Hash Chain based Key Management for Mobile Heterogeneous Sensor Network

Vipin Kumar
M. Tech 2nd Year
CSE Department
NIT Kurukshetra

Priyanka Ahlawat
Asst. Professor
CSE Department
NIT Kurukshetra

ABSTRACT

Key management is an important issue for wireless sensor networks because sensors have limited resources as memory, processing speed and battery power. Key management is one of the basic building blocks of sensor network security. Many protocol presented before didn't take much attention on mobility of node. In this paper, we proposed Hass Chain based key management scheme for Heterogeneous networks where Base Station and cluster head are fixed and other node moving around these heads. Compare to other basic key management scheme, proposed scheme increase resilience against node capture as well as consider memory limitations.

General Terms

General terms which are used in this paper are Security, Hash function collision, Hash Chain, Wireless Sensor Network, Key Management, Cluster Head, Mobility et. al.

Keywords

Heterogeneous wireless sensor network, key management, random key predistribution, hash chain.

1. INTRODUCTION

A Wireless Sensor Network consisting of a large number of small sensors with limited resources as battery power, processing speed and storage. Sensors are implemented in the environment and data collected by each sensor is communicated through the network to a single processing center. Applications of sensor network are in many areas as medical, army and battle fields. If data is collected in hostile area there is much need that data must be secure. Confidentiality and authentication of data is must for secure data. To achieve authentication and confideciality cryptography system use symmetric or asymmetric keys. The self-enforcing scheme depends on asymmetric cryptography. Communication nodes have a pair of public and private key. To use public key cryptography scheme as Diffie-Hellman[1] key or RSA in sensor network in not feasible due to their limited processing power of sensors.

1.1 Contribution of work

1. Network Model used in this paper is given by the Sarmad U Khan, L Lavango and C Pastrone in [10].

2. we have studied the a new type of improvement for random key predistribution by using the hash key chain pool the in [12] and applied to key management scheme discussed in [10]

3 we have shown that improved key management scheme has better resilience than basic one [10].

2. RELATED WORK

In this section, we provide background knowledge on the basic random key predistribution scheme and other related work in heterogeneous network. P. Samundiswary, Padma priyadarshini, et al. [2] proved that the performance of the heterogeneous sensor network is better than the homogeneous sensor network. On the average, the energy consumed is 92.5% lesser than the homogeneous sensor network. The end-to-end delay reduces 62.5%. Eschenauer and Gilgor[3] proposed the basic key predistributed scheme. In this scheme every sensor node pick up a set of m key from a key pool S before deployment. This set of m keys is called the node's key ring. The number of keys in the key pool, is chosen such that two random subsets of size m in S will share at least one key with some probability p. After the sensor nodes are deployed, a key-setup phase is performed. The nodes first perform key-discovery to find out with which of their neighbors they share a key. Such key discovery can be performed by assigning a short identifier to each key prior to deployment, and having each node broadcast its set of identifiers. Nodes which discover that they contain a shared key in their key rings can then verify that their neighbor actually holds the key through a challenge response protocol. The shared key then becomes the key for that link. After key-setup is complete, a connected graph of secure links is formed. Nodes can then set up path keys with nodes in their vicinity that they did not happen to share keys with in their key rings. If the graph is connected, a path can be found from a source node to its neighbor.

The source node can then generate a path key and send it securely via the path to the target node. Chan further extended this idea and developed two schemes to improve the resilience to node capture by modifying the basic scheme [4]. The first one is called q-composite keys scheme. This scheme employs q common keys to set up the common key with a hash function rather than only one. The second scheme is called multi-path key reinforcement. This scheme establishes the link key through multiple paths to strengthen the security. In [5], Blundo et al. proposed to use bivariate polynomials to achieve key distribution for dynamic conferences. To establish a pair-wise key between two nodes, the key setup server randomly generates a t-degree bivariate polynomial over a finite field F_q; where q is a predetermined prime number that is large enough to accommodate a cryptographic key. By choosing appropriate coefficients $a_{ij} = a_{ji}$, we can have the desired symmetric property f(x; y) = f(y; x). Due to the symmetry of the bivariate polynomial, the secure pair-wise key between nodes ni and nj is established as $K_{ij} = f(n_i; n_j) = f(n_j ; n_i)$. To improve this, Liu and Ning [8] developed a general framework for establishing pair-wise keys between sensors and two pair-wise key pre-distribution schemes: a random subset assignment key pre-distribution scheme and a grid-based key predistribution scheme. Literatures [5-8] have studied pre-distribute key management for distributed homogeneous sensor networks. At the same time, several research papers preliminarily discuss key management issues for heterogeneous sensor networks as well. Kejie Lu, et al., [9] proposed a unified framework for distributed key management schemes in heterogeneous wireless sensor networks. In [10], Du and Xiao, et al., proposed a novel routing-driven key management scheme, which only established shared keys for neighbor sensors that might communicate with each other. They utilized Elliptic Curve Cryptography to design an efficient key management scheme for HSN. H-Sensors should store all public keys of L-Sensors, and L-Sensors have to save all public keys of H-Sensors. The storage space and energy consumption of each node is high. Kausar, et al., [11], proposed a key management scheme based on random key pre-distribution for heterogeneous sensor network. In the scheme, all the keys of the key pool are assigned to H-Sensors. While the network is a dense graph, the storage requirement of H-Sensor is extremely large.

3. FRAMEWORK

We use the network model given in [10] and describe in [11]. There are two types of node in the network mobile node MN and fixed node FN and a base station BS. As a heterogeneous network FN node has more capabilities as processing power, memory and transmission capability. FN has additional transmission capability as IEEE 802.11 and directly communicates to other FN and base station. Mobile node MN moves freely around the FN and can communicate to FN when they are in the range of FN. They can move from one FN to other FN. In this model we have a large key pool and the BS select a sub key pool of size P as an authentication key pool.

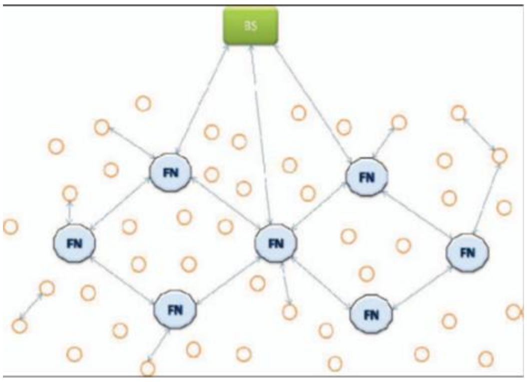

Fig 1: Network Model given in [10-11].

4. PROPOSED SCHEME

In the scheme discussed in [10] BS has a key pool of size P as an authentication key pool. BS assign random key ring of size S and assign to every FN and a key ring of size K assign to every MN. Where K<<S.

In our scheme, we have applied the improvement discussed in [12] to basic key management scheme in [10]. The BS has key pool consist of P non colliding hash chain of a length L and every value in the chain is considered as potentials key as given in [12].

$$\begin{tikzcd}
K_{11} \arrow[r, "H"] & K_{12} \arrow[r, "H"] & \cdots \arrow[r, "H"] & K_{1L} \\
K_{21} \arrow[r, "H"] & K_{22} \arrow[r, "H"] & \cdots \arrow[r, "H"] & K_{2L} \\
K_{31} \arrow[r, "H"] & K_{32} \arrow[r, "H"] & \cdots \arrow[r, "H"] & K_{3L} \\
\vdots & \vdots & & \vdots \\
K_{|P|1} \arrow[r, "H"] & K_{|P|2} \arrow[r, "H"] & \cdots \arrow[r, "H"] & K_{|P|L}
\end{tikzcd}$$

Fig 2: Key chain pool given in [12]

P randomly selects S chain and given a key ring of size s to FN as each key from a different chain. Same as P gives a key ring of size K to every MN that are randomly selected from K different chain. Two node shared a key if they have a the key from same key chain. A node with a value closer to the beginning of the key chain can traverse the chain downwards to find the shared key carried by the second node. Every FN can be reach to other FN and BS in a single hop. Each FN and BS has a pair of public and private keys. These keys are used for authentication and secrete communication to other FN and BS. The notations for the proposed scheme are

P	sub key pool of size |P|
S	keys assigned to FN
K	keys assigned to MN
Kplc	Network public key
Kprt	Network private key for FN
hn	Number of time key Hashed
Cid	Key Chain id
L	Length of key chain

Steps for the keys assignment are:

Initially we assume that base station BS and every fixed node FN has a pair of public/private keys for authentication and secrete communication. Every MN knows the public key Kplc of FN. Hash function is known all over the network.

1. BS generates a large key pool.

2. Base station BS selects a random sub key pool P of size |P| as a key pool for network.

3. BS generates a hash chain for every key in P of length L by applying hash function H. Every hash chain has a chain id Cid.

4. BS randomly selects K key chain and assign K keys(one key from each chain) to MN along with chain id and its hn.

5. FN key assignment

   i. BS randomly selects S key chain and assign S keys(one key from each chain) to FN along with chain id and its hn. S is much bigger than K (S>>K).

   ii. BS assign MN ids and associated key chain id along with hn information to all the FNs.

4.1 Authentication and Communication Key

Authentication between FN and BS is done by public private key pair and authentication of MN is done by FN very similar to [10]. When MN wants to communicate with FN these steps are fallow:

1. MN request to FN by sending its node id encrypted by Kplc of FN.

2. After receiving MN id, FN matches the key chain-ids of MN with its own key chain identifier list.

   i. if key chain match=0
   FN asks BS for communication
   key along with key chain-ids and
   hn used for generating
   communication key.

   ii. if key chain match=1
   communication key from the key
   chain that shared by FN and MN.

   iii. if key chain match =q
   communication key generated by
   concatenation of keys from shared
   q chain.

3. FN generates a authentication nonce, encrypt it with communication key and send to MN along with key chain id and hn. If FN key's hn is smaller than FN key's then FN hashed the key and send hn value 0.

4.2 Mobility of MN

When MN move from the range of one FN to other FN. MN send join request to FN with its id and previous FN id. If MN is not authenticating by above method New FN can authenticate MN by previous FN and take key chain ids from previous FN. New FN also authenticates MN by BS similar to basic key management scheme in [10].

5. PERFORMANCE

5.1 Connectivity

After key setup is complete a connected graph of secure link if formed. We want that every pair of node has shared key or for more secure at least q shared keys in q-composite scheme [4]. The probability that every pair of FN and MN shared exactly i keys is given by the

$$P_{\text{SharedExactly}}(i)=\frac{\binom{P}{i}\binom{P-i}{(S-i)+(K-i)}\binom{(S-i)+(K-i)}{(S-i)}}{\binom{P}{S}\binom{P}{K}}$$

If the key chains are non-colliding the probability is independent of their length L. The probability for link establishment that nodes share at least q keys is given by

$$P_{\text{LinkEstiblish}}=\sum_{i=q}^{K} PsharedExextly(i)$$

As compared to q-composite scheme probability of shared keys is same in the network but node can also communicate to each other if they have not shared key chain. But in this case traffic is more in the network and more bandwidth is consumed similar to as in [12].

5.2 Resiliency Against node capture

We use similar analysis given in [12] where if two node assign key from the given chain the probability that they establish ith shared key is (2i-1)/L^2. If key of a given key chain compromise in random capture node (k/p).(i/L). So according to [12] probability of chain compromise

$$P_{\text{chainComp}}=\sum_{i=1}^{L}\left(\frac{2i-1}{L^2}\right)\left(1-\left(1-\frac{K}{P}\frac{i}{L}\right)^x\right)$$

If a chain is compromised than the fraction of links that uses that chain is given by the ratio of number of links uses that chain to total link establish. So the probability of link compromised is given by the equation

$$P_{\text{LinkComp}}=\sum_{i=q}^{K}\left(P_{ChainCompr}\right)^i\frac{P_{SharedExactly(i)}}{P_{LinkEstabilsh}}$$

As compared to the basic q-composite scheme [3] the probability of link compromise is given by

$$P_{\text{LinkComp}}=\sum_{i=q}^{K}\left(1-\left(1-\frac{K}{P}\right)^x\right)^i\frac{P_{SharedExactly(i)}}{P_{LinkEstablish}}$$

there is much improvements is the resiliency. By this scheme we can achieve this level of resiliency by keeping the same level of connectivity. q-composite scheme efficient only when the number of node captured is low. But our scheme works for all the scenario as we only adjust the length of key chain to achieve high resiliency. Resilience is defined as probability to compromise a link using the keys of compromised nodes. Performance and compassion of q-composite scheme and hash chain shown in the fig-3. According to the graph node capture resilience after x randomly selected nodes have been captured, key ring size K = 41. Node captures resilience with Collision key improvement. We fixed P=800, S=100, q=3. Fig show the result for L=5 and L=10. Value of L will be discussed in next section. Basic q-composite scheme offers greater resilience against node capture when the number of nodes captured is small [4].

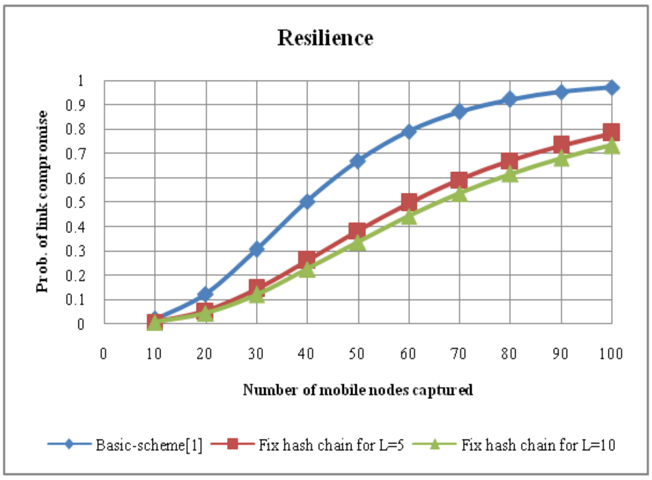

Fig 3: Node capture resiliency

5.3 Computation and Storage Overhead

Storage overhead only increase by hash function as hash function is also known to every FN and MN. The computational overhead also increase by calculating hash values of keys. But as compared to improvements in resilience it is not big overhead. Hashing has the property of one way evaluation so it is also used for authentication purpose. Some computation also changes on MN as hash function calculation can be come on either end so computation is distributed. But as compared MN, FN has more resources so computation must be done on fixed node very similar to [10].

5.4 Key Chain Length

The length of the key chain L is important security parameter of the key chain improvement. It is seen that large the value of L, the batter node capture resiliency. However as the length of the key chain increases the security gain obtain for a single unit increment decreases rapidly [12]. So key chain length L give better result for small values of L. It also affects the computational overhead. As the key chain length increases more computation is required but it give better resilience. So there is trade-off between resiliency and computation similar to [12].

6. CONCLUSSION

Key management is the most critical issue in the security of wireless sensor network. The basic scheme of random key predistribution is the hot topic of key management. There are many improvements in this scheme. The use the key chain pool instead of key pool, gives batter result on node capture resiliency by maintaining the length of key chain L. we have used the improvement of [12] in [10] .The result shows that larger the value of L, better the result but as the key chain length increases more computation required and less improvement for large L. The important part of key chain improvements is to find the hash chain that gives batter result. Also the value of q is important factor as the probability of key shirring decreases as q increases and large key pool required for batter overlapping of keys for FN and MN.

7. REFERENCESW.

[1] Diffie and M. E. Hellman 1976. New direction in cryptography. IEEE Trans. Inform. Theory, IT-22:644-654.

[2] P. Samundiswary, Padma priyadarshini, P. Dananjayan, Performance evaluation of heterogeneous sensor networks, Proc. 2009 Int. Conf. on Future Computer and Communication (ICFCC 2009), Kuala Lumpar, Malaysia, April, 2009, 264-267

[3] L. Eschenauer, V. D. Gligor, A key-management scheme for distributed sensor networks, Proc. Of the 9th ACM Conf. on Computer and Communications Security, Washington, DC, United States, Nov. 2002, 41-47

[4] H. Chan, A. Perrig, D. Song, Random key predistribution schemes for sensor networks, Proc. IEEE Symposium on Research in Security and Privacy, Berkeley, CA, United States, May 2003, 197-213

[5] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, M. Yung, Perfectly-secure key distribution for dynamic conferences, Proc. of the 12th Annual Int. Cryptology Conf. on Advances in Cryptology, California, USA, 1992, 471-486

[6] D. Liu, P. Ning, Establishing pairwise keys in distributed sensor networks, Proc. of the 10th ACM Conf. on Computer and Communications Security, CCS 2003, Washington, DC, United States, Oct. 2003, 52-61

[7] Kejie Lu, Yi Qian, Mohsen Guizani, Hsiao-Hwa Chen, A framework for a distributed key management schemes in heterogeneous wireless sensor networks, IEEE Transactions on Wireless Communications, 2(7), 2008, 639-647

[8] Xiaojiang Du, Yang Xiao, Song Ci, Mohsen Guizani, Hsiao-Hwa Chen, A routing-driven key management scheme for heterogeneous sensor networks, Proc. 2007 IEEE Int. Conf. on Communications, ICC'07, Glasgow, Scotland, United kingdom, June 2007, 3407-3412

[9] F. Kausar, M. Q. Saeed, A. Masood, Key management and secure routing in heterogeneous sensor networks, Proc. 4th IEEE Int. Conf. on Wireless and Mobile Computing, Networking and Communication, WiMob 2008, Avignon, France, Oct. 2008, 549-554

[10] Sarmad Ullah khan,Luciano L 2010. A key Management scheme Supporting Node Mobility in Hetrogeneous Sensor Network, 6th International Conference on Emerging Technologies (ICET)

[11] P. Traynor, R. Kumar, H. BinSaad ,G. Cao, T, La Porta 2007. "Efficient hybrid security machanisms for heterogeneous sensor network", IEEE Trans. On mobile Computing 6(6): 633-677

[12] Jiri Kur, Vashek Matyas, Petr Svenda "Two Improvement of Random Key Predistribution for Wireless Sensor Networks-Revised Version" Journal of Latex Class Files. Vol 6 no 1, January 2007