A new Group Key Management Scheme based on DMST for Wireless Sensor Networks

YingZhi Zeng[1], Yan Xia[1,2], JinShu Su[1]

[1]School of Computer
National University of Defense Technology
ChangSha Hunan, China
zyz1234@gmail.com

[2]School of Computer and Communication
Hu'nan University
ChangSha Hunan, China
xiayan@hnu.cn

***Abstract*—Communication via self-organization is a practical and most common model for wireless sensor network. Its security, efficiency and cost and corresponding key management are one of the key research topics on WSN security. This paper proposes a group key management scheme for WSN based on an original self-organized structure, grid-loop. The group which we called Grid-loop is constructed on distributed Minimum Spanning Tree. Our group key management scheme has many advantages over cluster-based scheme. The analysis and comparison demonstrates its feasibility, efficiency and security for key establishment and maintenance in Wireless Sensor Networks.**

Keywords- group-key; key management; wireless sensor networks; network security; DMST

I. INTRODUCTION

In a wireless sensor network (WSN), sensor nodes are typically deployed in adversarial environments such as military applications. Sensor nodes may be dropped from airplanes to the work places and need to communicate later with each other for data processing and routing. The unattended nature of the deployed sensor network lends itself to several attacks by the adversary. Unattended deployment also makes insider attack easier.

Thus the encrypted communication is crucial to the secure operation of sensor networks. Firstly a large number of keys need to be managed in order to encrypt and authenticate all sensitive data exchanged. Secondly the characteristics of sensor nodes and WSNs render most existing key management solutions developed for other networks infeasible for sensor networks. To provide security communication key in such a distribution environment, the well-developed public key cryptographic methods have been considered at first, but these demand excessive computation and storage from the resource extra-limited sensor nodes. The symmetric key cryptography is considered as the only feasible way for wireless sensor networks. Therefore, there must be a secret key shared between a pair of communicating sensor nodes. Sensor nodes can use pre-distributed keys directly, or use keying materials to dynamically generate pair-wise keys.

The broadcast communication mode and the resource-constrained feature of sensor nodes makes that generating a unique communication key for every two nodes is costly and unpractical. Due to the high communication cost and limited bandwidth, it is not feasible either for each node to send its sensor data directly or through relay nodes to the sink. The relations of sensor data generated by neighbor nodes lead out that data aggregation is needed. There is two kinds of communication situation should keep secure, the data transferred from normal nodes to the aggregator nodes; the aggregated data transferred from the aggregators to the sink.

The next important topic we should focus on is the network topology, which is related to the communication connectivity and the total coverage of target region. Since the network topology is unknown prior to deployment, the key pre-distribution scheme is put forward to provide communication keys for the data aggregating flow and relaying flow, where the keys are stored in the ROMs of sensor nodes before the deployment. After deployment, each sensor node should connect with its neighboring nodes and generate their security keys in a self-organized method.

The main contribution of our work is focused on the management mechanism of group-key in WSN. The propose scheme includes self-organized group forming algorithm, Grid-loop-based key management scheme and rekeying.

This paper is organized as following: Section Ⅱ describes the related works. Section Ⅲ describes the new scheme in detail. Section Ⅳ deals with the detailed analysis and comparisons. Section Ⅵ concludes the paper with future research directions.

II. RELATED WORKS

For the sake of generating key for security communication in WSN, many Key Establishment Schemes (KES) have been proposed in recent years. Due to the obvious shortcoming, Key Distribution Center scheme and PKI-based scheme are both unfeasible in realization. Key Pre-Distribution (KPD) is the hot spot in this area. The main KPD schemes include random schemes and determinate schemes.

Eschenauer and Gligor [1] proposed the first random key pre-distribution scheme EG. Each sensor node is assigned k keys out of a large pool P of keys in the pre-deployment phase. Neighboring nodes may establish a secure link only if they share at least one key, which is provided with a certain probability that is depended on the selection of k and P.

Liu-Ning ,Du and Choi-Youn scheme can be considered as determinate scheme in which any two nodes can share a certain common key according to some mathematics rules [2].

978-1-4244-5113-5/09/$25.00 © 2009 IEEE

Random schemes have a common security problem: they cannot keep remain nodes being safe if some nodes are compromised and those keys loaded on the nodes are exposed to enemy.

Determinate schemes rely on certain mathematics rules. If the number of compromised nodes is increased to a level, the enemy can use the known key material to deduce the mathematics rules for key generating.

Generated keys are distributed among sensor nodes to keep the communications secure. All of above schemes aim to find a best way to establish keys, but they ignore an important thing: nodes playing different roles should use different keys according to different situations. A node may be a normal sensor, a data aggregator or a relay node. The data flow can also be different. Which role nodes should be chosen is related to the topology of WSN directly. Most schemes simple use cluster topology as their basic organization among sensor nodes.

III. SELF-ORGANIZED TOPOLOGY OF WSN

Section 2 proposes the shortcoming of current KES schemes. From the data-center's viewpoint, the sensor data of WSN is collected and aggregated through the collaboration among neighbor nodes. So the self-organized topology of WSN is very important in group-key establishment phase.

1) Basic definitions of loop

In the graph theory, a loop is a non-directional path, which begins and ends with the same node. Since there is at most one connection between every two nodes in an undirected graph G=(V, E) [3], a path from v_i to v_j representing a wireless sensor network link can be defined as a sequence of vertices $\{v_i, v_{i+1}, \ldots, v_j\}$, where V representing the set of nodes and E is the set of connections.

Loop length: The length of a loop also can be called path length, is the number of hops from v_i to v_j. Let L be a loop. It is evident that if length (L)<3, either the node on L is isolated or L is a round trip between two nodes.

Loop type: In a large scale WSN, there may be some isolated nodes. A loop with only two nodes is also a special loop. For example, in Fig.1, L2 and L3 are typical loops and L1 is a special loop. In the following parts, nodes on the loops with greater length than 2 are called on-loop nodes.

2) Basic definitions of cluster

In the graph theory, let G = (V, E) be a connected graph and $C = \{ C_1, \ldots, C_k\}$ a partition of V . We call C a clustering of G and C_i a cluster; C is called trivial if either k = 1, or all clusters C_i contain only one element. We often identify a cluster C_i with the induced sub-graph of G, i.e., the graph $G[C_i]:= (C_i, E (C_i))$,where $E (C_i) := \{\{v\} | v \in C_i\}$. Then $E(C)=\bigcup_{i=1}^{k} E(C_i)$ is the set of intra-cluster edges and $E \setminus E(C)$ is the set of inter-cluster edges. The set $E(C_i,C_j) := \{\{v,w\} \in E: v \in C_i, w \in C_j\}$ is the set of edges that have one end-node in C_i and the other end-node in C_j .

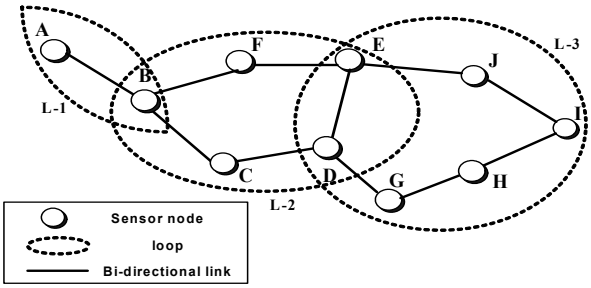

Figure 1. An example for loop-based WSN

In the research of sensor networks, a lot of applications use cluster as the basic organization. The feature of wireless broadcast communication introduces the concept of a cluster-header. A cluster-header is chosen among its neighbor nodes according to some rules. A node acting as a cluster-header would have the power in control of its cluster-members, such as node B, D and I in Fig2.

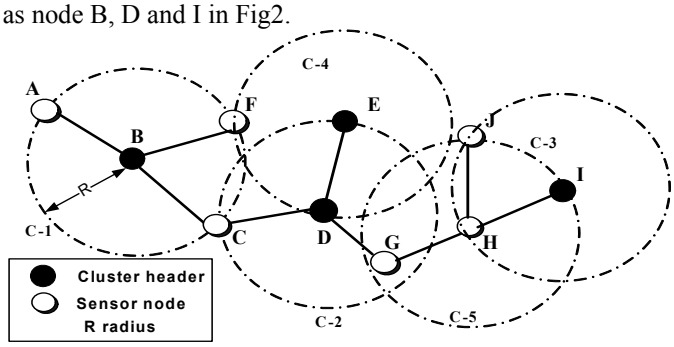

Figure 2. An example for cluster-based WSN

Cluster length: In a sensor networks, cluster length is the number of nodes in a cluster.

Cluster type: According to different cluster-header rules, there is also different cluster-header forming type: such as the lowest-ID cluster or the maximum connection-degree cluster. Neighbor clusters may share some common nodes: some cluster headers may be another cluster's members in some certain situations.

3) a new self-organized group strucure:grid-loop

Based on loop structure which has the convergence problem proposed in our previous work [7], this paper propose another new group structure of loop, grid-loop. Grid-loop is the smallest loop that is connected by neighbor nodes. A grid-loop cannot cover any sub-loop inside.

Definitions: Given T as a Minimum spanning tree (MST) of a connected, undirected graph G representing the network of wireless sensor nodes, the circle formed by some edges within T and one edge that does not belong to T is called a basic circle of the graph G, or a grid-loop.Grid-loop is the basic loop of G based on MST and the smallest unit for the network communication.

According to [4], research has shown that asymptotic connectivity results when every node is connected to its nearest 5.1774logn neighbors, while asymptotic disconnection

results when each node is connected to less than 0.074log nearest neighbors. With the node number N steadily increasing, the connectivity also steadily increase. Under condition with enough node density, we can make sure that grid-loop is the most suitable smallest group structure for WSN.

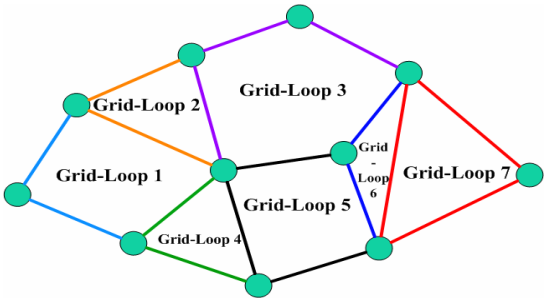

Figure 3. grid-loop structure in WSN

In figure 3, we can see that the overlay of neighbor grid-loops can be divided into two situations.

A. Sharing a point. Such as grid-loop 5 and 7are sharing a common node in figure 3.

B. Sharing a link. Grid-loop 1 and 2 are sharing a common link.

Basic features of Grid-loop are listed as follows.

(1) Grid-loop is the smallest loop.

(2) The data flow in Grid-loop is uniform.

(3) There is no special node or group leader in a grid-loop. Every node has the same right and power.

(4) As a smallest loop, grid-loop has the most stability of group structure.

(5) The length of a grid-loop is at least equal to 3.

4) Comparison between grid-loop and cluster

As a data-center network, the core function of WSN is aggregating data and forwarding data through relay nodes to the sink. So we consider the key establishment topology and the data process topology should not be separated.

Old KES are mainly based on cluster topology. There exist some key establishment schemes for WSN that are based on the cluster topology [5]. In a WSN, each sensor node acts either as a data producer or just as a router. In cluster-topology, each node should take part in a voting to choose some nodes acting as cluster headers (maybe choose itself). After the deployment and the CH's voting, the cluster headers play an important role in the next steps which include initializing keys, distributing cluster keys and rekeying. The most notable problem of cluster-based KES is the permission of nodes. If a cluster member acts as a local aggregator, then its cluster header also has to send sensor data to this member. Which one has the higher power? The header is the cluster controller that is selected from neighborhoods. The aggregator is the closest node to the sink. It is difficult to choose either of them to be the commander.

We take grid-loop as the basic unit and the entire network is grouped into self-organized inter-connected grid-loops. Within a grid-loop, nodes can exchange information with each other by forwarding messages along the loop in either of the two directions. For inter-loop communications, messages are first routed to the gateways nodes (router nodes joining multiple loops) and transferred from gateway to gateway till reaching the destination. As for inner Loop transmission, messages are finally being forwarded to the destination. Grid-Loop topology has many special benefits in WSN.

Efficiency: The grid-loop topology is more adapt to the physical positions of sensor nodes. It is obvious that data aggregation in grid-loop without the level of cluster header is more efficient and cost-saving.

Simple Structure: After the grid-loop topology is formed, there is no critical header node defined in a loop. This simple network topology never suffers from the chain change caused by the re-election of headers. The relationship of neighbor grid-loops is also very simple. On the contrary, the relationship between neighbor clusters is complex, especially during the cluster header's renewing time.

Robustness: Local grid-loop information can be reserved in every node on the grid-loop. This information redundancy enhances the network robustness. There exist two paths between every two nodes in the same grid-loop. This feature provides a backup route and authentication path for link failure during the transmission of aggregated data.

IV. THE PROPOSED KEY MANAGEMENT SCHEME

a) Creation of a grid-loop topology for key Establishment in WSN

Key Pre-distribution: Each node should be assigned some key materials, including a unique ID, a private key, a Hash function and a global key.

The problem of grid-loop construction in wireless sensor network can be modeled as the problem of creating the Minimum Spanning Tree in an undirected graph [6]. With the MST created, each edge that does not in the MST represents a new grid-loop constructed. We modified the distributed MST algorithm in [6] and improved it for grid-loop construction as following:

I. Basic idea of the distributed algorithm:

Step 1: Each node is initialized as a tree with single node and the tree Id is set to be the ID of the root.

Step 2: Each tree searches for edges to connect to another tree. If exists, they are combined to be a new tree labeled by the smaller value of two original Ids. Repeat Step 2 till all the nodes are connected.

II. Messages communicated between nodes:

COMPUTE and DIFFUSE contain information as:

1. label: the ID of the tree.
2. min_bridge_wt: the smallest weight from this tree to other trees(if no connection exists, this value is infinity. By the end of the algorithm this value is infinity).

GRID and UNGRID contain following information:

1.label: ID of the tree

2.gl_key: new generated group key

CONNECT and SUB are commands with no data.

III. The information items managed by each node are listed as followed.

1. label: the ID of the Tree that the node belongs to.
2. cand_mst_heap: ID tables of neighbor nodes belongs to different Trees.
3. loc_min_bdg_wt: the minimum weight between this node and neighbors of different trees or to be infinite when cand_mst_heap is empty.
4. mst_adj_list: the table of IDs of the neighbor nodes of the same tree.
5.mst_chd_list: all the child nodes that are connected to this node via edges in MST.
6.loc_gl_label: the labels of the grid-loops that this node belongs to.
7. loc_gl_key: the table for group keys corresponding to the labels in gl_label.

IV. Key steps of the algorithm:

1. Each tree root initializes COMPUTE.label to be its own ID，COMPUTE.min_bdg_wt to be its own loc_min_bdg_wt，then sends COMPUTE to all its children and waits for their replies.

2. Each child node that receives COMPUTE would first compare COMPUTE.label with its own label. If yes, then compare its own loc_min_bdg_wt with COMPUTE.min_bdg_wt if its own value smaller, then it updates it to COMPUTE. min_bdg_wt, otherwise no update. Then it passes COMPUTE to all its children. Leaf nodes would pass COMPUTE to their parents.

3. For a node that receives COMPUTE from its neighbor, if COMPUTE.label is less than its own label, it considers it as a message to combine trees. It updates its own label to be COMPUTE.label and goes to step 2. Those COMPUTE message with larger label would be ignored.

4. Finally the root would collect COMPUTE from all its children. It can find the smallest weight to the other trees via COMPUTE. min_bdg_wt. If it equals to its own loc_min_bdg_wt then the root would call the function add_MST_edge() to connect two trees. Otherwise it sends DIFFUSE with min_bdg_wt to all its children.

The edge weight w = (ID_i, ID_j) is a doublet that can be ordered by lexical order of ID_i and ID_j:

(ID_i, ID_j) > (ID_m, ID_n) iff $ID_i > ID_m$， or $ID_i = ID_m$ and $ID_j > ID_n$

(ID_i, ID_j) < (ID_m, ID_n) iff $ID_i < ID_m$， or $ID_i = ID_m$ and $ID_j < ID_n$

(ID_i, ID_j) = (ID_m, ID_n) iff $ID_i = ID_m$ and $ID_j = ID_n$

5. The child node that receives DIFFUSE would compare DIFFUSE.label to its own label. If equal, then it compares DIFFUSE. min_bdg_wt to its own loc_min_bdg_wt，if equal again, it calls the function add_MST_edge() to connect two trees. Otherwise it would send DIFFUSE to all its children.

6. The tree that calls the function add_MST_edge() would move the connecting neighbor node from its table of neighbors for different trees into the table of neighbors of the same tree, then send CONNECT along the edge to the other side. The other side would update its tables and finishes the connection.

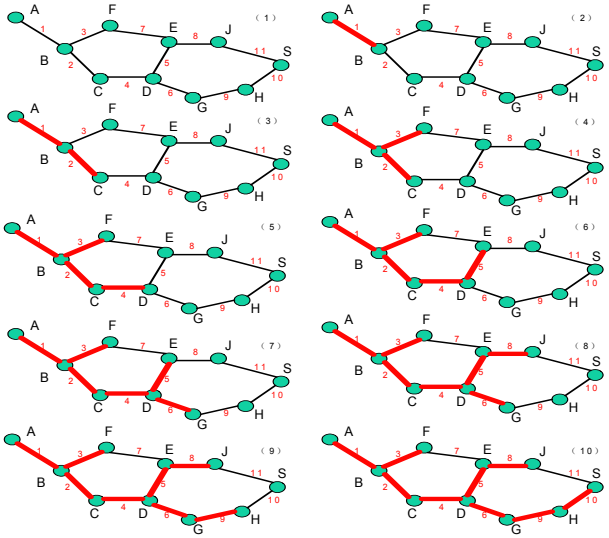

Figure 4. construction of grig-loop in WSN

b) The grid-loop key Establishment scheme

After step a), all the nodes are divided into different grid-loops and some nodes are shared between two neighbor loops. As shown in the figure 4, the construction process of grid-loops in Figure 1 is presented.

Based on the self-organized grid-loops, the key of a grid-loop can be created. Each grid-loop has only one link not including in the MST. The node on the link with a larger ID becomes a loop-creator and has the power to create a new key for those nodes in the grid-loop. We can set up the computing formula of loop-key as following.

According to the construction algorithm of the grid-loops based on MST, each grid-loop has only one edge that does not belong to the MST. The node of the larger ID value on the side of this edge becomes the creator node of the grid-loop and generates the group key for the grid-loop. Assuming the nodes on the sides of this edge to be node I and J, and node M is another neighbor of J. J with larger ID value is the creator of the grid-loop. The group key of the grid-loop is defined as:

$$Grid\text{-}loop\text{-}key = Hash\left([ID_I]_{Key(I\text{-}J)} \| [ID_J]_{Key(J\text{-}M)} \right) \quad (1).$$

Among above equation Key(I-J) and Key(J-k) are the pair keys between I-J and J-K respectively according to EG scheme[1] or L-D-C scheme[2]. This design makes sure that only the creator node J knows these two pair keys and node I is unaware of any of them. Any if other nodes except the

creator node J is compromised, the details of group key of the grid-loop would not be leaked to enemy, thus prevent the forge group key by the enemy.

After creation of keys, loop-creator will send the loop-key and the member list to its loop members. If the loop format is not special, the key messages will be sent to its two loop-neighbors at first. Each node on the loop will send the key to next node on the member list till some node receives same message from its two neighbor nodes.

After above steps, each node in WSN should belong to one grid-loop group at least and should keep a grid-loop-key shared with other loop members. Data aggregation and communication within the grid-loop should be encrypted with the grid-loop-key.

c) The Rekeying Scheme

As a resource-limited network, WSN cannot afford changing group keys continuously. But there are still two situations in that rekeying is needed for security. We define the situations as key maintenance triggers.

Situation I: If a grid-loop member is recognized as a defection node or the BS sends a command to get rid of a node from certain loop, the urgent work is to eliminate it form the loop's member list. First of all, such an abnormal message arrives at the closest safe loop member. The node becomes a temporary leader and will send a cleaning message to its two loop neighboring nodes. At the same time, the leader node will random generate a new key for the loop and send this rekeying message to replace the old loop-key.

Situation II deals with normal rekeying. If a loop member is out of battery and cannot work properly any more, it should be deleted from the loop list. The loop-key that it shared with other members should also be abandoned. So the first step is to clean old loop-key stored on every loop member. The second step is to set up new loop-key.

In one word, rekeying process is very important in WSN. Grid-Loop-key should be changed as quickly as possible if some defection nodes are found. At the same time, normal key updating is also a good method to keep WSN secure.

V. Analysis and Comparisons

a) Analysis of grid-loop architecture

In the undirected random graph of the wireless sensor network, connected nodes no less than three would form a grid-loop while two connected nodes cannot form a grid-loop, according to the principle of loop-priority, so we arrive at the apparent theorem as following:

Theorem 1: Grid-loop is the atomic loop with covering area.

According to the construction algorithm of grid-loop, we can induct following corollaries:

Corollary 1: For the same WSN the number of grid-loops constructed by the grid-loop algorithm is larger than the number of that of cluster algorithm.

Corollary 2: For the same WSN the number of nodes in grid-loops structure to be adjusted after some nodes are compromised is smaller than that in cluster structure.

b) Comparison on Communication

In the cluster algorithm for a N-noded and E-edged WSN, Assuming there are M nodes to be candidates of headers with total degrees of Em and M=O(N) the total communication cost is 2*E + 2*Em + 2*M = 2*E + 2 * Em + O(N).

According to the analysis [6] of grid-loop algorithm for a N-noded and E-edged WSN, the communication complexity is 2*E + C * N * log(N) + O(N) where C<5 is a constant.

We can see that the communication complexities of the two algorithms are almost the same. According to the theory of random graph [1,4], the degree of the random graph must satisfy E >= 5.1774*N*log(N) for the WSN to be asymptotic connected thus making the communication complexity of grid-loop algorithm less than 2.05*E. It's obvious that Em would be close to E in such dense graph, so the second item in the communication complexity equation of cluster algorithm would be greater than the counterpart in grid-loop algorithm. So the total cost of grid-loop is smaller and in advantage.

We designed the simulations for two algorithms in the same situations. The nodes of the WSN are chosen for 100, 200 and 300. Densities are randomly set and the results are shown in Fig 5, 6, and 7. From these figures we can see that grid-loop algorithm is economical than cluster algorithm.

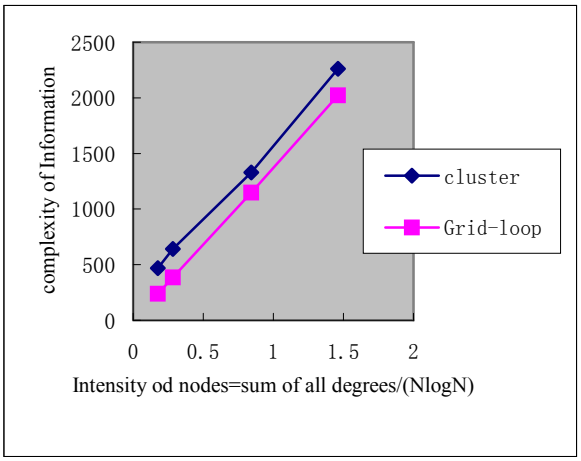

Figure 5. （N=100）comparison on complexity of information between grid-loopVS cluster under different indensity of nodes

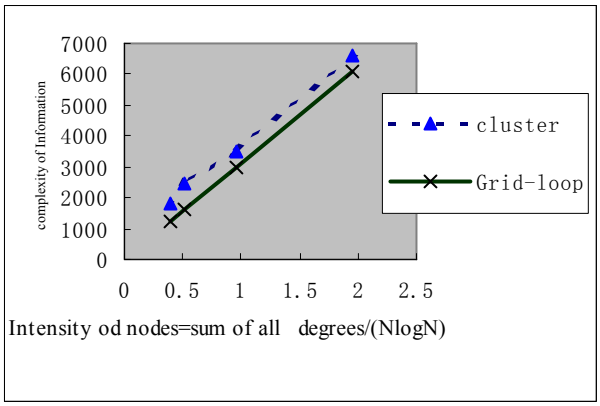

Figure 6. （N=200）comparison on complexity of information between grid-loopVS cluster under different indensity of nodes

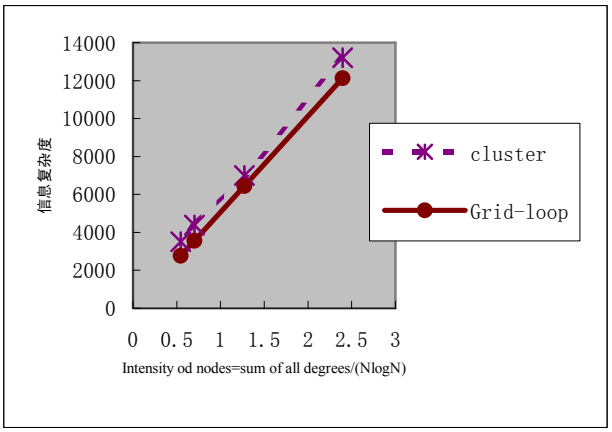

Figure 7. （N=300）comparison on complexity of information between grid-loopVS cluster under different indensity of nodes

c) Comparison on security

From the perspective of security, the grid-loop-based scheme is safer and more stable than the cluster-based scheme. These two schemes have different role assignment among sensor nodes. Cluster-based scheme assigns many important tasks on the cluster headers. A header node will control its cluster members all the time until it is replaced by another node. A grid-loop creator's identifier initializes a construction of a loop and has the right to generate a grid-loop key. After the grid-loop is constructed, there is no difference between normal nodes and the grid-loop creator.

TABLE I. COMPARISON OF PROBABILITY OF NODE BEING CAUGHT

Cluster-based key establishment and rekeying		Grid-Loop-based key establishment and rekeying	
identifier of Node being Caught	Impact to WSN	identifier of Node being Caught	Impact to WSN
Cluster header	(1) Lost control to all the cluster members under the control of that cluster header; (2) remain nodes have to start a new round cluster header election	Grid-Loop creator	same as grid-loop members
Cluster members	(1) The cluster header have to delete it from the member list and inform other members; (2) The cluster header start a rekeying	Grid-Loop members	(1) Neighbor nodes delete it from the loop sequence and broadcast cleaning message; (2) Neighbors nodes generate new loop key and spread it along the new loop sequence

TABLE II. COMPARISON OF IMPACT OF NODE BEING CAUGHT

Cluster-based key establishment		Grid-Loop-based key establishment		
Node identifier	Probability of being caught	Node identifier	Probability of being caught	
Cluster header	$\frac{C_n}{T_n}$	Grid-Loop creator	Before all grid-loops being formed	$\frac{1}{T_n} \times W (1 \le W < \lfloor \frac{T_n}{L_n} \rfloor)$
			after all grid-loops being formed	0
Cluster members	$\frac{(T_n - C_n)}{T_n}$	Grid-Loop mmbers	$\frac{1}{T_n}$	
	Cn: Cluster numbers Tn: total node numbers		Ln: Grid-loop numbers Tn: total node numbers	

According to the probability theory, each member in a grid-loop topology has equal probability to be caught by enemy. Once a grid-loop member is lost, its grid-loop-neighbors can set up new grid-loop quickly according to the rekeying scheme. If a cluster header is being caught, then its member nodes have to take part in a new cluster header's election. At the same time, the probability of a cluster header being caught is determined by the result that cluster numbers compare to the total node numbers. This probability is greater than that of a loop creator being caught. The probability and impact comparison results are listed in Table 1 and 2.From the point of graph theory, we can also take grid-loop as a special cluster without cluster-header. Grid-loop-based scheme can reduce the burden on the cluster-header and decrease relative security risk. At the same time, grid-loop-based topology is proved to be suitable for data aggregation in WSN.

VI. CONCLUSION

Establishing group key is one of the most important technologies in the security mechanism of WSN. In this study, we have developed an original scheme for key management and rekeying to WSN based on the self-organized structure, grid-loop. Based on grid-loop, we proposed new algorithms for key management, i.e., forming grid-loops via Minimum Spanning Tree and forming group key, which provides an original scheme to the WSN for creating loop keys and their maintenance and renewing. Differing from classic cluster topology, the construction of grid-loop topology not only has an efficient constructing mechanism, but also has a simple and robust structure. Comparing with existing cluster-based establishment schemes, our scheme based on DMST is proved to be more balanced, cost-saving, efficient and safe.

Future research would focus on reduction of communication cost during the key establishment and real time detection of compromised node based on self-organized grid-loop structure.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation for the Doctoral Program of Higher Education of China under grant No.20049998027, the National Science Foundation of China under grant No. 90104001 and No. 90604006.

REFERENCES

[1] Eschenauer, L.and Gligor, V. D.: A key-management scheme for distributed sensor networks. The 9th ACM conference on Computer and Communications, USA, Nov.2002.
[2] D. Liu and P. Ning, "Establishing pair-wise keys in distributed sensor networks," ACM Conference on Computer and Communications Security, pp 52--61, 2003.
[3] Yanping Li, XinWang, Florian Baueregger,Xiangyang Xue, and C.K.Toh, Loop-Based Topology Maintenance in Wireless Sensor Networks, ICCNMC 2005.
[4] Xue F, Kumar P R. The number of neighbors needed for connectivity of wireless networks [J]. Wirel. Netw. , 2004, 10 (2): 169-181
[5] Li Lin, Wang Ru-chuan, Jiang Bo, Huang Hai-ping, Research of Layer-Cluster Key Management Scheme on Wireless Sensor Networks, Journal of Electronics & Information Technology , Vol.28No.12, Dec.2006.
[6] H Abdel-Wahab I S F S. A Simple and Fast Distributed Algorithm to Compute a Minimum Spanning Tree in the Internet [J]. 1997.
[7] YingZhi Zeng, JinShu Su, Xia Yan, BaoKang Zhao, QingYuan Huang. LBKERS: A New Efficient Key Management Scheme for Wireless Sensor Networks[C].Beijing, China, MSN2007.