A Key Management Scheme in Distributed Sensor Networks Using Attack Probabilities

Siu-Ping Chan, Radha Poovendran and Ming-Ting Sun
spchan@u.washington.edu, {radha,sun}@ee.washington.edu
Department of Electrical Engineering,
University of Washington, Seattle, Washington, USA

***Abstract*— Clustering approaches have been found useful in providing scalable data aggregation, security and coding for large scale Distributed Sensor Networks (DSNs). Clustering (also known as subgrouping) has also been effective in containing and compartmentalizing node compromise in large scale networks. We consider the problem of designing a clustered DSN when the probability of node compromise in different deployment regions is known apriori. We make use of the apriori probability to design a variant of random key predistribution method that improves the resilience and hence the fraction of compromised communications compared to seminal works. We further relate the key ring size of the subgroup node to the probability of node compromise, and design an effective scalable security mechanism that increases the resilience to the attacks for the sensor subgroups. Simulation results show that by using our scheme, the performance can be substantially improved in the sensor network (including the resilience and the fraction of compromised communications) that only sacrifices a small extent in the probability of a shared key exists between two nodes, compared to those of the prior results.**

I. Introduction

Distributed Sensor Networks (DSNs) are being widely used in many applications such as real-time traffic monitoring, military sensing and tracking, wildlife monitoring and tracking, etc. DSNs are ad-hoc mobile networks that may include thousand of sensor nodes with limited computation and communications capabilities. DSN topology can be dynamic and allow addition and deletion of sensor nodes after deployment. Besides, they may be deployed in hostile areas and hence the sensor nodes can be vulnerable to attacks by the adversaries. Because of the limited computation and communication capabilities of the sensor nodes [1], it is difficult to bootstrap the establishment of a secure communications infrastructure from a collection of sensor nodes which may have been pre-initialized with some secret information but have had no prior direct contact with each other [3], [4].

To address the bootstrapping problem in DSNs, Eschenauer et al [3] firstly proposed the random key predistribution scheme that relies on probabilistic key sharing among the nodes of a DSN and uses simple protocols for shared key discovery and path key establishment. The basic idea is that a random pool of keys is selected from the key space. Each sensor node then receives a random subset of keys from the key pool before deployment. Any two nodes able to find a common key within their respective subsets can use that key as their shared secret to initiate communication and to set up the secure connection [1]. Authors in [3] identify that the connection setup process between two nodes can be modeled by the random graph theory given in [2]. A random graph $G(N, p)$ is a graph with N vertices where the edges are formed with probability p. When $p = 0$, the entire graph is disconnected and when $p = 1$, the graph is fully connected. By Erdös and Renyi [2], given N and a desired probability P_c which is defined as the probability that $G(N, p)$ is *connected* and it has a path between any two vertices, we can get the expected degree d of a node (i.e., the average number of edges connecting that node with its network neighbor) to form a *connected* graph as follows [3],

$$d = \frac{N-1}{N}(\ln(N) - \ln(-\ln P_c)), \tag{1}$$

and

$$p = \frac{d}{(N-1)}. \tag{2}$$

For example, if $P_c = 0.99999$ (that means the network will "almost certainly" be connected), and $N = 10000$, then from eq.(1) and (2), $d = 20.7$ and $p = 0.002$, where p represents the probability that a shared key exists between two sensor nodes in the sensor network, and N is the number of sensor nodes in the network. Chan et al [4] further strengthened the basic scheme [3] and proposed the q-composite random key predistribution scheme. The difference between the q-composite scheme and the basic scheme in [3] is that q common keys $(q \geq 1)$, instead of just a singe one, are needed to establish secure communications between a pair of nodes. It was shown in [4] that the q-composite scheme can achieve greatly strengthened security under a small scale attack while trading off increased vulnerability in the face of a large scale physical attack on network nodes. However, the basic scheme [3] and the q-composite scheme [4] considered the sensor deployment to be uniformly distributed and hence, did not make use of any *apriori* deployment knowledge. Later, Du et al. [5] proposed a random key predistribution scheme using deployment knowledge to avoid unnecessary key assignments.

Although prior schemes [3], [4], [5], [6], [7] suggested the use of the random keys to establish the secure connections between the nodes, the idea of different security needs for different locations of nodes is not considered. Besides, the limited key pool will be eventually used up if the number of nodes grow dramatically. The scalability of random key predistribution is a concern and was left unaddressed in the basic and q-composite schemes [3], [4]. We address these two problems in this paper. The main contributions of this paper are summarized in the following:

1) We propose a subgrouping approach to isolate the effect of node captures into one specific subgroup, and to provide scalability for random key predistribution in DSN.

This work was supported by the Croucher Foundation fellowship from Hong Kong.

[1] It is possible that two nodes may share more than one key. Various policies including the q-composite scheme in [4] can be used to generate common keys in this model.

Under the two-level hierarchical subgroup infrastructure, we describe how to perform random key predistribution. We also analyze the corresponding performance metrics including connectivity, resilience and fraction of compromised communications which are discussed in later sections.

2) We propose the idea of considering the probability of node compromise Pnc_i for each subgroup G_i in order to design a scalable security mechanism, such that resilience to the attacks for the sensor subgroup with larger probability of node compromise will be improved. The proposed scheme can maintain flexibility in providing different security concerns for different sensor subgroups. We also present detailed simulation studies to illustrate our approach.

The rest of the paper is organized as follows. We describe the proposed scheme in Section II. We then analyze and evaluate the performance of the proposed scheme in section III. The paper is concluded in Section IV.

II. Proposed Scheme

A. Subgrouping and random key predistribution

By using the deployment knowledge, we can subdivide the whole N nodes group into different subgroups G_i, each with M_i nodes, according to their deployment locations or the probability of node compromise as will be discussed in later sections. Different subgroup nodes can communicate with nodes in other subgroups through the controller node.

Within each subgroup G_i, our random key predistribution scheme consists of three phases, following the idea of the basic scheme [3], which are key predistribution, shared key discovery and path-key establishment. During the key predistribution phase, a large key pool of S keys is first generated. We then randomly pick up m_i keys out of S without replacement and store them into a key ring of each sensor node in the subgroup. The key identifiers of a key ring and the associated sensor identifiers are saved by a controller node.

As mentioned in [3], during the shared-key discovery phase, every node discovers its neighbors in the wireless communication range with which it shares keys. If there exist any common shared key between two nodes, the corresponding secure connection can be set up accordingly. Finally, the path-key establishment phase assigns a path-key to selected pairs of sensor nodes in the wireless communication range that do not share a key but are connected through other nodes at the end of the shared key discovery phase [3].

B. Probability of node compromise Pnc_i for a subgroup G_i

Since the sensor subgroups are located in different areas, they may have different chances of being attacked by the adversaries. Therefore, as discussed, we can actually assign different probability of node compromise Pnc_i into different subgroup G_i as shown in Figure 1.

The Pnc_i for a particular subgroup G_i, can also be defined as the normalized pre-assigned relative security weighting W_i of the subgroup G_i, i.e.

$$Pnc_i = \frac{W_i}{\sum_{i=1}^{G} W_i}, \qquad (3)$$

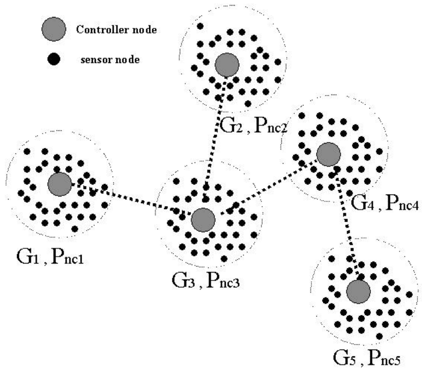

Fig. 1. Network Topology

such that $\sum_{i=1}^{G} Pnc_i = 1$. For example, W_i could be a security weighting between 1 and 10. Larger the W_i means larger the chance that this subgroup G_i will be attacked by the adversaries. For two special cases, if one particular subgroup G_i has the Pnc_i close to 1, that means this subgroup G_i will be mainly attacked by the adversaries and it will have the largest value of W_i, e.g., if only this particular subgroup is located in the hostile area, but not the other subgroups in the network. On the other hand, if all the subgroups share the same value of Pnc_i, i.e. $Pnc_i = 1/G$ where G is the number of subgroups in the network, that means all the subgroup will have the same chance of being attacked by the adversaries and they all have the same value of W_i. One scenario in this case is that every subgroup is located within a confined area and hence every subgroup faces the same chance of being captured by the adversaries. By using Pnc_i in different subgroups, we can actually design an effective scalable security mechanism, such that the resilience to the attacks for the sensor subgroup with larger probability of node compromise will be increased. Besides, it is more flexible to provide different security concerns in different sensor subgroups.

The basic idea is that after the subgrouping process, the whole network with N nodes is divided into G subgroups, each contains M_i members according to their locations. We can assign different Pnc_i values to different subgroups. In fact, we can also vary the size of key ring in a node m_i for a particular subgroup G_i according to Pnc_i. The objective is to improve the resilience R_i to that particular subgroup G_i. In this paper, R_i is defined as the probability that a given key in the subgroup G_i has not been compromised after x_i nodes in that subgroup are captured. However, there is a tradeoff between the probability that a shared key exists between two sensor nodes p_i and the resilience R_i in that particular subgroup G_i. The detailed analysis is presented in Section III.

The random key predistribution scheme is to establish the secure connections between each sensor node within the subgroup. In fact, we can further use the same random key predistribution scheme to securely connect different controller nodes or different subgroups together [2]. The objectives are to facilitate the efficient subgrouping for the subgroup nodes and the controller nodes in each subgroup. It also simplifies

[2]We do recognize that each subgroup must have more than one node able to perform group controller functionalities. However, we do not address this point in this paper.

the design of key distribution and management and provides scalability for node and subgroup addition or removal in the sensor network. According to [1], the controller nodes usually have larger computation and communication capabilities than other sensor nodes within the subgroup. Therefore, we can assume that all the controller nodes are fully connected and the connection links are not easily compromised and broken by the adversaries.

III. Analysis of the Proposed Scheme

A. The probability p_i that a shared key exists between two sensor nodes within a particular subgroup G_i

For simplicity, similar idea to the basic scheme [3] is used in each subgroup during the key setup. Any two nodes within a subgroup share one common key from their key rings can setup a secure link between each other. Although the derivation of the probability p_i that a shared key exists between two sensor nodes in the subgroup is the same as that in the basic scheme, we want to show that under the subgroup network structure, different subgroups can actually have different p_i and m_i.

Given the key pool size $|S|$ and the size of key ring in a node m_i for each subgroup G_i, we can actually calculate p_i as follows,

$$p_i = 1 - \Pr[\text{two nodes do not share any key in a subgroup}], \tag{4}$$

and we can obtain

$$p_i = 1 - \frac{(1-\frac{m_i}{|S|})^{2(|S|-m_i+\frac{1}{2})}}{(1-\frac{2m_i}{|S|})^{(|S|-2m_i+\frac{1}{2})}}. \tag{5}$$

The proof is given in Appendix A.

B. The resilience R_i and the fraction of compromised communications fc_i for a particular subgroup G_i after x_i nodes in that subgroup are captured

In this section, we evaluate the resilience R_i of the subgroup in terms of a node capture attack by calculating the fraction of links in the network that an attacker is able to eavesdrop on indirectly as a result of recovering keys from captured nodes. We define the fraction of compromised communications fc_i as the probability that any secure link setup in the key setup phase between two nodes is compromised when x_i nodes have been captured in the subgroup G_i.

Let the number of captured nodes in a particular subgroup G_i be x_i. We define the resilience R_i as the probability that a given key in the subgroup G_i has not been compromised after x_i nodes in that subgroup are captured. Since each node contains m_i keys, therefore,

$$R_i = \left(1 - \frac{m_i}{|S|}\right)^{x_i}, \tag{6}$$

and the fraction of compromised communications, fc_i for a particular subgroup G_i after x_i nodes in that subgroup are captured is

$$fc_i = 1 - R_i = 1 - \left(1 - \frac{m_i}{|S|}\right)^{x_i}. \tag{7}$$

By eq.(5) and (6), we can show that there is a tradeoff between the p_i and the R_i by varying the m_i. In our proposed scheme, we would like to vary m_i according to the given Pnc_i for that particular subgroup G_i in order to achieve better resilience R_i of the subgroup. However, we may need to sacrifice certain extent of p_i and that implies lower connectivity in the subgroup network. This is better illustrated in Figure 2 that if m_i for a particular subgroup G_i decreases, the resilience R_i will increase. In this simulation, $|S| = 100000$.

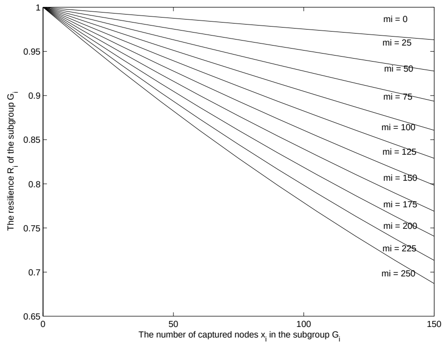

Fig. 2. The resilience of the subgroup G_i against the number of x node being captured with different m_i, $|S| = 100000$

C. The relationship between the probability of node compromise Pnc_i and the size of key ring in a node m_i for a particular subgroup G_i

As discussed in Section II, we would like to reduce the value of m_i, given the Pnc_i for a particular subgroup G_i, in order to increase the resilience R_i of the subgroup to the attack. However, there is a tradeoff between the resilience R_i of the subgroup and the probability of a shared key exists between two nodes in the subgroup, p_i. Later, by using the simulation results, we can show that this tradeoff is desirable.

Given the probability of node compromise Pnc_i for a particular subgroup G_i, the resilience R_i should be proportional to Pnc_i. As both R_i and Pnc_i are values in $[0, 1]$, we would like to find the m_i such that R_i (which is defined as the probability that a given key un the subgroup G_i has not been compromised after x_i nodes in that subgroup are captured) is larger than Pnc_i, i.e.

$$R_i = \left(1 - \frac{m_i}{|S|}\right)^{x_i} \geq Pnc_i. \tag{8}$$

From eq.(8),

$$m_i \leq |S|\left(1 - (Pnc_i)^{\frac{1}{x_i}}\right). \tag{9}$$

Therefore, we can obtain the upper bound or the maximum value of m_i, such that the resilience of that particular subgroup is larger than Pnc_i,

$$m_{imax} = |S|\left(1 - (Pnc_i)^{\frac{1}{x_i}}\right). \tag{10}$$

Eq.(10) states that in order for the probability of a shared key exists between two sensor nodes, m_i should not exceed m_{imax}. Let m_{io} be the value of m_i used to maintain the value of p_i for a subgroup and a given $|S|$ according to eq.(5). For example, if $p_i = 0.33$ and $|S| = 100000$, then $m_{io} = 200$ from eq.(5). Setting m_i to be smaller than m_{imax} will ensure the resilience R_i is greater than or equal to Pnc_i. However, a smaller m_i will cause a smaller p_i which will affect the connectivity. We set m_i to m'_i according to the following strategy which represents a reasonable compromise between the resilience and the connectivity.

$$R_i(Pnc_i) = \left(1 - \frac{m'_i}{|S|}\right)^{x_i}, \tag{11}$$

and

$$p_i(Pnc_i) = 1 - \frac{(1 - \frac{m'_i}{|S|})^{2(|S| - m'_i + \frac{1}{2})}}{(1 - \frac{2m'_i}{|S|})^{(|S| - 2m'_i + \frac{1}{2})}}, \tag{12}$$

where if $(m_{imax} < m_{io})$, then let

$$m'_i = m_{imax}, \tag{13}$$

and from eq.(12), if $(p_i(Pnc_i) \leq p_{imin})$ with m'_i calculated from eq.(13), then let

$$m'_i = m_{imin}, \tag{14}$$

such that $p_i(Pnc_i) = p_{imin}$.

On the other hand, if $(m_{imax} \geq m_{io})$, then let

$$m'_i = m_{io}. \tag{15}$$

For the simulations in this paper, we let p_{imin} to be $0.5p_i$. For example, if $m_{io} = 200$, $|S| = 100000$ and $p_i = 0.33$, then $p_{imin} = 0.165$ and $m_{imin} = 135$. Similarly, we can also determine the function of fc_i, given Pnc_i for a particular subgroup G_i,

$$fc_i(Pnc_i) = 1 - R_i(Pnc_i) = 1 - \left(1 - \frac{m'_i}{|S|}\right)^{x_i}. \tag{16}$$

D. The resilience, the fraction of compromised communications and the probability of a shared key exists between two nodes for the whole sensor network

Given Pnc_i for each subgroup G_i, we can compute m'_i for each subgroup G_i. As discussed before, with m'_i and $|S|$, we can calculate the function of the resilience $R_i(Pnc_i)$ of the subgroup to the attack, the fraction of compromised communications $fc_i(Pnc_i)$ and the probability of a shared key exists between two nodes $p_i(Pnc_i)$ of the subgroup, given that Pnc_i is known. Therefore, we can calculate the resilience R, the fraction of compromised communications fc and the probability of a shared key exists between two nodes p for the whole sensor network with G subgroups, respectively, i.e.

$$R = \sum_{i=1}^{G} R_i(Pnc_i)Pnc_i = \sum_{i=1}^{G} \left(1 - \frac{m'_i}{|S|}\right)^{x_i} Pnc_i, \tag{17}$$

$$fc = \sum_{i=1}^{G} fc_i(Pnc_i)Pnc_i = \sum_{i=1}^{G} \left(1 - \left(1 - \frac{m'_i}{|S|}\right)^{x_i}\right) Pnc_i, \tag{18}$$

$$p = \sum_{i=1}^{G} p_i(Pnc_i)Pnc_i = \sum_{i=1}^{G} \left(1 - \frac{(1 - \frac{m'_i}{|S|})^{2(|S| - m'_i + \frac{1}{2})}}{(1 - \frac{2m'_i}{|S|})^{(|S| - 2m'_i + \frac{1}{2})}}\right) Pnc_i. \tag{19}$$

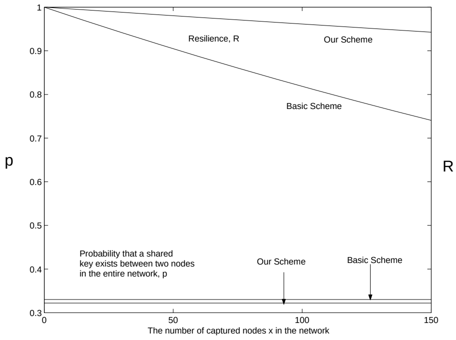

Fig. 3. The comparison between the basic scheme and our scheme in terms of R and p against the number of x node being captured , $|S| = 100000$, $m = 200$ keys, $p = 0.33$, $N = 10000$ nodes, $M_i = 2000$ nodes, $Pnc_i = \{0.2, 0.2, 0.2, 0.2, 0.2\}$

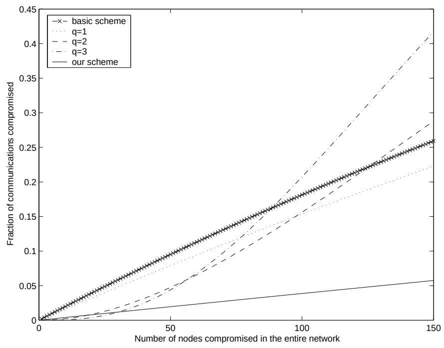

Fig. 4. The fraction of compromised communications fc against the number of x node being captured in the network for the basic scheme, the q-composite scheme and our proposed scheme, $m = 200$ keys, $p = 0.33$, $N = 10000$ nodes, $M_i = 2000$ nodes, $Pnc_i = \{0.2, 0.2, 0.2, 0.2, 0.2\}$

E. Simulation Results

We compare our results with the basic scheme [3] and the q-composite scheme [4] to evaluate the performance. Figure 3 shows the performance comparison between our proposed scheme and the basic scheme, where $|S| = 100000$, $m = 200$, $p = 0.33$, $N = 10000$ and $M_i = 2000$. The resilience R and the probability that a shared key exists between two sensor nodes p in the distributed sensor network are investigated. In this simulation, there are total 5 subgroups with the same value of Pnc_i. The result clearly shows that our scheme can actually achieve a continuous increase in resilience with the increasing number of captured nodes x in the whole network compared to those of the basic scheme. By using the ideas of subgrouping and Pnc_i, the performance gain is actually achieved by isolating the effect of captured nodes into one subgroup and using smaller m_i in each subgroup G_i in our scheme compared to that of other schemes. It also shows that our proposed scheme only needs to sacrifice a small constant decrease of the probability that a shared key exists between two sensor nodes in the sensor network compared to that of the basic scheme (e.g. 2.3% in this simulation), which is approximately only 0.01% decrease in P_c. Comparing to the gain of 27% in resilience, this tradeoff is desirable.

Similarly, Figure 4 shows that our proposed scheme substantially lowers the fraction of compromised communications compared to those of other prior schemes [3], [4] after x nodes are captured by the adversaries in the sensor network for the case of $m = 200$ and $p = 0.33$.

Another set of simulations for the case of $m = 200$ and $p = 0.33$ is done in Figure 5 and Figure 6 to investigate the effects of using different Pnc_i in different subgroups. In these simulations, there are total 5 subgroups with different values of Pnc_i in different subgroups. We let Pnc_i for the subgroup G_1 be 0.8 and all the others be 0.05. Figure 5 shows that our proposed scheme can also substantially lower the fraction of compromised communications compared to those of other prior schemes after x nodes are captured by the adversaries in the entire network. Besides, the result shows a larger fc than the case of all the subgroups with the same value of Pnc_i in Figure 4.

Figure 6 shows the effects on the resilience R_i and the probability that a shared key exists between two nodes p_i for different subgroups G_i with different Pnc_i after x nodes are captured by the adversaries in the entire network. Since the subgroup G_1 has the largest Pnc_i value among the other subgroups, the result shows that our proposed mechanism can lower the p_1 in order to achieve larger resilience R_1 for that particular subgroup. Besides, the p_1 value cannot be further reduced as there is a lower bound p_{imin} to maintain the minimum connectivity of the network. In this simulation, the entire network by using our scheme needs to sacrifice up to 38% in p, which is approximately only 0.04% reduction in P_c.

IV. CONCLUSIONS

This paper proposes a variant of random key predistribution scheme for bootstrapping the clustered Distributed Sensor Network (DSN) when the probability of node compromise in different deployment regions is known apriori. The cluster-based hierarchical topology not only isolates the effect of node compromise into one specific subgroup and provides scalability for node and subgroup addition, but more importantly, it simplifies the design of key management scheme for the sensor networks. With apriori knowledge of the probability of node compromise, an effective scalable security mechanism that increases the resilience to the attacks for the sensor subgroups is designed. Simulation results demonstrated the substantial performance improvement (including the resilience and the fraction of compromised communications) compared to that of the prior schemes [3], [4] by using the proposed scheme.

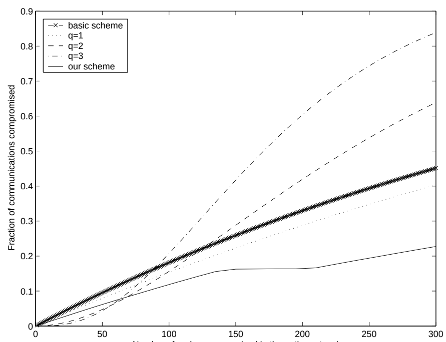

Fig. 5. The fraction of compromised communications fc against the number of x node being captured in the network for the basic scheme, the q-composite scheme and our proposed scheme (different Pnc_i in different subgroups), $m = 200$ keys, $p = 0.33$, $N = 10000$ nodes, $M_i = 2000$ nodes, $Pnc_i = \{0.8, 0.05, 0.05, 0.05, 0.05\}$

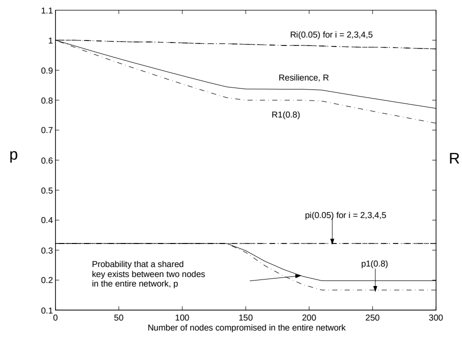

Fig. 6. The resilience R_i and the probability that a shared key exists between two nodes p_i for different subgroups G_i with different Pnc_i against the number of x nodes being captured in the network, $m = 200$ keys, $p = 0.33$, $N = 10000$ nodes, $M_i = 2000$ nodes, $Pnc_i = \{0.8, 0.05, 0.05, 0.05, 0.05\}$

REFERENCES

[1] D. W. Carman, P. S. Kruus and B. J. Matt,"Constraints and Approaches for Distributed Sensor Network Security," Sept 1, 2000. NAI Labs Technical Report No.00-010.

[2] J. Spencer, "The Strange Logic of Random Graphs, Algorithms and Combinatorics 22," *Springer-Verlag*, 2000, ISBN 3-540-41654-4.

[3] Laurent Eschenauer and Virgil D.Gligor, "A Key-Management Scheme for Distributed Sensor Networks," *Proc. of the 9th ACM conference on Computer and Communications Security*, Nov. 2002, pp.41-47.

[4] H. Chan, A. Perrig, and D. Song, "Random Key Predistribution Schemes for Sensor Networks," *IEEE Symposium on Security and Privacy*, May 2003, pp.197-213.

[5] W. Du, J. Deng, Y.S. Han, S. Chen, P.K. Varshney,"A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge," *IEEE INFOCOM 2004*.

[6] W. Du, J. Deng, Y.S. Han, and P.K. Varshney,"A Pairwise Key Predistribution Scheme for Wireless Sensor Networks," *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS)*, Oct. 27-31 2003, pp.42-51.

[7] D. Liu, and P. Ning,"Establishing Pairwise Keys in Distributed Sensor Networks," *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS)*, Oct. 27-31 2003, pp.52-61.

APPENDIX A

Given the key pool size $|S|$ and the size of key ring in a node m_i for each subgroup G_i, we can actually calculate p_i as follows,

$$p_i = 1 - \Pr[\text{two nodes do not share any key in a subgroup}]. \quad (1)$$

To compute the probability that two key rings do not share any key in a subgroup, we note that each key of a key ring is drawn out of a pool of $|S|$ keys without replacement. Thus, the number of possible key rings equals

$$\frac{|S|!}{m_i!(|S| - m_i)!}. \quad (2)$$

Assuming that the first ring is picked, therefore, the total number of possible key rings that do not share a key with key ring for the nodes in the subgroup G_i is the number of key rings that can be drawn out of the remaining $|S| - m_i$ unused key in the key pool,

$$\frac{(|S| - m_i)!}{m_i!(|S| - 2m_i)!}. \quad (3)$$

The probability that no key is shared between the two rings for the nodes in the subgroup G_i is the ratio of the number of rings without a match by the total number of possible key rings,

$$\frac{(|S| - m_i)!(|S| - m_i)!}{|S|!(|S| - 2m_i)!}. \quad (4)$$

Therefore, p_i, the probability that a shared key exists between two sensor nodes within a particular subgroup G_i equals

$$p_i = 1 - \frac{((|S| - m_i)!)^2}{|S|!(|S| - 2m_i)!}. \quad (5)$$

Since $|S|$ is very large, we can use Stirling's approximation to simplify the expression of p_i and obtain

$$p_i = 1 - \frac{(1 - \frac{m_i}{|S|})^{2(|S| - m_i + \frac{1}{2})}}{(1 - \frac{2m_i}{|S|})^{(|S| - 2m_i + \frac{1}{2})}}. \quad (6)$$

In order to test the accuracy of using Stirling's approximation to simplify the eq.(5), we try $|S| = 100000$ and $m_i = 200$ in both eq.(5) and eq.(6). The accuracy is 99.99%.