International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Key Pre Distribution for Multicast Groups using Two Server Authentication in WSN

Maddineni Ramchandra[1], T. Madhavi Kumari[2]

[1]MTech, ECE Department, Jawaharlal Nehru Technological University, Hyderabad, India

[2]Associate Professor, ECE Department, Jawaharlal Nehru Technological University, Hyderabad, India

Abstract: ***Presented the sensitivity in the potential WSN purposes and on account of resource constraints, key management emerges as being a challenging difficulty for WSNs. One of the many concerns when designing a crucial management scheme would be the network scalability. Certainly, the process should support a large number of nodes permit a significant scale deployment in the network. In this paper, we propose a new scalable crucial management program for WSNs which gives beneficial secure connection coverage. For this purpose, we use the unital layout theory. We show which the basic mapping via unitals to help key pre- distribution we can achieve large network scalability. Even so, this trusting mapping does not guarantee an increased key discussing probability. Therefore, we suggest an increased unital-based crucial pre-distribution program providing large network scalability along with good crucial sharing possibility approximately reduce bounded. We prolong this recommended system by providing new several group crucial management (MGKM) program, named the actual master-key-encryption-based MGKM (MKE-MGKM) program, which can reduce the rekeying cost from taking care of multiple collection keys. The real key idea in the MKEMGKM should be to employ an asymmetric encryption program, called the actual master crucial encryption (MKE), to further improve the rekeying efficiency by relieving the rekeying cost. new several group crucial management (MGKM) program, named the actual master-key-encryption-based MGKM (MKE-MGKM) program, which can reduce the rekeying cost from taking care of multiple collection keys. The real key idea in the MKEMGKM should be to employ an asymmetric encryption program, called the actual master crucial encryption (MKE), to further improve the rekeying efficiency by relieving the rekeying cost. To boost the security we all propose a new symmetric remedy for two-server PAKE. In every existing two-server PAKE standards, two servers are offered random pass word shares pw1 along with pw2 subject to pw1 along with pw2. Within our protocol, you can expect one server S1 through an encryption in the password along with another server S2 through an encryption in the password in which pk1 along with pk2 include the encryption tips of S1 along with S2, respectively. Furthermore, two servers are offered random pass word shares b1 along with b2, where H is a hash purpose.***

Keywords: WSN, Security, Key Management, Scalability

1. Introduction

Wireless sensor networks (WSNs) are increasingly used in critical apps within a number of fields as well as military, healthcare and business sectors. Given this sensitivity of those applications, sophisticated safety services are required. Key management is really a corner stone for most security services for example confidentiality as well as authentication which are needed to secure marketing and sales communications in WSNs. The institution of safeguarded links among nodes is a challenging difficulty in WSNs. As a consequence of resource disadvantages, symmetric critical establishment is one of the most ideal paradigms with regard to securing swaps in WSNs. On the other hand, because of the possible lack of infrastructure throughout WSNs, we have usually no trusted third party which can certainly attribute pairwise key keys for you to neighbouring nodes, that's why most active solutions provide key pre-distribution.

Multicast is an efficient method for transmitting data coming from a single source to many destinations. Particularly, in wireless networks having a broadcast channel, an individual transmission may be received through all nodes just a transmission variety, which makes it simple to put into practice the multicast. As a result, the multicast throughout wireless networks is expected to pave the best way for effective group marketing and sales communications, by which many group-based apps, such seeing that charged video on desire or video conferencing, may be commercialized. This requirement is helped by activities being taken on by bigger standards committees regarding wireless networks, including IEEE 802.16 and 3GPP, which may have standardized this multicast assistance named multicast sent out services (MBS) or multimedia broadcast/ multicast assistance.

The broadcasting channel, however, makes this wireless network susceptible to various safety attacks since anyone can simply eavesdrop in messages transmitted in the air. To help implement this multicast, i.e., the shipping of data simply to the members of the group, throughout wireless networks, we must have an admittance control mechanism for the broadcasted communications, which assures confidentiality, defends digital articles, and helps accurate accounting. Therefore, it is one of the key requirements for profitable commercialization of those multicast providers in wireless networks. The usual way to offer an admittance control mechanism for the secure class communication is to employ the symmetric critical, known to be a group critical, shared solely by class members. Communications, encrypted by a member creating a group critical, can end up being decrypted through other class members obtaining the same class key, that may guarantee safeguarded group conversation. Although this kind of mechanism, while using the shared class key, is an efficient solution to guarantee safety, it leads to some complications in maintaining an efficient key managing system because the group key have to be updated according to membership changes including the user leaving behind or getting started with, which is called rekeying.

These days, passwords are commonly used through people within a log throughout process in which controls having access to protected computer systems, mobile phones, cable TELEVISION decoders, automated teller machines and the like. A laptop or computer user might have to have passwords for most purposes: logging into computer accounts, retrieving e-mail through servers, being able to view programs, sources, networks, internet websites, and possibly reading this morning classifieds online.

Earlier password-based authentication devices transmitted the cryptographic hash in the password spanning a public channel which makes the hash worth accessible a great attacker. When that is done, in fact it is very popular, the attacker can function offline, rapidly testing possible passwords contrary to the true password's hash worth. Studies have consistently shown a large tiny proportion of user-chosen security passwords are readily guessed routinely. For example, according for you to Bruce Schneier, examining data coming from a 2006 phishing attack, 55 percent of Bebo passwords can be crack equipped in 8 hours having a commercially readily available Password Retrieval Toolkit efficient at testing two hundred, 000 security passwords per minute in 2006. Recent study advances throughout password-based authentication have allowed complaintant and the server mutually for you to authenticate having a password as well as meanwhile to establish a cryptographic critical for safeguarded communications immediately after authentication. In general, current solutions for private data based authentication abide by two versions.

2. Previous Work

The significant advances of hardware manufacturing technology and efficient software algorithms make a network composed of numerous, small, low-cost sensors, using wireless communication a wireless sensor network (WSN) a promising network infrastructure for many applications such as environmental monitoring, medical care, and home appliance management. This is particularly true for battlefield surveillance and homeland security scenarios because WSNs are easy to deploy for those applications. However, in many hostile and tactical scenarios and important commercial applications, security mechanisms are required to protect WSNs from malicious attacks. Therefore, the security in WSNs becomes an important and a challenging design task.

A WSN is a large network of resource-constrained sensor nodes with multiple preset functions, such as sensing and processing, to fulfill different application objectives. Usually, sensor nodes are deployed in a designated area by an authority such as the government or a military unit and then, automatically form a network through wireless communications. Sensor nodes are static most of the time, whereas mobile nodes can be deployed according to application requirements. One or several base stations (BSs) are deployed together with the network. A BS can be either static or mobile. Sensor nodes keep monitoring the network area after being deployed. After an event of interest occurs, one of the surrounding sensor nodes can detect it, generate a report, and transmit the report to a BS through multihop wireless links. Collaboration can be carried out if multiple surrounding nodes detect the same event. In this case, one of them generates a final report after collaborating with the other nodes. The BS can process the report and then forward it through either high-quality wireless or wired links to the external world for further processing. The WSN authority can send commands or queries to a BS, which spreads those commands or queries into the network. Hence, a BS acts as a gateway between the WSN and the external world. An example is illustrated in Figure below.

Because a WSN consists of a large number of sensor nodes, usually, each sensor node is limited in its resources due to the cost consideration in manufacturing. For example, MICA2 MPR400CB, which is the most popular sensor node platform, has only 128 KB of program memory and an 8-bit ATmega128L CPU. Its data rate is 38.4 kbaud in 500 feet, and it is powered by only two AA batteries. The constrained resource cannot support complicated applications. On the other hand, usually, BSs are well designed and have more resources because they are directly attached to the external world.

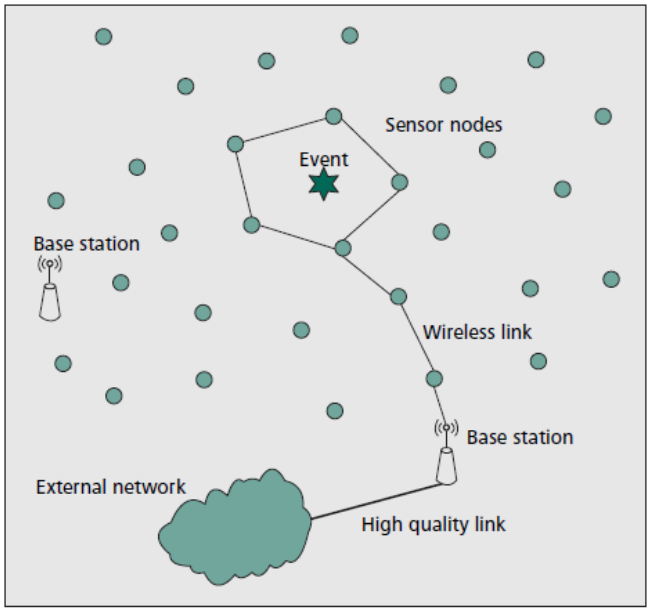

Figure 1. *A wireless sensor network.*

The harsh environments and the existence of threats demand more careful security considerations in the design of WSN protocols. Typically, one or more of the following security services should be provided:

- *Confidentiality* is a basic security service to maintain the secrecy of important data transmitted between sensor nodes. Usually, critical parts of a packet are encrypted before the packet is transmitted from the sending node and then, the parts are decrypted at the receiving node. Without the corresponding decryption keys, attackers are prevented from accessing the critical information. The kind of information that must be encrypted depends on the applications. In some cases, only the data part of a packet is encrypted, and in the other cases, the packet header also is encrypted to protect node identities.
- *Authenticity* is critical to provide the assurance of the identities of communicating nodes. Every node should

check whether a received message comes from a real sender. Without authentication, attackers easily can spoof node identities to spread false information into the WSN. Usually, an attached message authentication code (MAC) can be used to authenticate the origin of a message.

- *Integrity* should be provided to guarantee that the transmitted messages are not modified by attackers. Attackers can introduce interference to some bits of transmitted packets to change their polarities. A malicious routing node can also change important data in packets before forwarding them. Like a cyclic redundancy checksum (CRC) used to detect random errors during packet transmissions, a keyed checksum, such as a MAC, can protect packets against modification.
- *Availability* indicates another important capability of a WSN to provide services whenever they are required. However, attackers can launch attacks to degrade the network performance or even destroy the entire network. A denial of service (DoS) attack is the most detrimental threat to network availability; this occurs when attackers cause the network to lose the capability to provide services by sending radio interference, disrupting network protocols, or depleting the power of nodes through various tricky methods. [1]

There exist a number of key pre-distribution 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 pair wise 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. Another key pre-distribution scheme is to let each sensor carry N − 1 secret pair wise 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 pre-existing sensor network is difficult because the existing nodes do not have the new nodes' keys. [2]

Mitchel and Piper were the first to apply combinatorial designs in key distribution. C¸ amtepe and Yener applied combinatorial designs for key predistribution in WSN. A set system or design can be mapped to a key predistribution scheme in the following way. Let (X; A) be the set system with a set of elements (key identifiers) X and A set of subsets with elements from X. The subsets belonging to A are also called blocks of the design. Each sensor is associated with a block. Then the pool of key identifiers is the set X and the subsets of A are key chains of the sensor nodes. A special type of design is Balanced Incomplete Block Design, BIBD, with the parameters v, b, r, k and _, where v = jXj, b is the number of subsets (that is, the number of sensor nodes) of A or blocks, r is the number of blocks in which a particular element occurs, k is the size of each subset (the size of key chains), _ is the number of blocks (sensors) in which a given pair of elements occur. For a t-design, every t-subset of X occurs in _ blocks. t = 2 for BIBD. Combinatorial designs can be used to establish unique pairwise and triple keys respectively, between sensor nodes.

We consider here for the first time the scenario where three nodes want to communicate securely, and we call this concept the triple key distribution. It is a special type of group key distribution, where the group size is three. We describe few applications of this concept. A hierarchical network typically consists of a base station, cluster heads and small sensor nodes. A group of sensors nodes sense data and send to cluster head, which processes the data and sends to the base station. Sensor nodes share keys with the cluster head which helps them to securely send data to it. If cluster head needs to monitor the communication between two nodes in its cluster then these three nodes need to share a unique common key. [3]

The common approach is to assign each sensor node multiple keys, randomly drawn from a key-pool, to construct a key-chain to ensure that either two neighbouring nodes have a key in common in their key-chain, or there is a key-path. Thus the challenge is to decide on the key-chain size and key-pool size so that every pair of nodes can establish a session key directly or through a path. Keychain size is limited by the storage capacity of the sensor nodes. Moreover, very small key-pool increases the probability of key share between any pair of sensor nodes by decreasing the security in that, the number of the keys needed to be discovered by the adversary decreases. Similarly, very large key-pool decreases the probability of key share by increasing the security.

Eschenauer et al. propose a random key pre-distribution scheme where tens to hundreds of keys are uploaded to sensors before the deployment. In their solution, initially a large key pool of P and the key identities are generated. For each sensor, k keys are randomly drawn from the key-pool P without replacement. These k keys and their identities form a key-chain which is loaded in to the memory of the sensor node. Two neighbouring nodes compare list of identities of keys in their key-chain. Since only the identities are exchanged, this process can take place without any privacy mechanism. Eschenauer et al. also propose to employ Merkle Puzzle similar approach to secure key identities. After key identity exchange, common key(s) are used to secure the link in between two sensor nodes. It may be the case that some of the neighbouring nodes may not be able to find a key in common. These nodes may communicate securely through other nodes, through other secured links. Chan et al. propose a modification to the basic scheme of Eschenauer et al. They increase the amount of key overlap required for key-setup. That is, q common keys are needed instead of one to be able to increase the security of the communication between two neighbouring nodes. In common keys in the key-chains are used to establish multiple logical paths over which threshold key sharing scheme is used to agree on a new secret.

Random-pairwise key scheme is a modification of the pairwise key scheme. It is based on Erdos and Renyi's work; to achieve probability p of any two nodes are connected, in a network of n nodes, each node needs to store only a random

set of np pairwise keys instead of n − 1. Slijepcevic et al. propose that each sensor node shares a list of master keys, a random function and a seed. Every sensor uses shared random function and shared seed to select a network wise or group wise master key. In polynomial-based key predistribution protocol proposed for group key pre-distribution. In polynomial pool-based key pre-distribution is used for pairwise key establishment. For each sensor, random or a grid based pre-distribution scheme is used to select set of polynomials from a pool. [4]

3. Proposed System

3.1 Key Pre-distribution

Before the deployment step, we generate blocks of m order unital design, where each block corresponds to a key set. We pre-load then each node with t completely disjoint blocks where t is a protocol parameter that we will discuss later in this section. We demonstrate the condition of existence of such t completely disjoint blocks among the unital blocks. In the basic approach each node is pre-loaded with only one unital block and we proved that each two nodes share at most one key. Contrary to this, pre-loading each two nodes with t disjoint unital blocks means that each two nodes share between zero and t2 keys since each two unital blocks share at most one element.

After the deployment step, each two neighbours exchange the identifiers of their keys in order to determine the common keys. If two neighbouring nodes share one or more keys, we propose to compute the pairwise secret key as the hash of all their common keys concatenated to each other. The used hash function may be SHA-1 for instance. This approach enhances the network resiliency since the attacker has to compromise more overlap keys to break a secure link.

Otherwise, when neighbours do not share any key, they should find a secure path composed of successive secure links. The major advantage of this approach is the improvement of the key sharing probability. As we will prove in next subsection, this approach allows achieving high secure connectivity coverage since each node is pre-loaded with t disjoint blocks. Moreover, this approach gives good network resiliency through the composite pair wise secret keys which reinforce secure links. In addition, we show that our solution maintains high network scalability compared to existing solutions although it remains lower than that of the naive version.

We denote in what follows by t-UKP the unital-based key pre-distribution scheme of parameter t (t is the number of preloaded blocks at each node). When using the t-UKP scheme of order m, we pre-loaded each node with t(m+1) distinct keys. Indeed, from the construction, we can see that t blocks preloaded in a given node are completely disjoint. So, each two blocks within a key ring do not intersect at any key. So, the memory required to store keys is then equal to l×t×(m+1), where l is the key size.
Since each node is pre-loaded with t blocks from the m2 × (m2 −m+1) possible blocks of the unital design, it is obvious that the maximum number of key rings that we can reach is equal to n = m2 t (m2−m+1). This is the ideal case when all unital blocks are used. When using the random pre-distribution of unital blocks, we may generate a number of blocks slightly lower than this best value. We compute in what follows the minimum network size that can be supported by the random blocks distribution.

3.2 MKE Scheme

In the MGKM, a user can belong to one or more groups. Since the membership changes of a user having multiple group keys can affect all users of the corresponding groups, the effect of 1-affects-n may become much more severe. The reason why the existing schemes cannot efficiently alleviate the rekeying cost is that they use only symmetric key encryption for both the TEK and the KEK. In contrast, a new asymmetric key scheme, called the MKE scheme, has been introduced in KEKs to solve this problem. Since the TEK is used to encrypt/decrypt every packet, it should be based on symmetric key encryption that is much faster than asymmetric key encryption. However, since the KEK is used only for distributing the TEKs, it is reasonable that an asymmetric key is used as a KEK.

The MKE scheme is a RSA-based public-key cryptosystem proposed by Koyama where every user in the RSA system has a key pair that consists of a public key and a private key, each of which is used for encryption and decryption in an asymmetric pairwise manner. The concept of the MKE where the basic idea is to make a special key, called a master key, by using the Chinese remainder theorem (CRT). The master key can be used to encrypt messages, which can be decrypted by several different private keys or to decrypt messages encrypted with several different public keys. The most important feature of the MKE for MGKM is that one of the key pairs can be easily changed by modifying only the master key, without any changes to other users' key pairs. Through the asymmetry, the MKE scheme can alleviate the rekeying cost resulting from the symmetry of the TEK.

3.3 Two Server Password Registration

Prior to authentication, each client C is required to register both S1 and S2 through different secure channels. First of all, the client C generates decryption and encryption key pairs (xi, yi) where yi=g power xi for the server Si using the public parameters published by the two servers. Next, the client C chooses a password pwC and encrypts the password using the encryption key, according to ElGamal encryption. Then, the client C randomly chooses b1 from Zq. At last, the client C delivers the password authentication information Auth1to S1 through a secure channel, and the password authentication information Auth2 to S2 through another secure channel. After that, the client C remembers the password pwC only. The two secure channels are necessary for two server PAKE protocols, where a password is split into two parts, which are securely distributed to the two servers, respectively, during registration. Although we refer to the concept of public key cryptosystem, the encryption key of one server should be unknown to another server and the client needs to remember a password only after registration.

3.4 Key Exchange

In this module the two servers S1 and S2 have received the password authentication information of a client C during the registration, there are several steps for the two servers S1 and S2 to authenticate the client C and establish secret session keys with the client C in terms of parallel computation. The client C randomly chooses an integer r from Zq , computes R and then broadcasts a request message M to the two servers S1 and S2. On receiving M1, the server S1 randomly chooses an integer r1 from Zq and computes A2' and B2'. The server S2 randomly chooses an integer r2 from Zq and computes A1' and B1'. Then, S1 and S2 exchange M2 and M3. On receiving (A1', B1') the server S1 randomly chooses an integer r1' from Zq , computes R1, K1 and h1 and replies M4 to the client C. On receiving (A2', B2') the server S2 randomly chooses an integer r2' from Zq, computes R2, K2 and h2 and replies M5 to client C.

3.5 Authentication

In this module after receiving M4 and M5, the client C computes K'1 and K'2 and checks if PwC is same. If so, the two servers S1 and S2 are authentic. The client C computes h'1 and h'2 and broadcasts M6. At last, the client C sets the secret session keys with S1 and S2 respectively. On receiving M6, the server S1 checks if H() * b1 is same as h'1. If so, S1 concludes that the client C is authentic and sets the secret session key with the client C as SK1. The server S2 checks if H() * b2 is same as h'2. If so, S2 concludes that the client C is authentic and sets the secret session key with the client C as SK2

4. Results

The concept of this paper is implemented and different results are shown below, The proposed paper is implemented in NS 2.34 on fedora on a Pentium-IV PC with minimum 20 GB hard-disk and 1GB RAM. The propose paper's concepts shows efficient results and has been efficiently tested on different scenarios.

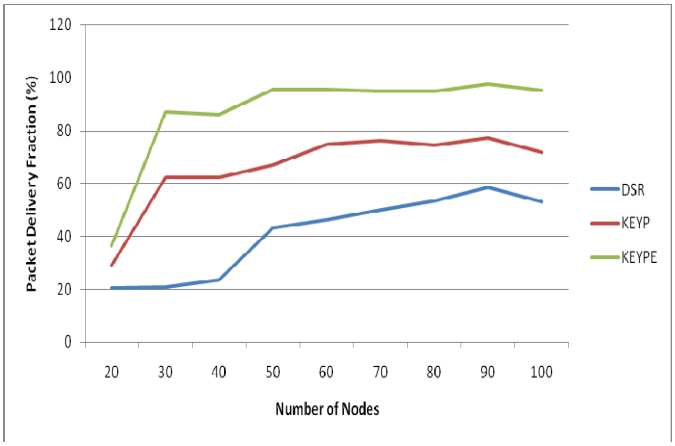

Figure 2: Number of Nodes Vs Packet Delivery Fraction

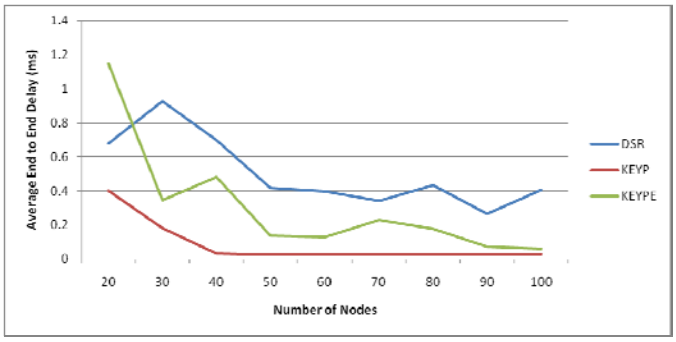

Figure 3: Number of Nodes Vs Average End to End Delay

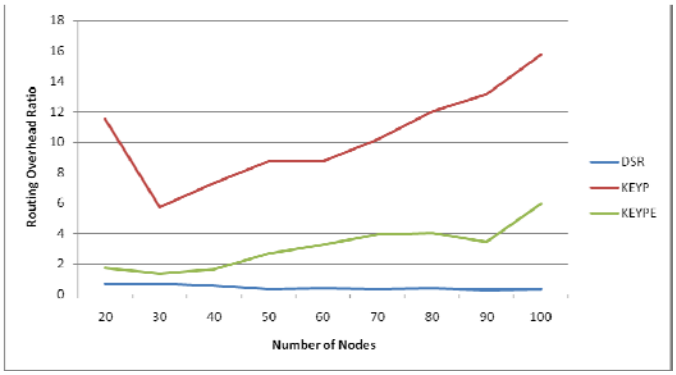

Figure 4: Number of Nodes Vs Routing Overhead

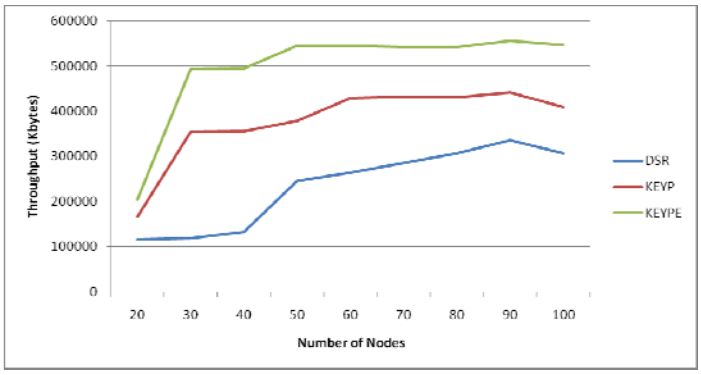

Figure 5: Number of Nodes Vs Throughput

5. Conclusions

We proposed, in this work, a scalable key management scheme which ensures a good secure coverage of large scale WSN with a low key storage overhead and a good network resiliency. We make use of the unital design theory. We showed that a basic mapping from unital to key pre-distribution allows achieving high network scalability while giving low direct secure connectivity coverage. We proposed then an efficient scalable unital-based key pre-distribution scheme providing high network scalability and good secure connectivity coverage. We discuss the solution parameter and we propose adequate values giving a very good trade-off between network scalability and secure connectivity. We conducted analytical analysis and simulations to compare our new solution to existing ones, the results showed that our approach ensures a high secure coverage of large scale networks while providing good overall performances.

References

[1] Y. Zhou, Y. Fang, and Y. Zhang, "Securing wireless sensor networks: a survey," *IEEE Commun. Surv. Tuts.*, vol. 10, no. 1–4, pp. 6–28, 2008.
[2] W. Du, J. Deng, Y. Han, S. Chen, and P. Varshney, "A key management scheme for wireless sensor networks using deployment knowledge," in *Proc. 2004 IEEE INFOCOM*, pp. 586–597.
[3] S. Ruj, A. Nayak, and I. Stojmenovic, "Fully secure pairwise and triple key distribution in wireless sensor networks using combinatorial designs," in *Proc. 2011 IEEE INFOCOM*, pp. 326–330.
[4] S. A. C¸ amtepe and B. Yener, "Combinatorial design of key distribution mechanisms for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 15, pp. 346–358, 2007.
[5] B. Maala, Y. Challal, and A. Bouabdallah, "Hero: hierarchcal key management protocol for heterogeneous WSN," in *Proc. 2008 IFIP WSAN*, pp. 125–136.
[6] W. Bechkit, Y. Challal, and A. Bouabdallah, "A new scalable key predistribution scheme for WSN," in *Proc. 2012 IEEE ICCCN*, pp. 1–7.
[7] S. A. C¸ amtepe and B. Yener, "Key distribution mechanisms for wireless sensor networks: a survey," Technical Report TR-05-07, Mar. 2005.
[8] S. Ruj and B. Roy, "Key predistribution using combinatorial designs for grid-group deployment scheme in wireless sensor networks," *ACM Trans. Sensor Netw.*, vol. 6, no. 4, pp. 1–4:28, Jan. 2010.
[9] M. Doddavenkatappa, M. C. Chan, and A. L. Ananda, "A dualradio framework for MAC protocol implementation in wireless sensor networks," in *Proc. 2011 IEEE ICC*, pp. 1–6.

Author Profile

Maddineni Ramchandra obtained his B.Tech degree from DRK college of Engineering and Technology, Bowrampet and Ranga Reddy District. Currently he is pursuing his M.Tech degree in Digital Systems and Computer Electronics. His research interests are in the area of ADHOC and Wireless Sensor Networks, Advanced Data Communications, VLSI Design, Microcontroller Design, Real Time Operating Systems and Digital System Design.

Mrs.Thoomati Madhavi Kumari obtained B.Tech. in ECE and M.Tech. in Computer Science from JNT University. Presently Mrs. Madhavi is pursuing Ph.D.in Data security for natural languages using FPGA. Mrs. Madhavi joined the faculty of ECE Department of JNU College of Engineering, Kukatpally, Hyderabad as Assistant Professor and served the department for 13 years. Later she was appointed as Assistant Director, UGC-ASC, JNT University, and Hyderabad. She was associated with the implementation of AICTE Sponsored project on computer networks.