

A Security Framework for Wireless Sensor Networks

Tanveer Zia and Albert Zomaya
School of Information Technologies
University of Sydney
Madsen Building F09, Camperdown NSW 2006
Email: {tanzia, zomaya}@it.usyd.edu.au

Abstract – Wireless sensor networks are result of developments in micro electro mechanical systems and wireless networks. These networks are made of tiny nodes which are becoming future of many applications where sensor networks are deployed in hostile environments. The deployment nature where sensor networks are prone to physical interaction with environment and resource limitations raises some serious questions to secure these nodes against adversaries. The traditional security measures are not enough to overcome these weaknesses. To address the special security needs of tiny sensor nodes and sensor networks as a whole we introduce a security framework. In our framework we emphasize on three areas: (1) cluster formation (2) secure key management scheme, and (3) a secure routing algorithm. Our security analysis shows that the framework presented in this paper meets the unique security needs of sensor networks.

Keywords – Wireless sensor networks security, secure key management, secure routing.

I. INTRODUCTION

Advancements in micro electro mechanical systems (MEMS) and wireless networks have made possible the advent of tiny sensor nodes called “smart dust” which are low cost small tiny devices with limited coverage, low power, smaller memory sizes and low bandwidth. Wireless sensor networks are consisting of large number of sensor nodes which are becoming viable solution to many challenging domestic, commercial and military applications. Sensor networks collect and disseminate data from the fields where ordinary networks are unreachable for various environmental and strategically reasons.

In addition to common network threats, sensor networks are more vulnerable to security breaches because they are physically accessible by possible adversaries, consider sensitive sensor network applications in military and hospitals compromised by adversaries. Many developments have been made in introducing countermeasures to potential threats in sensor networks; however, sensor network security remains less addressed area. In this paper we present a security framework for wireless sensor networks to provide desired security countermeasures against possible attacks. Our security framework consists of three interacting phases: cluster formation, secure key management and secure routing schemes.

We make three contributions in this paper:

- We discuss cluster formation and leader election in a multihop hierarchical cluster model
- We present a secure key management scheme
- We propose a secure routing mechanism which addresses potential threats in node to cluster leader and cluster leader to base station and vice versa communication.

The rest of paper is organized as follows. Section II provides summary of related work in key management and routing protocols in wireless sensor networks. Section III presents our security framework discussing the cluster formation and leader election process, secure key management scheme, secure routing and their algorithms. Section IV provides analysis of our security framework, and finally in Section V we conclude our paper providing the future research directions.

II. RELATED WORK

Researchers have addressed many areas in sensor network security. Some of the related work has been summarized in the following paragraphs.

Eschenauer et al. [1], present a probabilistic key pre-distribution scheme where each sensor node receives a random subset of keys from a large key pool before deployment. To agree on a key for communication, two nodes find one common key within their subsets and use that key as their shared key.

Chan et al [2], extended idea of Eschenauer et al. [14] and developed three key pre-distribution schemes; q-composite, multipath reinforcement, and random-pairwise keys schemes.

Pietro et al [3], Present a random key assignment probabilistic model and two protocols; ‘direct and cooperative’ to establish a pairwise communication between sensors by assigning a small set of random keys to each sensor. This idea later converges to pseudo random generation of keys which is energy efficient as compare to previous key management schemes.

Liu et al [4] propose a pairwise key schemes is based on polynomial pool-based and grid based key pre-distribution

schemes have high resilience against node captures and communication overhead.

Du et al [5] pairwise key pre-distribution is an effort to improve the resilience of the network by lowering the initial payoff of smaller scale network attacks and pushes adversary to attack at bigger scale to compromise the network.

Du et al [6] present a key scheme based on deployment knowledge. This key management scheme takes advantage of the deployment knowledge where sensor position is known prior to deployment. Because of the randomness of deployment, it is not feasible to know the exact neighbor locations, but knowing the set of likely neighbors is realistic, this issue is addressed using the random key pre-distribution of Eschenauer et al.

Adrian et al [7] have introduced SPINS (Security Protocols for Sensor Networks). SPINS is a collection of security protocols (SNEP) and mirco-TESLA. SNEP (Secure Network Encryption Protocol provides data confidentiality and two-way data authentication with minimum overhead. Micro-TESLA, a micro version of TESLA (Time Efficient Streamed Loss-tolerant Authentication) provides authenticated streaming broadcast.

SPINS leaves some questions like security of compromised nodes, DoS issues, network traffic analysis issues. Furthermore, this protocol assumes the static network topology ignoring the ad hoc and mobile nature of sensor nodes.

Chen et al [8] proposed two security protocols. First, *base station to mote confidentiality and authentication* which states that an efficient shared-key algorithm like RC5 be used to guarantee the authenticity and privacy of information. Second, *Source authentication*, by implementing a hash chain function similar to that used by TESLA (timed efficient stream loss-tolerant authentication) to achieve mote authentication.

Jeffery et al [9] proposed a light weight security protocol that operates in the base station of sensor communication where base station can detect and remove an aberrant node if it is compromised.

This protocol does not specify any security measures in case of any passive attacks on node where an adversary is intercepting the communication.

III. THE SECURITY FRAMEWORK

Our security framework consists of three interacting phases: cluster formation, secure key management and secure routing.

A. Cluster formation

As soon as sensor nodes are deployed, they broadcast their ID's and listens to the neighbors, add the neighbor ID's in its routing table and count the number of neighbor it could listen to. Hence these connected neighbors become a

cluster. Each cluster elects a sensor node as a leader. All inter-cluster communication is routed through cluster leaders. Cluster leaders also serve as fusion nodes to aggregate packets and send them to the base station. The cluster leader receives highest number of messages, this role changes after reaching an energy threshold, hence giving opportunity to all the nodes becoming a cluster leader when nodes move around in a dynamic environment. Coverage of clusters depends on the signal strength of the cluster leader. Cluster leader and its neighbor nodes form a parent-child relationship in a tree-based network topology. In this multi hop cluster model, data is collected by the sensor nodes, aggregated by the cluster leader and forwarded to the next level of cluster, eventually reaching the base station. Figure 1 below shows a network of 200 sensor nodes forming 10 clusters.

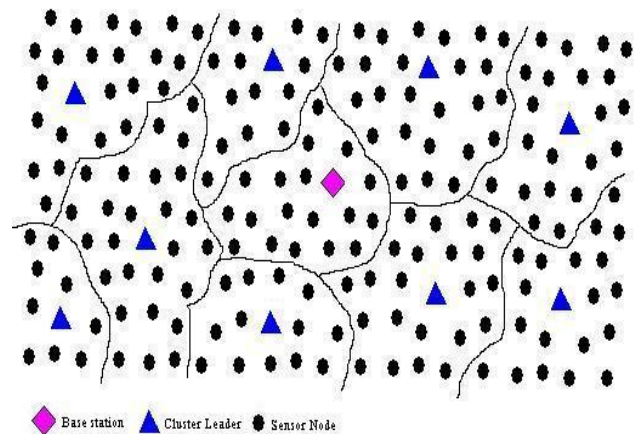


Fig 1: Cluster formation

B. Secure key management scheme

Key management is critical to meet the security goals of confidentiality, integrity and authentication to prevent the Sensor Networks being compromised by an adversary. Due to ad-hoc nature and resource limitations of sensor networks, providing a right key management is challenging. Traditional key management schemes based on trusted third parties like a certification authority (CA) are impractical due to unknown topology prior to deployment. Trusted CA is required to be present all the times to support public key revocation and renewal [10]. Trusting on a single CA for key management is more vulnerable, a compromise CA will risk the security of entire sensor network. Fei et al [10] decompose the key management problem into:

Key pre-distribution – installation of keys in each sensor node prior to distribution

Neighbor discovery – discovering the neighbor node based on shared key

End-to-end path key establishment – end to end communication with those nodes which are not directly connected

Isolating aberrant nodes – identifying and isolating damaged nodes.

Re-keying – re-keying of expired keys

Key-establishment latency – reducing the latency resulted from communication and power consumption.

The core problem we realize in wireless sensor network security is to initialize the secure communication between sensor nodes by setting up secret keys between communicating nodes. In general we call this *key establishment*. There are three types of key establishment techniques [5, 6]: trusted-server scheme, self enforcing scheme, and key pre-distribution scheme. The trusted server scheme depends on a trusted server e.g., Kerberos [11]. Since there is no trusted infrastructure in sensor networks, therefore trusted-server scheme is not suitable in this case. The self-enforcing scheme depends on asymmetric cryptography using public keys. However, limited computation resources in sensor nodes make this scheme less desirable. Public key algorithms such as Diffie-Hellman [12] and RSA [13] as pointed out in [6, 7] require high computations resources which tiny sensors does not provide. The key pre-distribution scheme, where key information is embedded in sensor nodes before the nodes are deployed is more desirable solution for resource starved sensor nodes. A simple solution is to store a master secret key in all the nodes and obtain a new pairwise key. In this case capture of one node will compromise the whole network. Storing the master key in tamper resistant sensor nodes increases the cost and energy consumption of sensors. Another key pre-distribution scheme [5] is to let each sensor carry $N - 1$ secret pairwise keys, each of which is known only to this sensor and one of the other $N - 1$ sensors (N is the total number of sensors). Extending the network makes this technique impossible as existing nodes will not have the new nodes keys.

In our security framework we introduce a secure hierarchical key management scheme where we use three keys: two pre-deployed keys in all nodes and one in network generated cluster key for a cluster to address the hierarchical nature of sensor network.

K_n (network key) – Generated by the base station, pre-deployed in each sensor node, and shared by the entire sensor network. Nodes use this key to encrypt the data and pass onto next hop.

K_s (sensor key) – Generated by the base station, pre-deployed in each sensor node, and shared by the entire sensor network. Base station uses this key to decrypt and process the data and cluster leader uses this key to decrypt the data and send to base station.

K_c (cluster key) – Generated by the cluster leader, and shared by the nodes in that particular cluster. Nodes from a cluster use this key to decrypt the data and forward to the Cluster Leader.

By providing this key management scheme we make our security framework resilient against possible attacks on the sensor network.

In this key management scheme base station uses K_n to encrypt and broadcast data. When a sensor node receives the message, it decrypts it by using its K_s . In this key calculation, base station uses $K_{n1..nn}$ to broadcast the message. This process follows as: Base station encrypts its own ID, a current time stamp TS and its K_n as a private key. Base station generates a random seed S and assumes itself at level 0. Sensor node decrypts the message received from the base station using K_s .

When a node sends a message to cluster leader, it constructs the message as follows:

$$\{ID, K_s, TS, MAC, S (\text{message})\}$$

Cluster leader checks the ID from the packet, if the ID in the packet matches the ID it holds, verifies the authentication and integrity of the packet through MAC. Otherwise, packet is dropped by the cluster leader. Node builds the message using the fields below:

Cluster leader aggregates the messages received from its nodes and forwards it to next level cluster leader or if the cluster leader is one hop closer to the base station, it directly sends to the bases station. Receiving cluster leader checks its routing table and constructs the following packet to be sent to next level cluster leader or base station. Cluster leader adds its own ID, its network and cluster key in incoming packet and rebuilds the packet as under:

$$\{ID, K_n, k_c, [ID, K_s, TS, MAC, S (\text{Aggr message})]\}$$

Here ID is the ID of receiving cluster leader which aggregates and wraps the message, and sends it to the next hop cluster leader or to the base station if directly connected. Next hop cluster leader receives the packet and checks the ID, if the ID embedded in the packet is same as it holds, it updates the ID for the next hop and broadcast it, else the packet is discarded.

Base station receives the packet from its directly connected cluster leader; it checks the ID of sending cluster leader, verifies the authentication and integrity of the packet through MAC. Cluster leader directly connected with base station adds its own ID along with the packet received from the sending cluster leader. Packet contains the following fields:

$$\{ID[ID, K_n, k_c, [ID, K_s, TS, MAC, S (\text{Aggr message})]]\}$$

C. Secure Routing

In our secure routing mechanism, all the nodes have a unique ID#. Once the network is deployed, base station builds a table containing ID#s of all the nodes in the network. After self organizing process base station knows the topology of the network. Using our secure key management scheme nodes collect the data, pass onto the cluster leader which aggregates the data and sends it to the

base station. We adapt the energy efficient secure data transmission algorithms by [15] and modify it with our secure key management scheme to make it more resilient against attacks in wireless sensor networks. Following two algorithms: sensor node and base station algorithms are presented for secure data transfer from node to base station and base station to node communication:

Node algorithm performs the following functions:

- Sensor nodes use the K_n to encrypt and transmit the data
- Transmission of encrypted data from nodes to cluster leader
- Appending ID# to data and then forwarding it to higher level of cluster leaders
- Cluster leader uses K_c to decrypt and then uses its K_n to encrypt and send the data to next level of cluster leaders, eventually reaching the base station

Base station algorithm is responsible of following tasks:

- Broadcasting of K_s and K_n by the base station
- Decryption and authentication of data by the base station

Node algorithm

- Step 1: If sensor node i wants to send data to its cluster leader, go to step 2, else exit the algorithm
- Step 2: Sensor node i requests the cluster leader to send the K_c .
- Step 3: Sensor node i uses K_c and its own K_n to compute the encryption key $K_{i,cn}$.
- Step 4: Sensor node i encrypts the data with $K_{i,cn}$ and appends its ID# and the TS to the encrypted data and then sends them to the cluster leader.
- Step 5: Cluster leader receives the data, appends its own ID#, and then sends them to the higher-level cluster leader or to the base station if directly connected. Go to Step 1.

Base Station Algorithm

- Step 1: Check if there is any need to broadcast the message. If so, broadcast the message encrypting it with K_n .
- Step 2: If there is no need to broadcast the message then check if there is any incoming message from the cluster leaders. If there is no data being sent to the base station go to step 1.
- Step 3: If there is any data coming to the base station then decrypt the data using K_s , ID# of the node and TS within the data.
- Step 4: Check if the decryption key K_s has decrypted the data perfectly. This leads to check the credibility of the TS and the ID#. If the decrypted data is not perfect discard the data and go to step 6.

Step 5: Process the decrypted data and obtain the message sent by sensor nodes

Step 6: Decides whether to request all sensor nodes for retransmission of data. If not necessary then go back to step 1.

Step 7: If a request is necessary, send the request to the sensor nodes to retransmit the data. When this session is finished go back to step 1.

Flow chart below in figure 4 illustrates the base station to node algorithm:

This routing technique provides stronger resilience towards spoofed routing information, selective forwarding, sinkhole attacks; Sybil attacks wormholes and HELLO flood attacks presented in [16].

Flow chart below in figure 2 illustrates the base station to node algorithm:

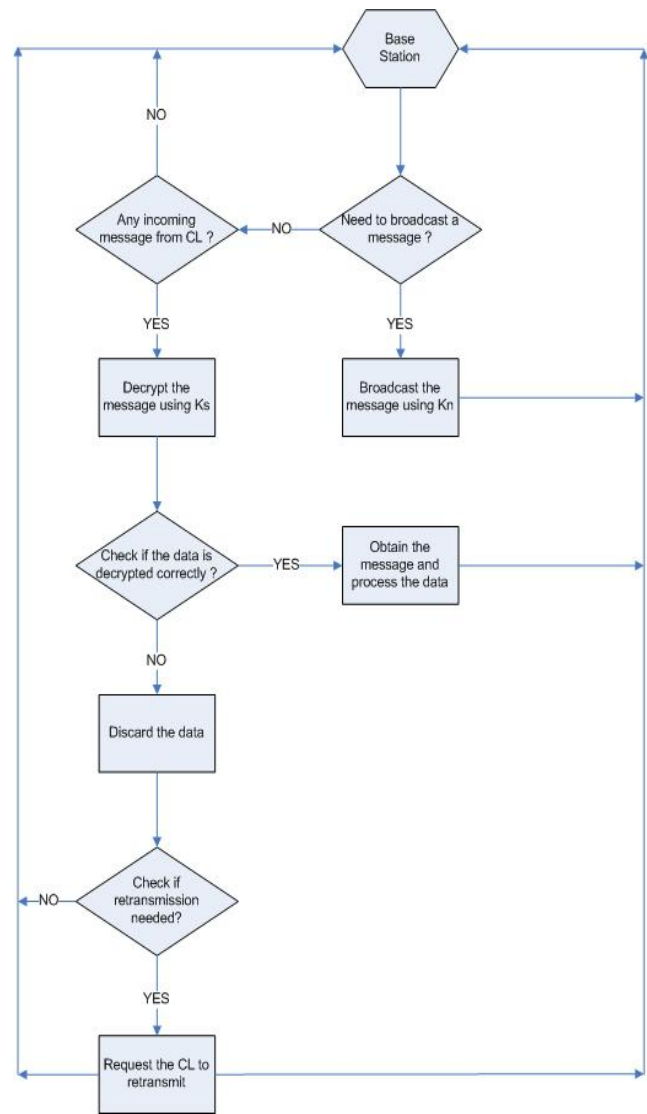


Fig 2: Base station to node communication

IV. ANALYSIS OF PROPOSED FRAMEWORK

This section presents an analysis to explain the features of our security framework which make this framework feasible to implement.

In our security framework packet format in a typical node to cluster leader communication would be as under:

IDs	Keys	TS	S	Data	MAC
(3)	(3)	(1)	(1)	(0..31)	(4)

This gives us 44 bytes of data packet to transmit. Taking into account 128K program memory of ATmega128L MICA2Dot our framework can be best implemented in a network of up to 3000 sensor nodes. Going beyond this number we might need to have a tradeoff between the security and performance which is highly unlikely because most of the applications so far do not deploy sensor nodes at that large quantity. Assuming the ongoing developments in enhancing the program memory this framework will be feasible in even larger and denser networks.

The algorithms presented in this framework takes into consideration the nodes and cluster leaders which are not participating in sending and aggregating the data. These nodes forward the data packets without applying any further cryptographic operation, thus further saving the processing power and memory.

V. CONCLUSION AND FUTURE WORK

In this paper we have presented a security framework for wireless sensor network which is composed of three phases: cluster formation, secure key management scheme and secure routing. Cluster formation process has described the topology formation and self organization of sensor nodes, leader election and route selection towards base station. We have presented a hierarchical secure key management scheme based on three levels of pre-deployed keys and lastly we have presented a secure routing mechanism which provides a stronger resilience towards susceptible attacks on sensor networks. We plan to implement this security framework in Berkeley's motes having confidence that this framework will provide added security in wireless sensor network communication.

REFERENCES

- [1] L. Eschenauer and V. Gligor, "A Key-management Scheme for Distributed Sensor Networks", Proceedings of the 9th ACM conference on Computer and Communication Security 2002, Washington DC, USA
- [2] P. Ganesan, R. Venugopalan, P. Peddabachagari, A. Dean, F Mueller, and M Sichitiu, "Analyzing and Modeling Encryption Overhead for Sensor Network Nodes", WSNA'03, September 19, 2003, San Diego, California, USA
- [3] R. Pietro, L. Mancini, and A. Mei, "Random key-Assignment for Secure Wireless Sensor Networks", ACM SANS 2003.
- [4] D. Liu and P. Ning, "Establishing Pairwise Keys in Distributed Sensor Networks", ACM CCS 2003.
- [5] W. Du, J. Deng, Y. S. Han, and P. K. Varshney, "A Pairwise Key Pre-Distribution Scheme for Wireless Sensor Networks", ACM CCS 2003.
- [6] W. Du, J. Deng, Y. S. Han, S. Chen, and P. K. Varshney, "A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge", IEEE InfoCom 2004.
- [7] A. Perrig, R. Szewczyk, V. Wen, D. Culler, J. D. Tygar. SPINS: Security Protocols for Sensor Networks, in Wireless Networks Journal (WINE), September 2002.
- [8] H. Chan, A. Perrig, and D. Song, "Random Key Predistribution Schemes for Sensor Networks". In Proceedings of the IEEE Symposium on Security and Privacy, Oakland, California USA
- [9] J. Undercoffer, S. Avancha, A. Joshi, and J. Pinkston, "Security for Sensor Networks" 2002 CADIP Research Symposium
- [10] F. Hu, J. Ziobro, J. Tillett, and N. Sharma, "Wireless Sensor Networks: Problems and Solutions" Rochester Institute of Technology, Rochester, New York USA.
- [11] B. C. Neuman and T. Tso., "Kerberos: An authentication service for computer networks. IEEE communications 32(9):pgs33-38, 1994.
- [12] W. Diffie and M. E. Hellman, "New directions in cryptography. IEEE transactions on information theory, 22:644-654, 1976.
- [13] R. L. Rivest, A. Shamir, and L. M. Adleman, "A method for obtaining digital signatures and public key cryptosystems. Communications of the ACM, 21(2):120-126, 1978
- [14] T. Li, H. Wu and F. Bao, "SenSec Design", Institute for Infocomm research, Singapore, 2004
- [15] H. Cam, S. Ozdemir, D. Muthuavinashiappan, and P. Nair, "Energy Efficient Security Protocol for Wireless Sensor Networks", 2003 IEEE
- [16] C. Karlof and D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures", University of California at Berkeley, USA 2003.