A Scalable Grouping Random Key Predistribution Scheme for Large Scale Distributed Sensor Networks

Po-Jen **Chuang,** Tun-Hao Chao, and Bo-Yi Li
Department of Electrical Engineering, Tamkang University
Tamsui, Taipei Hsien, Taiwan 25137, R.O.C.
pjchuang@ee.tku.edu.tw

Abstract

The security issue in a distributed sensor network (DSN) has been drawing considerable research attention in recent years. Key management, a basic security service, becomes the core design for various security services, such as encryption and authentication. This paper proposes a scalable grouping random key predistribution scheme which divides all nodes into several groups and uses the one-way function to generate group-to-group pairwise keys to increase the connectivity of each key and to enlarge the maximum supportable network size.

1. Introduction

The development of wireless sensor networks has become an important research topic in recent years due to such critical application needs as emergency response, medical monitoring, military tracking, energy management and pollution monitoring. **As** sensor networks are often deployed in unattended environments, they can be easily assaulted or compromised by adversaries. To enhance the security of sensor networks, researchers have come up with a number of important security services, including key management [1-21.

The following are basic features of a distributed wireless sensor network:

- Low bandwidth and computing power
- Limited memory and energy resources
- Prone to failure
- Large scale of sensor nodes
- Possibly without a central device

Limited by the processing power, battery life, communication bandwidth and memory space of sensor nodes, the key management protocol should be simple and light. Traditional key exchange mechanisms in wired networks do not fit in DSNs due to high computational overhead and considerable memory requirement. Key distribution protocols relying on infrastructures or trusted third parties are also impractical because of restricted communication ranges and inability to learn about the network topology before deployment. **As** each node may only communicate with its neighbor nodes, the random key predistribution scheme **[3]** is appropriate for large scale DSNs.

There are two extreme cases of key predistribution: (1) each node stores a single master key and (2) each node stores all the other nodes' pairwise keys. In case (1), each node consumes only a little memory space. But when the adversary compromises a node, the entire network gets disrupted. Case **(2)** is able to provide the highest security because the key that every two nodes use to communicate with each other is unique. This is nevertheless unfeasible due to limited memory resources of the node. For instance, assuming n is the number of nodes in the network, it will be difficult for each node to store $n-1$ keys when n **is** large.

Using the concept of random key predistribution, this paper presents a Scalable Grouping (SG) random key predistribution scheme, briefed as the SG scheme. The proposed scheme, a modification of the Unique Assigned One-way **(UAO)** function scheme [4], divides all nodes into several groups and uses the one-way function to generate group-to-group pairwise keys. To assure that the link key used to secure a link is nearly unique, the ***SG*** scheme uses the exclusive-OR operation to compose the link key when two nodes have two or more shared keys. Simulation results show that the SG scheme is able to support larger networks and yield better resilience against node capture. The scalability of the ***SG*** scheme also proves higher than that of other random key based schemes.

Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05)
0-7695-2316-1/05 $20.00 © 2005 **IEEE**

2. Previous random key based schemes

2.1 The random key predistribution scheme [3]

This basic scheme includes three phases: key pre-distribution, shared-key discovery and path-key establishment. In the first phase, each node picks r keys from a large key pool of S and stores into its memory before deployment (r is called the node's key ring). In the second phase, each node tries to discover whether its neighbors share a key with it. If so, the shared key becomes the link key for their transmission, and a graph is gradually formed. In the last phase, if two neighbor nodes do not have a shared key, they will try to establish a path-key through other secure links.

The basic scheme uses the Random graph theory [5] to analyze the suitable parameter. Consider a random graph with n nodes $G(n, p_l)$. Assume the probability that a link exists between any two nodes is p_l: When $p_l = 0$, the graph has no edges; when $p_l = 1$, the graph is fully connected. In [5], we find the monotone property that there exists a threshold value of p_l and that the property will move from "likely false" to "likely true" on whether graph G is fully connected. The threshold function p_l is defined as follows, where P_c is the desired probability that graph G is fully connected:

$$p_l = \frac{\ln(n) - \ln(-\ln(P_c))}{n}$$

The expected number of secure links for a node will be

$$d = p_l \times (n-1) = \frac{(n-1)\times(\ln(n) - \ln(-\ln(P_c)))}{n}$$

Now assume the average number of neighbor nodes

$$p = \frac{d}{n'}$$

for each node is n'. Since the expected degree d is calculated, we obtain the required probability p for two neighbor nodes to establish a secure link successfully

2.2 The q-composite keys scheme [6]

The q-composite keys scheme works similarly as the basic scheme. To enhance security against smaller-scale attacks, the scheme employs a parameter q. For any two nodes, it requires at least q shared keys to establish a secure link, and the link key K is generated as the hash of all shared keys,

$K = hash(K_1 \| K_2 \| \ldots \| K_{q'})$, where $q' \geq q$.

As the probability for any two nodes to share at least q keys may be less than the required probability p, the key pool size needs to shrink until the probability of connectivity equals p.

2.3 The random-pairwise keys scheme [6]

The basic idea of the random-pairwise keys scheme is to store sufficient pairwise keys in each node to form a connected graph, instead of storing all the other nodes' painvise keys. Assume each node stores at most

$$n = \frac{m}{p}$$

m keys and the required probability for two neighbor nodes to setup a secure link is p. The maximum supportable network size n can be calculated as

The random-pairwise keys scheme is able to provide node-to-node authentication properties as it uses pairwise keys instead of picking keys from a large key pool. The fact that each key used to secure links is unique also helps the scheme produce more desirable resilience against node capture than the above two schemes.

2.4 The Unique Assigned One-way (UAO) function scheme [4]

The UAO scheme also uses pairwise keys, derived from a unique one-way function in each node, to establish the secure link. It can support larger networks than the random-pairwise keys scheme due to less required node memory. Before nodes are deployed, the key decision algorithm is performed. Suppose each sensor node SN_i has a unique identifier ID_i and is assigned a unique one-way function F_i. Each node first randomly selects r node identifiers (r is the required number of keys) to achieve the connected graph, calculates r pairwise keys K_j by the following equation $K_j = F_j(ID_i)$ (j being the selected node identifier), and then memorizes r pairs of K_j and ID_j.

After the deployment, each node performs the node-to-node authentication protocol to set up the secure link with its neighbors. Each node first broadcasts its identifier ID_i to the neighbor nodes. Nodes that receive the identity information will verify whether ID_i is combined with any key in their key rings. If node j finds a key K_s combined with ID_i, it will send a request message encrypted by K_s and its identifier ID_j to node i. When node i receives the request message, it obtains the key K_s by computing $F_i(ID_j)$. Thus both nodes i and j can verify the link key K_s through a challenge-response process, and establish a secure link between them.

3. The proposed Scalable Grouping (SG) random key predistribution scheme

The proposed *SG* scheme aims to support more nodes in the sensor network and to provide desirable resilience against node capture. As mentioned, the random-pairwise keys scheme *[6]* achieves the highest security at the cost of large node memory space. To reduce such memory requirement, the UAO scheme uses the one-way function to assist the forming of link keys and attains as favorable security as the random-pairwise keys scheme (because each key used to secure a link is unique).

Different from the UAO scheme, the proposed SG scheme divides all nodes into several groups to increase the connecting ability of each key and to enlarge the maximum supportable network size. It also takes the concept that the link key is composed of some shared keys to improve resilience against node capture.

The features of the *k*-SG scheme are listed below.

(1) Each node in the network has a group identifier.
(2) Each group has at most *k* nodes.
(3) Nodes of the same group have a shared group key.
(4) Two nodes of different groups use the group-to-group pairwise key to establish the secure link.
(5) If two nodes have two or more shared keys. the link key is the composite of these shared keys using the exclusive-OR operation.

The *k-SG* scheme consists of three phases:

3.1 Initialization

The initialization phase runs off-line. First we divide all nodes into several groups, each group having at most k nodes. **All** the nodes of the same group i have a group identifier GID_i, a group key K_i and a one-way function F_i. Each sensor node randomly selects r group identifiers (r being the required number of keys to achieve the connected graph), calculates r group painvise keys K_{ji}, (j being the selected group identifier GID_j) and stores r pairs of K_{ji} and GID_j. The equation for generating group-to-group painvise keys is

$$K_{ji} = F_j(GID_i)$$

3.2 Link key setup

After deployment, sensor nodes perform the link key setup. Each sensor node broadcasts its GID_i to the neighbor nodes and verifies the received GID_i. If in the same group, the neighbor node returns the message of the same group and the list of its key ring; otherwise, returns its *group* identifier GID_j. Both nodes then execute the link key setup algorithm. Assuming the group identifiers of nodes **A** and B are respectively GID_i and GID_j, the algonthm can be defined as follows:

```
if ( GID_i = GID_j){ /* node A belongs to the same group with
                        node B */
   if (node A has other shared keys with node B)
      Link Key K_s = K_i ⊕ shared keys ;
   else
      K_s = K_i ;
}
else { /* node A and node B are not in the same group */
   switch (){
      case1 (node A has K_ji and node B has K_ij) :
         node A calculates K_ij = F_i(GID_j) ;
         node B calculates K_ji = F_j(GID_i) ;
         K_s = K_ij ⊕ K_ji ;
         break ;
      case2 (node A has K_ji but node B doesn't have K_ij) :
         node B calculates K_ji , then K_s = K_ji ;
         break ;
      case3 (node B has K_ij but node A doesn't have K_ji) :
         node A calculates K_ij , then K_s = K_ij ;
         break ;
      case4 (node A doesn't have K_ji and node B doesn't
      have K_ij) :
         node A cannot setup a secure link with node B ;
         break ;
} }
```

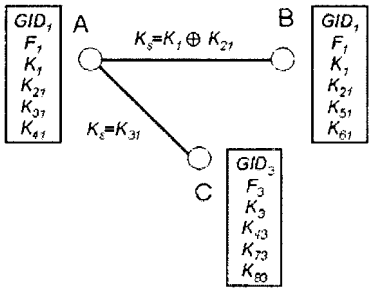

Figure 1. An example of link key setup.

Now suppose nodes **A,** B and C are neighbors in the network, and their group identifiers are GID_1, GID_1 and GID_3, respectively. The key rings that the three nodes store are shown in Figure 1. As we can see, the link key K_s generated by nodes A and B is composed of the group key K_1 and one shard key K_{21}; the link key K_s generated by nodes **A** and C is the group-to-group painvise key K_{31} generated by node C using $F_3(GID_1)$.

3.3 Secure link establishment

After execution of the link key setup algorithm, if two nodes have a link key K_s, they will verify the link key through a challenge-response protocol. If it turns out correct, a secure link will be established between the two nodes for communication.

4. Performance evaluation

Simulation runs have been carried out to evaluate the performance of the **SG** scheme and other schemes. Performance evaluation is analyzed based on the security strength, the maximum supportable network size under limited memory resources, and the conformity to the limited global payoff requirement.

The involved notations are defined as follows.

n : Number of sensor nodes in the network
n' : Average number of neighbor nodes for each node
r : Key ring size (i.e. number of keys in each node)
p : Probability for two neighbors to setup a secure link
k : Group size of the ***k-SG*** scheme (i.e. maximum number of nodes for each group)
g : Number of groups in the network, equal to n / k
X : Number of directly compromised links in the network (i.e. at least one of the two nodes on this link has been compromised)
Y : Number of additionally compromised links in the network (i.e. neither of the two nodes on this link is compromised, but the link key can be recovered by the adversary)

4.1 The security strength

The security of the SG scheme is evaluated in terms of resilience against node capture. The fraction of links in the network that an adversary is able to eavesdrop on indirectly by recovering keys from the captured nodes is estimated as

$$\frac{Y}{number\ of\ links - X}$$

The following are steps for calculating this fraction:

(1) Randomly pick i nodes -- i is the expected number of compromised nodes.
(2) Take all group keys and group-to-group pairwise keys stored in i nodes into a database.
(3) Take all one-way functions stored in the i nodes into the database, so all the keys derived from these functions are compromised.
(4) For each link that is not directly compromised, analyze the link key to see if it will be recovered from the database.

The key ring size (r) for each scheme is listed in Table I. The size of the ***k-SG*** scheme includes a group key, k being the group size.

Table I. The key ring size for each scheme (p = 0.33)

	n =1000	n =2000
Basic scheme	200	200
2-composite keys scheme	200	200
3-composite keys scheme	200	200
2-SG scheme	90+1	180+1
3-SG scheme	60+1	120+1

Figures 2 and 3 show the resilience against node capture under different numbers of sensor nodes for various schemes. Note that the random-pairwise keys scheme *[6]* and the UAO scheme [4] are represented as the line y=0 because the adversary cannot get the global information from local node capture.

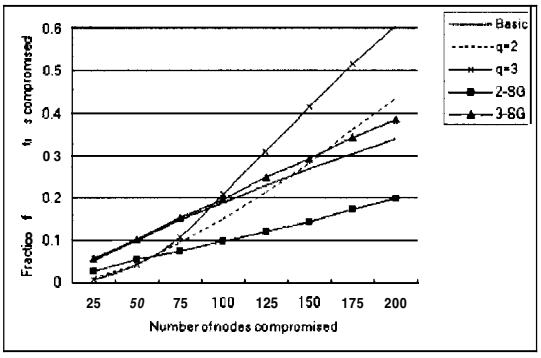

Figure 2. Number of compromised nodes versus fraction of additional compromised links for various schemes (n = 1000, n' = 60, p = 0.33).

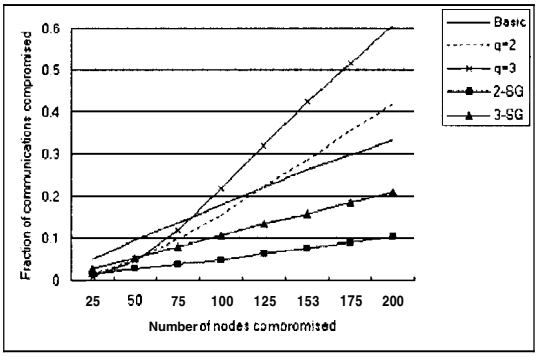

Figure 3. Number of compromised nodes versus fraction of additional compromised links for various schemes (n = 2000, n' = 60, p = 0.33).

Figure *2* shows that the ***2-SG*** scheme has the largest secunty strength. This is because the **SG** scheme uses the group key and the group-to-group pairwise key, not picking keys from a large key pool. That is, whenever an adversary compromises a node, it can approximately get the information about other nodes in the same group only. As a result, the fraction of additional compromised links can be significantly decreased,

Figure **3** shows that when the network size extends (to $n = 2000$ in this case), the SG scheme provides even stronger security. This is because the ratio of information obtained by the adversary to the whole network gets smaller in large-sized networks.

4.2 The maximum supportable network *size*

As the SG scheme uses pairwise keys to establish secure links, the key ring size directly determines the probability for two neighbor nodes to set up a secure link. Scalability is hence restricted. Take the random-pairwise keys scheme as an example. Assuming that the network size $n = 10000$ and the probability of connectivityp = 0.33, the required key ring size r will be $r = n \times p = 3300$, which is quite impractical in sensor networks.

Let p_a be the probability for two neighbor nodes with the same group identifier to setup a secure link and p_b be the probability for two neighbor nodes with the different group identifiers to setup a secure link. p in the SG scheme can be replaced by the following eauation

$$p = \frac{1}{g} \times p_a + \left| 1 - \frac{1}{g} \right| \times p_b$$

$1/g$ **is** the probability for any two nodes to have the same group identifier, p_a is one because the two nodes have a shared group key, and p_b is one minus the probability that neither node has the key derived from the other's one-way function. Since the key ring size is r, the probability for any node *to* get a key derived from a particular node's one-way function is $r/(g-1)$. Thus

$$p_b = 1 - \left| 1 - \frac{\quad}{\quad} \right.$$

From the above, the probability of connectivity p in the ***SG*** scheme can be derived as

$$(1) \qquad p = \frac{1}{g} + \left(1 - \frac{1}{g}\right) \times \left[1 - \left(1 - \frac{r}{g-1}\right)^2\right]$$

or be simplified by $g = n/k$ and becomes

$$(2) \qquad p = \frac{k}{n} + \left(1 - \frac{k}{n}\right) \times \left| 1 - \left(1 - \frac{\quad}{\quad} \right.\right.$$

Equation **(2)** is then used to evaluate the maximum supportable network size for the *k-SG* scheme.

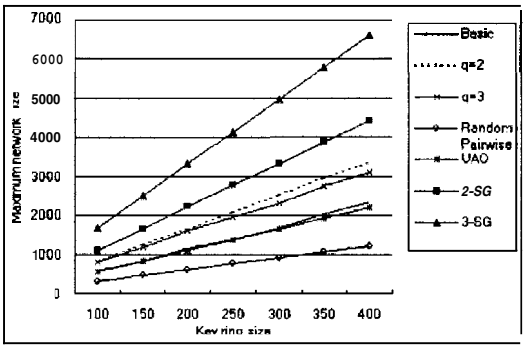

Figure 4. Maximum supportable network sizes for various schemes **(p = 0.33, the key length = 128 bits and the size of the one-way function = 160 bits).**

Figure **4** exhibits maximum supportable network sizes for various schemes. **Among** all schemes, the SG scheme is shown to support the largest-sized network than other schemes, and when k increases, the maximum supportable network size also grows. However, it will be more practical to keep $k < 4$ because of the limited global payoff requirement.

4.3 The limited global payoff requirement

The limited global payoff requirement *[6]* is applied to the proposed SG scheme because its secure **links** may be compromised indirectly, like the basic scheme or the q-composite scheme. The main purpose of such a requirement is to prevent the adversary from gaining too much at too little expense. Take the basic scheme as an example. In a network with size $n = 10000$, when an adversary compromises 50 nodes, there will be about 9.5% additionally compromised links. **As** the ratio of compromised nodes to the whole network size is only 0.5%, the random key based schemes need to meet the following requirement to avoid such a situation:

" The number of additional compromised links in the network $\leq$ the number of directly compromised links in the network (i.e. $Y \leq X$). "

The limited global payoff ratio ρ is thus defined to be Y / X. When $\rho \leq 1$, it indicates the limited global requirement has been satisfied.

Figure *5* shows the limited global payoff requirement for the ***2-SG*** scheme. X and Y are evaluated in 5(a), 5(b) and 5(c) for different network sizes; ratio ρ is

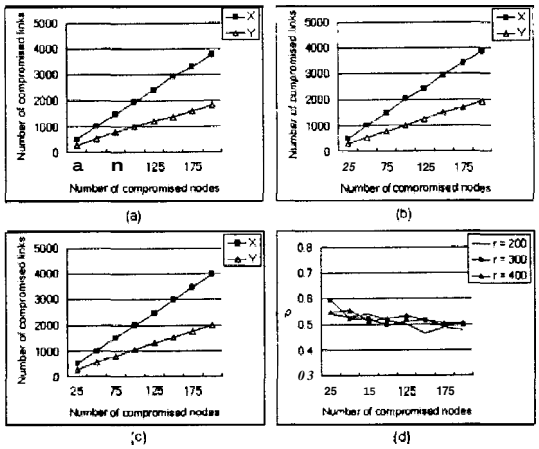

Figure 5. Evaluation of the limited global payoff requirement in 2-SG, where (a) n = 2210, (b) n = 3312, and (c) n = 4414. The ratio ρ is calculated in (d).

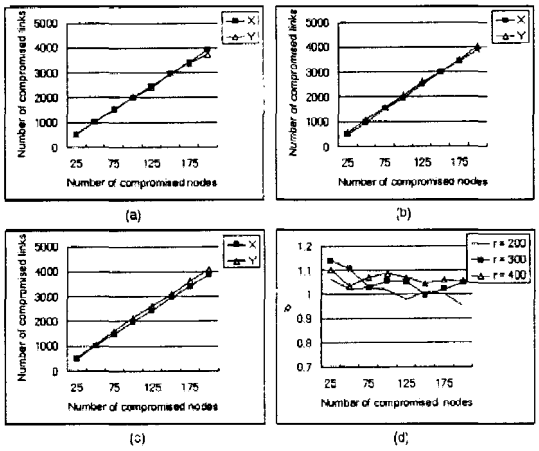

Figure 6. Evaluation of the limited global payoff requirement in 3-SG, where (a) n = 3315, (b) n = 4968, and (c) n = 6621. The ratio ρ is calculated in (d).

calculated in 5(d). We can see that regardless of the key ring size and the number of compromised nodes, the 2-SG scheme constantly meets the limited global payoff requirement (because $\rho \leqq 1$), while the ***3-SG*** scheme, whose ρ value ranges around 1 and up as Figure ***6*** depicts, barely meets the requirement (although it supports the largest network size among all schemes). Note that we do not discuss the k-SG scheme with **k > 3** because in such cases the ρ value is always bigger than one – which apparently disagrees with the limited global payoff requirement.

5. Conclusion

This paper presents a Scalable Grouping ***(SG)*** random key predistribution scheme for large scale DSNs. The ***SG*** scheme divides all nodes into several groups and uses the one-way function to generate group-to-group pairwise keys to increase the connectivity of each key and to enlarge the maximum supportable network size. To improve resilience against node capture, it takes the concept that the link key is composed of some shared keys. When two nodes have two or more shared keys, the scheme uses the exclusive-OR operation to compose the link key -- assuring that the link key used to secure a link is nearly unique. Expenmental results show that, due to its simple and effective designs. the SG scheme is able to yield more enhanced resilience against node capture in large-scale networks, generate higher scalability than existing random key based schemes, and limit global payoff from local compromised nodes.

6. References

[1] S. Zhu, S. Setia, and S. Jajodia, "LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks," ***Proc. 10th ACM conf. on Computer and communication security,*** Oct. ***2003,*** pp.62-72.

[2] **A.** Wadaa. S. Olariu, L. Wilson, and M. Eltoweissy, "Scalable Cryptographic Key Management in Wireless Sensor Networks," ***Proc. 24th Int'l Conf. on Distributed Computing Systems Workshops,*** Mar. 2004, pp. 796-802.

[3] L. Eschenauer, and V. D. Gligor, **"A** key-Management Scheme for Distnbuted Sensor Networks," ***Proc. 9th ACM Conf. on Computer and Communication Security.*** Nov. 2002, pp. 41-47.

[4] S. Y. Wu and S. P. Shieh. "Adaptive Random Key Distribution Schemes for Wireless Sensor Networks." ***Proc.*** 2003 ***Int'l Workshop on Advanced Developments in Software and Systems Security,*** Dec. ***2003.***

[5] J. Spencer, ***The Strange Logic*** of ***Random Graphs,*** Algorithms and Combmatorics.22, Springer-Verlag, 2000.

[6] H. Chan, A. Pemg, and D. Song, "Random Key Predistribution Schemes for Sensor Networks," ***Proc.*** IEEE ***Symp. an Research in Security and Privacy,*** May 2003, pp. 197–213.