무선 센서 네트워크망에서의 효율적인 키 관리 프로토콜 분석

김정태

목원대학교

Analyses of Key Management Protocol for Wireless Sensor Networks in Wireless Sensor Networks

Jung-Tae Kim

Mokwon University

E-mail : jtkim3050@mokwon.ac.kr

요 약

In this paper, we analyses of Key Management Protocol for Wireless Sensor Networks in Wireless Sensor Networks. Wireless sensor networks have a wide spectrum of civil military application that call for security, target surveillance in hostile environments. Typical sensors possess limited computation, energy, and memory resources; therefore the use of vastly resource consuming security mechanism is not possible. In this paper, we propose a cryptography key management protocol, which is based on identity based symmetric keying.

I. Introduction

The demand for personal communication systems, including cellular phones and cordless access services, has recently been growing rapidly, not only in Korea, but also throughtout the world. To support global mobility, roaming service must be provided both in the home network and in the roamed network. Mobility is a function that enables a user to move inside and around networks, and the network providing network this function called the mobility network. The mobility network includes the GSM, USDC and PDC systems, and the FDMA system is used in more than 70 countries around the world and helps to promote globalization. While the exiting mobility network expands, the network architecture technology for the personal communication systems capable of user's global mobility is also advancing dramatically/ The universal personal telecommunication and the FPLMTS(future public land mobile telecommunication systems), based on intelligent network technology, are being studied energetically in order to provide personal communication service flexibly and effectively.

II. Sensor Network Architecture

We adopt the sensor network model proposed by Younis. In this model, a sensor network consists of a large number of sensors distributed over an area of interest. There is a command node in charge of the networks mission. The model also introduces super nodes called gateways, in addition to the sensor nodes. The gateways have considerably high energy resources compared to the sensors, and are equipped with high performance

processors and more memory. As shown in Figure 1, the gateways partition the sensors into distinct clusters, using a clustering algorithm. Each cluster is compared of a gateway and a set of sensor nodes, which gather information and transmit to the gateway of their cluster. The gateway fues the data from the different sensors, performs mission-related data processing, and sends to the command node via long-haul transmission.

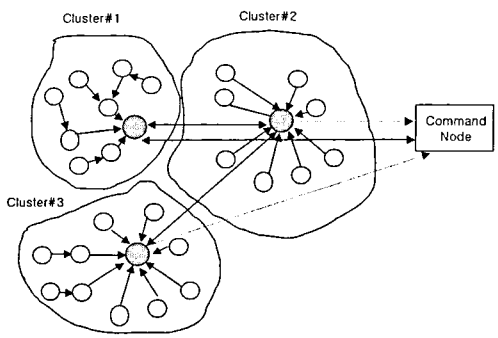

Fig. Multi-gateway, clustered sensor network

III. System Model

Base stations interface sensor network to the outside network. Sensor nodes are assumed to be immobile and also they do not have a specific architecture when deployed over a specific geographic area, based on self-organizing clustering techniques. A cluster head is chosen from each cluster to handle the communication between the cluster nodes and the base station. Sensor nodes are assigned a secret key(Ki) and a unique ID number that identifies itself in the network. As wireless transmission is not completely trustable, assigning secret keys to sensor IDs are assigned during the manufacturing phase. Base station is then given all the ID numbers and K_i used in the network before the deployment of network. Having a complete list of sensors in the base station protects sensor network from malicious sensor nodes. In addition, base station generates a session key(K_b), at certain time intervals and broadcasts to all the sensor nodes have to re-generate their new secret session keys($K_{i,\ b}$). The built-in keys in sensor nodes avoid the distribution if secret keys in the wireless environment as well as providing substantial security.

IV. Secure Data Transmission Algorithms for Wireless Sensor Networks

The security protocol achieves secure data transmission on wireless sensor networks by implementing the following five phases in two algorithm:
Algorithm A and B are implemented in the sensor nodes and in the base station respectively.

1) Broadcasting session key by base station, performed in algorithm B.
2) Generation of cryptographic keys in sensor nodes, performed in algorithm A.
3) Transmission of encrypted data from sensor nodes to cluster heads using NOVSF code-hopping technique, performed in algorithm A.
4) Appending the ID# to data and then forwarding it to higher level cluster heads, performed in algorithm A.
5) Decryption and authentication of data by the base station, performed in algorithm B.

The base station, periodically broadcasts a new session key to maintain data freshness.

sensor nodes receive broadcasted session key K_b and computes their node-specific secret session key ($K_{i,b}$) by XORing K_i with K_b, $K_{i,b}$ is used for all the consequent data encryption and decryption during that session by both algorithms. Since each sensor node calculates $K_{i,b}$ using its unique built in key, encrypting data with $K_{i,b}$ also provides data authentication in the proposed architecture. Changing encryption keys time-to-time guarantees data freshness in the sensor network, moreover it helps to maintain confidentiality of transmitted data by preventing the use of the same secret key at all times.

Algorithms A: Implemented in Sensor Nodes

Step 1: If sensor node i wants to send data to its cluster head, go to next step, otherwise exit the algorithm.

Step 2: Sensor node i requests the cluster head to send the current session key K_b.

Step 3: Sensor node i XORs the current session key(K_b) with its built-in key K_i to compute the encryption key $K_{i,b}$.

Step 4: Sensor node i encrypts the data with $K_{i,b}$ and appends its ID# and the time stamp to the encrypted data and then sends them to the cluster head using NOVSF code-hopping technique.

Step 5: Cluster head receives the data, appends its own ID#, and then sends them to the higher-level cluster head or the base station. Go to step 1.

Algorithm B: Implemented in Base Station

Step 1: Check if there is any need to broadcast the session key K_b to all sensor nodes. If so, broadcast K_b to all sensor nodes.

STEP 2: If there is no need for a new session key then check if there is any incomming data from the cluster heads. If there is no data being sent to the base station go to step 1 after the session is complete.

Step 3: If there is any data coming to the base station then compute the encryption key, $K_{i,b}$, using the ID# of the node and the time stamp within the data. Base station then uses the $K_{i,b}$ to decrypt the data.

Step 4: Check if the current encryption key $K_{i,b}$, has decrypted the data perfectly. This leads to check the creditability of the time stamp 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 the sensor nodes

Step 6: Decides whether to request to all sensor nodes for transmission 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.

V. Implementation and Results

The amount of computational energy consumed by a security function on a given microprocessor is primarily determined by the number of clocks needed by the processor to compute the security function. The number of clocks necessary to perform the security function mainly depends on the efficiency of the cryptographic algorithms. In both

algorithms-MAC protocol is used to provide data integrity, where as node authentication is granted by using periodically changing user specific session keys, $K_{i,b}$ and NOVSF codes assigned to each node.

Table I. Performance Analysis Results

Algorithm.	Key Length(bit)	Total Time for 16B data input	Throughput (Kbps)
TEA	128	8,402185	1904.267
AES	128	7,639798	2094.296
DES	56	8,218642	1946.794
Blowfish	128	7,781995	2056.028

The cryptographic algorithm, Blowfish, necessitate 1KB memory; in addition to that we need 400-Byte for key setup. CBC-MAC also requires 580-byte in the smallest case.

VI. Conclusion and Future Work

We have presented an energy conserving method to provide key management for sensor networks. The method uses pre-deployed symmetric keying. A critical observation is that sensor-to-sensor secure channel establishment is not necessary for many monitoring applications. Therefore, pre-deployed keying has become sufficient, cost-effective approach to provide a keying infrastructure for security protocols that use those key. The overhead per sensor appears to be feasible , and none of the sensors are required to store large numbers of keys. Therefore, our approach supports key revocation and renewal mechanism, as well. Formal analysis about the security strength of the proposed scheme remains as future work.

References

[1] W. Fumy and P. Landrock,"Principles of key management," IEEE J. of Selected Areas in Communication, vol.11, pp.785-793, June 1993

[2] V.Raghunathan, C. Schurgers, S.park, "Energy Aware Wireless Microsensor Networks", IEEE Signal Processing Magazine, March 2003

[3] A. Sinha and A. Chandrakasan, "Dynamic power management in wireless sensor networks," IEEE Design and Test of Computers, pp.62-74, 2001