An Efficient Key Management Scheme with Key Agreement to Mitigate Malicious Attacks for Wireless Mesh Network

Vijay H. Kalmani
Principal
B.T.Patil & Sons Polytechnic

Sojwal S. Kulkarni *alias* R.M.Jogadand
Assoc. Prof, CS&E
GIT, Belgaum

H.M.Rai, PhD.
Ex-Prof., NIT, Kurukshetra

ABSTRACT

In the recent times the IT industry has been witnessing rapid development in wireless communication technologies such as 3 G, 4 G, etc. Wireless Mesh Network is one such promising wireless communication technology which provides high bandwidth internet connectivity in a sizable geographic area at a much lower cost than with classic WiFi networks. WMN consists of wireless entities such as Mesh Routers (MRs), Mesh Clients (MCs), Internet Gateways (IGs) are organized in an arbitrary mesh topology and form the wireless mesh back bone. Due to the mobility of the Mesh Clients, great challenges arise in securing the WMNs from various kinds of attacks. In this work a security mechanism is presented to ensure that induced network is connected and well protected from potential eaves dropping attacks, and this is accomplished by introducing an efficient key management scheme with key agreement. The simulation results show that the scheme implemented in this work, out performs previous schemes thus providing a network that is resistant against malicious eavesdropping attack.

Keywords

Wireless Mesh Network, Efficient Key management Scheme, malicious eaves dropping

1. INTRODUCTION

Wireless Mesh Networks (WMN) is a promising wireless technology that complements high bandwidth communication connectivity of a wired infrastructure with wireless backbone for to the mobile nodes (MNs). Different from flat network architectures such as an ad-hoc network, WMN is primarily hierarchal network architecture. The network essentially comprises of mobile nodes or clients in the lowest level, wireless mesh routers in the intermediate level and gateways connected to internet servers at the highest level. In an attempt to standardize the WMN technology, the IEEE 802.11s has been consistently working on the mesh network ingredients till date. The mesh network depends on the multi hop communication technique for the traffic to and from underlying devices.

The architecture of WMN is shown in fig. 1. The following security challenges aspects of the WMN can are highlighted due to the distrusted architecture of WMN [5]. Firstly wireless links in WMN makes it prone to active attacks, passive attacks and message distortion. Moreover passive attacks would compromise confidentiality and active attacks would result in violating availability, integrity, authentication, and non-repudiation. Secondly, due to the lack of physical protection we have the probability of node there is possibility being compromised. This makes the network unprotected from malicious attacks from outside network and also from the attacks launched from within the network. Thirdly, a due to dynamic topology and its membership can cause the trust relationship among nodes to change also. Finally, the traditional schemes for achieving security are not applicable due to and computational constraints.

Figure 1. WMN Architecture

2. RELATED WORK

Farah Kandah, Weiyi Zhang, Xiaojiang Du, , Yashaswi Singh proposed a key management scheme for wireless mesh networks. The work provided a key assignment scheme between the nodes by assigning *K* available encryption keys among all nodes in a common neighborhood. *Drawback:* The keys were getting distributed among the nodes themselves, without a mechanism to keep track of the key allotted. This leads to repetition of keys in the network, thus increasing the malicious eavesdropping

Du *et al.* in [6] proposed a key management scheme for heterogeneous sensor networks. Three phases have beed defined in the work *pre-distribution phase*, *discovery phase*, *key setup phase*. In the *pre-distribution phase* each high-end sensor is preloaded with *M* keys, and each low-end sensor is preloaded with *L* keys ($M >> L$), where the keys are randomly picked from a pool of keys P without replacement. The *discovery phase*, is used to check if neighboring sensors have a shared key, and the *key setup phase* is used to find a shared key between any two neighboring sensors when the discovery phase returns that there is no common key between them. *The pool size can affect this proposed scheme*, where with a large pool size and a small *K* keys randomly selected from P to be

stored in each node, a better security can be provided [6]. On the other hand with small pool size, there will be a chance of having more nodes shared common keys in the neighborhood which might harm the network due to various adversary attacks. Moreover, not all the generated keys in the pool are being used nor the keys in high-end or low-end sensors. Our proposed scheme use the least used of keys (*K*) from Key Distribution Centre (KDC) to be assigned *among* all the nodes, without generating too many unnecessary keys, and keep the network as secure as possible.

Zhao *et al.* in [14] propose an elliptic curve cryptosystem (ECC)-based self-certified public key cryptosystem for constructing WMNs security infrastructure; and its related security schemes, with a few modifications, are used for designing the Authentication and Key Agreement (AKA) protocol. Both authentication and key agreement between a Mesh Router (MR) and a Mesh Client (MC) can be simultaneously finished during one interaction using the AKA protocol; furthermore, a registered MC can access any WMN domain through any connectable MR independently, all of which improve the efficiency, convenience and fault tolerance of the system.

Eschenauer and Gligor [7] propose a key management scheme based on probabilistic key sharing among the nodes of random graph. Key distribution consists of three phases: key pre-distribution, shared-key discovery, and path key establishment. In key pre-distribution phase, each node randomly selects *k* keys from a key pool. In the shared key discovery phase, each sensor node discovers the shared keys with its neighbors within its transmission range. If two nodes have no shared keys, they will establish a shared key via two or more links in path key establishment phase. In this scheme, once a node is compromised, the key ring will be revoked. What is more, these keys should also be removed from other node key rings. This will decrease the link connectivity of the network, and the affected nodes need to reconfigure their links.

Another study in [5] considers the problem of designing a key management scheme in a clustered distributed sensor network, where the probability of node compromise in different deployment regions is known in advance. Different probability of node compromise values are assigned to different subgroup. Our proposed scheme differs in that, the repetition factor of the keys is least *k.*.

3. MODEL AND SYSTEM DESIGN

3.1 Network Model

In the model [1] the Mesh routers (MR)s in the WMN are stationary and without energy constraints. All MRs use the same fixed transmission power ($R > 0$). To model the WMN use an undirected bi-connected graph G (V; E) V is the set of n nodes and E is the set of m links in the network. There exist a undirected edge $e \in E$ if and only if $d(u; v) \leq R_u$, where $d(u; v)$ is the Euclidean distance between u and v, and R_u is the transmission range of node u, for each pair of nodes (u; v). The wireless link between nodes u and v in the network corresponds to each edge between any pair of nodes (u; v) in G. In this work we assume *nodes in* the transmission range of each other do not communicate between any two neighboring nodes unless they shared a common encryption key.

3.2 System Design

In this work it is intended to provide a key assignment mechanism between the Mesh Routers by assigning K available encryption keys among all nodes in a common neighborhood to be as different as possible such that the malicious eavesdropping attack can be reduced. In this work an algorithm for secure efficient key management mechanism with key agreement (EKeMS with KA) that seeks to minimize the malicious eavesdropping ability (MEA) in the network is to be introduced.

In Proposed System fig. 2, the KDC (Key Distribution Center) is introduced which keep the key of all the nodes, along with the mutual encryption, the key got by KDC also used thus more security is provided, which definitely going to reduce the MEA i.e. increase the security.

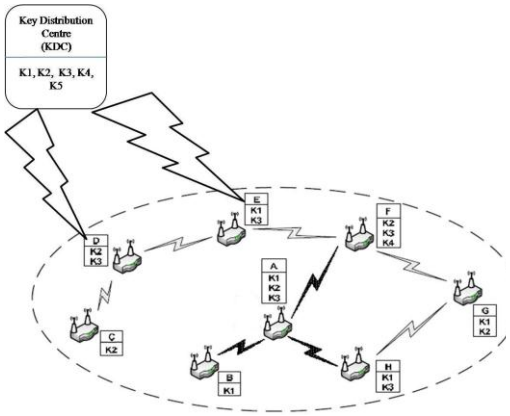

Figure 2. Key distribution using KDC

4. PROBLEM DEFINITION

To introduce an effective key management mechanism which seeks an encryption key assignment such that induced network is connected and well protected against potential eavesdropping.

Definition 4.1: ***(Shared encryption key** ($Sk_{u,v}$))*: Given any two neighboring nodes $u, v \in G$, let Key Distribution Centre (KDC) choose key K and allot to node u and v. Node u requests for key to KDC to share with node v. The KDC then finds least key used and allots to pairs u and v. Thus we can say that there exists a shared encryption key $Sk_{u,v}$ between node u and node v.

Definition 4.2: ***(2-hop compromised nodes** ($2CN_u$)):* Given nodes $u, v,w \in G$, where v is a 1-hop neighbor of u, and w is a 2-hop neighbor of u via v. If node u has been compromised, the *2-hop compromised nodes* of node u ($2CN_u$) is defined as the set of nodes (w), for which node v sends messages encrypted by any key $k \in Sk_{u,v} \cap Sk_{v,w.}$ [1]

Definition 4.3: ***(Node compromise ability** (NCA(u))):* Given a network G, we define the node compromise ability (NCA) for a compromised node $u \in G$, as the number of nodes in the set $2CN_u$ [1].

This is given in Eq. 1.

$$NCA(u) = |2CN_u| \quad (1)$$

Definition 4.4: ***(Malicious eavesdropping ability** (MEA)):* Given a network G with n nodes, where each node has been loaded by a set of encryption keys. The *malicious eavesdropping ability* in the network is defined as the maximum shared keys of node u and v allotted by KDC.

This is shown in Eq. 2.

$$MEA = max\{KDC(SK_{u,v})\} \quad (2)$$

Definition 4.5: (***Probability of repetition of keys P***): Given a network G and a set of encryption keys (K) allotted by KDC to nodes the key sharing probability is the average total number of shared keys $SK_{u,v}$ in the KDC.

This is shown in Eq. 3.

$$P = \sum \{KDC(SK_{u,v})\} \mid Number\ of\ Keys\ \} \quad (3)$$

Definition 4.6: *(***EKeMS** *problem):* Given a network *G* and a set of encryption keys (*K*) allotted by KDC to nodes, the Efficient Key Management Scheme (**EKeMS**) seeks a key assignment design *A* such that the *MEA* in the network is minimized using |*K*| encryption keys.

5. METHODOLOGY

1. Establish network topology
2. Select Node with maximum neighbors
3. Generate shared keys
4. Maintain keys t in the node
5. Make the following calculations
   a) *2-hop compromised nodes ($2CN_u$)*
   b) *Node compromise ability (NCA(u)*
   c) *Malicious eavesdropping ability (MEA)*
   d) *Probability of repetition of keys P*

6. IMPLEMENTATION

Efficient Key Management Scheme with Key Agreement (EKeMSwithKA) that seeks to minimize the malicious eavesdropping ability (*MEA*) in the network

Notation used in our scheme description

Notation	*Description*
u, *v*, *w*	Nodes
NIR(*u*)	*u*'s neighbors that have no common keys with *u*
K	A set of available encryption keys
k	An encryption key
keys(*u*)	A set of keys in node *u*

Table 1 Notation used in our scheme

Algorithm for An Efficient Key Management Scheme with Key Agreement (*G*, *K*)

1. *At KDC*
2. *for k number of keys*
3. *Keyalloted(k)=null*
4. *end*
5. *for each node u belonging to G do keys(u) = Null;*
6. *end for*
7. *for all nodes in G do*
8. *for each node u belonging to G do*
9. *Find NIR(u);*
10. *Calculate |NIR(u)|;*
11. *end for*
12. *Choose node u from G with the highest |NIR(u)|;*
13. *for each node v Belong NIR(u) do //Assign keys between node u and node v belonging NIR(u) based on the and maintain the keys allotted in KDC using following rules:*
14. *if keys(u) = null and keys(v) = null then*
15. *Choose k as the least used key from K and KDC;*
16. *Add k to keys(u) and keys(v);*
17. *Add k to keyalloted in KDC*
18. *else if keys(u) != null and keys(v) = null then*
19. *Choose k as the least used key from K not in keys(w), where w is a neighbor of u, if applicable, else choose k as the least used key from K and least used at KDC;*
20. *Add k to keys(u) and keys(v) ;*
21. *else if keys(u) != null and keys(v) != null then*
22. *Choose k as the least used key from K not on w where w from NIR(u) from NIR(v), if applicable, else choose the least used key from K and least key used from KDC;*
23. *Add k to keys(u) and keys(v);*
24. *end if*
25. *end for*
26. *end for*

7. REULTS

To illustrate the performance of our scheme, we implemented our solution (denoted by **EKeMS with KA** in the figures), and compared it with previous scheme in [1] (denoted by S**KeMS** in the figures). We considered static WMN with *n* nodes uniformly distributed in a square playing field of 1000*m* x 1000*m.*

The results shown are the average of 3 test runs for various scenarios. The first metric used for performance evaluation is *malicious eavesdropping ability ratio* (denoted as *MEA ratio* in the figures), which is calculated as the neighbor compromise ability (*NCA*) divided by the total number of neighboring nodes that are vulnerable to eavesdropping attack (discussed in Chapter 3 subsection 3.2). *Having smaller MEA ratio indicates that the network is more secured and more resistant against malicious eavesdropping attacks*. In our first tested scenario, we randomly distributed 300 and 400 nodes in a $10x10^5$ square meters. To achieve better security for KMS scheme, we provide different pool sizes ranges from (200−*400*) keys. *Note that, having different pool sizes doesn't affect our EKeMS with KA scheme*. Our first scenario's results are shown in fig. 3

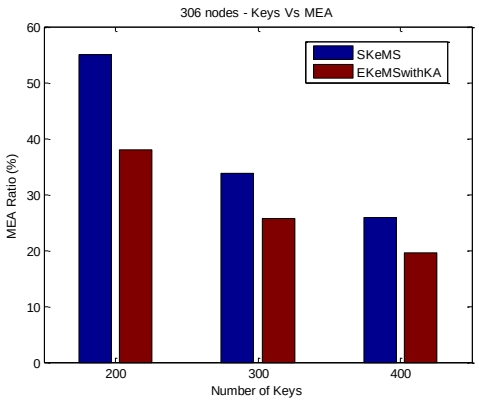

Figure 3. 306 nodes in 1000m X 1000m

Fig. 3 shows the *MEA ratio* versus different number of keys ranges from (200−400) keys. In our proposed scheme, we used least key used from KDC to be assigned keys among all nodes. On the other hand, in the SKeMS used available keys assigned keys among all nodes. From the fig.3 we can observe that as the Key size increases *MEA ratio* minimizes in our schemes and thus out performs the previous scheme i.e SKeMS. For example, with 200 keys chosen, we have an *MEA ratio* of SKeMS scheme is 55.06 %, while with 300 keys size *MEA* ratio of 33.76 % and with 400 keys size *MEA* ratio of 25.89 %. In our scheme, with 200 keys, our scheme

has an *MEA ratio* of 37.90 %, while with 300 keys size *MEA* ratio of 25/70 % and with 400 keys size *MEA* ratio of 19.60 %. These results also show that, by increasing the number of available keys, we can provide a better *MEA ratio*. The same results' trend can be seen in fig. 4, where we distribute 400 nodes in a 1000*m* square field.

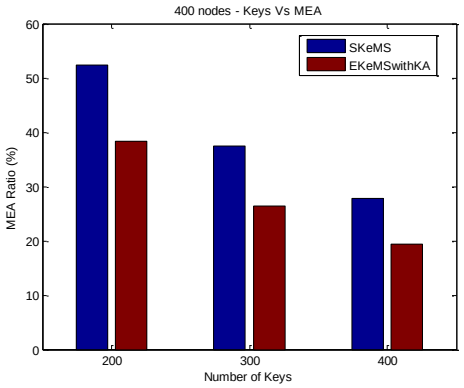

Figure 4. 400 Nodes in 1000 *m* x 1000 *m*

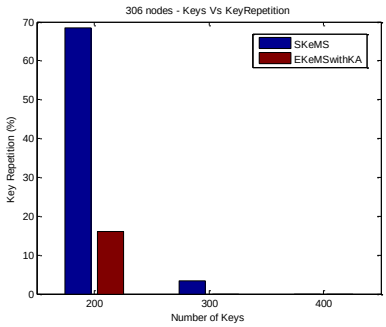

Figure 5. Key repetition for 306 nodes

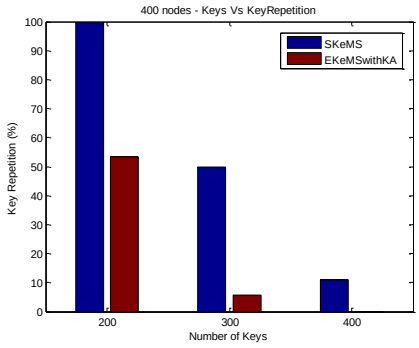

Figure 6. Key repetition for 400 nodes

Fig. 5 show the results of our third scenario in which we studied the schemes' performance with respect to key repetition factor when having different number of nodes in an area size of 1000*m* x 1000*m.* For example 200 keys and observed that probability of the keys of getting repeated in our scheme is 16% compared to SKemS which is 68.52%, and with 300 keys probability of the keys of getting repeated in our scheme is 0 %. The same results' trend can be seen in fig. 6, where we distribute 400 nodes in a 1000*m* square field

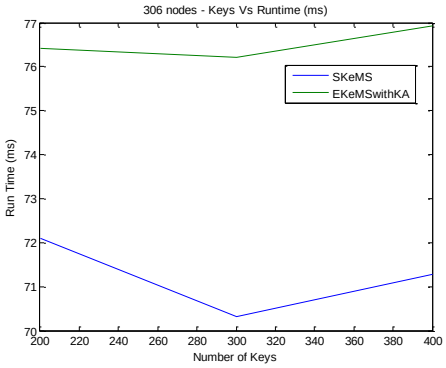

Figure 7. Running time for 306 nodes

In fig.7 it can be seen that running time of EKeMS with KA scheme is 76.42*ms* for 306 nodes in 1000*m* square compared to SKeMS scheme is 72.09*ms* which is little more than our scheme. This is due to the computational time taken by KDC in assigning the least key used to the nodes in the network. But this can be discarded as the security of the network with respect to MEA ratio is decreased using the EKeMS with KA scheme. The same results' trend can be seen in fig. 8, where we distribute 400 nodes in a 1000*m* square field.

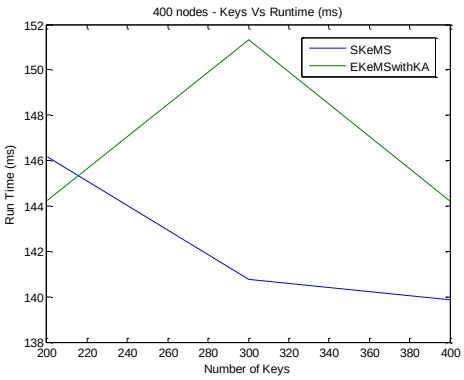

Figure 8. Running time for 400 nodes

8. CONCLUSION

In this work we defined the Efficient Key Management Scheme with Key Agreement (EKeMS with KA) problem, and presented an effective solution that provide a key assignment to a wireless mesh network. Our solution is resistant against malicious eavesdropping attacks. Simulation results showed that our solution performs well in terms of smaller malicious eavesdropping ability ratio and less key repetition factor. To sum up, in this work, we showed that a good key management scheme can ensure a more secure network.

9. REFERENCES

[1] Farah Kandah, Weiyi Zhang, Xiaojiang Du, Yashaswi Singh, A Secure Key management Scheme in Wireless Mesh Networks, *Communications (ICC), 2011 IEEE International Conference*, June 2011.

[2] I. F. Akyildiz, X. Wang, W. Wang, Wireless mesh networks: a survey; *Elsevier Journal of Computer Networks*, vol.47, Issue.4, pp.445-487, 2005.

[3] N. Asokan, P. Ginzboorg, Key Agreement in Ad Hoc Networks; *Computer Communications*, vol.23, pp.1627-1637, 2000.

[4] M. Cagalj, J. Hubaux, C. Enz, Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues; *ACM MobiCom'02*, Atlanta, Georgia, USA.

[5] S. P. Chan, R. Poovendran, M. T. Sun, A key management scheme in distributed sensor networks using attack probabilities; *IEEE GLOBECOM' 05*, vol.2, pp.5, St. Louis, MO, USA.

[6] X. Du, Y. Xiao, M. Guizani, H. H. Chen, An effective key management scheme for heterogeneous sensor networks; *Ad Hoc Networks, Special Issues in Sensor and Ad Hoc Networks*, vol.5, Issue.1, pp.24-34, 2007.

[7] L. Eschenauer, V. D. Gligor, A key-management scheme for distributed sensor networks; *ACM CCS'02*, Washington, DC, USA.

[8] P. Loree, K. Nygard, X. Du, An efficient post-deployment key establishment scheme for heterogeneous sensor networks; *IEEE GLOBECOM'09*, Honolulu, Hawaii, USA.

[9] A. Raniwala, T. Chiueh, Architecture and algorithms for an IEEE 802.11- based multi-channel wireless mesh network; *IEEE INFOCOM'05*, vol.3, pp.2223- 2234, Miami, Fl., USA.

[10] A. Raniwala, K. Gopalan, T. Chiueh, Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks; *ACM MobiCom'04*, Vol.8, Issue.2, pp.50-65, Philadelphia, PA, USA.

[11] J. Shi, R. Zhang, and Y. Zhang, Secure range queries in tiered sensor networks; *IEEE INFOCOM'09*, pp.945-953, Rio de Janeiro, Brazil.

[12] J. Tang, G. Xue, W. Zhang, Interference-Aware Topology Control and QoS Routing in Multi-Channel Wireless Mesh Networks; *ACM Mobi-Hoc'05*, pp. 68-77, Urbana-champaign, IL, USA.

[13] W. Zhang, F. Kandah, J. Tang, K. Nygard, Interference-Aware Robust Topology Design in Multi-Channel Wireless Mesh Networks; *IEEE CCNC'10*, pp.6-10, LAS Vegas, NV, USA.

[14] X. Zhao, Y. Lv, T. H. Yeap, B. Hou, A Novel Authentication and Key Agreement Scheme for Wireless Mesh Networks; *In Proceedings of NCM'09*, pp.471-474, Washington, DC, USA.

[15] Dr. M.S.Aswal, Paramjeet Rawat, Tarun Kumar, Threats and Vulnerabilities in Wireless Mesh Networks, International Journal of Recent Trends in Engineering, Vol 2, No. 4, November 2009