NEIGHBOUR DISCOVERY IN ASYNCHRONOUS WIRELESS SENSOR NETWORKS

Yegireddi Ramesh[1], **Krishnaveni Paidi** [2]

[1]*Assoc.professor, Dept of CSE, Aditya Institute of Technology And Management – Tekkali, Srikakulam, A.p.*
[2]*Final M Tech Student, Dept of CSE, Aditya Institute of Technology And Management,Tekkali, Srikakulam,A.p.*

Abstract

Many sensor networks (especially networks of mobile sensors or networks that are deployed to monitor crisis situations) are deployed in an arbitrary and unplanned fashion. Thus, any sensor in such a network can end up being adjacent to any other sensor. In the network. To secure the communications between every pair of adjacent sensors in such a network, each sensor x in the network needs to store n - 1 symmetric keys that sensor x shares with all the other sensors, where n is the number of sensors in the network. This storage requirement of the keying protocol is rather severe, especially when n is large and the available storage in each sensor is modest. Earlier efforts to redesign this keying protocol and reduce the number of keys to be stored in each sensor have produced protocols that are vulnerable to impersonation, eavesdropping, and collusion attacks. In this paper, We are enhancing the present secure keying protocol fully secure keying protocol where each sensor needs to store (n+1)/2 keys, which is much less than the n-1 keys that need to be stored in each sensor in the original keying protocol by sharing the other sensor storage virtually in the sensor region by identifying the idle of the each sensor by using shortest path techniques and enhances the secure communication store the keys.

Keywords: *sensor network; keying protocol; secure communication; optimal; uniform*

--***--

1. INTRODUCTION

Many wireless sensor networks are deployed in arbitrary and unplanned fashion. Examples of such networks are networks of mobile sensors [1] and networks that are deployed in a hurry to monitor evolving crisis situations [2] or continuously changing battle fields [3]. In any such network, any deployed sensor can end up being adjacent to any other deployed sensor. Thus, each pair of sensors, say sensors x and y, in the network need to share a symmetric key, denoted Kx;y,that can be used to secure the communication between sensors x and y if these two sensors happen to be deployed adjacent to one another. In particular, if sensors x and y become adjacent to one another, then these two sensors can use their shared symmetric key Kx;y to authenticate one another (i.e. defend against impersonation) and to encrypt and decrypt their exchanged data messages (i.e. defend against eavesdropping). such a network is required to store n - 1 symmetric keys, where n is the total number of sensors in the network and each stored key is shared between sensor x and a different sensor in the network. This requirement that each sensor in the network stores n-1 symmetric keys, where n is the number of sensors in the network, is rather severe especially when n is large and the available storage to store keys in every sensor is modest.

In the proposed system we are enhancing the protocol to share the memory virtually from the other sensor which sre in idle ,we are using the shortest path technique to find the nearest idle sensors around the communication path .this process automatically accepts the any number of keys in peak stage.

2 RELATED WORK

There are two main keying protocols that were proposed in the past to reduce the number of stored keys in each sensor in the network. We refer to these two protocols as the probabilistic keying protocol and the grid keying protocol.

In the probabilistic keying protocol [4], each sensor in the network stores a small number of keys that are selected at random from a large set of keys. When two adjacent sensors need to exchange data messages, the two sensors identify which keys they have in common then use a combination of their common keys as a symmetric key to encrypt and decrypt their exchanged data messages. Clearly, this protocol can probabilistically defend against eavesdropping Unfortunately, the probabilistic keying protocol suffers from the following problem[a]. In the grid keying protocol [5], [6], [7], and [8], each sensor is assigned an identifier which is the coordinates of a distinct node in a two-dimensional grid. Also each symmetric key is assigned an identifier which is the coordinates of a distinct node in two-dimensional grid. Then a sensor x stores a symmetric key K iff the identifiers of x and K satisfy certain given relation. When two adjacent sensors need to exchange data messages, the two sensors identify which keys they have in common then use a combination of their common keys as a symmetric key to encrypt and decrypt their exchanged data messages. The grid keying protocol has two advantages (over the probabilistic protocol)[a].

This situation raises the following important questions: Is it possible to design a keying protocol, where each sensor stores less than n-1 symmetric keys and yet the protocol is deterministically secure against impersonation, eavesdropping, and collusion?

In this paper, we worked on that In particular, we follow a new keying protocol where each sensor stores only (n+1)/2 symmetric keys, and yet the protocol is deterministically secure against impersonation, eavesdropping, and collusion. We also show that this new protocol is optimal by showing that each sensor, in any keying protocol that is deterministically secure against impersonation, eavesdropping, and collusion, needs to store at least (n / 1)/2 symmetric keys.

2.1 Sensor Networks and Adversaries:

We investigate a sensor network whose topology is not planned in advance, prior to the deployment of the network. Thus, when the network is deployed, any sensor can end up being adjacent to any other sensor in the network. In this network, when a sensor x is deployed, it first attempts to identify the identity of each sensor adjacent to x, then starts to exchange data with each of those adjacent sensors. Any sensor z in this network can be an "adversary", and can attempt to disrupt the communication between any two legitimate sensors, say sensors x and y, by launching the following two attacks[a].

2.2 The Keying Protocol:

Let n denote the number of sensors in our network. Without loss of generality, we assume that n is an odd positive integer. Each sensor in the network has a unique identifier in the range 0 …. n - 1. We use ix and iy to denote the identifiers of sensors x and y, respectively, in this network. Each two sensors, say sensors x and y, share a symmetric key denoted Kx;y or Ky;x. Only the two sensors x and y know their shared key Kx;y. And if sensors x and y ever become neighbors in the network, then they can use their shared symmetric key Kx;y to perform two functions[a].

2.3 A Data Exchange Protocol:

After two adjacent sensors x and y have authenticated one another using the mutual authentication protocol described in the previous section, sensors x and y can now start exchanging data messages according to the following data exchange protocol [a].

2.4 Optimality of the Protocol:

According to keying protocol, described above. each sensor in the network is required to store only (n+1)/2 keys. Thus, the total number of keys that need to be stored in the sensor network is n(n+1)/2. Despite the big saving in storage, that is achieved by previous keying protocol, one wonders ”Is there another keying protocol that requires the network to store much less than n(n + 1)/2 keys?” The following theorem indicates that the answer to this question is ”No.Theorem 6. Each keying protocol requires the sensor network to store at least n(n - 1)/2 keys [a], Theorem 7. Each uniform keying protocol requires each sensor in the network to store at least (n -1)/2 keys[a].

3. PROPOSED MODEL

Many wireless sensor networks are deployed in arbitrary and unplanned fashion In any such network, any deployed sensor can end up being adjacent to any other deployed sensor. Thus, each pair of sensors, say sensors x and y, in the network need to share a symmetric key, denoted Kx;y, that can be used to secure the communication between sensors x and y if these two sensors happen to be deployed adjacent to one another. In particular, if sensors x and y become adjacent to one another, then these two sensors can use their shared symmetric key Kx;y to authenticate one another (i.e. defend against impersonation) and to encrypt and decrypt their exchanged data messages. We keep mind the above technique we are enhancing the above technique.

In this paper , we followed the previous keying protocol and enhancing the keying protocols to share the memory virtually by implementing the shortest path technique identify the nearest sensor implement the technique on sensor whose need more memory to uninterrupted the communication between the source and the destination node while avoiding the adversaries.

3.1 Enhanced Keying Protocol Work Nature :

The enhanced keying protocol maintain in the sensor network each sensor inherits the property of finding the nearest idle sensor to share the memory virtually to To secure the communications between every pair of adjacent sensors in such a network. This proposed enhanced protocol increases the performance and secure from the adversaries.

4. CONCLUSIONS

Typically, each sensor in a sensor network with n sensors needs to store n - 1 shared symmetric keys to communicate securely with each other. Thus, the number of shared symmetric keys stored in the sensor network is n(n - 1). However, the optimal number of shared symmetric keys for secure communication, theoretically, is (n/2) = n(n - 1)/2. Although there have been many approaches that attempt to reduce the number of shared symmetric keys, they lead to a loss of security: they are all vulnerable to collusion. In this paper we enhanced the previous keying protocol for sensor networks, that needs to store only (n + 1)/2 shared symmetric keys to each sensor. The number of shared symmetric keys stored in a sensor network with n sensors is n(n + 1)/2, which is close to the optimal number of shared symmetric keys for any key distribution scheme that is not vulnerable to collusion bay virtually sharing the memory by implementing the shortest path technique for neighbor node which are in idle process. It may be noted that in addition to the low number of keys stored, and the ability to resist collusion between sensors. As our protocol has many desirable properties, such as efficiency, uniformity and security, we call this protocol the best keying protocol for sensor networks.

REFERENCES

[1]. The Best Keying Protocol for Sensor Networks, 978-1-4577-0351-5/11-IEEE

[2] A. Howard, M. J. Mataric, and G. S. Sukhatme, “Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem,” in Proceedings of the International Symposium on Distributed Autonomous Robotic Systems (DARS), 2002, pp. 299–308.

[3] S. Sana and M. Matsumoto, “Proceedings of a wireless sensor network protocol for disaster management,” in Information, Decision and Control (IDC), 2007, pp. 209 –213.

[4] S. Hynes and N. C. Rowe, “A multi-agent simulation for assessing massive sensor deployment,” Journal of Battlefield Technology, vol. 7, pp. 23–36, 2004.

[5] 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 (CCS), 2002, pp. 41–47.

[6] L. Gong and D. J. Wheeler, “A matrix key-distribution scheme,” Journal of Cryptology, vol. 2, pp. 51–59, January 1990.

[7] S. S. Kulkarni, M. G. Gouda, and A. Arora, “Secret instantiation in ad-hoc networks,” Special Issue of Elsevier Journal of Computer Communications on Dependable Wireless Sensor Networks, vol. 29, pp. 200–215, 2005.

[8] A. Aiyer, L. Alvisi, and M. Gouda, “Key grids: A protocol family for assigning symmetric keys,” in Proceedings of IEEE International Conference on Network Protocols (ICNP), 2006, pp. 178–186.

[9] E. S. Elmallah, M. G. Gouda, and S. S. Kulkarni, “Logarithmic keying,” ACM Transactions on Autonomic Systems, vol. 3, pp. 18:1–18:18, December 2008.

BIOGRAPHIES

I am Krishnaveni Paidi doing MTech in Aditya Institute of Technology And Management, Tekkali ,Srikakulam, A.P.,and intresting research areas are Datamining and network security.

Mr. YEGIREDDI RAMESH is MCA (Computer Applications) from Osmania University and M.Tech(CSE) from JNTUH Hyderabad, Andhra Pradesh, India. He is working as Associate professor in Computer Science & Engineering department in Aditya Institute of Technology and Management, Tekkali, Andhra Pradesh, India. He has 12 years of experience in teaching Computer Science and Engineering related subjects. He is a research scholar and his area of interest and research include Computer Networks, Wireless LANs & Ad-Hoc Networks and Cloud Computing. He has published several Research papers in national and international journals/conferences. He has guided more than 80 students of Bachelor degree, 25 Students of Master degree in Computer Science and Engineering in their major projects. He is a member of ISTE and CSI. He can be reached at **ramesh_yegireddi@yahoo.com**