The Key Management Scheme for the WMSN-based Post-Disaster Road Monitoring System

Xiaoling Sun, Xuguang Sun, Shanshan Li, Fuming Chen, Qiuge Yang
Department of Disaster Information Engineering
Institute of Disaster Prevention
Sanhe,China
{sunxiaoling, sunxuguang, lishanshan, chenfuming, yangqiuge }@fzxy.edu.cn

***Abstract*—In order to obtain the availability of roads and bridges after earthquake, a real-time monitoring system is build by wireless sensor networks. And a new key management scheme is proposed to protect the safety of the system. The scheme is based on the polynomial and the Chinese Remainder Theorem, divided into three stages which are system initialization, key establishment, node join and revocation. Analysis shows that the new scheme greatly reduces the storage space and the energy consumption of calculation and communication of node. It is suitable for post-disaster road monitoring system.**

Keywords- sensor networks, key management scheme, post-disaster road , monitoring system

I. INTRODUCTION

Wireless sensor networks(WSN) collect the perceived target information within the coverage area through large numbers of nodes deployed in monitoring regional，then the information is transmitted to end-users after processed through multi-hop communication methods．For the nodes have the characteristics of random deployment and the dynamic changes of network topology, wireless sensor networks have broad application prospects in environmental monitoring, precision agriculture, national defense, military and commercial field. As the ordinary sensor nodes are limited by the energy, computing and storage capacity, communication bandwidth and transmission distance, they are vulnerable to security threats such as monitoring , capturing nodes, wormhole attacking. The content must be encrypted and authenticated to protect its safety communications. Key management becomes a very worthy of study.

At present, many scholars carried out relevant research on key management scheme for WSN. Eschenauer and Gligor[1] proposed a probabilistic key predistribution scheme for pairwise key establishment. The main idea was to let each sensor node randomly pick a set of keys from a key pool before deployment, so any two sensor nodes have a certain probability of sharing at least one common key. Chan [2]et al. further extended this idea and developed two key predistribution techniques: q-composite key predistribution and random pairwise keys scheme. The q-composite key predistribution also uses a key pool but requires two sensors compute a pairwise key from at least q predistributed keys they share. The random pairwise keys scheme randomly picks pairs of sensors and assigns each pair a unique random key. Du and Liu were partly based on Blom matrix [3] and polynomial model [4] to propose corresponding threshold solutions, these shemes effectively improved the ability of node to anti-capture under the case of increasing communications and computing. Zhu et al [5] proposed LEAP program to establish four types of communication key. Although the LEAP program achieved certain of the safety performance, but did not solve the problem of energy consumption for key updating.

II. APPLICATION OF WSN IN POST-DISASTER ROAD MONITORING SYSTEM

After the earthquake, roads and bridges are damaged, transmission and communication are interrupted, so the manpower and material for rescue can not be implemented in the first time to the disaster-affected area. In order to obtain the damage intensity of roads, we need a real-time monitoring system. There are two types of node in the system. One is the comman sensor node with less power energy, limited storage space and communication range. The other is the cluster node with high computing power and energy. The cluster node is responsible for data processing and forwarding. It collects the information sent by comman nodes and forward the to the base station.

A. Node Layout

The spacing between the arranged nodes is relevant to the expectation of system for coverage vulnerability and reliability. In this project, the spacing between the arranged nodes is determined by the coverage of the sensor and the redundancy of the node arrangement. The coverage of the sensor includes the perception coverage and connectivity coverage of the target area.

1) Coverage

a) Perception Coverage

Assuming the perception radius of seismic motion sensor is R_s, and the perception radius of sound sensor is R_w, then the perception radius of node is $R_p = min(R_s, R_w)$. Assuming the width of monitored road is L_R, if the perception radius R_p of node is larger than the width L_R of monitored road, see Figure 1, the sensor nodes only need to be laid out along one side of the road. If the perception radius R_p of node is smaller than the width L_R of monitored road, then the sensor nodes need to be laid out along both sides of the road, and the perception radius of sensor node on each side is $L_R/2$. In short, the perception radius of sensor node is ensured to be larger than the width of the monitored road.

Published by Atlantis Press, Paris, France.

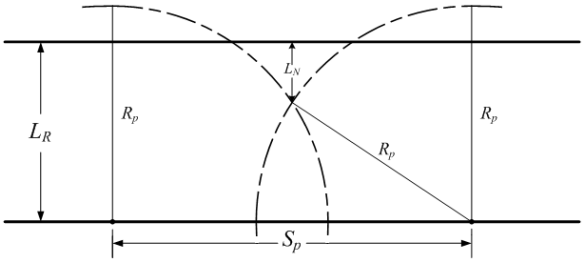

Figure 1 Schematic diagram of node sensing coverage

When the sensor node is monitoring the road, the uncovered area is called coverage vulnerability. In the post-disaster road monitoring system, if the allowed maximum width of the coverage vulnerability is L_N, see Figure 1, then in order to achieve the required perception coverage of the system, the maximum distance S_p between two arranged nodes is:

$$S_p = 2\sqrt{R_p^2 - (L_R - L_N)^2} \qquad (1)$$

b) Connectivity Coverage

Assuming the communication radius of nodes is R_c, then the maximum distance S_c between connectivity coverage nodes which complete the monitoring objectives is:

$$S_c = R_c \qquad (2)$$

Considering both the perception coverage and connectivity coverage, the maximum distance between nodes should be $S_{max} = min(S_p, S_c)$ to achieve the coverage of monitoring objectives. Assuming the length of monitored road is D_R, then the minimum number of sensor nodes needed to achieve the coverage of monitoring objectives is $N_{min} = D_R / S_{max}$.

2) Redundancy

Considering that the sensor nodes may lose efficacy due to uncontrollable reasons after the layout to the environment, if we use the minimum number of sensor nodes to accomplish the coverage, there would be coverage vulnerability exceeding the allowable range for the monitored road when one sensor node lose efficacy. So we consider laying out the sensor nodes with redundant. Taking cost, performance and other factors into account, we lay out the sensor nodes with double redundant in the system. If the minimum number of sensor nodes needed to achieve the coverage of monitoring objectives is N_{min}, then there need $2N_{min}$ nodes for layout with double redundancy. So the distance between two sensor nodes is $S = S_{max} / 2$.

If we use the minimum number of sensor nodes to accomplish the coverage, there would be coverage vulnerability when one sensor node lose efficacy. But if we lay out the sensor nodes with double redundant, there would be coverage vulnerability only if two adjacent nodes lose efficacy at the same time. So the probability of appearing coverage vulnerability of the system is significantly reduced.

Considering coverage and redundancy of the system described in 1.2.1 and 1.2.2, the formula of calculating the distance S between nodes is:

$$S = S_{max} / 2 = min(S_p, S_c) / 2 \qquad (3)$$

If $S_p < S_c$, by (1) and (3), we know that the distance S between nodes is:

$$S = S_p / 2 = \sqrt{R_p^2 - (L_R - L_N)^2} \qquad (4)$$

If $S_p > S_c$, by (2) and (3), we know that the distance S between nodes is:

$$S = S_c / 2 = R_c / 2 \qquad (5)$$

So in this system, the distance S between nodes is calculated by (4) and (5) depending on the perception radius R_p of sensors, the communication radius R_c of nodes, the width L_R of monitored road, and the maximum monitored vulnerability L_N allowed by the system.

B. Network Topology

The nodes are deployed along the road on both sides, and seperated in cluster based on the communication range of cluster nodes. See Figure 2. Sense nodes in the same cluster can only communicate with its neighbor nodes and the cluster node. Cluster nodes communicate with station directly, and need not communicate with each others.

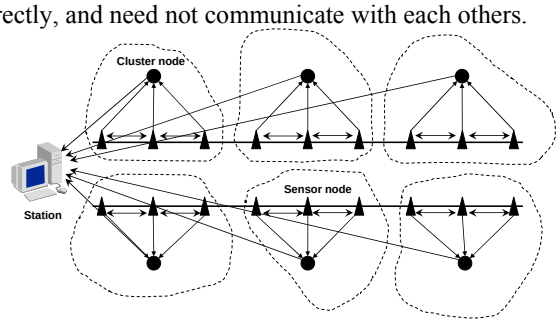

Figure 2 network topology

III. THE KEY MANAGEMENT SCHEME

As the network is deployed in the affected area, the node can be physical damaged easily. The key management scheme is adopted to be able to better support dynamically adding and deleting nodes, to ensure that the replacement of cluster node does not affect network topology and key establishment. For the special nature of network application, we use the polynomial-based key predistribution protocol[4] to establish the shared key between cluster node and station. Each cluster node shares one key with the station seperately, so capturing one cluster node will not affect the safty of other cluster nodes. We use the key management scheme based on Chinese Remainder Theorem[6] to establish the shared key between cluster node and sense nodes, to reduce the number of keys computated and storaged in cluster nodes and save energy and storage space for cluster nodes. In our new scheme, we combined the two schemes through intermediate variable. The station is responsible for large amount of computation to reduce energy consumption of cluster nodes and sense nodes.

A. System Initialization Phase

At the initialization phase, it is assumed that each node has a unique *ID*, for example, node u has ID_u. The key setup server randomly generates a bivariate t-degree polynomial

$f(x,y)=a_{ij}x^iy^j$, it has the property of $f(x,y)=f(y,x)$. The polynomial $f(x,y)$ is pre-storaged in station and cluster nodes, $f(ID_i, y)$ is pre-storaged in cluster node i, and the master key K_{share} is pre-storaged in station, cluster nodes and sense nodes. Nodes are deployed according to network topology, and sense nodes chose nearly cluster node according to some rules. About how to chose cluster node , there are many effective solutions, and it will not be discussed here.

B. Key Establishment Phase

After the network is set up, the shared key should be consulted dynamicly between sense node and its cluster node, between sense node and its neighbor nodes, between cluster node and station to ensure the reliability and confidentiality of information.

Step 1: The cluster node i collects the ID of each sense node ID_u (u=1,2,…, n, n is the number of sense nodes in the same cluster).And then sends the message $M=\{E_{Kshare}(ID_1), E_{Kshare}(ID_2),\cdots E_{Kshare}(ID_n), E_{Kshare}(ID_i)\}$ to the station.

Step 2: The station decrypts M to achieve ID_i of cluster node i and ID_u (u=1,2,$\cdots$,n) of sense nodes. Randomly choses a cluster key K, according to Chinese Remainder Theorem, we have:

$X\equiv k_1 \pmod{ID_1}$
$X\equiv k_2 \pmod{ID_2}$
…
$X\equiv k_u \pmod{ID_u}$
…
$X\equiv k_n \pmod{ID_n}$
$X\equiv k_i \pmod{ID_i}$

In which, $k_u=K\oplus ID_u$, (u=1,2,…,n), $k_i=K\oplus ID_i$. The station computes X and $f(X, ID_i)$, encrypts X by K_{share} and sends it to cluster node i.

Step 3: Cluster node i gets X and computes $f(ID_i,X)=f(X, ID_i)$ as the shared key between cluster node i and station. Then computes $K=(X \bmod ID_i) \oplus ID_i$, and sends X to each sense node in the cluster.

Step 4: Sense node u(u=1,2,…,n) gets X and computes $K=(X \bmod ID_u) \oplus ID_u$ seperately as the shared key between cluster node and the sense node.

Then we established shared keys between station, cluster node and sense node seperately.

C. Adding and Deleting of Nodes

1) Adding and Deleting of Sense Nodes

After key agreement is completed, if a new sense node joins the cluster, the cluster node collects ID_{new} of the new sense node, computes new value of M and forwards to the station. The station computes new encryption parameters through the Chinese Remainder Theorem:

$X_{new}\equiv k_1 \pmod{ID_1}$
$X_{new}\equiv k_2 \pmod{ID_2}$
…
$X_{new}\equiv k_u \pmod{ID_u}$
…
$X_{new}\equiv k_n \pmod{ID_n}$
$X_{new}\equiv k_{new} \pmod{ID_{new}}$
$X_{new}\equiv k_i \pmod{ID_i}$

The station sends X_{new} to cluster node, cluster node sents X_{new} to sense nodes, then cluster node and sense nodes computes shared key $K_{new}=(X_{new} \bmod ID_x) \oplus ID_x$ seperately. So there is a new shared key in the cluster, and the shared key between cluster node and station is the same as before.

When the sense node exits the cluster because of energy depletion, physical damage, being captured et al, assume that the station can detect the occurrence of this event through the information returned by cluster node, the station will delete the ID of this exiting node, recompute X, and send X to each node to renew the shared key in the cluster.

In the case that wireless sensor networks is used for collecting post-disaster information, the node exits mainly due to energy depletion or physical damage. When a certain number of nodes exit the network, we need to re-deploy nodes. The station will add the ID of new node, delete the ID of exiting node, compute the new value of X. Then all nodes in the cluster will compute new shared key.

2) Replace of Cluster Node

In our application environment, the exiting of individual sense node does not cause much of the network function, but the damage to the cluster nodes influenced more. So once the cluster node exits, it must be updated in time. If the replace of cluster node occurs, then recompute the shared key between cluster node and station, between cluster node and sense nodes according to section 3.2.

D. Performance Analysis

1) Analysis of Security Performance

The network deploys in the affected areas where roads and bridges are severely damaged. In this case, the probability of sensor node to be physical damaged is much more than to be captured. Another major security threat is information monitoring. The communication range between cluster node and sense nodes is too small to be monitored. If one node is captured, the station will delete its ID, and recompute the shared key in the cluster. The time of decrypting key is longer than the time of updating key. The communication range between cluster node and station is so long that it can be monitored easily, but the keys between each cluster node and station are different, so the failure of a cluster node will not affect other nodes.

2) Storage Space

Common sense node only need store its own ID, shared master key K_{share} and the shared key K. Cluster node need store its own ID, shared master key K_{share}, polynomial value of $f(ID_i, y)$, shared key $f(ID_i, X)$ with the station, shared key K with the sense nodes in the cluster.

Assumed that in E-G program, the total number of keys in the key pool is 10 times the number of network nodes, and the connected probability is 0.5. We compared this scheme with E-G program, see figure 3. As can be seen from the figure, the number of keys stored in ordinary nodes and cluster head node in this scheme is more less than the one in E-G program. And this scheme also support for the expansion of the network size better.

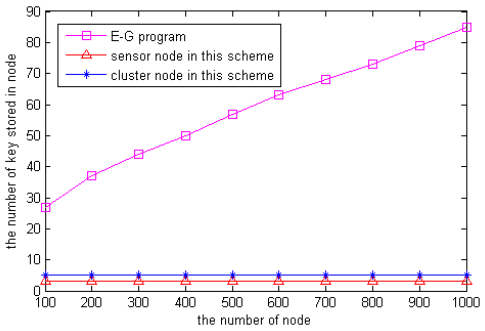

Figure 3 the number of keys stored in node under different schemes

3) Analysis of Energy Consumption

The power energy of sensor nodes is limited, so the key management scheme should try to reduce energy consumption while ensure the security of the network. The energy consumption mainly includes the calculation consumption and communication consumption. The required energy of sensor node for transmitting information is greater than the energy for calculating. The required energy for transmitting 1bit information can approximately calculate 3000 instructions. Therefore, the amount of information transmission should be limited to prolong the network lifetime.

In computational terms, common sense node just do one encryption operation to send *ID*, do one decryption operation to achieve *X*, and do one modulo operation and one XOR operation to calculate a shared key *K*. The cluster node need do one encryption operation to send *ID* within cluster, do one decryption operation to achieve *X*, do one polynomial operation to calculate shared key $f(ID_i, y)$ with station, and do one modulo operation and one XOR operation to calculate a shared key *K*. The *X* value and most of the polynomial operation are calculated by the station. So the energy consumption of nodes are saved significantly.

In communications, common sense node only need transmit its *ID* and receive *X* for one time, cluster node need receive *ID* for *n-1* times(*n* is the total number of nodes inside the cluster), send *M* to the station for one time, receive *X* from station for one time and send *X* to *n-1* nodes inside the cluster.

Assume the energy consumption of send and receive one message is e_S and e_R separately, the energy consumption of key looking is e_L, the energy consumption of symmetric encryption or decryption is e_{Sym}, the energy consumption of Polynomial arithmetic is $e_{f(x,y)}$, the energy consumption of modular arithmetic and XOR arithmetic is e_{mod} and e_{xor} separately, table 3.1 compares the energy consumption of common sense node and cluster node in this scheme with the one in E-G program. Assume that network size is a single cluster, *k* is the number of keys stored in one node in E-G program, *n* is the total number of nodes, *p* is the probability of establishing shared key directly between two nodes.

The computational and communicational energy consumption of common sense nodes in this sheme is smaller than E-G program. The communicational energy consumption of cluster node in this sheme is slightly larger than the E-G program, but energy reserves of cluster node in this sheme is higher than the sensor nodes in E-G program. Overall, the scheme greatly reduces the amount of information transmission and effectively saves the energy consumption of nodes as compared to other key management scheme.

TABLE Ⅰ ENERGY CONSUMPTION OF NODES

	communication consumption	calculation consumption
E-G program	$e_S+ne_R+npe_S$	npe_L+ke_{Sym}
Sensor node in this scheme	e_S+e_R	$2e_{Sym}+e_{mod}+e_{xor}$
Cluster node in this scheme	ne_S+ne_R	$2e_{Sym}+ e_{f(x,y)}+e_{mod}+e_{xor}$

IV. CONCLUSION

In this paper, for the wireless sensor networks used in the situation of collecting post-disaster information, we proposed a new key management scheme combined with the polynomial-based key pre-distribution scheme and the Chinese Remainder Theorem based key distribution scheme. The program is divided into three stages: system initialization, key establishment, adding and deleting of nodes. In the system initialization phase, each node prestores relevant variables. In the key establishment phase, the station and nodes consult shared keys with stored variables, including the key between cluster node and sense node, the key between cluster node and station. In the last phase, we discuss the issues of key update in the case of adding and deleting nodes. The new scheme can better support dynamically adding and deleting of node to ensure the security of network communications, can better save the storage space, computing power and communication energy of nodes. The new scheme is suitable for monitoring system for the availability of devastated road.

ACKNOWLEDGEMENT

Foundations:

Special Fund of Fundamental Scientific Research Business Expense for Higher School of Central Government (Projects for young teachers) No. ZY20110211;

Teachers' Scientific Research Fund of China Earthquake Administration No.20110112;

Scientific Research Plan Projects for Higher Schools in Hebei Province No.Z2012143;

REFERENCES

[1] Eschenauer L, Gligor V D. A key～management scheme for distributed sensor networks [A]. Proc 9th ACM Conf on Computer and Communication Security [C]. Washington DC,2002:41～47.

[2] Chan H W, Perrig A, Song D. Random Key predistribution schemes for sensor networks [A]. Proc 2003 IEEE Symp on Security and Privacy[C]. Berkeley, California, 2003:197.

[3] Du Wen-liang, Deng Jing. A pairwise key pre-distribution scheme for wireless sensor networks[C]. Proc of the 10th ACM Conference on Computer and Communications Security. 2003.

[4] Liu Dong-gang， Ning Peng. Establishing pairwise keys in distributed sensor networks[C]. Proc of the 10th ACM Conference on Computer and Communication Security. New York: ACM Press, 2003:52-61.

[5] ZHU S, SETIA S, JAJODIA S. Leap: efficient security mechanisms for large-scale distributed sensor networks[C]. Proc of the 10th ACM Conference on Computer and Communications Security.[S.I.]: ACM Press, 2003:62-72.

[6] Zheng Xin-liang, Huang C T， Matthews M. Chinese remainder theorem based group key management[C]. Proc of ACMSE 2007. 2007:266-2