Key Management Schemes in Wireless Sensor Networks

Soundarya.P and Varalakshmi .L.M
Department of electronics and communication engineering
Sri Manakula Vinayagar Engineering College, Pondicherry, India.

Abstract— Key establishment in sensor networks is a challenging problem because existing security schemes are unsuitable for use in resource constrained sensor nodes, and also because the nodes could be physically compromised by an adversary. In this paper two key establishment scheme are presented using the framework of pre-distributing a random set of keys to each node. One is the random-pairwise key scheme, which perfectly preserves the secrecy of the rest of the network when any node is captured, and also enables node-to-node authentication(EG SCHEME) and other is EG SCHEME with deployment knowledge) and going to compare the basic scheme(EG SCHEME) with the deployment model(EG SCHEME with deployment knowledge).

***Keywords*— sensor nodes, security, random key predistribution, deployment knowledge.**

I.INTRODUCTION

RECENT advances in electronic and computer technologies have paved the way for the proliferation of wireless sensor networks (WSN). Sensor networks usually consist of a large number of ultra-small autonomous devices. Each device, called a sensor node, is battery powered and equipped with integrated sensors, data processing, and short-range radio communication capabilities. In typical application scenarios, sensor nodes are spread randomly over the deployment region under scrutiny and collect sensor data. Sensor networks are being deployed for a wide variety of applications, including military sensing and tracking, environment monitoring, patient monitoring and tracking, smart environments, etc. When sensor networks are deployed in a hostile environment, security becomes extremely important as they are prone to different types of malicious attacks. For example, an adversary can easily listen to the traffic, impersonate one of the network nodes (in this paper, we use the terms sensors, sensor nodes, and nodes interchangeably), or intentionally provide misleading information to other nodes. To provide security, communication should be encrypted and authenticated. An open research problem is how to bootstrap secure communications among sensor nodes, i.e., how to set up secret keys among communicating nodes. This key agreement problem is a part of the key management problem, which has been widely studied in general network environments. There are three types of general key agreement schemes: the trusted-server scheme, the self-enforcing scheme, and the key predistribution scheme. The trusted-server scheme depends on a trusted server for key agreement between nodes, e.g., Kerberos [1]. This type of scheme is not suitable for sensor networks because there is usually no trusted infrastructure in sensor networks. The self-enforcing scheme depends on asymmetric cryptography, such as key agreement using public key certificates. However, limited computation and energy resources of sensor nodes often make it undesirable to use public key algorithms [2]. The third type of key agreement scheme is key predistribution, where key information is distributed among all sensor nodes prior to deployment. If we know which nodes are more likely to be in the same neighborhood before deployment, keys can be decided a priori. However, because of the randomness of deployment, it might be infeasible to learn the set of neighbors a priori. There exist a number of key predistribution schemes. A naive solution is to let all the nodes carry a master secret key. Any pair of nodes can use this global master secret key to achieve key agreement and obtain a new pairwise key. This scheme does not exhibit desirable network resilience: If one node is compromised, the security of the entire sensor network will be compromised. Some existing studies suggest storing the master key in tamper-resistant hardware to reduce the risk, but this increases the cost and energy consumption of each sensor. Furthermore, tamper resistant hardware might not always be safe [3]. Another key predistribution scheme is to let each sensor carry N -1 secret pairwise keys, each of which is known only to this sensor and one of the other N - 1 sensors (assuming N is the total number of sensors). The resilience of this scheme is perfect because compromising one node does not affect the security of communications among other nodes; however, this scheme is impractical for sensors with an extremely limited amount of memory because N could be large. Moreover, adding new nodes to a preexisting sensor network is difficult because the existing nodes do not have the new nodes' keys.

II.PROBLEM STATEMENT AND EVALUATION METRICS.

In this section, the topology and architecture of a typical sensor network is discussed, then the technical properties of typical sensor networks that makes the bootstrapping problem a challenge is also discussed . Finally, the goals and evaluation metrics for a successful sensor network security bootstrapping scheme is presented.

A. Sensor network architecture

The sensor nodes are usually scattered in the sensor field. Each of these scattered sensor nodes has the capabilities to collect data and route data back to the sink. Data are route back to the sink by a multihop infrastructure less architecture through the sink. The sink may communicate with the task manager node via internet or satellite. A typical sensor network has hundreds to several thousand sensor nodes. Each sensor node is typically low-cost, limited in computation and information storage capacity, highly power constrained, and communicates over a short range wireless network interface.

B. Sensor network limitations

The following characteristics of sensor networks complicate the design of secure protocols for sensor networks, and make the bootstrapping problem highly challenging.

1) Impracticality of public key cryptosystems: The limited computation and power resources of sensor nodes often makes it undesirable to use public-key algorithms, such as Diffie-Hellman key agreement [4] or RSA signatures [5]. Currently, a sensor node may require on the order of tens of seconds up to minutes to perform these operations . This exposes a vulnerability to denial of service (DoS) attacks.

2) Vulnerability of nodes to physical capture: Sensor nodes may be deployed in public or hostile locations (such as public buildings or forward battle areas) in many applications. Furthermore, the large number of nodes that are deployed implies that each sensor node must be low-cost, which makes it difficult for manufacturers to make them tamper-resistant. This exposes sensor nodes to physical attacks by an adversary. In the worst case, an adversary may be able to undetectably.

3) Limited resource: The amount of key-storage memory in a given node is highly constrained; it does not possess the resources to establish unique keys with every one of the other nodes in the network. Typical sensor network platforms have very low bandwidth.

c. The problem of bootstrapping security in sensor networks

Based on the limitations, a bootstrapping scheme for sensor networks needs to satisfy the following requirements: Deployed nodes must be able to establish secure node to node communication, the scheme should be functional without involving the base station as an arbiter or verifier. Additional legitimate nodes deployed at a later time can form secure connections with already-deployed nodes. This implies that bootstrapping information must always be present and cannot simply be erased after deployment to prevent compromise in the event of capture. Unauthorized nodes should not be able to establish communications with network nodes and thus gain entry into the network. The scheme must work without prior knowledge of which nodes will come into communication range of each other after deployment. The computational and storage requirement of the scheme must be low, and the scheme should be robust to DoS attacks from out-of-network sources.

D. Evaluation metrics

Sensor networks have many characteristics that make them more vulnerable to attack than conventional computing equipment. Simply assessing a scheme based on its ability to provide secrecy is insufficient. We present several criteria that represent desirable characteristics in a key-setup scheme for sensor networks.

1) Resilience against node capture: We assume the adversary can mount a physical attack on a sensor node after it is deployed and read secret information from its memory. Evaluate a scheme's resilience toward node capture by estimating the fraction of total network communications that are compromised by a capture of x nodes not including the communications in which the compromised nodes are directly involved.

2)Resistance against node replication: Whether the adversary can insert additional hostile nodes into the network after obtaining some secret information (e.g. through node capture or infiltration). This is a serious attack since the compromise of even a single node might allow an adversary to populate the network with clones of the captured node to such an extent that legitimate nodes could be outnumbered and the adversary can thus gain full control of the network.

3) Revocation: Revocation is that whether a detected misbehaving node can be dynamically removed from the system.

IV.ESCHENAUER GLIGOR (EG SCHEME)

Eschenauer and Gligor proposed a random key predistribution scheme: Before deployment, each sensor node receives a random subset of keys from a large key pool[6]. To agree on a key for communication, two nodes find one common key within their subsets and use this key as their shared secret key. There are three phases in EG SCHEME.

A. Key Pre-distribution phase

The key pre distribution phase consists of generation of a large pool of P keys (e.g., 2^{17} - 2^{20} keys) and of their key identifiers; random drawing of k keys out of P without replacement to establish the key ring of a sensor; loading of the key ring into the memory of each sensor; saving of the key identifiers of a key ring and associated sensor identifier on a trusted controller node; and for each node, loading the i-th controller node with the key shared with that node[7].

B. Shared-key discovery phase

The shared-key discovery phase takes place during DSN initialization in the operational environment where every

node discovers its neighbors in wireless communication range with which it shares keys. The simplest way for any two nodes to discover if they share a key is that each node broadcast, in clear text, the list of identifiers of the keys on their key ring. This approach does not give an adversary any attack opportunity that he does not already have. For ex- ample, if an adversary captures a node he can discover which key of that node is used for which link by decrypting communications; and if he does not capture a node, the adversary can mount a traffic analysis attack in the absence of key identifiers.

C.PATH KEY ESTABLISHMENT PHASE

The path-key establishment phase assigns a path-key to selected pairs of sensor nodes in wireless communication range that do not share a key but are connected by two or more links at the end of the shared-key discovery phase. Path keys need not be generated by sensor nodes. The design of the DSN ensures that, after the shared-key discovery phase is finished, a number of keys on a key ring are left unassigned to any link.

V. EG SCHEME WITH DEPLOYMENT KNOWLEDGE

Although the above proposed schemes provided viable solutions to the key predistribution problem, they have not exploited an important piece of information that might significantly improve their performance. This piece of information is node deployment knowledge, which, in practice, can be derived from the way that nodes are deployed. Let us look at a deployment method that uses an airplane to deploy sensor nodes. The sensors are first prearranged in a sequence of smaller groups. These groups are dropped out of the airplane sequentially as the plane flies forward. This is analogous to parachuting troops or dropping cargo in a sequence. The sensor groups that are dropped next to each other have a better chance of being close to each other on the ground. This spatial relation between sensors derived prior to deployment can be useful for key predistribution. The goal of this scheme is to show that knowledge regarding the actual non uniform sensor deployment can help to improve the performance of key predistribution .This deployment model is called as grid based deployment or group based model. Knowing which sensors are close to each other is important for key predistribution. In sensor networks, long distance peer-to-peer secure communication between sensor nodes is rare and unnecessary in many applications. The primary goal of secure communication in wireless sensor networks is to provide such communications among neighboring nodes. Therefore, the most important knowledge that can benefit a key-predistribution scheme is the knowledge about the nodes that are likely to be the neighbors of each sensor node. If we know perfectly the neighbors of each node in the network, key predistribution becomes trivial: For each node ni's, we just need to generate a pairwise key between ni's and each of its neighboring nodes, and save these keys in ni's memory. This guarantees that each node can establish a secure channel with each of its neighbors after deployment.

A. Modeling of the Deployment knowledge

We assume that sensor nodes are static once they are deployed. We define deployment point as the desired point where a sensor is to be deployed[8]. This is not likely the location where the sensor resides eventually. The sensor node can reside at points around this desired point according to a certain pdf. As an example, let us consider the case where sensors are deployed by being dropped from a helicopter. The deployment point is the location of the helicopter. We also define resident point for a sensor as the point where the sensor finally resides.

B. Group-Based Deployment model

In practice, it is quite common that nodes are deployed in groups, i.e., a group of sensors are deployed at a single deployment point, and the pdfs of the final resident points of all the sensors in each batch (or group) are the same. In this work, we assume such a group-based deployment and we model the deployment knowledge as follows (we call this model the group-based deployment model)

1) N sensor nodes to be deployed are divided into $t \times n$ equal size groups so that each group, $G_{i,j}$, for $i = 1; \ldots ; t$ and $j = 1; \ldots ; n$, is deployed from the deployment point with index (i,j). Let (x_i,y_j) represent the deployment point for group $G_{i;j}$.

2) The deployment points are arranged in a grid. Note that the scheme we develop for grid-based deployment can be easily extended to different deployment strategies. We choose this specific strategy because it is quite common in realistic scenarios

3) During deployment, the resident points of the node k in group $G_{i;j}$ follow the pdf $f(x,y|k \in G_{i,j})$. An example of the pdf is a two-dimensional Gaussian distribution.

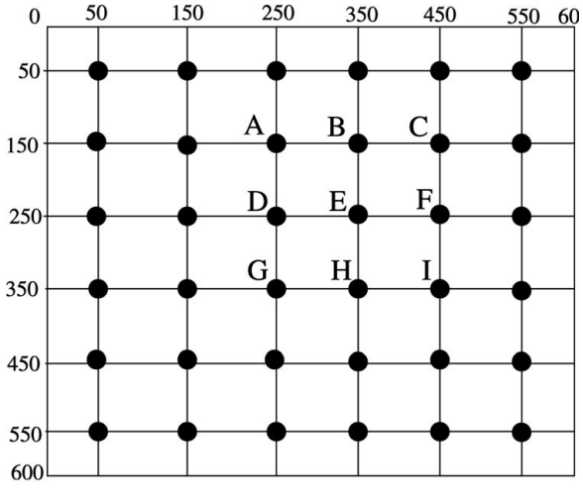

Fig 1 grid based deployment model

VI.SIMULATION RESULTS

A. *Connectivity analysis for EG Scheme*

The probability that two key rings share at least a key is 1 - Pr [two nodes do not share any key]. To compute the probability that two key rings do not share any key, each key of a key ring should be drawn out of a pool of P keys without replacement. Thus, the number of possible key rings is:

$$\frac{P!}{k!(P-K)!}$$

Select the first key ring. The total number of possible key rings that do not share a key with this key ring is the number of key-rings that can be drawn out of the remaining P – k unused key in the pool, namely:

$$\frac{(P-K)!}{k!(P-2k)!}$$

Therefore, the probability that no key is shared between the two rings is the ratio of the number of rings without a match by the total number of rings. Thus, the probability that there is at least a shared key between two key rings is:

$$\frac{k!(P-K)!(P-K)!}{P!k!(P-2k)!}$$

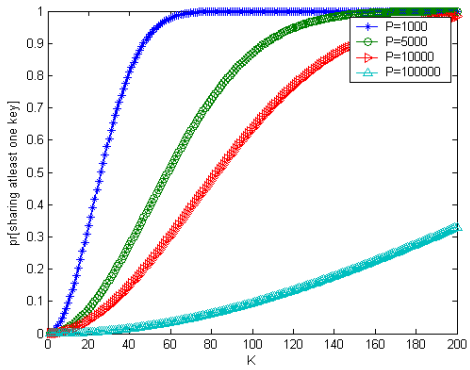

Fig 2 connectivity analysis of EG SCHEME

Figure 2 illustrates a plot of this function for various values of P. For example, one may see that for a pool size P = 10,000 keys, only 75 keys need to be distributed to any two nodes to have the probability p = 0.5 that they share a key in their key ring. If the pool is ten times larger, namely P = 100,000, the number of keys required is 250, which is only 3.3 times the number of keys distributed in the case P = 10,000. This provides intuition for the scalability of this approach. Of course, to determine the final the size of the key ring we need to provision for addition of new nodes, revocation, and re-keying. The scalability properties of the solution indicate that such provisioning will have minimal impact on the size of key rings.

B.COMPARISION WITH DEPLOYMENT KNOWLEDGE

From figure 3 it is understood that the memory usage is reduced in EG SCGEME WITH DEPLOYMENT KNOWLEDGE than the basic scheme.

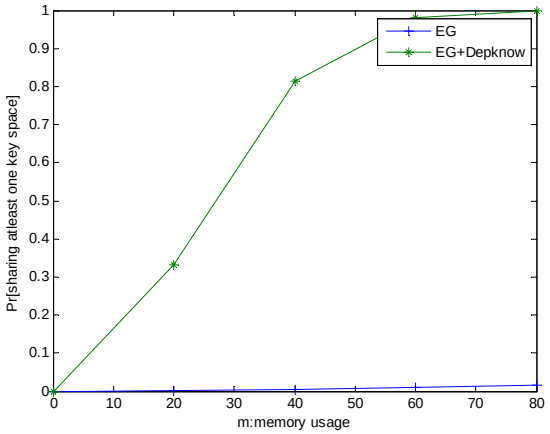

Fig 3connectivity analysis comparison

C.SECUIRTY ANALYSIS

Security analysis is the number of nodes need to be compromised to compromise the entire network. Security analysis of EG SCHEME is going to be compared with deployment model

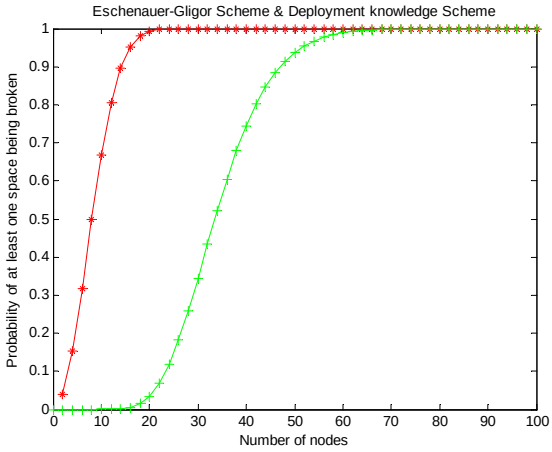

Fig 4 security analysis comparison

The figure 4 states that for EG scheme the attacker needs to compromise 20 nodes to achieve the probability of breaking atleast one key space and for deployment model the attacker needs to compromise more than 50 nodes to

compromise the entire network. Therefore EG SCHEME WITH DEPLOYMENT KNOWLEDGE is more advantageous than EG SCHEME both in security and connectivity.

VII.CONCLUSION

A new random key pre-distribution scheme for wireless sensor networks has been presented in this paper. This scheme has a number of appealing properties. First, this scheme is scalable and flexible. For a network that uses 64-bit secret keys, this scheme allows up to N = 264 sensor nodes. These nodes do not need to be deployed at the same time; they can be added later, and still be able to establish secret keys with existing nodes. Second, compared to existing key pre-distribution schemes, this scheme is substantially more resilient against node capture. The analysis and simulation results have shown, for example, that to compromise 10% of the secure links in the network secured using EG SCHEME WITH DEPLOYMENT KNOWLEDGE an adversary has to compromise 5 times as many nodes as he/she has to compromise in a network secured by Eschenauer- Gligor Scheme.

ACKNOWLEDGMENT

First and foremost, I would like to thank my guide M**rs. L.M.VARALAKSHMI** ***Assistant Professor, Department of Electronics and Communication Engineering***, for the valuable guidance and advice. She inspired me greatly to work in this publication. Her willingness to motivate me contributed tremendously to my work .I would like to take this opportunity to express our gratitude ***Mr.S.ARUNAGIRI Head of the Department, Electronics and Communication Engineering,*** for giving us suggestions then and there. He has always been a source of inspiration and encouragement towards this publication.

REFERENCES

[1] B.C. Neuman and T. Tso, "Kerberos: An Authentication Service for Compute Networks," IEEE Comm., vol. 32, no. 9, pp. 33-38, Sept. 1994.

[2] A. Perrig, R. Szewczyk, V. Wen, D. Cullar, and J.D. Tygar, "Spins: Security Protocols for Sensor Networks," Proc. Seventh Ann. ACM/ IEEE Int'l Conf. Mobile Computing and Networking (MobiCom), pp. 189-199, July 2001.

[3] R. Anderson and M. Kuhn. Tamper resistance - a cautionary note. In *Proceedings of the Second Usenix Workshop on Electronic Commerce*, pages 1–11, November 1996.

[4]W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Trans. Inform. Theory*, IT-22:644–654, November 1976.

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

[6] L. Eschenauer and V. D. Gligor. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM conference on Computer and communications security*, November 2002.

[7] H. Chan, A. Perrig, and D. Song, "Random Key Predistribution Schemes for Sensor Networks," IEEE Symp. Security and Privacy, pp. 197-213, 2003.

[8] W. Du, J. Deng, Y.S. Han, S. Chen, and P. Varshney, "A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge," Proc. IEEE INFOCOM '04, pp. 586- 597, 2004.