Elliptic Curve based "r out of n" Key Distribution Scheme for Hierarchy Wireless Sensor Networks

C.Porkodi, K.Sangavai

***Abstract*— Abstract— Wireless Sensor Networks (WSNs) widely used in many sensitive fields like target tracking, battle field surviellance, intruder detection, health care and hostile environments. In such applications, the security of data transmission is an important issue and it is achieved through cryptographic algorithms. Key establishment is a primary security service in sensor networks and it facilitates sensor nodes to communicate securely with each other. In this paper, a 'r out of n" key distribution scheme based on elliptic curve cryptography is proposed. The computational overhead of the nodes in the network is reduced and the survival time of the network is extended in the proposed scheme as the base station is involved in the majority of the computations.**

***Index Terms*— Elliptic curves, Key distribution, Key management, polynomial interpolation Wireless sensor networks.**

I. INTRODUCTION

Wireless sensor networks WSN find application in various fields like industrial automation, automated and smart homes, video surveillance, traffic monitoring, medical device monitoring, monitoring of weather conditions, air traffic control, and robot control and so on. A wireless sensor network is consists of a large number of sensor nodes; each node is a small, inexpensive wireless device with limited battery power, memory storage, data processing capacity and short radio transmission range. In a Hierarchical WSN (HWSN), based on its capabilities each node assumes one of the responsibilities: base station, cluster head or sensor node. Sensitive data are collected and transmitted through sensor nodes to cluster heads deployed in hostile environments. Cluster heads have more resources than sensor nodes, and their objective is to collect and merge the readings from neighboring sensors, routing the resultant data to a base station. Base stations perform costly operations on behalf of constrained nodes and manage entire network.

The sensitive data are transmitted by the nodes, and hence there is a need to provide security measures confidentiality, integrity, authentication and availability. This can be attained through cryptographic algorithms and it involves mechanisms like key distribution and key management. Key management is a set of techniques that maintain the establishment and maintenance of keying relationships between authorized persons.

In recent years much research is focused on key distribution, and several schemes [1, 2, 3, 12] have been proposed. AtaUllah Ghafoor [4] proposed a key distribution scheme, in which the actual key is splitted into fragments and these fragments are transmitted through intermediate nodes to the receiver node and the actual key is constructed usng these key fragments and XOR operation. In the random key pre-distribution scheme proposed by L. Eschenauer, V.D. Gligor [5], each node randomly selects 'm' keys from a large key pool, such that any two sensor nodes will share at least one common key with high probability. A q-composite scheme key was developed by H. Chan et.al. [6] in which, any couple of nodes can create a symmetric key only if they share at least q starting keys. The two nodes execute a hash function on the concatenation of all shared starting keys and use the result as a new pairwise key. Ting Yuan et. al. [7] proposed a matrix-based random key pre-distribution scheme, which uses linear algebraic operations to derive common keys and resilient against node collusion.

W. Du et.al [2] developed a threshold-based key pre-distribution scheme in which a unique pairwise key is established between any pair of neighboring nodes. Sencun Zhu et al [8] developed a scheme that establish pair wise key between two nodes that is secure against a collusion attack up to a certain number of compromised nodes. Y. Zhou, Y. Zhang, Y. Fang [9] presented a location-based link-layer key establishment scheme, in which a hexagonal-grid-based deployment model and a polynomial-based key establishment model are combined to establish a link-layer key between two neighboring nodes. A group-based key establishment scheme was developed by L. Zhou, J. Ni, C.V. Ravishankar [10] to reduce the communication overhead of the scheme proposed in Chan et al. [6]. Parakh and Kak [11] applied a data partitioning scheme involving the roots of a polynomial in finite field in which the partitions are stored on randomly chosen servers on the network and they need to be retrieved to recreate the original data. Data reconstruction requires access to each server, login password and the knowledge of the servers on which the partitions are stored.

Now a days sensors become powerful in terms of CPU and memory power and thus elliptic curve cryptography (ECC) [13,14, 15] provides new opportunities to utilize public-key cryptography in sensor networks. In this paper, a key distribution scheme for HWSN based on ECC is developed. The base station selects a secret key at random and partitions it into n- parts and the partitions are preloaded into the nodes in the network. The scheme reduces the overhead of computing and storage of nodes, hence it saves the time and the energy of encrypting data. The scheme is appropriate for WSN and it extends the survival time of the network. The security of the scheme is based on the computational hard elliptic curve discrete logarithm problem. The rest of the paper is organized as follows: Section 2 describes the proposed key distribution scheme, section 3 discusses the security requirements and analysis, and finally conclusion is given.

Dr.C.Porkodi, Department of Mathematics, PSG College of Technology, Coimbatore, India

Dr.K.Sangavai, Department of Mathematics, PSG College of Technology, Coimbatore, India

II. PROPOSED KEY DISTRIBUTION SCHEME

In this section, key pool generation, key ring loading, generation of session key and message encryption and decryption for a particular transaction.

2.1 Key Pool Generation and Preloading

Assume that the network is composed of 'n+1' nodes, including base station (BS). BS is assigned the unique identity ID_{BS} and the remaining nodes N_i are assigned unique identies ID_i for i =1, 2, …, n respectively. BS selects an elliptic curve E (F_p): $y^2 \equiv x^3+ax+b \pmod p$ and a base point P of the elliptic curve E (F_p) of order q. BS randomly selects a polynomial $y = f(x) = a_0+a_1x + a_2x^2 + \cdots + a_rx^r$, with $r < n$, $a_i \in Z_q^*$ for i = 1, 2, …, n. The secret key of BS is $f(0) = a_0$. The secret key of the i^{th} node N_i is generated as (y_i, x_i), where

$$y_i = f(ID_i)(\mathrm{mod} q), x_i = \prod_{\substack{i,j=1 \\ i \neq j}}^{n} \frac{(0 - ID_j)}{(ID_i - ID_j)}\, y_i \ (mod q)$$

The nodes N_i for i = 1, 2, …n are preloaded with key pair (x_i, y_i) respectively. With 'r' or more than 'r' number of secret keys x_i's the secret key of BS is a_0 can be constructed as

$$a_0 = f(0) = \sum \left(\prod_{i \neq j} \frac{(0 - ID_j)}{(ID_i - ID_j)}\, y_i \ (mod q) \right)$$

using the 'r' or more than 'r' number of identities, y values and Lagrange interpolation formula the polynomial f(x) can be constructed as,

$$f(x) = \sum \left(\prod_{i \neq j} \frac{(x - ID_j)}{(ID_i - ID_j)}\, y_i \right)$$

2.2 Session key Generation, Message Encryption and Decryption

On receiving a query from a user, the BS broadcasts the query to all nodes in the network. This query packet traces the path information (through which nodes). The number of nodes in the path may be less than the degree of the polynomial r or may be greater than or equal to r. Key generation, message encryption and decryption for the two cases are discussed in detail below. Suppose that k^{th} node N_k desires to return the message point $M \in E(F_p)$ to the BS, it returns an acknowledgement packet (Ack) and places a request for session key.

Case 1: The number of nodes in the path $\geq r$

BS computes a part of the session key as follows.

- BS randomly selects a point Q on E(F_p), computes a_0Q and transmits a_0Q to the k^{th} node N_k via the same path.
- Node N_k encrypts the message point M as $C_{k,1} = x_kP + a_0Q + M$ and encrypts the hash value of the message H(M) as $C_{k,2} = y_kH(M)$. The node transmits this ciphertext $(C_{k,1}, C_{k,2})$ to the adjacent node say N_j in the path from node N_k to BS.
- The node N_j encrypts the received cipher text $(C_{k,1}, C_{k,2})$ as $(C_{j,1}, C_{j,2})$, where $C_{j,1} = x_jP + C_{k,1}, C_{j,2} = y_j(C_{k,2})$. In turn, $C_{j,1} = x_jP + x_kP + a_0Q + M$, $C_{j,2} = y_j(y_kH(M))$. The node N_j transmits $(C_{j,1}, C_{j,2})$ to its adjacent node in the path.
- In this way, the encryption of the plain text M and the hashvalue H(M) is done by all nodes in the path from node N_k to the base station BS. Finally the cipher text received by BS is (C_1, C_2), where

$$C_1 = \left(\sum_{N_i epath\ from\ N_k to\ BS} x_i \right) P + a_0Q + M$$

$$C_2 = \left(\prod_{N_i epath\ from\ N_k to\ BS} y_i \right) (H(M))$$

The decryption is done by the BS as follows.

- BS computes the message, base station computes $C_1 - a_0P - a_0Q$ to get the plain text M.

$$C_1 - a_0P - a_0Q$$

$$= \left(\sum_{N_i epath\ from\ N_k to\ BS} x_i \right) P + a_0Q + M - a_0P - a_0Q$$

$$= a_0P + a_0Q + M - a_0P - a_0Q = M$$

Since,

$$\sum_{N_i epath\ from\ N_k to\ BS} x_i = a_0$$

- BS computes the hash value of the decrypted message and validates the data integrity by verifying the equation

$$\left(\prod_{N_i epath\ from\ N_k to\ BS} y_i \ mod q \right) (H(M)) = C_2$$

Case 2: The number of nodes in the path say $m < r$

BS computes a part of the session key as follows.

- BS randomly selects a point Q on E(F_p), computes a_0Q. Also randomly selects (r-m) secret pairs (y_i, x_i) of the nodes those do not lie on this path, computes and transmits the partial session key $k_{partial\ session}$

$$k_{partial\ session} = \left(\sum_{for\ (r-m) nodes\ not\ in\ the\ path} x_i mod q \right) P + a_0Q$$

to the k^{th} node N_k via the same path.

- Node N_k encrypts the message point M as $C_{k,1} = x_kP + k_{partial\ session} + M$ and encrypts the hash value of the message H(M) as $C_{k,2} = y_kH(M)$. The node transmits this ciphertext $(C_{k,1}, C_{k,2})$ to the adjacent node say N_j in the path from node N_k to BS.
- The node N_j encrypts the received cipher text $(C_{k,1}, C_{k,2})$ as $(C_{j,1}, C_{j,2})$, where $C_{j,1} = x_jP + C_{k,1}, C_{j,2} = y_j(C_{k,2})$. In turn, $C_{j,1} = x_jP + x_kP + a_0Q + M$, $C_{j,2} = y_j(y_kH(M))$. The node N_j transmits $(C_{j,1}, C_{j,2})$ to its adjacent node in the path.
- In this way, the encryption of the plain text M and the hashvalue H(M) is done by all nodes in the path from node N_k to the base station BS. Finally the cipher text received by BS is (C_1, C_2), where

$$C_1 = \left(\sum_{N_i epath\ from\ N_k to\ BS} x_i \right) P + k_{partial\ session} + M$$

$$C_2 = \left(\prod_{N_i epath\ from\ N_k to\ BS} y_i \right) (H(M))$$

- The decryption is done by the BS as follows.

BS computes the message, base station computes $C_1 - a_0P - a_0Q$ to get the plain text M.

$$C_1 - a_0P - a_0Q$$

$$= \left(\sum_{N_i epath\ from\ N_k to\ BS} x_i \right) P + k_{partial\ session} + M - a_0P - a_0Q$$

$$= \left(\sum_{N_i epath\ from\ N_k to\ BS} x_i \right) P$$

$$+ \left(\sum_{for\ (r-m) nodes\ not\ in\ the\ path} x_i\ mod q \right) P$$

$$+ a_0Q + M - a_0P - a_0Q$$

$$= a_0P + a_0Q + M - a_0P - a_0Q = M$$

In the computation of C_1, totally r-secret keys are involved and thus these secret keys constructs a_0 implicitly.

- To validate data integrity, BS computes H(M) and verifies whether the equation

$$\left(\prod_{N_i epath\ from\ N_k to\ BS} y_i\ mod q \right) (H(M)) = C_2$$

III. SECURITY ASPECTS AND ANALYSIS

In this section, we discuss how the proposed scheme satisfies the basic security aspects, namely data confidentiality, data integrity, data authenticity and availability.

3.1 Confidentiality

The implicit security is used to generate key partitions and with the knowledge of at least 'r out n' these secret key partitions a session key between the base station and the node can be created. The session key is constructed using elliptic curves and it is used for encryption. As the data is encrypted before transmission, the privacy of the message is maintained. For every transaction, as the base station randomly selects a point Q, the session key is uniquely is generated for each transaction even though the path contains same nodes. So chosen cipher text attack is not possible.

3.2 Data Integrity

The proposed scheme addresses the unauthorized alteration of data in the transaction. The attackers can modify the query packet; even they do not know the content of the packet. The cipher text along with signature of the messages C_2 provides data integrity.

3.3 Authentication and Availability

The proposed service confirms, the source of data is from a legitimate node, the network is survival and the data is available. The base station knows the topology of the network and it records the key partitions of all nodes. To validate the authentication of the nodes, the base station performs challenge response protocol discussed below.

Challenge response Protocol

- BS randomly selects a "challenge" c and broadcasts in a time interval
- Each i^{th} node signs 'c' with its own secret key y_i, s_i:=Sign(y_i, c), and sends the "response" s_i to the base station.
- BS accepts i^{th} node's identity, if Verify (s_i, y_i, c) = ok.

BS can ensure the node is not a forgery node and observe the survival status of nodes of the network.

3.4 Security Analysis

The computationally hard elliptic curve discrete logarithm problem (ECDLP) plays an important role in the security of the proposed scheme. Because of ECDLP from the partial session key a_0Q key it is computationally infeasible to find the value of the secret key a_0 . In the worst case, suppose if atleast r out of n nodes of the network except base station are compromised, ie. If an attacker knows the secret keys x_i for atleast r-nodes it is feasible to find BS's secret key a_0. So an attacker cannot impersonate as a base station unlesss otherwise he/she can capture r out of n nodes.

By observing the partial session keys of two transactions $y_1 = a_0P_1$ and $y_2 = a_0P_2$ an attacker may try recover the secret key of the BS a_0. To achieve such an attack, he/she has to solve the ECDLP, i.e. from $y_1 - y_2 = a_0(P_1 - P_2)$ he/she has to recover a_0.

For every transaction to generate the session key, the BS selects randomly a point Q from $E(F_p)$ of order q with probability 1/q. Thus the probability for generating the same session key for two different transactions is 1/q, provided the path information remains the same for both transactions.

3.5 Performance analysis

A comparison of key sizes for various cryptosystems providing the same level of security against brute-force attacks. is given below in Table 1. Currently cryptographic protocols apply asymmetric algorithms such as RSA and ECC because of their flexibility and enhanced ability to manage keys. ECC is considered to be suitable technique for WSN which provides a good tradeoff between key size and security.

SECURITY BITS	SYMMETRIC ENCRYPTION ALGORITHM	MINIMUM SIZE (BITS) OF PUBLIC KEYS		
		RSA	ECC	KEY SIZE RATIO OF RSA VS ECC
80	SKIPJACK	1024	160	6:1
112	3DES	2048	224	9:1
128	AES-128	3072	256	12:1
192	AES-192	7680	384	20:1
256	AES-256	15360	512	30:1

Table 1. Security Comparison for Various Algorithm-key Size Combinations

In cryptographic algorithms, to provide high security generally key lengths are increased over time as the computation available to attackers is increased. If elliptic curves are used for the key management (i.e. the encryption/decryption session key) of an AES-256 session, then a 512-bit elliptic curve session key would be required. To attain the same level of security with RSA encryption, 15,360 bit keys are required, which is computationally infeasible in embedded systems today. This key comparison shows that ECC is best suitable for embedded systems.

From the table1, it is obvious that the elliptic curve cryptosystem requires a considerably shorter key and offer the same level of security as RSA which need much larger keys. Generally the performance of RSA is reported to be 10 times slower than ECC at 128-bit security levels and 50 to 100 times slower at 256-bit security levels for secret key operations such as signature generation or key management. Also public keys and signatures are 6 times larger and private keys are 12 times larger for RSA than for ECC at 128-bit security level. The key size is significantly important in the cost of secure storage of the keys. As the sensor nodes are resource constrained energy wise, bandwidth wise, storage and so on, the proposed key distribution scheme based on ECC is best suitable for HWSN than the exponentiation based schemes. Presently most of the cryptographic protocols utilize ECC because is considered to be best suitable for embedded devices cost wise and performance wise.

IV. Conclusion

In this paper, a 'r out of n' key management scheme based on ECC is proposed for a HWSN and is implemented using matlab. The proposed scheme provides all basic internal security aspects, namely privacy, data integrity, authentication and availability in a network. As the underlying principle of ECC is based on the computational hard discrete logarithm problem, the secret key for data transmission cannot be attained by an attacker. ECC is utilized for key management, further reduction in storage space, computational overheads; power consumption could be achieved because of its shorter key length.

References

[1] N. Canh, Y.-K. Lee, S. Lee, HGKM: a group-based key management scheme for sensor networks using deployment knowledge, Proceedings of the Sixth Annual Communication Networks and Services Research Conference (CNSR'08), IEEE Computer Society, Los Alamitos, CA, USA, 2008, pp. 544–551.

[2] W. Du, J. Deng, Y. Han, P. Varshney, J. Katz, A. Khalili, A pairwise key pre-distribution scheme for wireless sensor networks, Proceedings of the 10th ACM conference on Computer and communications security (CCS'03), ACM, New York, NY, USA, 2003, pp. 42–51.

[3] Filippo Gandino, Renato Ferrero, Maurizio Rebaudengo, A Key Distribution Scheme for Mobile Wireless Sensor Networks: *q*-*s*-Composite, IEEE Transactions on Information Forensics and Security, Vol. 12, NO. 1, Jan 2017, pp.34-47.

[4] AtaUllah Ghafoor, Muhammad Sher, Muhammad Imran, and Abdelouahid Derhab, Secure Key Distribution Using Fragmentation and Assimilation in Wireless Sensor and Actor Networks, International Journal of Distributed Sensor Networks, Vol. 2015, pp. 1–15.

[5] L. Eschenauer, V.D. Gligor, A key-management scheme for distributed sensor networks, Proceedings of the 9th ACM Conference on Computer and Communications Security, November 2002.

[6] H. Chan, A. Perrig, D. Song, Random key pre-distribution schemes for sensor networks, Proceedings of IEEE Symposium on Security and Privacy, Berkeley, California, 2003, pp. 197–213.

[7] Ting Yuan, Shiyong Zhang, Yiping Zhong, A Matrix-Based Random Key Pre-distribution Scheme for Wireless Sensor Networks, 7th IEEE International Conference on Computer and Information Technology, 2007, pp 33-38.

[8] Sencun Zhu, Shouhuai Xu, Sanjeev Setia, Sushil Jajodia, Establishing pair-wise keys for secure communication in ad hoc networks: a probabilistic approach, Proceedings of 11th IEEE International Conference on Network Protocols, November 2003, pp.326-335.

[9] Y. Zhou, Y. Zhang, Y. Fang, LLK: a link layer key establishment scheme in wireless sensor networks, Proceedings of IEEE Wireless Communications and Networking Conference, March 2005, pp. 29–42.

[10] L. Zhou, J. Ni, C.V. Ravishankar, Efficient key establishment for group-based wireless sensor deployments, Proceedings of the 4th ACM Workshop on Wireless security, September 2005, pp. 1–10.

[11] A. Parakh and S. Kak, "Online data storage using implicit security, "Information Science, vol. 179, no. 19, pp.3323-3331, Sep. 2009.

[12] Walid Bechkit, Yacine Challal, Abdelmadjid Bouabdallah, and Vahid Tarokh, A Highly Scalable Key Pre-Distribution Scheme for Wireless Sensor Networks, IEEE Transactions on Wireless Communications, Vol. 12, No. 2, 2013, pp.948-959.

[13] N. Koblitz, Elliptic Curves Cryptosystems," Mathematics of computation, vol. 48, pp. 203-209, 1987.

[14] A. Liu and P. Ning, "TinyECC: a configurable library for elliptic curve cryptography in wireless sensor networks," Proc. IPSN 2008, Washington, DC, Apr. 2008, pp. 245-256

[15] A.J. Menezes, P.C. van Oorschot, S.A. Vanstone, Handbook of Applied Cryptography, CRC Press, Boca Raton, FL, 1999.

Dr.C.PORKODI received her M.Sc. degree in Mathematics from Bharathiar University, Coimbatore, Tamil Nadu, India in 1992, and M.Phil degree in Mathematics from Madurai Kamaraj University, Madurai, Tamil Nadu, India in 2002. She is a rank holder in BSc and M.Sc degree courses. She completed her Ph.D degree in Mathematics with specialization "Cryptography", from Anna University, Chennai India in 2010. She has been serving as a faculty member in the Department of Mathematics, PSG College of Technology, India since 2000. Her research interest includes Number theory, Cryptography, and Wireless Sensor Networks. Based on her present work, she has published seven papers in the International Journals.

Dr. K.SANGAVAI received her M.Sc. degree in Mathematics from Bharathidasan University, Trichy Tamil Nadu, Indiain 1992 and M.Phil degree in Mathematics from Anna University, Chennai, Tamil Nadu, India in 1993. She completed her Ph.D degree in Applied Mathematics with specialization "Graph Theory", from Bharathiar University, Coimbatore, Tamil Nadu, India in 2012.She is a rank holder in M.Sc degree course. She has been serving as a faculty member in the Department of Mathematics, PSG College of Technology, India, since 1999. Her research interest includes Graph Theory, Networks and Algorithms, Applications of Graph Theory and Statistics in Wireless Sensor Network. Based on her present work, she has published seven papers in International Journals and three papers in conferences.