Key Distribution in Mobile Heterogeneous Sensor Networks

Vijay Bulusu, Arjan Durresi, Vamsi Paruchuri, Mimoza Durresi
Department of Computer Science
Louisiana State University
Baton Rouge, LA 70803
Email: durresi@csc.lsu.edu

Raj Jain
Department of Computer Science and Engineering
Washington University in St. Louis
St. Louis, MO 63130
Email: jain@cse.wustl.edu

***Abstract*— Key predistribution is a popular technique for key distribution in sensor networks. The schemes available in current literature using this approach are for nodes with no or limited mobility. In this paper we present two key predistribution based scheme for heterogeneous networks i.e. networks which consist of nodes which are stationary as well as highly mobile. The existing schemes make use of only one key pool to establish links between the staionary and the mobile nodes. This restricts the mobility of nodes to one specific network. If the same key pool is used in multiple networks, the compromise of keys in one network would lead to compromise of keys in all the networks. We present two different solutions to this problem. The first approach uses a separate disjoint key pool to establish links between the stationary and mobile nodes of the network. In the second approach we take a large key pool and segment it into smaller key pools. Each of these segments acts as the key pool for different stationary sensor networks. The mobile nodes get keys from the aggregate of all these segments. The aggregate key pool can have some segments which can be used for future deployments. We compare the two schemes and analyze their performance. The schemes only deal with secure key distribution between the mobile and stationary nodes. It is assumed that the stationary nodes of the sensor networks are securely connected.**

I. Introduction

A Distributed Sensor Network (DSN) consists of a large number of autonomous, self-organizing sensors with limited battery power, computational power, communication range and memory. These nodes communicate through the wireless medium. Each node is equipped with integrated sensors, data processing capabilities and short-range radio communications.

Sensor Networks can be used in a variety of applications like military sensing and tracking, environmental monitoring, patient monitoring and tracking, smart environments, Disaster Management etc. The sensor nodes are deployed in large numbers in or close to the phenomenon [1]. These nodes typically sense the physical environment and send relevant data to a base station. In many applications like protection from forest fires, chemical attacks, military surveillance, home automation [2], [11] etc the use of mobile sensor nodes is fundamental. The nodes themselves may not move, but may be placed on the mobile objects which move in the network. For e.g. sensor nodes on a mobile tank of hazardous chemicals would communicate with other sensor nodes in case of a leak. To detect and extinguish forest fires, a sensor node may be placed on a fire truck which would interact with other stationary sensor nodes on the ground and guide the truck to the exact location of the fire. Several other military applications can be thought of where these mobile nodes could be very useful. Another possible application is navigation using these networks. We envisage many applications where people could navigate through sensor networks using common omnipresent devices like cellular phones. For e.g. a man stuck in a building on fire may use his cellular phone to interact with the stationary sensor networks deployed in the building to find the best escape route. All these problems can be modelled as mobile nodes interacting with stationary nodes in a sensor network.

Many of these applications transmit critical data over the network which makes security important. The resource starved nature of these nodes and the fact that they communicate in the wireless medium makes data confidentiality and integrity non-trivial. Traditional schemes involving asymmetric key cryptography are not feasible because of their energy requirements[3]. In such an environment, key distribution is one of the most difficult tasks because it has to be accomplished in an unsecured environment. The limited power and computation power make schemes like Diffie-Hellman[6] and RSA[12] undesirable. These limitations make key predistribution a viable, practical and scalable alternative [4], [5], [7], [8], [9], [10], [14]. It involves loading of key information before their deployment. The main disadvantage of key predistribution is that when a node is physically captured all the keys present in that node are known to the adversary. This not only compromises the links established by the captured nodes but also compromises links between uncompromised nodes.

All existing schemes make use of the same key pool for stationary and mobile nodes. Although this approach works fine when the mobile nodes are restricted to one network, they fail when the mobile nodes need to move through multiple networks a great geographical distances. The use of the same key pool in all networks is not possible because the capture of nodes in one scheme would compromise the secure links established in other networks. To address these problems we propose two schemes for secure key predistribution between the stationary nodes and the mobile nodes of the sensor networks. The first scheme uses a separate key pool for links between mobile and static nodes. From this key pool the

mobile and stationary nodes randomly select m keys and $e(e << m)$ keys respectively. Having fewer keys in the stationary nodes ensures that the capture of a stationary node compromises a small fraction of the mobile key pool. The second scheme uses a large key pool which is segmented into smaller key pools. All the nodes of a particular stationary network select m keys randomly from one of the small segments whereas the mobile nodes select m nodes from the entire key pool. It is ensured that the probability that a mobile node would have some keys from each of the segments is high.

We analyze the performance, merits and demerits of both these schemes and compare their performace through mathematical analysis and simulations. Both schemes assume secure well connected stationary networks. To minimize overhead and increase security our scheme does not attempt to connect the mobile node with all the stationary nodes. Our scheme instead allows the mobile node to communicate with some(not all)stationary nodes from all points in a network. This is ensured by the unequal sharing of keys between the mobile and stationary nodes.

Our schemes are designed to minimize the compromise of secure links mobile and stationary nodes by the capture of both stationary and mobile nodes. The capture of a mobile nodes in the scheme that uses a separate mobile key pool would compromise a larger portion of the key pool than the capture of a stationary node because the mobile nodes have a more keys from the mobile key pool. In the segmented key pool based scheme, the compromise of a mobile node would compromise a small portion of the keys from each segment of the key pool. Therefore, the total number of keys compromised when a mobile node is captured is lesser than the separate mobile key pool based scheme. The stationary nodes are deployed in hostile inaccessible regions whereas the mobile nodes are typically deployed of objects of importance. Based on this we believe that the capture of mobile nodes is much harder than the capture of a stationary node. Also the number of mobile nodes is going to be considerably less than the stationary nodes.

The remainder of this paper is organized as follows. Section 2 discusses some existing schemes in current literature which are relevant to our scheme. We then present the two schemes in section 3. Section 4 has all the mathematical analysis, simualations and comparison of the two schemes. Future work and conclusion are provided in section 5.

II. Previous Work Done

To the best of our knowledge this is the first attempt at developing a key distribution scheme for mobile nodes with unrestricted mobility over multiple sensor networks. This scheme assumes that the networks of stationary nodes are well connected using any of the existing schemes. We now discuss some of the existing schemes that are relevant to the our schemes.

Eschenauer and Gligor proposed the random key predistribution scheme[8]. This is based on the interesting properties observed in random graphs where $G(n,p)$ is a graph of n vertices with p being the probability of there being an edge between any two vertices of the graph. For monotone properties there exists a value of p such that the probability of the graph being connected moves from "non-existent" to "certainly true" [13]. When two neighboring nodes share a key, then they are able to establish a session key.This is further improved by Chan et al. [5] by increasing the number of keys to be shared between nodes to $q\,(q > 1)$.

The most important information that can benefit key predistribution is the knowledge of nodes that are likely to be neighbors after deployment. No such information is assumed in the above schemes. The scheme presented by Du et al.[7] uses this knowledge to improve security and connectivity. In these schemes the probability of two nodes sharing a key is based on the probability of the nodes being neighbors. Nodes in different parts of the network have different keys which makes them unsuitable for networks with mobile nodes. These schemes make use of deployment knowledge to offer better security and connectivity.

We want the mobile nodes to have the ability to operate in multiple sensor networks each of which would have keys from a separate key pool. This would make the existing schemes ineffective because the mobile nodes would be able to operate in only a small portion of the network. Schemes that assume deployment knowledge face the same problem and hence can not be used with mobile nodes. We address this problem by using a different key pool for connecting mobile nodes to static nodes and also by using disjoint segments of a large key pool for static nodes and the whole key pool for the mobile nodes. Our schemes achieves this with very little memory overhead on the static nodes of the network.

III. Our Schemes

Mobile nodes operate in multiple networks of stationary nodes. When a mobile node moves into a particular network of stationary nodes it interacts with them. This paper presents two schemes which are able to establish secure links between the mobile nodes and stationary nodes of a sensor network. Our schemes ensures confidentiality and integrity of the messages transmitted between the mobile and stationary nodes. Both these schemes minimize the storage overhead on the stationary nodes of the network. We also present a tradeoff between security and connectivity for both our schemes. We now describe the two schemes in detail.

A. Separate Key pool based scheme

In this scheme we use a separate key pool to connect mobile nodes with the stationary nodes of the network. Each mobile node randomly selects some keys from this key pool. All stationary nodes also select some keys from this key pool randomly before they are deployed. The number of keys from the mobile key pool in each mobile node is far greater than the number of these keys in each stationary nodes. This not only reduces the overhead on stationary nodes but also reduces the number of keys compromised when stationary nodes are captured. The advantage of this scheme is that

the communication between mobile and stationary nodes is independent of the key distribution scheme used to securely connect the stationary network. We divide our scheme into different stages which are key predistribution, key discovery and location key establishment. We now present each of these stages briefly

- ***Key predistribution:*** This stage is performed before the nodes are deployed. A mobile key pool S of size $|S|$ is generated along with the key identifiers. All mobile and stationary nodes are given m and $e\,(e << m)$ keys from S respectively.
- ***Session Key discovery:*** When a mobile node wants to talk to the stationary nodes of the network, it broadcasts the list of its key identifiers. The static nodes match the list of broadcasted identifiers with their own identifiers. If a static node shares a key with the mobile node it establishes a secure session key with the mobile node. The mobile nodes may establish more than one links (if possible) to increase redundancy and reliability.
- ***Location key establishment:*** Once a mobile node establishes a session key at a particular location, it can store the key in its memory. Whenever the node visits that particular location again, it could reuse the session key. This would make session key discovery a one time overhead. This would make key Discovery for a location, a one time overhead. If the overhead of storing keys at all the locations is high, the mobile nodes could store the session keys for only the frequently visited locations.

Having fewer keys in stationary nodes reduces the probability of a mobile node sharing a key with a particular stationary node. But, we assume that the density of the stationary nodes in the deployment region is high. As a result the probability that the mobile node would establish secure links with some of its stationary neighbors is high. The mobile nodes can communicate with all the stationary nodes of the network through these nodes.

B. Segmented Key pool based scheme

The idea behind this scheme is to give the mobile nodes a small number of keys from the key pools of all the stationary networks with which the mobile nodes may interact. The number of keys from each of the key pools may depend on the frequency with which the mobile node visits a particular network. The number of keys from the key pool of a stationary network in a mobile node is much less than the number of nodes that are present in a stationary node. Like the previous scheme we leverage on the fact that a mobile node has several stationary nodes in its communication range at any point inside the network. Even though the probability of a mobile node sharing a key with a particular stationary node is small, the probability of sharing a common key atleast some nodes in its neighborhood is high. This allows the mobile nodes to interact with the stationary nodes without any memory overhead on the static nodes.

Like the previous scheme using Separate key pools, this scheme also has three stages which are key predistribution, session key discovery and location key discovery. In Key predistribution we generate a large key pool S if size $|S|$. This pool is divided into segments $S_1, S_2...S_n$ and one of these segments is assigned to each sensor network. All of the static nodes of a network i randomly select m keys from the key pool S_i. The mobile nodes on the other hand randomly select m keys from S. The session and location key discovery stages are exactly same as in the case of Separate key pools.

On an average the fraction of keys in a mobile node from a particular segment S_i is $1/n$. It is not mandatory that all n segments be in use. Some segments can be kept for future deployments. This makes the segmented approach extremely flexible. We compare the two schemes extensively in the next section.

IV. Analysis and Simulations

The metrics for the analysis of this scheme are

- **Security:** It is the probability of a secure link between a mobile and stationary being compromised with the capture of a node.
- **Connectivity:** It is the probability of a mobile node establishing q secure links with the stationary nodes from any point in the network.
- **Overhead:** It is the memory overhead on the stationary nodes to store the keys of the mobile key pool.

A. Mathematical Analysis

In this section we look at the performance of Key predistribution using separate and segmented key pools using the metrics discussed above.

1) Key Predistribution using separate key pool: In this scheme we have a separate mobile key pool from which the mobile and stationary nodes randomly select keys. These keys are used to establish secure links between the stationary nodes and the mobile nodes. Let e be the number of keys from the pool S in each stationary node. We analyze the situation where a mobile node needs to establish q secure connections from a point in the deployment region. Let M be a mobile node and K_s be the union of all the keys from the mobile key pool in the stationary nodes within the communication range of M. Let K_m be the number of keys from the mobile key pool S of size $|S|$ in M. The probability of establishing q secure links can be obtained by

$$P = \frac{\left(\begin{array}{c} |S| \\ q \end{array}\right) \cdot \left(\begin{array}{c} |S|-q \\ K_m+K_s-q \end{array}\right) \cdot \left(\begin{array}{c} K_m+K_s-q \\ K_m-q \end{array}\right)}{\left(\begin{array}{c} |S| \\ K_m \end{array}\right) \cdot \left(\begin{array}{c} |S| \\ K_s \end{array}\right)} \tag{1}$$

In this equation we know the values of S,K_m and q. By fixing the value of P in the equation we can obtain the value of K_s. Key distribution in static nodes should be done so that the combination of all the static nodes in the neighborhood of a mobile node should have atleast K_s keys. Each node of the stationary network has e keys out of the key pool S of size $|S|$. The probability of a particular key from the key pool

being in any node of the network is $\frac{e}{S}$. The probability of that key not being present in one node of the network is $\left(1-\frac{e}{S}\right)$. The probability of that particular key being present in the x stationary nodes in the neighborhood of the mobile node is

$$P = 1 - \left(1 - \frac{e}{|S|}\right)^x \tag{2}$$

Therefore the total number of keys in the x neighbors of a mobile node K_m are

$$K_s = S.\left[1 - \left(1 - \frac{e}{|S|}\right)^x\right] \tag{3}$$

By fixing the value of P in the above equation, we obtain the value of x. This gives us the minimum number of stationary nodes within the communication range of the mobile node for it to establish a session key with probability P. If R is the communication range of the mobile and stationary nodes, then the total area in their communication range is πR^2 . For the mobile node to share q keys it needs x stationary nodes in its neighborhood. Based on this the required density of stationary nodes d is

$$d = \frac{x}{\pi R^2} \tag{4}$$

The value d gives the minimum number of stationary nodes per unit area which would allow the mobile node to have q secure links from all points in the network with a probability P.

We now analyze the affect of node capture on the security of the scheme. Let c static nodes be captured. The capture of a static node compromises e keys. The probability of a key being compromised by the capture of a static node is $\frac{e}{|S|}$. The probability of a key not being compromised is $\left(1-\frac{e}{|S|}\right)$. The probability of a key not being compromised after the capture of c static nodes is $\left(1-\frac{e}{|S|}\right)^c$. Hence the probability of a key being compromised after the capture of c nodes is

$$P = 1 - \left(1 - \frac{e}{|S|}\right)^c \tag{5}$$

Equation (3) shows that an increase in the value of x would increase the value of K_s. Equations (1) and (5) show the tradeoff between security and connectivity. According to equation (1) an increase in K_s increases the probability of a mobile node sharing q keys with a stationary node in its neighborhood. On the other hand equation (5) shows that an increase in the value of $e\,(\propto K_s)$ increases the probability of a key being compromised incase of node capture. This gives us the tradeoff between security and connectivity.

2) Key Predistribution using segmented key pool: In this scheme we have a large key pool which is divided into segments. Each of these segments is assigned to a sensor network. All stationary nodes randomly select keys from one of the segments whereas the mobile nodes select keys from the union of all these individual segments. Each mobile node randomly selects m keys from a key pool S. This key pool S is divided into n mutually disjoint segments $S_1, S_2...S_n$. Each stationary node belonging to a network i obtains m keys from the segmentS_i. If a mobile node wants to establish q secure links with the network from any point of deployment and K_s is the union of all the keys in the nodes in the neighborhood of a mobile node. The probability of a mobile node with m keys and n segments establishing q secure links from a particular location in the sensor network is

$$P = \frac{\begin{pmatrix} |S_i| \\ q \end{pmatrix} \cdot \begin{pmatrix} |S_i| - q \\ K_s + \frac{m}{n} - q \end{pmatrix} \cdot \begin{pmatrix} K_s + \frac{m}{n} - q \\ K_s - q \end{pmatrix}}{\begin{pmatrix} |S_i| \\ K_s \end{pmatrix} \cdot \begin{pmatrix} |S_i| \\ \frac{m}{n} \end{pmatrix}} \tag{6}$$

By fixing the value of P,S_i,q,m and n in this equation we can obtain the value of K_s. Using the value of K_s and replacing e with m in equation (3)we can obtain the value of x which is the number of stationary nodes in the neighborhood of a mobile node which would allow the mobile node to establish q secure links with the stationary nodes with a probability P. Using the value of x in equation (4) we can obtain the density of node deployment.

The capture of nodes reveals the keys present in those nodes to the attackers. If the attacker captures c_i nodes from the network i then the probability of a key being compromised is

$$P = 1 - \left(1 - \frac{m}{|S_i|}\right)^{c_i} \tag{7}$$

An increase in the value of m would improve connectivity but worsen security. This tradeoff can be seen in equations (6) and (7). This is similar to the tradeoff seen between equations (1) and (5).

B. Simulations

In this section we analyze the performance of key predistribution using separate and segmented key pools. Our simulation considers a square deployment area of 200x200m^2 with the communication range of each stationary and mobile node being $20m$. We assume all links to be symmetric meaning that if node A is within the communication range of node B then node B is in the communication range of node A. The capture of nodes by the adversary leads to the compromise of keys. In key predistribution using separate key pools the size of the mobile key pool is assumed to be 10000. In the segmented key pool based scheme the size of each segment is taken as 10000 and the number of keys in the mobile and stationary nodes is assumed to be the same. The number of keys in each mobile node are assumed to be 100. For clear understanding, these simulations we assume that the number of mobile nodes is equal to the number of stationary nodes although we believe that the number of mobile nodes would be much lesser.

In fig.1 we show the relation between connectivity and the density of nodes. In this simulation we increase the number of nodes deployed and analyze the corresponding connectivity. Here connectivity is expressed as a fraction of links established to the total stationary nodes within the communication range of a mobile node. We calculate this value by placing the

mobile node in 100 different locations of the deployment region. The increase in the number of nodes will increase the number of links formed because the mobile node can get connected to more nodes. But an increase in the number of nodes also means that the number of stationary neighbors to a mobile node increase. As a result the ratio of links established to the total neighbors is almost constant with the increase in stationary nodes. This figure also shows that key predistribution using segmented key pools has the best connectivity. This is due to the fact that a stationary node uses the same key pool to establish links with the mobile and stationary nodes.

Through these simulations we want to present the tradeoff's between security, connectivity and overhead. We plot graphs for all possible pairs of these values. fig.2 shows the connectivity with respect to the overhead. For this simulation the value of node capture was kept constant. The increase in overhead results in better connectivity. This is expected because greater the number of keys stored, greater is the probability of the mobile node getting connected to the stationary nodes. Our simulations show that the the increase in the overhead is about the same as the increase in the number of links established.

In fig.3 the relation between secure links compromised and the nodes captured is shown. As the number of nodes captured increases the attacker obtains more key information from the mobile pool and as a result more secure links are compromised. In case of key predistribution using a segmented key pool, the number of links compromised due to node compromise is very high because the same key pool is used by the stationary nodes to connect with other stationary and mobile nodes. For schemes with high rates of node capture, this scheme is would not be suitable. In the case of capture of mobile nodes, the segmented key pool scheme has an advantage because the number of keys from each segment of the key pool is small.

In fig.4 we derive the relation between the increase in overhead and links compromised. We can see that as the overhead increases the links compromised also increase. An increase in overhead means that the number of keys stored in the nodes is increased. Although this results in better connectivity, the capture of one node would reveal a greater portion of the key pool to the adversary. As a result the capture of a node would compromise a lot more keys. We can see that when the nodes captured is kept constant the number of links compromised with the capture of each node increases with the overhead.

In key predistribution using segmented key pools, the mobile nodes must store keys from all the different segments. Each segment is assigned to a different sensor network. As the number of different sensor networks which need to interact with the mobile node increases, the number of keys from the key pool of each segment goes down. This results in reduced connectivity between the mobile and stationary nodes of one particular sensor network. This trend is shown in fig.5. The number of segments does not affect the links compromised because only stationary nodes are vulnerable to node capture.The

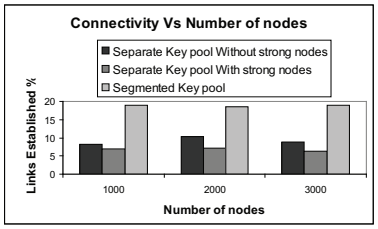

Fig. 1. Relation between connectivity and the density of nodes. Here the memory overhead per node and the node capture are kept constant

number of sensor networks does not influence the number of keys compromised by the capture of each node.

C. Comparison of the Schemes

In this section we analyze the relative strengths and weaknesses of the key predistribution with separate key pools and key predistribution in segmented key pools.

In key predistribution with separate key pools, the number of keys stored in the stationary nodes is much less than in mobile nodes. The capture of stationary nodes leads to the compromise of a very small portion of the network. This scheme scales very well with the increase in sensor networks. The main disadvantage of this scheme is that the mobile key pool must be known before the deployment of stationary nodes. The overhead of this scheme on the stationary nodes is due to the extra memory required to store the keys from the mobile key pool. This overhead is not there in the scheme using segmented key pools. Moreover the connectivity offered by using separate key pool for mobile nodes is less than that offered by the use of segmented key pools.

In key predistribution with segmented key pools, a large key pool is divided into disjoint segments and each of these segments is assigned to a sensor network. The stationary nodes randomly select keys from the key pool segment assigned to their sensor network and the mobile nodes randomly select keys from the whole key pool. This scheme allows the stationary nodes to communicate with other stationary and mobile nodes using the same set of keys stored in their memory. As a result this scheme avoids the overhead of storing extra keys unlike the schemes using a separate key pool. This also ensures better connectivity between the stationary and mobile nodes. The capture of a mobile node would compromise fewer keys between the mobile nodes and a particular stationary network. Also, unlike the previous schemes the keys compromised by the capture of stationary nodes in one network can not be used to compromise the links of another network and incase a network is extensively captured by the attacker, the mobile nodes can stop interacting with that network. The main disadvantage of this scheme is that it is not scalable if the number of networks becomes high. Although the use of one key pool means better connectivity, the number of links compromised incase of node capture is also much higher than the previous scheme. If the probability of the capture of the stationary nodes is higher the mobile nodes, the separate key pool scheme may be used. Otherwise the segmented key

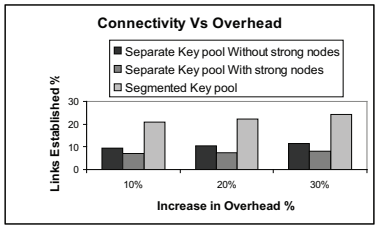

Fig. 2. Relation between connectivity and overhead.In this case the number of nodes captured is kept constant

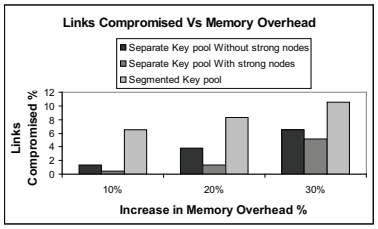

Fig. 4. Relation between overhead and security. In this case the number of nodes captured is kept constant

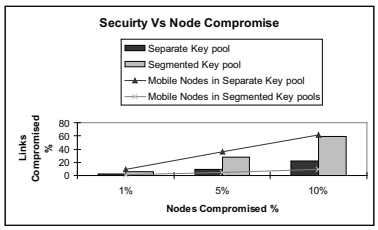

Fig. 3. Relation between connectivity and security.In this case the memory overhead for the keys stored in the static nodes is kept constant.

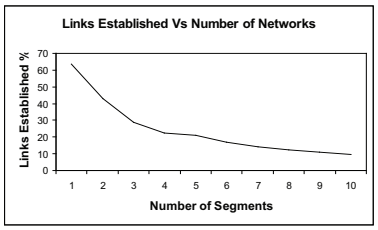

Fig. 5. Relation between connectivity and the number of sensor networks. In this case the total keys in the mobile node is kept constant.

pool based scheme may be used provided that the number of networks in which the mobile nodes need to operate is low.

V. Conclusion

Sensor networks with heterogeneous nodes have a wide range of applications. These applications need to establish secure connectivity between the mobile and the stationary nodes of the network. The mobile nodes may need unrestricted movement through different sensor networks. The existing key predistribution schemes restrict the mobility of the nodes to only one network. In this paper we present two schemes namely, key predistribution using separate key pool and key predistribution using segmented key pool. They allow the mobile nodes to interact with the stationary nodes of different networks. In key predistribution with separate key pool, a separate key pool is used to connect the mobile nodes to the stationary nodes. In key predistribution with segmented key pools, a large key pool is divided into disjoint segments and each of these segments is assigned to a different sensor network. We have performed extensive analysis and simulations to validate these schemes and compare their performance.

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. *Computer Networks*, 38(4):393–422, 2002.

[2] Ian F. Akyldiz and Ismail H. Kasimoglu. Wireless sensor and actor networks: research challenges. *ADHOC Networks*, 2004.

[3] D. W. Carman, P. S. Kruus, and B. J. Matt. Constraints and approaches for distributed sensor network security. Technical Report 00-010, NAI Labs Technical Report, September 1,2000.

[4] H. Chan, A. Perrig, and D. Song. Key distribution techniques for sensor networks. *Wireless Sensor Networks ISBN:1-4020-7883-8*, pages 277–303.

[5] Haowen Chan, Adrian Perrig, and Dawn Song. Random key predistribution schemes for sensor networks. *IEEE Symposium on Security and Privacy*, pages 197–213, May 11–14,2003.

[6] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22:644–654, 1976.

[7] Wenliang Du, Jing Deng, Yunghsiang S. Han, Shigang Chen, and Pramod K. Varshney. A key management scheme for wireless sensor networks using deployment knowledge. *In Proceedings of the IEEE INFOCOM'04*, pages 586–597, March 7–11,2004.

[8] Laurent Eschenauer and Virgil D. Gligor. A key-management scheme for distributed sensor networks. *in Proceedings of the 9th ACM conference on Computers and communications security*, pages 41–47, November 18–22,2002.

[9] Donggang Liu and Peng Ning. Location-based pairwise key establishments for static sensor networks. *ACM workshop on Security in Ad Hoc and Sensor Networks*, 2003.

[10] Donggang Liu, Peng Ning, and Rongfang Li. Establishing pairwise keys in distributed sensor networks. *10th ACM conference on Computers and Communication Security (CCS 03)*, pages 52–61, October 2003.

[11] E.M. Petriu, N.D. Georganas, D.C. Petriu, D. Makrakis, and V.Z.Groza. Sensor based information appliances. *IEEE Instrumentation and Measurement Magazine*, 3(4):31–35, 2000.

[12] R. L. Rivest, A. Shamir, and L. M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[13] J. Spencer. *The Strange Logic of Random Graphs ISBN: 3-540-41654-4*. Springer-Verlag, August 9,2001.

[14] Sencun Zhu, Sanjeev Setia, and Sushil Jajodia. Leap: Efficient security mechanisms for large-scale distributed sensor networks. *10th ACM conference on Computers and Communication Security (CCS 03)*, pages 62–72, October 2003.