Chapter 7

ADAPTIVE RANDOM KEY DISTRIBUTION SCHEMES FOR WIRELESS SENSOR NETWORKS

Shih-I Huang
Department of Computer Science and Information Engineering
National Chiao Tung University

Shiuhpyng Shieh
Department of Computer Science and Information Engineering
National Chiao Tung University

S.Y. Wu
Department of Computer Science and Information Engineering
National Chiao Tung University

Abstract Wireless Sensor Networks (WSNs) are formed by a set of small devices, called nodes, with limited computing power, storage space, and wireless communication capabilities. Most of these sensor nodes are deployed within a specific area to collect data or monitor a physical phenomenon. Data collected by each sensor node needs to be delivered and integrated to derive the whole picture of sensing phenomenon. To deliver data without being compromised, WSN services rely on secure communication and efficient key distribution . In this paper, we proposed two key distribution schemes for WSNs, which require less memory than existing schemes for the storage of keys. The Adaptive Random Pre-distributed scheme (ARP) is able to authenticate group membership and minimize the storage requirement for the resource limited sensor nodes. The Uniquely Assigned One-way Hash Function scheme (UAO) extends ARP to mutually authenticate the identity of individual sensors. The two proposed schemes are effective for the storage of keys in a wireless sensor network with a large number of sensors.

1. Introduction

Wireless Sensor Network (WSN) [Akyilidiz et al., 2002, Estrin et al., 1999] is a kind of network composed of nodes associated with sensors. Each node has the characteristics of small size, limited power, low computation power and wireless access. The sensor node is responsible for collecting and delivering data over wireless network, and it is desirable to keep the delivered data confidential along the wireless transmission path from one node to another. [Tilak et al., 2002, Kong et al., 2001]

To ensure secure peer-to-peer wireless communication [Slijepcevic et al., 2002, He et al., 2003, Heinzelman et al., 1999, Intanagonwiwat et al., 2000, Zhou et al., 1999, Luo et al., 2002, Hubaux et al., 2001, Basagni et al., 2001], the shared session key between any two nodes must be derived [Asokan et al., 2000, Yi et al., 2002, Carman et al., 2000]. Some protocols use a trusted third party to deliver keys to every node [Yi et al., 2003], while other protocols pre-distribute communication keys to all nodes]. [Chan et al., 2003] Since WSNs are self-organized, and trusted third party may not be available, key pre-distribution protocols are often adopted in such networks. However, key pre-distribution protocols need to store session keys in every node. This may be difficult to achieve in a sensor network where thousands of nodes are deployed with limited storage space only enough to store a small number of session keys. It is desirable to design a new key pre-distribution protocol, which can reduce the storage space of session keys for a large WSN without degrading its security.

Much research has been done on key distribution in WSN over the past few years. Carman et al. [Carman et al., 2002] analyzed various conventional approaches for key generation and key distribution in WSN on different hardware platforms with respect to computation overhead and energy consumption [Hodjat et al., 2002, Heinelman et al., 2000]. The results showed that conventional key generation and distribution protocols are not suitable for WSN. To cope with the problem, a key management protocol [Carman et al., 2002] is proposed for sensor networks, which is based on group key agreement protocols and identity-based cryptography . This protocol used Diffie-Hellman key exchange scheme to perform group key agreement . However, the high storage and high computation requirements make it difficult to use.

Perrig et al. [Perrig et al., 2001] proposed a security protocol for sensor networks named SPINS . SPINS uses base station as a trusted third party to set up session keys between sensor nodes. Liu and Ning [Liu et al., 2003] extended Perrig's scheme and proposed an efficient broadcast authentication method for sensor networks. Their scheme uses multi-level key chains to distribute the key chain commitments for the broadcast authentication. Undercoffer et al. [Undercoffer et al., 2002] proposed a resource-driven security protocol , which consider the trade-off between security levels and computational resources. However, in a randomly dispersed wireless sensor network, the base station is not always available for all nodes. Without the base station, a sensor network using SPINS may be disconnected. Therefore, these schemes are not well suitable for sensor networks due to the need of base station. Eschenauer and Gligor [Eschenauer et at., 2002] proposed a key management scheme based on Random Graph Theory . [Chan et al., 2003, Erdoos et al., 1960, Spencer, 2000] The Random Graph Theory is defined as follows. A random graph $G(n, p)$ is a graph with n nodes, and the probability that a link exists between any two nodes in the graph is p. When p is equal to 0, the graph G has no edges, whereas when p is equal to 1, the graph G is fully connected. Erdõs and Rẽnyi [Erdoos et al., 1960] showed the monotone properties of a random graph $G(n, p)$ that there exists a threshold value of p, over which value the property exhibits a "phase transition", i.e. the probability for G to have that property will transit from "likely false" to "likely true". The threshold probability is defined by:

$$p = \frac{ln(n) - ln(-ln(P_c))}{n} \tag{7.1}$$

where P_c stands for desired probability of the property. Furthermore, the expected degree of a node can be calculated by:

$$d = p * (n-1) = \frac{(n-1)ln(n) - ln(-ln(P_c))}{n} \tag{7.2}$$

Therefore, the scheme only needs to select d keys to keep a network connected under probability p. It can then significantly reduce the key space. However, it is discovered that the degree d is proportional to the number of nodes n under the same connectivity probability p. That is, when more nodes are deployed, more storage space is needed in each sensor node. Since the storage space in each node is fixed, the maximum number of nodes that can be deployed is also fixed in this scheme. This characteristic restricts the deployment of sensor nodes and therefore the scalability of this scheme is somewhat limited. To improve the scalability, we propose two key distribution schemes: Adaptive Random Pre-distributed scheme (ARP) and Uniquely Assigned One-way Hash Function scheme (UAO) . Both ARP and UAO schemes pre-distribute keys in each sensor node before its deployment. When the number of sensor nodes increases, both key distribution schemes dynamically adjust itself according to remaining storage space in each sensor node without reducing the connectivity probability p. Both schemes minimize the storage requirement for key management under the same connectivity probability p, and can work well even when a large number of sensor nodes are deployed. In contrast, ARP scheme needs the smallest storage space, while UAO scheme provides the capability of

The rest of this paper is organized as follows: The Adaptive Random Pre-distributed scheme and the Uniquely Assigned One-way Hash Function scheme are presented in Sections II and III, respectively. The evaluation of the schemes are provided in Section IV. Finally, Section V concludes the paper.

2. Adaptive Random Pre-distribution Scheme

ARP scheme is composed of two parts. One is the key pool, and the other is the key selection algorithm . The key pool is used to store randomly generated keys, and the key selection algorithm is to select a set of keys from the key pool. Every node needs to select a set of keys from the key pool by using key selection algorithm before its deployment. These selected keys are saved in each node's storage space. Any two nodes shares a common key is able to securely communicate with each other by using this shared key. In ARP, the key pool is a two-dimensional key pool in which keys are generated in two phases, and are arranged in two-dimensional ordered matrix. The key is pre-generated as follows:

2.1 Key Pool Generation Algorithm

- **Step 1:** Randomly generate t keys, called seed keys, and any t one-way hash functions.
- **Step 2:** For every seed key and one-way hash function, an one-way key chain is generated.

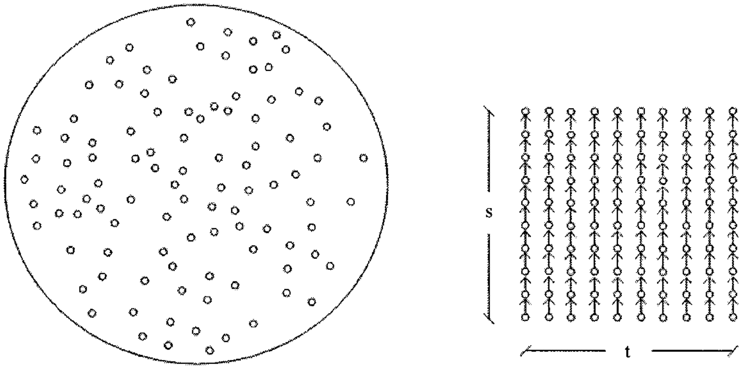
It uses $K_{i,0}$ as initial input, and computes the generated key with an one-way hash function F_i . The generated key is fed back into F_i to generate a third key. The procedure $K_{i,j+1} = F(K_{i,j})$ is repeated until the entire key chain is generated.

Consequently, the key chain KC_0 of length s, is composed of a series of keys, $K_{i,0}$, $K_{i,1}$, . . . , $K_{i,s-1}$. With t seed keys and t one-way hash functions, t key chains generated, namely KC_0, KC_1,. . . , KC_{t-1} .

Figure 7.1 demonstrates the difference between the conventional random key pool and the Two-Dimension Key Pool. As shown in Figure 7.1(a), the original random key pool can be regarded as a set of keys disorderly spread into a large pool. In Figure 7.1(b), keys of the Two-Dimension Key Pool are arranged in an s by t matrix.

2.2 Key Selection Algorithm

After key generation, a key pool of size st is generated. Each sensor needs to randomly choose keys from the key pool by using key selection algorithm described here. The number of keys can affect the connectivity of the entire sensor network and the storage requirement of each sensor node. Fewer keys



(a) The original key pool

(b) The Two-Dimension key pool

Figure 7.1. Unordered key pool and the Two-Dimension key pool with $t = 10$, $s = 10$.

can save storage but lower the probability for a sensor network to be connected. More keys can guarantee higher connectivity probability but at the same time increase the storage requirement. We'll discuss the the relationship of keys, connectivity probability and storage requirement later in the paper.

The key selection algorithm is used to select a set of communication keys by all nodes before its deployment. The detail of the key selection algorithm for ARP scheme is described as follows.

- Step1: Let r be the number of keys each node needed to achieve connectivity among n sensor node with probability p. r can be chosen as d in eq.2. Each sensor node randomly picks up an one-way key chain $KC_i = (KC_{i,0}, KC_{i,1}, \ldots, KC_{i,s-1})$ from the two-Dimension key pool, and use the keys in the key chain.

- Step2: Each sensor node randomly selects the remaining $r' = r - s$ keys from different key chains.

- Step3: Each sensor node has chosen one key chain KC_i and r' single keys. For each sensor node, it will only need to memorize those r' keys and the one-way hash function F_i and seed key $KC_{i,0}$ of the key chain KC_i.

Figure 7.2 shows an example of key selection, where $t = 10$, $s = 10$, and $r' = 5$. The randomly selected one-way key chain is KC_3, and the rest ri' randomly picked keys are $KC_{0,6}$, $KC_{5,8}$, $KC_{6,3}$, $KC_{8,7}$, and $KC_{9,4}$.

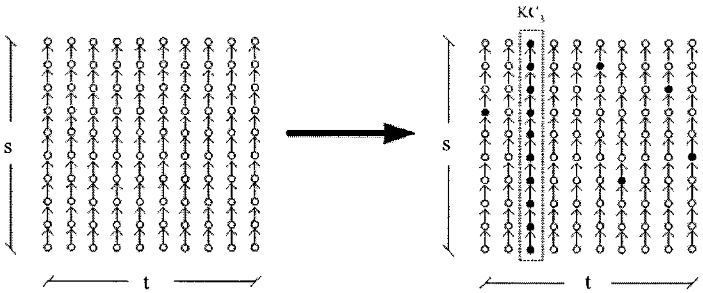


Figure 7.2. A key selection example

3. Uniquely Assigned One-Way Hash Function Scheme

In ARP, any two nodes shared a can directly communicate with each other in a secure way. However, a key in ARP may be shared by more than two nodes, and therefore a node may not be able to authenticate with the shared key the identity of an individual. To cope with the problem, UAO extends ARP to authenticate individual sensor node identities. The detail of UAO is describes as follows.

Each sensor node SN_i is assigned a unique identity ID_i and a uniquely assigned one-way hash function F_i before its deployment. In contrast to ARP key selection algorithm, UAO scheme does not select a key. Instead, it uses ID_i and F_i to decide a key, where ID_i can be the node's MAC address or identifier; and F_i is an one-way hash function. The UAO key decision algorithm is as follows:

3.1 Key Decision Algorithm:

- Step1: Assume the required number of keys to achieve the Random Graph Theory is r. For each sensor node SN_i in the network, SN_i will randomly select r sensor nodes, excluding itself, in the network, denoted as $SN_{v1}, SN_{v2}, \ldots, SN_{vr}$.

- Step 2: For each sensor node SN_{vj}, where j ranges from 1 to r, SN_{vj} uses its unique one-way hash function F_j to generate a unique K_j for SN_i. The K_j is generated by the following equation:

$$K_j = F_j(ID_i)$$

SN_i will memorize all pairs of K_j and ID_j in its key ring.

3.2 Mutual Authentication

After applying key decision algorithm , every node is deployed in a WSN. For communication between two nodes, SN_i and SN_j, SN_i shares unique session key K_j with SN_j, and SN_j shares unique session key K_i with SN_i. Mutual authentication is achieved because SN_i is the only node that owns the unique one-way hash function F_i. If SNi can correctly calculate K_j and decrypt the cipher, then SN_j can authenticate the identity of SN_i. Due to K_j is derived from F_i and ID_j, if SN_j really owns the key K_j then it will make the correct response. Therefore the SN_i will be able to authenticate SN_j with ID_j.

4. Evaluation

To evaluate ARP scheme and UAO scheme, both schemes are analyzed in terms of connectivity and storage space.

4.1 Evaluation of ARP Scheme

To evaluate ARP scheme, the connectivity probability is analyzed because it was observed in the preceding section that ARP is proposed based on Random Graph Theory. If the connectivity probability of different schemes is the same, the scheme requires smaller storage space to store keys.

To evaluate the required probability of connectivity, the network size n and the expected probability Pc of forming a connected graph must be determined. By given n and P_c, we can calculate the threshold probability p and the expected degree d by Equation 7.1 and 7.2. Moreover, since a sensor node cannot communicate with all other nodes in the network, only a limited number of neighbor nodes n' can be contacted. Therefore, the probability of sharing a common key between any two nodes in a neighborhood is:

$$p' = \frac{d}{n'} \tag{7.3}$$

Also, the required key ring size s and the key pool size K to achieve the probability of neighborhood connectivity can be determined.

We denote the probability of any two nodes in the neighborhood sharing at least one common key in Two-Dimension Key Pool Selecting scheme as p'. It is proved that p' is related to the number of key chains t, key chain length s, and the number of selected keys r'. The p' can be calculated by one minus the probability that any two nodes in the neighborhood do not sharing any key. To calculate the probability that any two nodes A and B do not sharing any key, the calculation can be categorized into four parts:

1 A's one-way key chain does not match with B's one-way key chain.

2 A's one-way key chain does not match with any B's selected keys.

3 A's selected keys do not match with B's one-way key chain.

4 A's selected keys do not match with any B's selected keys.

Since B selects one hash function and r' selected keys in different key chains, A's one-way key chain must belong to the rest of the $t-(r'+1)$ key chains. Therefore, the probability of matching both the first and the second conditions are $\frac{t-(r'+1)}{t}$.

For the third condition, we randomly choose r' key chains from the key pool. A's r' selected keys must not belong to A's key chain. As to match the third condition, it must not also belong to B's key chain. Thus the probability can be calculated as

$$\frac{\begin{pmatrix} t-2 \\ r' \end{pmatrix}}{\begin{pmatrix} t-1 \\ r' \end{pmatrix}} = \frac{t-r'-1}{t-1}$$

For the fourth condition, it is assumed that A and B have exactly i selected keys belonging to the same i key chains and the probability that A and B have exactly i selected keys belonging to the same i key chains as $p(i)$. There are $\begin{pmatrix} r' \\ i \end{pmatrix}$ ways to pick i common key chains from B's selected key ring, and there are only $(t-2-r')$ key chains to pick up the remaining A's $(r'-i)$ selected keys. This is because we have to eliminate A's and B's key chains and the other r' key chains that B's r' selected keys belong to. Thus there are $\begin{pmatrix} t-2-r' \\ r'-i \end{pmatrix}$ ways to pick up the remaining $(r'-i)$ key chains. The total number of ways for A to choose r' key chains is $\begin{pmatrix} t-2 \\ r' \end{pmatrix}$. Therefore we get the following equation:

$$p(i) = \frac{\begin{pmatrix} r' \\ i \end{pmatrix}\begin{pmatrix} t-2-r' \\ r'-i \end{pmatrix}}{\begin{pmatrix} t-2 \\ r' \end{pmatrix}}$$

Moreover, considering that A and B have exactly i selected keys belonging to the same key chains, the probability that A's selected keys do not match with any B's selected keys becomes:

$$p(i)(1-\frac{1}{s})^t$$

Hence, to calculate the probability of matching the fourth condition, we have to consider all possible value of i, where $i = 0, 1, 2, \ldots, r'$. Thus the probability for the fourth condition is:

$$\sum_{i=0}^{r'} p(i)(1 - \frac{1}{s})^t$$

By Summarizing the above four conditions, we can calculate the probability p' by the following equation:

$$p' = 1 - \left(\frac{t - (r' + 1)}{t}\right)\left(\frac{t - r' - 1}{t - 1}\right)\left(\sum_{i=0}^{r'} p(i)(1 - \frac{1}{s})^t\right)$$

Figure 7.3 shows the probability of connectivity with different configurations of number of key chains t and the key chain length s.

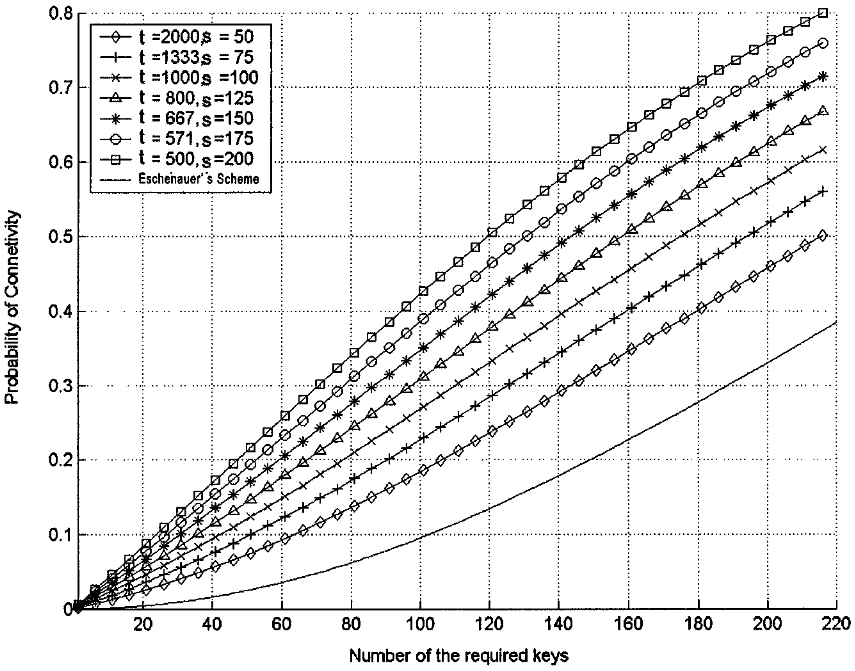


Figure 7.3. Comparison of different configured Two-Dimension Key Pool Selecting Schemes and Eschenauer's scheme (key pool size is 100,000)

As Figure 3 shows, under the same connectivity probability, the ARP scheme requires fewer keys than Eschenauer's scheme . In other words, the ARP scheme demands for less storage space than the Eschenauer's scheme does. Moreover, with different h and y value, the ARP scheme needs different storage space. This can be left as an option for deployment consideration.

4.2 Evaluation of UAO

In this section, evaluation of the probability of connectivity and the maximum supported network size are analyzed. The maximum supported network size stands for maximum sensor node capacity that can achieve mutual authentication under the same memory storage space attached in every sensor node. In addition, we also make a comparison with the random-pairwise scheme in terms of maximum supported network size and the probability of connectivity.

- **Probability of Connectivity:**

  In UAO scheme, the probability of any two nodes in the neighborhood sharing a common key can be evaluated by one minus the probability of that either nodes does not have any key derived from the other's unique one-way function. The probability for any node to get a key derived from a particular node's one-way function is $\frac{r}{n-1}$. Because each node gets r keys in the key ring, those keys are derived from other nodes in the network. The probability of any two nodes in the neighborhood sharing a common key will be

$$p' = 1 - (1 - \frac{r}{n-1})^2 \tag{7.4}$$

- **Maximum Supported Network Size:**

  By combining Equation 7.3 and 7.4, the following the equation can be derived.

$$\frac{d}{n'} = 1 - (1 - \frac{r}{n-1})^2$$

  Furthermore, by using Equation 7.2, the above equation becomes:

$$\frac{(n-1)(ln(n) - ln(-ln(P_c)))}{n * n'} = 1 - (1 - \frac{r}{n-1})^2$$

  The equation can be simplified to:

$$r^2 - 2(n-1)r + (n-1)^2 \frac{(n-1)(ln(n) - ln(-ln(P_c)))}{n * n'} = 0$$

By calculating the root of the above quadratic equation, we can get:

$$r = (n-1)(1 - \sqrt{1 - \frac{(n-1)(ln(n) - ln(-ln(P_c)))}{n * n'}}) \tag{7.5}$$

It can be more simplified as:

$$r = (n-1)(1 - \sqrt{1 - \frac{d}{n'}})$$

In comparison with the Random-Pairwise scheme , we assume the network size is n, expected degree of graph connectivity is d, the number of neighbor nodes is n', and the key ring size is r. According to the definition of pairwise scheme, there are only r nodes having common shared keys with each sensor node and it still has to achieve the expected degree in the neighborhood. Then we can find the following equation:

$$d = \frac{r * n'}{n} \Rightarrow r = \frac{d * n}{n'} \tag{7.6}$$

To analyze the relationship between memory space and network size, first we combine Equation 7.2 and Equation 7.6 to obtain the following equation:

$$r = \frac{(n-1)}{n'}(ln(n) - ln(-ln(P_c))) \tag{7.7}$$

According to the Equation 7.7, it is clear that the complexity of memory space requirement for the Random-Pairwise scheme is $O(nln(n))$. In addition, according to the Equation 7.5, it is found that the complexity of memory space requirement for the UAO scheme is $O(n\sqrt{ln(n)})$.

Figure 4 shows the comparisons of UAO scheme and Random-Pairwise keys distribution scheme in memory space requirement and the maximum supported network size. As Figure 4 shows, UAO scheme achieves better performance in maximizing network size under the same memory requirement. Therefore, with the same sensor node hardware equipment, UAO can adapt more sensor nodes in a network while providing better security than Random-Pairwise key distribution scheme.

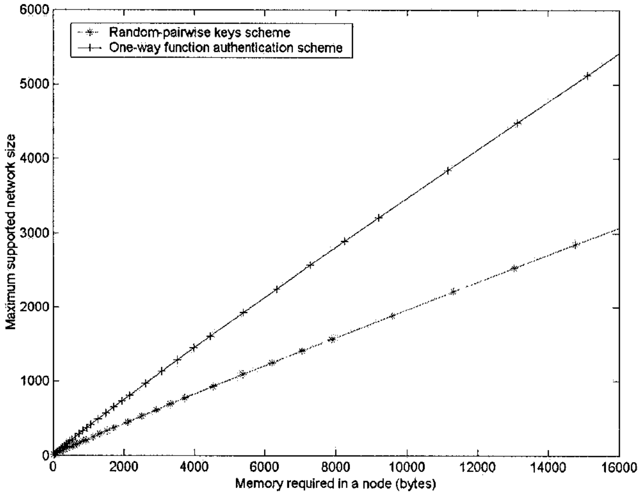


Figure 7.4. Comparison of Random-pairwise keys scheme and UAO scheme in memory requirement and maximum supported network size.

5. Conclusion

Key distribution is a critical and fundamental issue for the security service in wireless sensor networks. The pre-distributed and symmetric cryptography based key management system is well suitable for the resource limited sensor network. Two efficient schemes are proposed which are based on the Random Graph Theory to provide key distribution for the secure sensor network services.

Adaptive Random Pre-distributed scheme needs less memory space than existing schemes. ARP can be used in the WSN with a large number of nodes where each node contains limited storage space. On the other hand, Uniquely Assigned One-Way Hash Function scheme possesses the characteristics of mutual authentication . The tradeoff between these two schemes depends on security requirement, network size and available memory space. If mutual authentication of individuals is desirable, Uniquely Assigned One-Way Hash Function scheme should be used. Otherwise, the Adaptive Random Pre-distributed scheme should be used because it needs smaller storage space.

References

[Akyilidiz et al., 2002] F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. (2002). A Survey on Sensor Networks. In *IEEE Communications Magazine*, pages 102-114, August.

[Estrin et al., 1999] D. Estrin, R. Govindan, J. Heidemann and S. Kumar. (1999). Next Century Challenges: Scalable Coordination in Sensor Networks, In *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, August.

[Slijepcevic et al., 2002] S. Slijepcevic, M. Potkonjak, V. Tsiatsis, S. Zimbeck, and M. B. Srivastava. (2002). On communication security in wireless ad-hoc sensor network, *Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET-ICE'02)*, June.

[Tilak et al., 2002] S. Tilak, N. Abu-Ghazaleh, and W. Heinzelman. (2002). A taxonomy of wireless microsensor network models, *ACM Mobile Computing and Communications Review (MC2R 2002).*

[Hodjat et al., 2002] Hodjat and I. Verbauwhede. (2002). The energy cost of secrets in adhoc networks, *IEEE CAS Workshop on Wireless Communications and Networking*, September.

[He et al., 2003] T. He, J. A. Stankovic, C. Lu, and T. F. Abdelzaher. (2003). Speed: A stateless protocol for real-time communication in sensor networks, In *International Conference on Distributed Computing Systems (ICDCS 2003)*, May.

[Heinelman et al., 2000] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. (2000). Energy efficient communication protocols for wireless microsensor networks, *Proc. Hawaaian Int'l Conf. on Systems Science*, January.

[Heinzelman et al., 1999] W. Heinzelman, J. Kulik, and H. Balakrishnan. (1999). Adaptive protocols for information dissemination in wireless sensor networks, In *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, August.

[Intanagonwiwat et al., 2000] C. Intanagonwiwat, R. Govindan, and D. Estrin. (2000). Directed diffusion: A scalable and robust communication paradigm for sensor networks, In *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks (MobiCOM '00)*, August 2000.

[Zhou et al., 1999] L. Zhou and Z. J. Haas. (1999) Securing ad hoc networks, *IEEE Networks Magazine*, vol. 13, no. 6, pages 24-30, November.

[Kong et al., 2001] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang. (2001). Providing robust and ubiquitous security support for mobil ad-hoc network, *Network Protocols Ninth International Conference on ICNP 2001.*

[Luo et al., 2002] H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang. (2002). Self-securing ad hoc wireless networks, In *Proceedings of Seventh International Symposium on Computers and Communications (ISCC 2002)*, pages 567-574.

[Hubaux et al., 2001] J.-P. Hubaux, L.Buttyan, and S. Capkun. (2001). The quest for security in mobile ad hoc networks, In *Proceedings of the 2001 ACM International Symposium on Mobile Ad Hoc Networking and Computing*, October.

[Asokan et al., 2000] N. Asokan and P. Ginzborg. (2000). in ad hoc networks, *Computer Communications*, vol. 23, pages 1627-1637.

[Yi et al., 2003] S. Yi and R. Kravets. (2003). Moca: Mobile certificate authority for wireless ad hoc networks, *2nd Annual PKI Research Workshop Program (PKI03)*, April.

[Yi et al., 2002] Seung Yi and Robin Kravets. (2002) Key management for heterogeneous ad hoc wireless networks, *The 10th IEEE International Conference on Network Protocols (ICNP2002).*

[Basagni et al., 2001] S. Basagni, K. Herrin, E. Rosti, D. Bruschi, and E. Rosti. (2001). Secure pebblenets, In *Proceedings of the 2001 ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 156 - 163.

[Carman et al., 2000] D. W. Carman, P. S. Kruus, and B. J. Matt. (2000). Constraints and approaches for distributed sensor network security, *NAI Labs Technical Report #00- 010*, September.

[Carman et al., 2002] D. W. Carman, B. J. Matt, and G. H. Cirincione. (2002). Energy-efficient and low-latency key management for sensor networks, In Proceedings of 23rd Army Science Conference, December 2002.

[Perrig et al., 2001] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, Spins: Security protocols for sensor networks, In *Proceedings of the Seventh Annual International Conference on Mobile Computing and Networking*, July.

[Liu et al., 2003] D. Liu and P. Ning. (2003). Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks, *The 10th Annual Network and Distributed System Security Symposium*, February.

[Undercoffer et al., 2002] J. Undercoffer, S. Avancha, A. Joshi, and J. Pinkston. (2002). Security for sensor networks, *2002 CADIP Research Symposium.*

[Eschenauer et at., 2002] L. Eschenauer and V. D. Gligor. (2002). A key-management scheme for distributed sensor networks, In *Proceedings of* *the 9th ACM Conference on Computer and Communication Security*, pages 41-47, November.

[Chan et al., 2003] H. Chan, A. Perrig, and D. Song. (2003). Random key predistribution schemes for sensor networks, *IEEE Symposium on Security and Privacy*, May.

[Erdoos et al., 1960] P. Erdõs and A. Rẽnyi. (1960). On the evolution of random graphs, *Publ. Math. Inst. Hungat. Acad. Sci.*, vol. 5, pages 17-6.

[Spencer, 2000] J. Spencer. (2000). The Strange Logic of Random Graphs. *Springer-Verlag*.

III

INTRUSION DETECTION, DEFENSE, MEASUREMENT