N[2]

IV Sem M.Tech [DCN] Student, UBDTCE, Davangere, India[1]

Assistant Professor, Dept. of ECE, UBDTCE, Davangere, India [2]

ABSTRACT: There are wide variety of applications deployed in wireless sensor networks (WSNs) such as military sensing and tracking, patient status monitoring, traffic flow monitoring, where sensory devices often move between different locations. Securing data and communications requires suitable encryption key protocols. In this paper, we analyzed a certificate less-effective key management (CL-EKM) protocol for secure communication in dynamic WSNs characterized by node mobility. Securing data and communications requires suitable encryption key protocols. In this paper, we propose a certificate less-effective key management (CL-EKM) protocol for secure communication in dynamic WSNs characterized by node mobility. The CL-EKM supports efficient key updates when a node leaves or joins a cluster and ensures forward and backward key secrecy. The protocol also supports efficient key revocation for compromised nodes and minimizes the impact of a node compromise on the security of other communication links. A security analysis of our scheme shows that our protocol is effective in defending against various attacks.

KEYWORDS: Wireless sensor networks, certificateless public key cryptography, key management scheme.

I. INTRODUCTION

The emerging field of wireless sensor networks (WSN) combines sensing, computation and communication into a single tiny device. Through advanced mesh networking protocols, these devices form a sea of connectivity that extends the reach of cyberspace out into the physical world. As water flows to fill every room of a submerged ship, the mesh networking connectivity will seek out and exploit any possible communication path by hopping data from node to node in search of its destination. While the capabilities of any single device are minimal, the composition of hundreds of devices offers radical new technological possibilities. The power of wireless sensor networks lies in the ability to deploy large numbers of tiny nodes that assemble and configure themselves. Usage scenarios for these devices range from real-time tracking, to monitoring of environmental conditions, to ubiquitous computing environments, to in situ monitoring of the health of structures or equipment. While often referred to as wireless sensor networks, they can also control actuators that extend control from cyberspace into the physical world. The most straightforward application of wireless sensor network technology is to monitor remote environments for low frequency data trends. For example, a chemical plant could be easily monitored for leaks by hundreds of sensors that automatically form a wireless interconnection network and immediately report the detection of any chemical leaks. Unlike traditional wired systems, deployment costs would be minimal. The CL-EKM does not require the use of certificates and yet does not have the built-in key escrow feature of ID-PKC. It is a model for the use of public key cryptography that is intermediate between traditional PKI and ID-PKC. A CL-EKM system still makes use of a trusted third party which is called the key generating center (KGC). By way of contrast to the PKG in ID-PKC, the KGC does not have access to the user's private key. Instead, the KGC supplies a user with a partial private key that the KGC computes from the user's identity and a master key. The user then combines the partial private key with some secret information to generate the actual private key. The system is not identity-based, because the public key is no longer computable from a user identity. When Alice wants to send a message to Bob in a CL-EKM system, she must obtain Bob's public key. However, no authentication of Bob's public key is necessary and no certificate is required. In this paper, we present a certificateless effective key management (CL-EKM) scheme for dynamic WSNs. In certificateless public key cryptography (CL-PKC) [12], the user's full private key is a combination of a partial private key generated by a key generation center (KGC) and the user's own secret value. The special organization of the full private/public key pair removes the need for certificates and also resolves the key escrow problem by removing the responsibility for the user's full private key.

DOI:10.15662/IJAREEIE.2016.0507080

We also take the benefit of ECC keys defined on an additive group with a 160-bit length as secure as the RSA keys with 1024-bit length.

In order to dynamically provide both node authentication and establish a pairwise key between nodes, we build CL-EKM by utilizing a pairing-free certificateless hybrid sign cryption scheme (CL-HSC) proposed by us in an earlier work [13], [14]. Due to the properties of CL-HSC, the pairwise key of CL-EKM can be efficiently shared between two nodes without requiring taxing pairing operations and the exchange of certificates. To support node mobility, our CL-EKM also supports lightweight processes for cluster key updates executed when a node moves, and key revocation is executed when a node is detected as malicious or leaves the cluster permanently. CL-EKM is scalable in case of additions of new nodes after network deployment. CL-EKM is secure against node compromise, cloning and impersonation, and ensures forward and backward secrecy.

The remaining of this paper is organized as follows: In Section 2, we briefly discuss Literature work. In Section 3, we provide

our network model and adversary model. In Section 4, we provide an overview of our CL-EKM. In Section 5, we introduce the details of CL-EKM. In Section 6, we analyze the security of CL-EKM. In Section 7, we evaluate the performance of CL-EKM, conduct the simulation in Section 8, and conclude in Section 9.

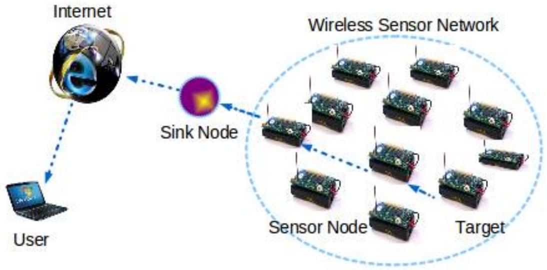


Fig.1. Dynamic wireless sensor network.

II. LITERATURE WORK

Symmetric key schemes are not viable for mobile sensor nodes and thus past approaches have focused only on static WSNs. A few approaches have been proposed based on PKC to support dynamic WSNs. Thus, in this section, we review previous PKC-based key management schemes for dynamic WSNs and analyze their security weaknesses or disadvantages.

Chuang et al. [7] and Agrawal et al. [8] proposed a two-layered key management scheme and a dynamic key update protocol in dynamic WSNs based on the Diffie-Hellman (DH), respectively. However, both schemes [7], [8] are not suited for sensors with limited resources and are unable to perform expensive computations with large key sizes (e.g. at least 1024 bit). Since ECC is computationally more efficient and has a short key length (e.g. 160 bit), several approaches with certificate [5], [10], [15] have been proposed based on ECC. However, since each node must exchange the certificate to establish the pairwise key and verify each other's certificate before use, the communication and computation overhead increase dramatically. Also, the BS suffers from the overhead of certificate management. Moreover, existing schemes [5], [10], [15] are not secure. Alagheband et al. [5] proposed a key management scheme by using ECC-based signcryption, but this scheme is insecure against message forgery attacks [16]. Huang et al. [15] proposed a ECC-based key establishment scheme for self-organizing WSNs. However, we found the security weaknesses of their scheme. In step 2 of their scheme, a sensor node *U* sends

$$z = qU \cdot H(MacKey) + dU\ (modn)$$

to the other node *V* for authentication, where *qU* is a static private key of *U*. But, once *V* receives the *z*, it can disclose *qU*, because *V* already got *MacKey* and *dU* in step 1. So, *V* can easily obtain *qU* by computing

$$qU = (z - dU) \cdot H(MacKey)_{-}1.$$

Thus, the sensor node's private key is exposed to the other node during the key establishment between twonodes. Zhang et al. [10] proposed a distributed deterministic key management scheme based on ECC for dynamic WSNs. It uses the symmetric key approach for sharing the pairwise key for existing nodes and uses an asymmetric key approach to share the pairwise keys for a new node after deployment. However, since the initial key *KI* is used to compute the individual keys and the pairwise keys after deployment for all nodes, if an adversary obtains *KI*, the adversary has the ability to compute all individual keys and the pairwise keys for all nodes. Thus, such scheme suffers from weak resilience to node compromises. Also, since such scheme uses a simple ECC-based DH key agreement by using each node's long-term public key and private key, the shared pairwise key is static and as a result, is not secure against known-key attacks and cannot provide re-key operation. Du et al. use a ECDSA scheme to verify the identity of a cluster head and a static EC-Diffie- Hellman key agreement scheme to share the pairwise key between the cluster heads. Therefore, the scheme by Du et al. is not secure against known-key attacks, because the pairwise key between the cluster heads is static. On the other hand, Du et al. use a modular arithmetic-based symmetric key approach to share the pairwise key between a sensor node and a cluster head. Thus, a sensor node cannot directly establish a pairwise key with other sensor nodes and, instead, it requires the support of the cluster head. In their scheme, in order to establish a pairwise key between two nodes in the same cluster, the cluster head randomly generates a pairwise key and encrypts it using the shared keys with these two nodes. Then the cluster head transmits the encrypted pairwise key to each node. Thus, if the cluster head is compromised, the pairwise keys between non-compromised sensor nodes in the same cluster will also be compromised. Therefore, their scheme is not compromise-resilient against cluster head capture, because the cluster head randomly generates a pairwise key between sensor nodes whenever it is requested by the nodes. Moreover, in their scheme, in order to share a pairwise key between two nodes in different clusters, these two nodes must communicate via their respective cluster heads. So, after one cluster head generates the pairwise key for two nodes, the cluster head must securely transmit this key to both its node and the other cluster head. Thus, this pairwise key should be encrypted by using the shared pairwise key with the other cluster head and the shared key with its node, respectively. Therefore, if the pairwise key between the cluster heads is exposed, all pairwise keys of the two nodes in different clusters are disclosed. The scheme by Du et al. supports forward and backward secrecy by using a key update process whenever a new node joins the cluster or if a node is compromised. However, the scheme does not provide a process to protect against clone and impersonation attack.

Most recently, Rahman et al. [4] and Chatterjee et al. have proposed ID-PKC based key management schemes supporting the mobility of nodes in dynamic WSNs which removes the certificate management overhead. However, their schemes require expensive pairing operations. Although many approaches that enable pairing operations for sensor nodes have been proposed, the computational cost required for pairing is still considerably higher than standard operations such as ECC point multiplication. For example, NanoECC, which uses the MIRACL library, takes around 17.93s to compute one pairing operation and around 1.27s to compute one ECC point multiplication on the MICA2(8MHz) mote [17].

III. OVERVIEW OF THE CERTIFICATELESS EFFECTIVE KEY MANAGEMENT SCHEME

In this paper, we propose a Certificate less Key Management scheme (CL-EKM) that supports the establishment of four types of keys, namely: a certificate less public/private key pair, an individual key, a pairwise key, and a cluster key. This scheme also utilizes the main algorithms of the CL-HSC scheme [13] in deriving certificateless public/private keys and pairwise keys. The purpose of these keys.

A. *Types of Keys*

• *Certificateless Public/Private Key:* Before a node is
deployed, the KGC at the BS generates a unique certificateless private/public key pair and installs the keys in the node. This key pair is used to generate a mutually authenticated pairwise key.

• *Individual Node Key:* Each node shares a unique
individual key with BS. For example, a *L*-sensor can use the individual key to encrypt an alert message sent to the BS, or if it fails to communicate with the *H*-sensor. An *H*-sensor can use its individual key to encrypt the message

corresponding to changes in the cluster. The BS can also use this key to encrypt any sensitive data, such as compromised node information or commands. Before a node is deployed, the BS assigns the node the individual key.

• *Pairwise Key:* Each node shares a different pairwise key with each of its neighboring nodes for secure communications and authentication of these nodes.

• *Cluster Key:* All nodes in a cluster share a key, named as cluster key. The cluster key is mainly used for securing broadcast messages in a cluster, e.g., sensitive commands or the change of member status in a cluster. Only the cluster head can update the cluster key when a *L*-sensor leaves or joins the cluster.

Advantages of Proposed System:

- To support node mobility, our CL-EKM also supports lightweight processes for cluster key updates executed when a node moves, and key revocation is executed when a node is detected as malicious or leaves the cluster permanently.
- CL-EKM is scalable in case of additions of new nodes after network deployment. CL-EKM is secure against node compromise, cloning and impersonation, and ensures forward and backward secrecy. The security analysis of our scheme shows its effectiveness.

IV. DESIGN OF CERTIFICATELESS EFFECTIVE KEY MANAGEMENT SCHEME

The CL-EKM is comprised of 5 phases: *Environment setup, Pair wise key generation, cluster formation, key update, and node movement.*

A. Environment Setup

Before the topology build up Base Station generates system parameters and Registers the node by including in a member list.

In this Networks, there is no one-for-all scheme that works well in scenarios with different network sizes, traffic overloads, and node mobility patterns. Routing performance is compared based on simulation results. Ns-2 is a discrete event simulator using in networking research. NS-2 used for wired and wireless network to provides significant support for simulation of TCP, routing and multicast protocols. It is combination of two simulation tools. The network simulator (ns) contains all commonly used IP protocols. The network animator(nam), which is use to visualize the simulations. Ns-2 can fully simulates a layered network from the physical radio transmission channel to high-level applications.

Table: 1 SIMULATION PARAMETERS

Simulation Parameters	**Value**
Simulator	Ns-2 (2.35)
Topology	1000*1000
No. of Nodes	0- 33
Bandwidth	3 Mbps
Queue length	1000
Packet Size	512 bytes
Energy	100J
Simulation Time	15 s
Routing Protocol	DSR

B. Pairwise Key Generation

In this phase, two symmetric shared keys, a secret key and a public key, are encrypted by the pre-distributed key and are distributed locally. Keys are shared by a cluster head to all its cluster members, which is mainly used for

DOI:10.15662/IJAREEIE.2016.0507080

securing locally broadcast messages, e.g., routing control information, or securing sensor messages. Moreover, in order to form a secure communication channel between the gateways of adjacent clusters, a symmetric shared key may be used to encrypt the sending message. In this phase, another challenge encrypted by a key may be made to guard against adversaries. Therefore, the security of intra-cluster communication and inter-cluster communication are established upon shared keys, respectively.

C. Cluster Formation

The network is evenly divided into small grids. Each grid has a relative location based on the grid information. The node in each grid with the highest energy level is selected as the head node or message forwarding. The head node can be reelected if the energy level becomes lower than other nodes in the grid. In addition, each node in the grid will maintain its own attributes, including location information, remaining energy level of its grid, as well as the attributes of its adjacent neighboring grids. The information maintained by each sensor node will be updated periodically. We assume that the sensor nodes in its direct neighboring grids are all within its direct communication range. We also assume that the whole network is fully connected through multi-hop communications. In addition, through the maintained energy levels of its adjacent neighboring grids, it can be used to detect and filter out the compromised nodes for active routing selection.

D. Key Update

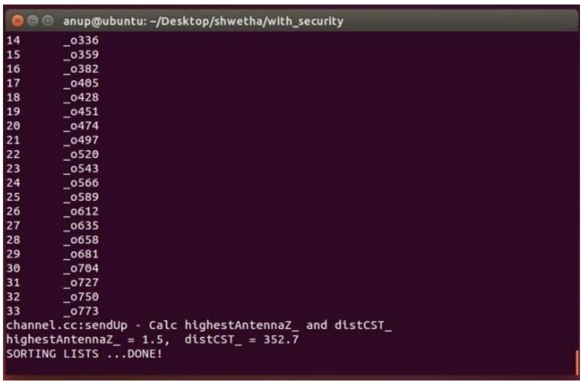
Using the same encryption key for extended periods may incur a cryptanalysis risk. To protect the sensor network and prevent the adversary from getting the keys, key update may be necessary. Initially all cluster heads (CHs) choose an originator to start the "key updates", and then it will send the index to all cluster heads in the network. After selecting the originator, it initializes the "Key update" process and sends the index to its neighboring clusters by heads. Then the cluster head refreshes the two keys from the key pool and distributes the two new keys to their cluster members locally.

E. Node Movement

When a node moves between clusters, the *H*-sensors must properly manage the cluster keys to ensure the forward/backward secrecy. Thus, the *H*-sensor updates the cluster key and notifies the BS of the changed node status. Through this report, the BS can immediately update the node status in the M. We denote a moving node as n_{Lm}.

V. SIMULATION RESULTS

Initial Setup is done in terminal which is shown in the below figure.



```
14       _o336
15       _o359
16       _o382
17       _o405
18       _o428
19       _o451
20       _o474
21       _o497
22       _o520
23       _o543
24       _o566
25       _o589
26       _o612
27       _o635
28       _o658
29       _o681
30       _o704
31       _o727
32       _o750
33       _o773
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 352.7
SORTING LISTS ...DONE!
```

Fig.2.setup in terminal window

Performance Evaluation

In this section, evaluate the performance of simulation. We are using the xgraph for evaluate the performance. We choose the some evaluation metrics: Packet delivery ratio – the ratio of the total number of packets received by the destination node to the number of packet sent by the source, Energy Consumption – the energy consumed by the source node and destination node and also calculate the Throughput and PDR. Along these evaluation metrics we have to evaluate the simulation performance in xgraph.

Packet Delivery Ratio:

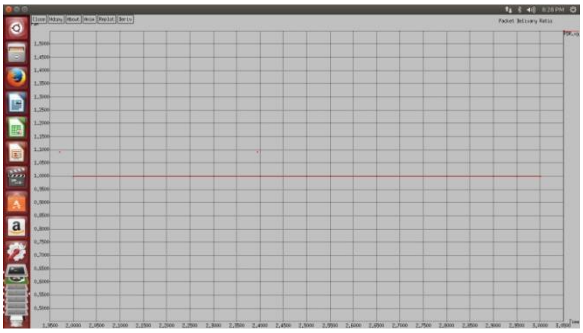


Fig.3.Packet Delivery Ratio

Throughput :

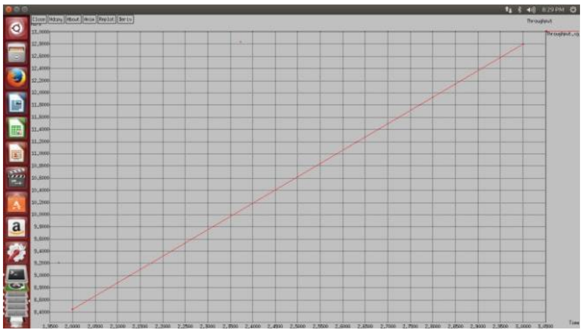


Fig.4.Throughput

Energy Consumption :

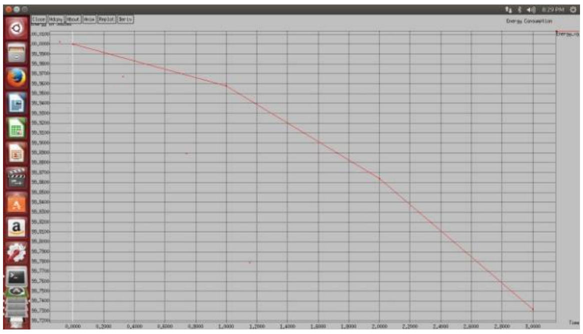


Fig.5.Energy Consumption

DOI:10.15662/IJAREEIE.2016.0507080

Packet Drop :

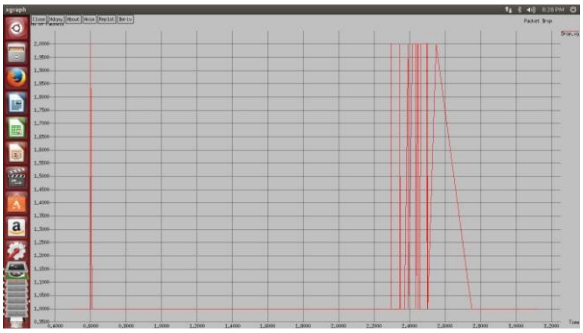


Fig.6.Packet Drop

VI. CONCLUSION

We proposed the certificate less effective key management protocol (CL-EKM) for secure communication in dynamic WSNs. CL-EKM supports efficient communication for key updates and management when a node leaves or joins a cluster and hence ensures forward and backward key secrecy. Our scheme is resilient against node compromise, cloning and impersonation attacks and protects the data confidentiality and integrity. The experimental results demonstrate the efficiency of CL-EKM in resource constrained WSNs.

REFERENCES

[1] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proc. IEEE Symp. SP*, May 2003, pp. 197–213.

[2] W. Du, J. Deng, Y. S. Han, and P. K. Varshney, "A key predistribution scheme for sensor networks using deployment knowledge," *IEEE Trans. Dependable Secure Comput.*, vol. 3, no. 1, pp. 62–77, Jan./Mar. 2006.

[3] W. Du, J. Deng, Y. S. Han, P. Varshney, J. Katz, and A. Khalili, "A pairwise key predistribution scheme for wireless sensor networks," *ACM Trans. Inf. Syst. Secur.*, vol. 8, no. 2, pp. 228–258, 2005.

[4] M. Rahman and K. El-Khatib, "Private key agreement and secure communication for heterogeneous sensor networks," *J. Parallel Distrib.Comput.*, vol. 70, no. 8, pp. 858–870, 2010.

[5] M. R. Alagheband and M. R. Aref, "Dynamic and secure key management model for hierarchical heterogeneous sensor networks," *IET Inf. Secur.*, vol. 6, no. 4, pp. 271–280, Dec. 2012.

[6] D. S. Sanchez and H. Baldus, "A deterministic pairwise key predistribution scheme for mobile sensor networks," in *Proc. 1st Int. Conf. SecureComm*, Sep. 2005, pp. 277–288.

[7] I.-H. Chuang, W.-T. Su, C.-Y. Wu, J.-P. Hsu, and Y.-H. Kuo, "Twolayered dynamic key management in mobile and long-lived clusterbased wireless sensor networks," in *Proc. IEEE WCNC*, Mar. 2007, pp. 4145–4150.

[8] S. Agrawal, R. Roman, M. L. Das, A. Mathuria, and J. Lopez, "A novel key update protocol in mobile sensor networks," in *Proc. 8th Int. Conf. ICISS*, vol. 7671. 2012, pp. 194–207.

[9] S. U. Khan, C. Pastrone, L. Lavagno, and M. A. Spirito, "An energy and memory-efficient key management scheme for mobile heterogeneous sensor networks," in *Proc. 6th Int. Conf. CRiSIS*, Sep. 2011, pp. 1–8.

[10] X. Zhang, J. He, and Q. Wei, "EDDK: Energy-efficient distributed deterministic key management for wireless sensor networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2011, pp. 1–11, Jan. 2011.

[11] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz, "Comparing elliptic curve cryptography and RSA on 8-bit CPUs," in *Proc. 6th Int. Workshop Cryptograph. Hardw. Embedded Syst.*, 2004, pp. 119–132.

[12] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Proc. 9th Int. Conf. ASIACRYPT*, vol. 2894. 2013,pp. 452–473.

[13] S. Seo and E. Bertino, "Elliptic curve cryptography based certificateless hybrid signcryption scheme without pairing," CERIAS, West Lafayette, IN, USA, Tech. Rep. CERIAS TR 2013-10, 2013.

[14] S. H. Seo, J. Won, and E. Bertino, "POSTER: A pairing-free certificatelesshybrid sign-cryption scheme for advanced metering infrastructures," in *Proc. 4th ACM CODASPY*, 2014, pp. 143–146.

[15] Q. Huang, J. Cukier, H. Kobayashi, B. Liu, and J. Zhang, "Fast authenticated key establishment protocols for self-organizing sensor networks," in *Proc. 2nd ACM Int. Conf. WSNA*, 2003, pp. 141–150.

[16] X.-J. Lin and L. Sun, "Cryptanalysis and improvement of a dynamic and secure key management model for hierarchical heterogeneous sensor networks," in *Proc. IACR Cryptol. ePrint Archive*, 2013, pp. 698–698.

[17] P. Szczechowiak, L. B. Oliveira, M. Scott, M. Collier, and R. Dahab, "NanoECC: Testing the limits of elliptic curve cryptography in sensor networks," in *Proc. 5th Eur. Conf. WSN*, vol. 4913. 2008, pp. 305–320.