LOKS: Low-Overhead Forward and Backward Key Secrecy in WSNs

Malvika Ashok
CSE Department
M. M. Engineering College
M. M. University
Mullana, Ambala, Haryana, India-133207

Rohit Vaid
CSE Department
M. M. Engineering College
M. M. University
Mullana, Ambala, Haryana, India-133207

ABSTRACT

Security plays an important role in designing a wireless sensor networks (WSNs). As the medium is wireless in nature which is more vulnerable to adversaries attacks in the network. Key management is used to achieve security in WSNs. If symmetric key is used in the network then one key is more enough to secure the network but the issue is that once the key is compromised, entire network gets compromised. On the other hand, if an individual key is provided to every sensor in the network then sensor to sensor communication is not possible in the network. Therefore sensors are grouped together to form a cluster. Each cluster is assigns a cluster key shared by every cluster member of the group. If this key is fixed then key compromising affects the security principle in the entire group. So to avoid this type of attack, key is updated after a fix interval of time. But if key updating is done by the base station, communication overheads in updating the keys are increased in the network. Therefore resource constrained wireless sensor networks; the concept of key generation is used instead of key distribution. In key generation process, the key is generated by applying a one way hash function on a given secret. But the problem is that if this secret is compromised, all the keys which are generated in past or to be generated in the future are immediately compromised. In this research paper, we present a LOKS: **L**ow-**O**verhead Forward and Backward **K**ey **S**ecrecy scheme to secure WSNs. This scheme updates the keys of each cluster in the network with resiliency to attack. Simulation results proves that presented scheme takes less number of communication overheads as compared to existing schemes given in literature to update the group keys for every groups after every round.

Keywords

Backward Secrecy, Cluster Head, Forward Secrecy, Key Generation, Key Secrecy, Sensor nodes, WSNs.

1. INTRODUCTION

Due to the advancements in the technology wireless networks are used in intense and difficult situations where humans can't reach easily. Due to its self-configured nature it is used in various applications. But sensor networks are resource constrained, so the main challenge in these networks is to provide secure communication in an efficient manner. There are many protocols that have been developed to utilize these kinds of networks in an efficient way so to increase the life time of the network. One of such technique is also presented in this paper to provide secure communication over the network with low communication overhead.

As the sensor network is more prone to inside and outside attacks because the medium is wireless. So Security is one of the big issues in Wireless sensor network. As the data is communicated through the wireless medium so the attacker or intruder can easily hack or tamper the data, make the nodes to be compromised, introduce malicious nodes in the network. All these activities make the network less secure and also decrease the lifetime of the network as they work on battery. So it becomes very important to make the network secure and generate the methods which should increase the network lifetime.

To provide security, it is very important that the data that is being communicated must be encrypted and verified. The data encryption means hiding the data or information into some code words so that no unauthorized person could harm the data. The same data when reached securely to the receiver side it is decrypted first i.e. changing the data into its original form. But the problem is how to secure the information between the sender and the receiver and how to keep the network away from the reach of the attackers or intruders.

All this process of encryption and decryption involves the keys. Keys are the secret number or code that is used to provide security to the information in the network.

1.1 Forward and Backward Key Secrecy

The process of key generation is shown in Fig 1. The key which is used in each round by different sensor node in a particular round is generated by applying a one way hash function on round key of previous round, i.e. i^{th} round key is used to generate the round key of $(i+1)^{th}$ round. Key generation is used in replace of key distribution to reduce complexity in the network.

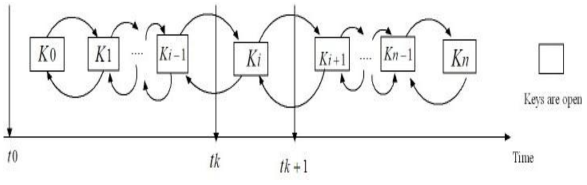

Fig 1: Key Generation Process

The drawback of key generation scheme is that once a key of some round is compromised, the key of any round is easily calculated by an attacker as the function used to generate a key is static in nature and is open to all sensors. The problem is shown in Fig 2 where the key of i^{th} round is compromised which results to disclose the round key of all the rounds.

Table 1. Key Updating [5]

Current Round	Seed value generated by sensor node S1	Seed value generated by sensor node S2	Sensor node S1 send the encrypted value to sensor node S2	Sensor node S2 send the encrypted value to sensor node S1	Generation of round Key
1	S_1^1	S_1^2	$S_1 \to S_2 : E(S_1^1, S_0^2)$	$S_2 \to S_1 : E(S_1^2, S_1^1)$	$Key_1 = \int(S_1^1, S_1^2)$
2	S_2^1	S_2^2	$S_1 \to S_2 : E(S_2^1, S_1^2)$	$S_2 \to S_1 : E(S_2^2, S_2^1)$	$Key_2 = \int(S_2^1, S_2^2)$
3	S_3^1	S_3^2	$S_1 \to S_2 : E(S_3^1, S_2^2)$	$S_2 \to S_1 : E(S_3^2, S_3^1)$	$Key_3 = \int(S_3^1, S_3^2)$
:	:	:	:	:	:
:	:	:	:	:	:
:	:	:	:	:	:
i_{th}	S_i^1	S_i^2	$S_1 \to S_2 : E(S_i^1, S_{i-1}^2)$	$S_2 \to S_1 : E(S_i^2, S_i^1)$	$Key_i = \int(S_i^1, S_i^2)$

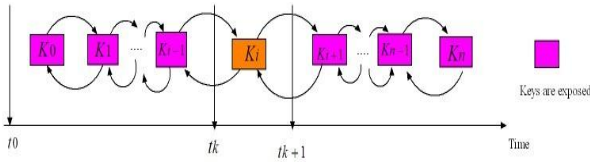

Fig 2: All key of the System are compromised due to one key compromising

So the process of key generation is implemented in such a way that the principle of 'Forward and Backward Secrecy' is maintained in the network. According to the principle of 'Forward Secrecy', the compromise of a key does not disclose the round key of any round that is used in future. Similarly the principle of 'Backward Secrecy' states that compromise of a key does not disclose the round key of any round that is used in past. The principle of 'Forward and Backward Secrecy' is shown in Fig 3. The system is highly secured by this technique.

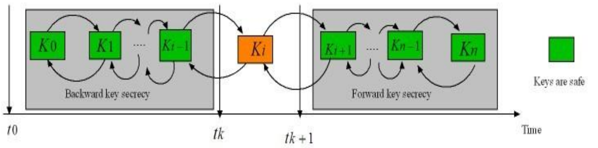

Fig 3: Forward and Backward Key Secrecy

2. EXISTING SCHEME TO ACHIEVE FORWARD AND BACKWARD KEY SECRECY IN WSNs [5]

Let there are two sensor nodes S1 and S2 deployed in wireless sensor networks. These two sensor nodes want to share a round key K_i in round 'i' with each other. Every sensor node generates a random seed value in each round. This seed value is used in generating the round key of current round in which it is generated. Let S_0^1 is the initial seed value generated by sensor node S1 before the starting of first round and similarly S_i^1 is the seed value generated by sensor node S1 in round 'i'. Let S_0^2 is the initial seed value generated by sensor node S2 before the starting of first round and similarly S_i^2 is the seed value generated by sensor node S2 in round 'i'. It is also assumed that both sensor nodes already share its own seed value with other sensor before the starting of first round. In first round sensor node S1 generates its random seed value S_1^1 and similarly sensor node S2 generates its own random seed value S_1^2. Now S1 encrypts S_1^1 with the help of a S_0^2 and send it to S2 as shown in Eq. 1.

$$S_1 \to S_2 : E(S_1^1, S_0^2) \qquad \ldots \qquad (1)$$

Similarly S2 encrypts S_1^2 with the help of S_1^1 and send it to S1 as shown in Eq. 2.

$$S_2 \to S_1 : E(S_1^2, S_1^1) \qquad \ldots \qquad (2)$$

Now both sensor nodes generates a round key used in first round by applying a one way hash function as shown in Eq. 3.

$$Key_1 = \int(S_1^1, S_1^2) \qquad \ldots \qquad (3)$$

So now in round 'i' sensor node S1 generates its random seed value S_i^1 and encrypt it with S_{i-1}^2 and send it to sensor node S2 as shown in Eq. 4.

$$S_1 \to S_2 : E(S_i^1, S_{i-1}^2) \qquad \ldots \qquad (4)$$

Similarly S2 encrypts S_i^2 with the help of S_i^1 and send it to S1 as shown in Eq. 5.

$$S_2 \to S_1 : E(S_i^2, S_i^1) \qquad \ldots \qquad (5)$$

Now both sensor nodes generates a round key used in round 'i' by applying a one way hash function as shown in Eq. 6.

$$Key_i = \int(S_i^1, S_i^2) \qquad \ldots \qquad (6)$$

Proceeding in this way the nodes generate new key in every round and use that key to communicate with each-others. The key updating method is shown in Table 1.

2.1 Issues in Existing Scheme

The existing system works well with two nodes only. As soon as the number of nodes in the network increases, the complexity of the system increases very high. The number of messages that communicates between all the nodes in the network to share and generates the key also increases. If the existing scheme to generate the keys is used then key generation in the system becomes the bottleneck of the network and becomes more difficult to exchange key generation parameters in the network. Each node generates its own random number and broadcast in the network. Fig 4 shows that the sensor node ‘A’ transmitting its random number Ra1 to each sensor in the network.

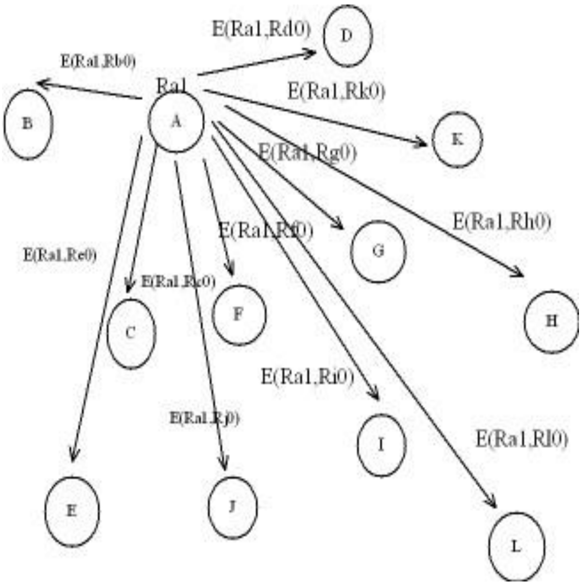

Fig 4: Sensor node ‘A’ broadcast its random number in the Network

This random number is encrypted with the random number of first sensor node with the help of a random number generated in previous round by first sensor node and transmitted to first sensor node then sensor node ‘A’ encrypts Ra1 with the help of a random number generated by second sensor node in previous round and transmit to second sensor node and so on. This process is achieved by all the sensors in the network. Each sensor generates its random number and sent it to each and every sensor one by one as shown in Fig 5.

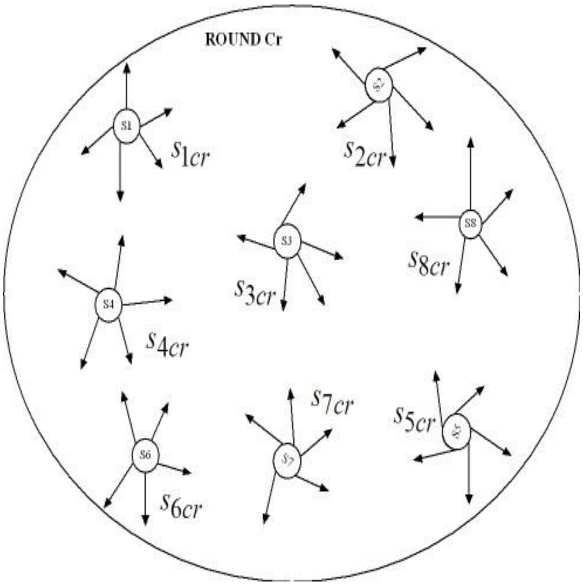

Fig 5: Each sensor nodes broadcasting its random number in the network

The drawback of this scheme is that broadcasting is not possible in this system, as the random numbers of each sensor in previous round are different. Only the round key of previous round is common between all the sensors. So the only method to broadcast the ransom number of current round is to encrypt it with the help of a round key used in previous round. But the drawback of this scheme is that once a key of any round is compromised, the principle of Forward and Backward Secrecy is not maintained in the network. So broadcasting is not used in this system. If each sensor broadcast its random number with the round key of previous round.

The second method to broadcast the random number is that each sensor node encrypts its random number with the help of its own random number that is used in previous round, as random number of all the sensor in previous round are known to each sensor in the network.

2.2 Communication Overheads in Existing Scheme

Let there are ‘n’ number of sensors in the network that wants and generate a key shared among all of them. To generate a key with the given scheme, each sensor node sent the random number by encrypting this random number with the previous round random number of individual sensor. This way each sensor transmits its random number to all the other sensors in the network one by one. So each sensor sent expectedly ‘n-1’ number of messages in the network, thus total ‘n*n-1’ messages communicated in the network as shown in Fig 6. As the number of sensors increases in the network, the communication overheads increase at very high speed. So this scheme is not efficient for more than two sensors.

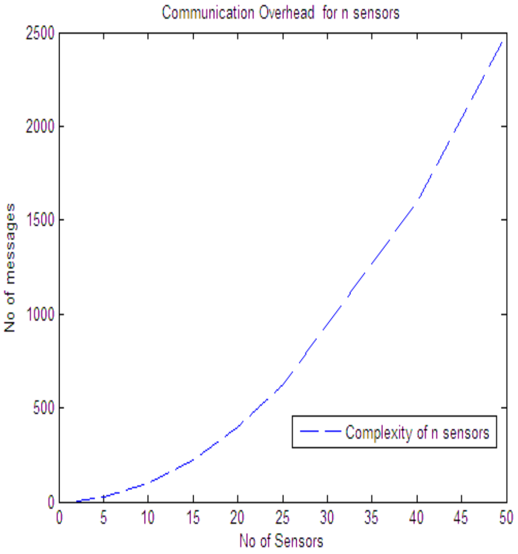

Fig 6: Communication Overheads in Existing Scheme

Table 2. List of abbreviations

S. No	Name	Description
1.	CH_{cr}	Cluster head for current round
2.	CH_{cr+1}	Cluster head in next round
3.	CH_{cr-1}	Cluster head in previous round
4.	R_{cr}	Random number generated by cluster head of current round
5.	R_{cr+1}	Random number generated by cluster head of next round
6.	R_{cr-1}	Random number generated by cluster head of previous round
7.	K_{cr}	Round key used in current round
8.	E_k	Encryption key

3. PRESENTED KEY GENERATION SCHEME WITH FORWARD AND BACKWARD KEY SECRECY

The problem in the existing system was the complexity. As the number of nodes in the system increases the system becomes more and more complex. The presented scheme for key generation is very efficient in terms of communication overheads by maintaining the principle of 'Forward and Backward Key Secrecy' in the network. Various notations used in understanding the presented scheme are given in Table 2.

In presented system, communication overheads are reduced by limit the number of nodes parcipating in key generation process to three only. In each round all the nodes in the group does not participate in generating the random number that is used in key generation process, only three nodes will generate the random numbers. These three nodes are cluster head nodes in three different consecutive rounds, i.e. cluster head of previous round (CH_{cr-1}), cluster head of current round (CH_{cr}) and cluster head in next round (CH_{cr+1}). CH_{cr} becomes CH_{cr-1} in round number (cr+1). So in every round a new cluster head is elected that is known as cluster head of next round (CH_{cr+1}). This cluster head becomes the current cluster head in round number (cr+1). But in first round only two cluster heads are choosen which participate in key generation process by generating their random numbers. From these two cluster heads, one cluster head plays the role of current cluster head and other play ther role of next round cluster head. In this round there is no previous round cluster head. In this round, CH_{cr+1} sends the generated random number (RCH_{cr+1}) to CH_{cr} as shown in Fig 7.

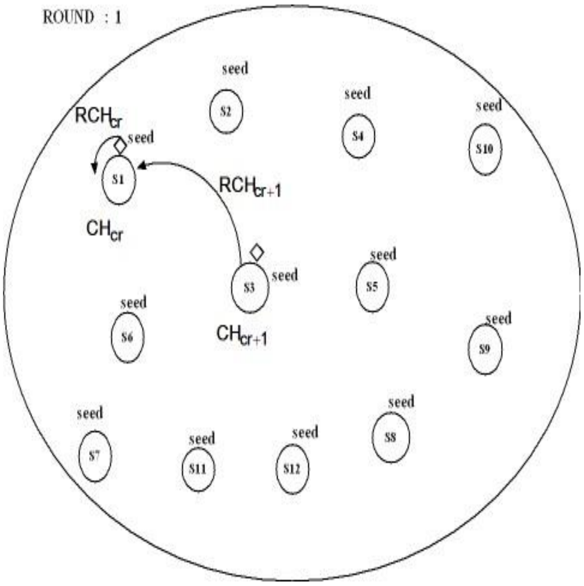

Fig 7: Next round CH sending its random number to current CH

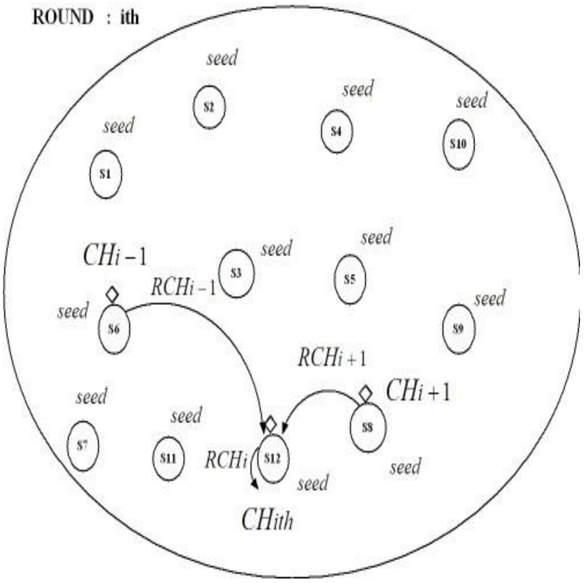

Fig 8: Previous and next round CH sending random numbers to i$_{th}$ round CH

Fig 8 shows the concept of three cluster heads. In this figure, cluster head CH_{cr-1} and CH_{cr+1} sent seed values to cluster head CH_{cr} . Now after receiving the random numbers from CH_{cr-1} and CH_{cr+1} , the cluster head of current round CH_{cr} generate the key K_{cr} used in current round by applying a one way hash function on the values RCH_{cr-1} , RCH_{cr} and RCH_{cr+1}. After generating the round key, CH_{cr} broadcast this round key (K_{cr}) in the network as shown in Fig 9.

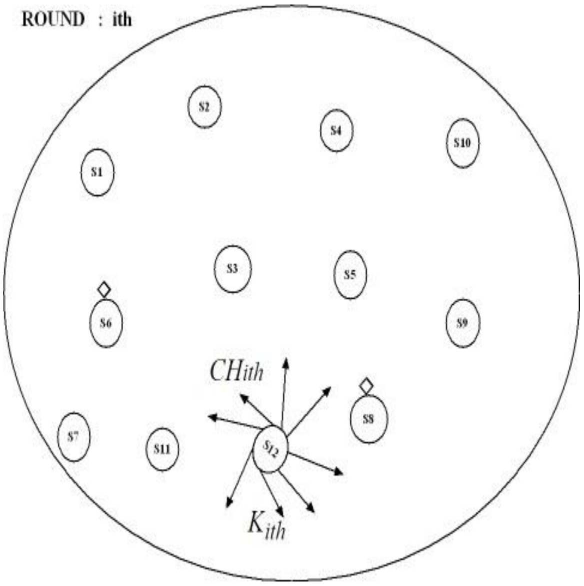

Fig 9: CH broadcasting key to all sensors in the network

This round key is broadcasted in the network in encrypted form. This key is encrypted with encryption key. This encryption key (KE_{cr}) is generated by applying a one way hash function on round key of previous round and seed value of current round as shown in Eq. 8. Where seed value in each round is generated by apply a one way hash function on seed value of previous round and round key of previous round as shown in Eq. 9.

$$K_{cr} = Hf1(RCH_{cr-1}, RCH_{cr}, RCH_{cr+1}) \quad \ldots \quad (7)$$

$$KE_{cr} = Hf2(K_{cr-1}, sd_{cr}) \quad \ldots \quad (8)$$

$$sd_{cr} = Hf3(K_{cr-1}, sd_{cr-1}) \quad \ldots \quad (9)$$

After receiving the encrypted key, individual sensor decrypt it with the decryption key to get the round key of current round. Decryption key is generated with same technique that is choosen by the current round cluster head as shown in Eq. 7. As encrypted key is broadcasted in the network so there is no chance that this key is hacked by the adversary because the encryption key is generated with the help of a seed value which is never communicated over the network. So the presented system provides a scalable forward and backward key secrecy in WSNs.

3.1 Key Generation Process

The following algorithm shows the steps involved in presented key generation scheme

A. $CH_{cr-1} \Rightarrow CH_{cr} : RCH_{cr-1}$

Cluster head in previous round sending its random number to current round cluster head.

B. $CH_{cr+1} \Rightarrow CH_{cr} : RCH_{cr+1}$

Cluster head for next round sending its random number to current round cluster head.

C. $CH_{cr} : RCH_{cr}$

Cluster head in current round generating its random number in current round.

D. $CH_{cr} : K_{cr} = Hf1(RCH_{cr-1}, RCH_{cr}, RCH_{cr+1})$

Cluster head in current round generating the current round key by applying one wat hash function on the random numbers of previous , current and next round cluster heads.

E. $CH_{cr} : KE_{cr} = Hf2(K_{cr-1}, sd_{cr})$

Cluster head in current round generating key Encryption key for current round by applying one way hash function on previous round key and the current round seed value.

F. $CH_{cr} \Leftrightarrow: KE_{cr}(K_{cr})$

Current round cluster head broadcasting current round key to all its cluster members by encrypting the key with the encryption key

G. $sd_{cr+1} = Hf3(K_{cr}, sd_{cr})$

Every sensor updates its seed value for next round by applying a one way hash function on current round key and the current round seed value.

In this way in each round new cluster head is selected and new key is generated for the new round. The seed value is never communicated on the network, it always remain hidden from the adversary. So if an adversary somehow hack the medium and gets the random number of any sensor node, it may never be able to generate the key. Due to the seed value that is involved in the encryption key the adversary may not be able to get the keys of any round nor past neither future keys. So the network is secure and the concept of forward and backward secrecy is maintained in the network.

4. COMMUNICATION OVERHEADS

Communication overheads in the proposed scheme are reduced due to number of sensors involved in key generation process is limited to three only. So in presented scheme, communication overheads are given below in Eq. 10.

CO=2+ (n-1) … (10)

Where CO is the communication overhead and 'n' is the total number of sensor nodes in the network.

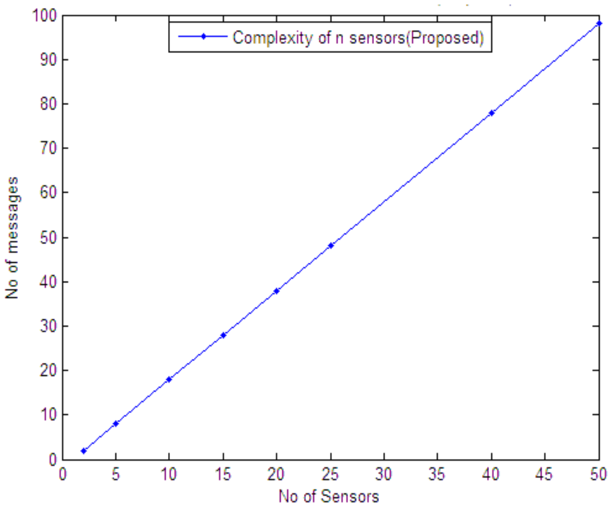

Fig 10: Communication overheads in presented system

Fig 10 shows the communication overheads in presented system with variable number of sensors in the network. Comparison between communication overheads in presented v/s existing scheme is shown in Fig 11.

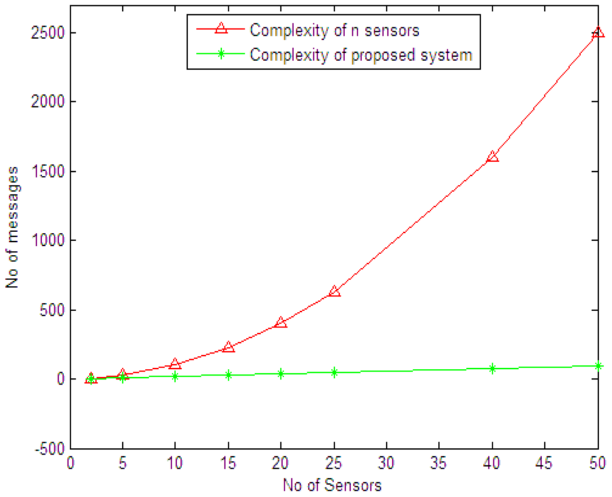

Fig 11: Communication overhead in presented v/s existing scheme

5. CONCLUSION

In this paper, we present a model to achieve the principle technique of forward and backward key secrecy in the WSNs. The presented model is scalable for large network with high number of sensor nodes. Presented scheme reduces the number of communication overheads as compared to existing scheme with reducing number of communication overheads. The scheme is very beneficial for cluster based approach where a group of nodes share a key with cluster head node. In this scheme, cluster heads are selected in each round where a current head of current round is responsible for generating and broadcasting a round key in entire cluster. In future, the scheme is applied for dynamic clustering where a node moves from one cluster to another cluster by crossing the cluster boundary.

6. REFERENCES

[1] Ali Bagherinia, Akbar Bemana, Sohrab Hojjatkhah and Ali Jouharpourr, "A Key Management Approach for Wireless Sensor Networks," IJITMC, pp. 1-9, 2014.

[2] Baojiang Cui, Ziyue Wang, Bing Zhao, Xiaobing Liang and Yuemin Ding, "Enhanced Key Management Protocols for Wireless Sensor Networks," pp. 1-10, September 2014

[3] Hani Alzaid, DongGook Park, Juan Gonzalez Nieto, Colin Boyd and Ernest Foo, "A Forward & Backward Secure Key Management in Wireless Sensor Networks for PCS/SCADA," Volume 24, , pp 66-82, 2010.

[4] Roberto Di Pietro , Luigi V. Mancini and Sushil Jajodia, "Providing secrecy in key management protocols for large wireless sensors networks," Elsevier, Vol. 1, pp. 455-468, November 2003.

[5] Vinayak Naik, Anish Arora, Sandeep Bapat and Mohamed Gouda, "Whisper: Local Secret Maintenance in Sensor Networks," pp. 1-16, 2003.

[6] Reza Azarderskhsh and Arash Reyhani-Masoleh, " Secure Clustering and Symmetric Key Establishment in HeterogeneousWireless Sensor Networks," Hindawi Publishing Corporation EURASIP Journal onWireless Communications and Networking, pp. 1-12, 2 October, 2010.

[7] Wensheng Zhang , Minh Tran, Sencun Zhu and Guohong Cao, "A Random Perturbation-Based Scheme for Pairwise Key Establishment in Sensor Networks," ACM 978-1-59593-684, pp. 1-10, Sept 2007.

[8] Eric Ke Wang, Lucas C.K HUI and S.M. Yiu, "A New Key Establishment Scheme for Wireless Sensor Networks," International Journal of Network Security and its Applications(IJNSA), Vol. 1, No. 2, pp. 17- 27, July 2009.

[9] Wenliang Du, Jing Deng, Yunghsiang S.Han, Pramod K. Varshney, Jonathan Katz and Aram Khalili, "A Pairwise Key Pre-Distribution Scheme for Wireless Sensor Networks," ACM Journal Name, Vol.4, pp. 1-29, September 9, 2010.

[10] Mu Kun and Li Li, " An Efficient Pairwise Key Predistribution Scheme for Wireless Sensor Networks," JOURNAL OF NETWORKS, Vol. 9, No. 2, pp. 1-6, February 2014.

[11] Donggang Liu, Peng Ning, "Establishing Pairwise Keys in Distributed Sensor Networks," ACM 158113738, 2003.

[12] Seyit Ahmet Camtepe, Bulent Yener, Moti Yung, "Expander Graph based Key Distribution Mechanisms in Wireless Sensor Networks," IEEE International Conference, pp. 1-6, 2006.

[13] Wenliang Du, Jing Deng, Pramod K. Varshney, et al., "A Pairwise Key Pre-Distribution Scheme for Wireless Sensor Networks," ACM Journal Name, Vol. V, pp. 228-258, 2 May 2005.

[14] Chi Yuan Chen and Han Chieh Chao, "A survey of key distribution in wireless sensor networks," Wiley Online Library, pp. 1-14, 2011.

[15] Rohit Vaid and Vijay Kumar, "Security issues and Remedies in Wireless Sensor Networks- A Survey," International Journal of Computer Applications, Vol. 79, No. 4, October 2013.

[16] Saurabh Singh, Dr. Harsh Kumar Verma, "Security for Wireless Sensor Network," International Journal for Computer Science and Engineering (IJCSE), Vol. 3, No. 6, pp. 1-7, 2011.