B.Tech. Project Report

IT-414

on

Secure Key Management Scheme with Vulnerability Constrained Routing for Wireless Sensor Networks

BY

RISHITA JAGGI (1140213)

EKTA (1140221)

SWAPNIL BHATT (1140275)

Under the Supervision of

Mrs. Priyanka Ahlawat, Asst. Prof.

DEPARTMENT OF COMPUTER ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
KURUKSHETRA – 136119, HARYANA (INDIA)
Dec 2017- April 2018

CERTIFICATE

I hereby certify that the work which is being presented in this B.Tech. Major Project (IT-414) report entitled **"Secure Key Management Scheme with Vulnerability Constrained Routing for Wireless Sensor Networks"**, in partial fulfilment of the requirements for the award of the **Bachelor of Technology in Information Technology** is an authentic record of my own work carried out during a period from December 2017 to April 2018 under the supervision of **Mrs. Priyanka Ahlawat**, Assistant Professor, Computer Engineering Department.

The matter presented in this project report has not been submitted for the award of any other degree elsewhere.

Signature of Candidate

RISHITA JAGGI (1140213)

EKTA (1140221)

SWAPNIL BHATT (1140275)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Date:

Signature of Supervisor

Mrs. Priyanka Ahlawat

Assistant Professor

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Wireless sensor networks (WSNs) use is increased due to recent advances in electronic and computer technologies. The applications of WSNs are mainly seen in critical tasks such as military. Thus, the security issues in WSNs is major concern for research areas. Security issues in WSNs is more complicated Compared to other wireless networks, such as ad hoc wireless LAN and cellular networks because it has the bound capabilities of sensor node hardware and the characteristics of the deployment environment.

Considering this scenario, our aim is to provide a procedure which can be applicable in WSNs to decrease Path Compromising Ratio by providing an algorithm to calculate the vulnerabilities of different sensor nodes by studying the factors like gradient attack distribution and neighboring nodes distribution in a WSN. As a result of this algorithm, one can easily detect the most vulnerable node and can make the WSN more secure from the adversaries.

One more constituent will be key management, under which the key pool under nodes will be hashed to decrease the number of compromised links and consequently decrease the power of adversary attacking model as compared to the basic model where no hashing techniques are applied to prevent nodes which have the same key pool. The results will include trends between:

- Path Compromising Ratio after every iteration.
- Comparison of links compromised in basic scheme and developed scheme.

1. INTRODUCTION

With the development of network and communication technology, the problem of wiring is solved with WSN; especially it has wide use and practicability in the area of remote sensing military sensing and tracking, environment monitoring, patient monitoring and tracking, and smart environments [1].Key distribution is a major issue in wireless sensor network (WSN) design [2]. WSNs are the networks of sensor nodes having very small, memory-constraint and battery-power, which have the capability of wireless communication over the communication range. They need to be well arranged to build a fully functional network due to memory and power constraints. The proposed approach is mainly based on the vulnerability constrained routing, where the vulnerability of each sensor node will be calculated using some factors. Key pre-distribution is the process of distribution of keys onto nodes before deployment by the key server from the large key pool. Therefore, the nodes build up the network using their secret keys after deployment, that is, when they reach their target position. There is a fundamental question as to how one can design a WSN that is secure, efficient and reliable. Achieving secure communication even in presence of the attacker is one of the aim of the WSN security. The key management scheme (KMS) plays a very important role in these applications. Many current key management proposals do not consider the attack distribution. They assume the attack probability to be the same for every node. Besides improving the key pre-distribution step of key management, we have applied our model in which all the links having same key will not be compromised and the network will be more reliable.

2. MOTIVATION

With the development of network and communication technology, the problem of wiring is solved with WSN; especially it has wide use and practicability in the area of remote sensing military sensing and tracking, environment monitoring, patient monitoring and tracking, and smart environments [1]. Compared to other wireless networks, such as ad hoc wireless LAN and cellular networks, security issues in WSNs is more complicated due to the bound capabilities of sensor node hardware and the characteristics of the deployment environment.

The routing schemes which are already present do not consider the probability of node getting attacked which is dependent on various factors such as gradient, neighbors etc. Due to ignorance of such factors the routing scheme does not find the safest possible path.

WSNs use key distribution to link with each other as they are not physically linked. Therefore, there is probability of keys getting hacked by attackers. All the links which share that key are compromised.

The above stated problems led us to research in the field of security in a WSN. So we came up with a solution of Vulnerability Constrained Routing and secure Key management scheme using hashing approach.

3. LITERATURE SURVEY

In a WSN, security guaranteeing secure applications relies on the confidentiality and integrity of the message exchange process [8]. To ensure WSN security, safe and reliable data transfer is required from one sensor node to another one in a deployed wireless sensor network. In [9], Eschenauer and Gligor have demonstrated that confidentiality and integrity of WSN can be easily destroyed by physically capturing sensor nodes and extracting the cryptographic keys from their memories [10]. This type of threat is called node capture attack [9], which is possible due to the unattended operation of wireless nodes and the prohibitive cost of tamper-resistant hardware. It destroys the privacy, safety and reliability of the network [11]. If no action is taken to safeguard against such an attack, the sensitive data or private messages transmitted in the network will be exposed, leading to catastrophic consequences. In order to model the effect of the node capture attack, several literatures [12-14] discussed the attacking methods when the network is configured with the random key pre-distribution schemes [18]. In general, the approaches of formalizing the effect of the node capture attack can be categorized into several types: probability analysis [12-14], system theoretic approach [9], epidemic theory [15, 16] and vulnerability evaluation [16-18].Probability analysis [12-14] calculated the characteristic parameters of the network by utilizing probabilistic model, but it neglected to consider the attacking efficiency and the resource expenditure in mounting an attack. System theoretic approach [9] provided a control theoretic framework to model physical node capture and a network response strategy to guarantee the network connectivity and stability under node capture attack. However the consequence of attack was ignored. Epidemic theory [15, 16] formalized the node capture attack as an epidemic propagation process and provide analytical results of the attack based on different node deployment strategy. But they didn't consider the dynamicity and the intelligence of the attacker in modelling the attack. They assume that the attacker captures the nodes independently at random. In[16], Tague, P. et al. point out that the adversary can compromise a node intelligently to improve the efficiency of the node capture attack publicly available information, which is learned through eavesdropping on insecure message exchange throughout the network. Hence, when evaluating the effect of the node capture attack, the intelligence of the attacker must be taken into consideration. In vulnerability evaluation method [16-18], Tague, etal. proposed a formal method to formalize the vulnerability of the network by using circuit theoretic analysis. They devised a greedy

node capture approximation using vulnerability evaluation (GNAVE) to approximate the minimum cost. However, GNAVE neglected to analyse the impact of the nodes who do not belong to any routes. Therefore, the attacking efficiency is relatively low. After the brief summary about node capture attack,Yu, C-M., Li, C-C., Lu, C-S[18] focused on an application driven attack. For the non – uniform sensor deployment an attack probability-based deterministic key pre-distribution method [18] is discussed which focuses on neighbour constraint. According to that, two sensor node within the same deployment group communicate more frequently in comparison to nodes belongs to different deployment group. After the literature survey regarding node capture along with vulnerability of each sensor node Xiangqian Chen [2] proposed some attacking models for WSN. According to basic attack distribution model, when the attack probability and the frequency are comparatively small, the correlation of attacking among neighbours can be neglected. Under this condition, basic models are accurate enough to estimate the attack probability. Uniform attack distribution model [2] explains that each sensor node have approximately same attacking probability but in real world it is not possible , so Chen[2] proposed gradient based attack distribution model which gives the reliable information regarding attacking probability of each sensor node in a deployed network is different and depends on location of each sensor node. After studying more about sensor nodes and their attacking probability, now the major concern is to secure WSN. In WSN security, the KMSs play a crucial role in providing security to the WSNs. Recently, many KMSs have been proposed for WSNs. The first KMS was proposed by Eschenauer and Gligor [9] called EG scheme. It is a probabilistic key predistribution scheme. In this scheme, the keys are selected from a large pool of keys with some probability. If two sensor nodes have a shared key and are in communication range, they can establish a secure link. The E–G scheme is based on random graph theory. The sensor nodes and the links are represented by the vertices and edges of a graph. The shortest path connecting the two nodes is called the key path. It is demonstrated that the integrity and confidentiality of the network traffic is destroyed by physically capturing the node and extracting the cryptographic keys. The E–G scheme is further strengthened by Q-composite scheme in which the link key is composed of more than one key (at least q keys) [9]. Anita et al. [10] proposed a triple key management scheme, in which the key pool based KMS is combined with Q composite scheme. This scheme has high network resilience and high probability of the key connectivity. Du et al. proposed deployment based key predistribution scheme [22]. Some

optimizations and enhancements to further improve the key management schemes are done in [23-25]. Hybrid key management schemes to strengthen the security of network are given by Qiu et al. [26].

4. VULNERABILITY CONSTRAINED ROUTING

Preliminaries

In this section, various models and definitions related with the proposed VCR are included. The notations and their related descriptions are listed in Table 1.

1. Key distribution scheme

Key predistribution [2] is the technique of distributing the keys onto the sensor nodes involved in network traffic nodes before deployment of the sensor nodes in network traffic. That is why, the nodes build up the network using their secret keys after deployment, that is, when they reach their target position. Key predistribution schemes are the various techniques that have been invented by researchers and an interesting subject for research in WSN field. Basically a key predistribution scheme has 3 phases:

1. Key distribution
2. Shared key discovery
3. Path-key establishment

During these phases, secret keys are generated, placed in sensor nodes and each sensor node searches the area in its communication range to find another node to communication. Whenever two nodes have same assigned key then the link is formed between the nodes. Afterwards, paths are established connecting these links, to create a connected graph. The result is a wireless communication network functioning in its own way, according to the key predistribution scheme used in creation.

2. Spatial correlation

In wireless sensor networks (WSNs), dense and huge deployment of sensor nodes makes the sensor observations highly correlated in the space domain. In other words, the existence of spatial correlation implies that the readings from sensor nodes which are geographically close to each other are expected to be largely correlated. Due to spatial correlation, neighbor of attacked node has high attacking probability. Spatial arrangement of sensor nodes is therefore a matter of concern in network deployment.

3. Performance Metrics

1. *Vulnerability coefficient* : This is computed for every sensor node in order to capture the most vulnerable. This coefficient is also used to determine the vulnerability rank of all the sensor nodes.
2. *Link compromise ratio* : This is the ratio of number of links compromised due to the capture of most vulnerable node by our proposed model to the number of links compromised by other model.

4. Network Model

The network is expressed in terms of a graph where the vertices denote the sensor nodes and the edges denote the links established between the nodes. Each sensor node has a set of keys that are randomly assigned from a large key pool. These assigned keys are known as the distributed keys of the sensor node. A link can only be formed between two sensor nodes if and only if there is at least one common key between their distributed keys.

Table 1- Symbols and Notations

Symbol	Notation
C_k	The set of compromised keys
d_i	Distance of *i*th node from sink node
K	The set of keys in the key pool
K_i	The set of keys assigned to *i*th node
$k_{i,j}$	Link key between *i*th node and *j*th node
KR_i	Key supremacy ranking of ith key
L	The set of links in the network
LCR	Link compromised ratio
LF_i	Location factor of *i*th node and it is in interval [0,1]
$l_{i,j}$	Link formed between the *i*th node and *j*th node
NF_i	Neighbor factor of *i*th node and is is in interval [0,1]
n_i	The *i*th node in the network
P	Number of paths formed between a source and sink pair

PF_i	Path factor of *i*th node
R_i	Rank of *i*th node
S,D	Source node and Sink node respectively
VC_i	Vulnerability Coefficient of *i*th node

5. Approach

The vulnerability of the sensor node is computed in terms of two variables namely location factor (LF), neighbor factor (NF) and path factor (PF). These variables are numeric and are solely defined by the network designer based on the physical properties of the node. The value of these variables is in the interval of (0, 1]. These variables are as follows:

(1) Location Factor of the sensor node (LF)

The nodes which are located near the sink node are more prone to adversarial attacks then the nodes which are located far away from sink node. The reason behind this is that the sink node is most protected node in the network as it is the only gateway for forwarding the aggregated data to the base station, so there is less probability of sink node to be captured due to its security and more probability of the nodes to be captured which are present close to the sink node.

(2) Neighbor Factor of the sensor node (NF)

Due to spatial correlation neighbor of attacked node has high attacking probability.

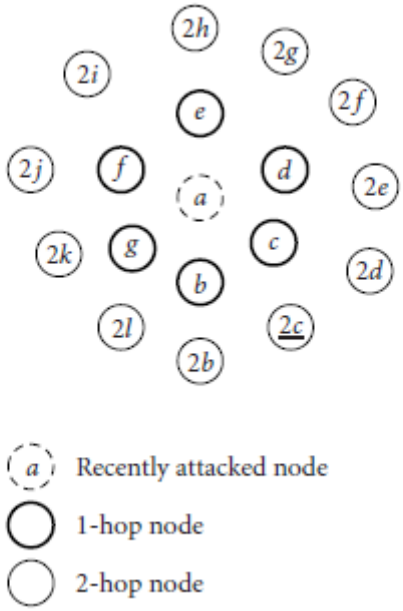

Figure 1- Arrangement of sensor nodes

We assume 1-hop neighbors of the given node are the nodes which are the immediate neighbor nodes of the given node and can directly connect to this node; 2-hop neighbors of the given node are the nodes which can contact the given node at least by two hops, and so on. We call all the 1-hop neighbors of the given node as 1-hop layer nodes, and all the 2-hop neighbors as 2-hop layer, and so on. In dense WSNs, the distances between a given node and its 1-hop neighbors are nearly equal. Therefore, we suppose that each 1-hop benign neighbor of a recently attacked node has the same probability of being chosen as the attacking target of an adversary which corresponds to this recently attacked node. Similarly, we make the same assumption of 2-hop neighbors, 3-hop neighbors, and so on. While the probability that one of 1- hop layer nodes being chosen as the attacking target is larger than the probability of 2-hop layer node, and so on.

For a given recently attacked node , the probability of a corresponding adversary choosing a 1-hop layer node as the attacking target is larger than the probability to choose a 2-hop layer node (i.e.,$p1 > p2 > p3 \ldots$) [12] . Mathematically NF can be calculated as:

$$NF_i = \frac{2*(\text{No. of 1-hop neighbors of } n_i) + (\text{No. of 2-hop neighbors of } n_i}{2*(\text{No. of 1-hop neighbors of } n_i + \text{No. of 2-hop neighbors of } n_i)}$$

(3) Path Factor of sensor node(PF)

It is the third metric that is included in the vulnerability factor of every sensor node it indicates the number of paths from source to sink which include that sensor node because the vulnerability of sensor node is directly proportional to the number of paths in which the sensor node is present.

After calculating the above metrics the vulnerability of the sensor node is computed by vulnerability coefficient (VC) i.e.

$$VC_i = 0.5*(LF_i + NF_i) + 0.5*PF_i$$

We have also proposed an algorithm for calculating the rank of the node (R_i). It is given below-

Algorithm 1: Algorithm to find rank of node

1. **Input**: *G(N)*
2. **Output**: Rank of each sensor node(R_i)
3. **for all** $n_i \in$ N
4. $LF_i = d_i$
5. Calculate NF_i
6. Calculate PF_i
7. $VC_i = 0.5(LF_i + NF_i) + 0.5*PF_i$
8. **end for**
9. **for each** n_i
10. **if** $VC_i > VC_j$ where ($j \in$ N except i)
11. $R_i > R_j$
12. **end if**
13. **end for**
14. **return** R_i where $i \in \{1,N\}$

5. KEY MANAGEMENT

1. Key supremacy ranking computation

During the pre-distribution phase of key distribution scheme, a large number of keys are picked randomly and assigned to the nodes from the key pool by the key server. It is possible that the key may get repeated, i.e. server assigns some keys more number of times than other keys into the wireless sensor nodes. These keys, when captured by adversary reveal larger fraction of network traffic and thus, cause maximum destructiveness. Thus, an intelligent attacker by eavesdropping, tries to capture the high dominance keys to disrupt the network traffic.

For the WSN security, it is necessary to store the count of keys to compute the vulnerability of each path. In WSN, due to random assignment of keys, there is a large probability of overlapping keys which may help the attacker to destroy network by destroying paths having same keys.

Algorithm 2 uses the overlapping link keys to compute a variable called **key supremacy ranking,** which is based on the number of times the link keys are overlapped in different paths in the network.

Algorithm 2: To compute key supremacy ranking

---

1. **Input** : $G(n)$
2. **Output** : Key supremacy ranking of each assigned key KR_i
3. **For all** $K_i \epsilon K$
4. **For all** $n_j \epsilon N$
5. **if** the key K_i is contained in the node n_j
6. *Count* K_i ++
7. **For all** $K_i \epsilon K$
8. *Key supremacy ranking* KR_i = *total no of keys / count* K_i
9. **End for**
10. **Return** KR_i where $i \epsilon \{1,K\}$

Algorithm 2 proceeds as follows- we have to compute key supremacy ranking of each node. For that, firstly we compute the count which is actually the count of repetition of any key.

After computing the count for each pre-distributed key, we compute the key supremacy ranking for each node.

2. Disruption of network traffic on the basis of key supremacy ranking

An intelligent attacker by eavesdropping, tries to capture the high supremised keys to disrupt the network traffic. As attacker capture the key of high supremacy ranking, can easily detect the path having overlapped highly supremised key. As attacker capture the high supremised key, there will be more probability of more disruption due to more path captured.

Algorithm 3: To compute link compromised ratio on the basis of key supremacy ranking

---

1. **Input** : *G(n)*
2. **Output** : Link compromised ratio *LCR*
3. Capture key having highest supremacy ranking *KH*
4. **For all** edge of *G(n)*
5. **If** edge contains *KH*
6. *Count++*
7. **End if**
8. **End for**
9. *Link compromised ratio (LCR) = count / total edge in graph*
10. **Return** *LCR*

Algorithm 3 proceeds as follows- to compute the link compromised ratio LCR on complete disruption of network traffic, firstly we need to find the key having highest supremacy ranking. An intelligent attacker now tries to capture this key. On capturing this key all the links having highest supremacy ranked overlapped key got compromised. Now the final link compromised ratio will be total count of edges having highest supremised key divided by total no of edges in the network traffic of wireless sensor network.

3. Hash value computation

As an intelligent attacker by eavesdropping, tries to capture the high supremised keys to disrupt the network traffic, to secure the network our proposed approach is to introduce

hashing concept. Pre-distributed keys are hashed using hash function h(k) and pre-distributed keys are now converted into hashed keys to secure network traffic.

Algorithm 4: To compute hash value of keys

1. **Input** : *G(n)*
2. **Output :** Hash value of each key of edges is stored
3. *Hash number = (Total No of nodes)*^{2}
4. **For all** edge $E_i \epsilon E$
5. *Power = average of rank of the nodes in the edge* E_i
6. *Hash value* $h_i(E_i)$ *= [Hash number %Key of edge* E_i *]* ^{power}
7. **End for**

Algorithm 4 proceeds as follows- to compute the hash value of assigned key to each edge firstly we compute hash number which is nothing but square of total no of nodes in graph. Then for all edges power will be simply average of rank assigned to nodes of connecting edge. Hash value is computed as mentioned in algorithm at step 6. These hash values for each edge keys is stored for less disruption of network traffic.

4. Disruption of network traffic on the basis of hashing approach

After converting normal pre-distributed keys into hash keys, an attacker tries to capture key having high supremised ranking but probability of disruption is less using hashing approach in comparison to normal pre-distribution scheme.

Algorithm 5: To compute link compromised ratio on the basis of key supremacy ranking including hashing approach

1. **Input** : *G(n)*
2. **Output** : Link compromised ratio *LCRH*
3. **For all** $K_i \epsilon K$
4. Compute hash value for each edge using algorithm 4
5. **End for**
6. Capture key having highest supremacy ranking *KH* with hash value *h(KH)*
7. Capture edge *Ec* having hash value *h(KH)*

8. **For all** $_i$ edge which contains key *KH*
9. **If** $h(E_i) > h(Ec)$
10. *countH ++*
11. **End for**
12. *Link compromised ratio LCRH = countH / total edge in graph*
13. **Return** *LCRH*

Algorithm 5 proceeds as follows- to secure network traffic hash keys for each edge using hash function which is predefined. Capture the key having highest supremised ranking and store the hash value assigned to particular captured edge. Now to compute link compromised ratio after including hashing concept, count the no of edges which have same key as captured key but having less hash value in comparison to captured edge hash value. Finally Link Compromised Ratio is computed by dividing total captured edges to the total number of edges in graph.

6. DATA FLOW DIAGRAM

1. Level 0 DFD

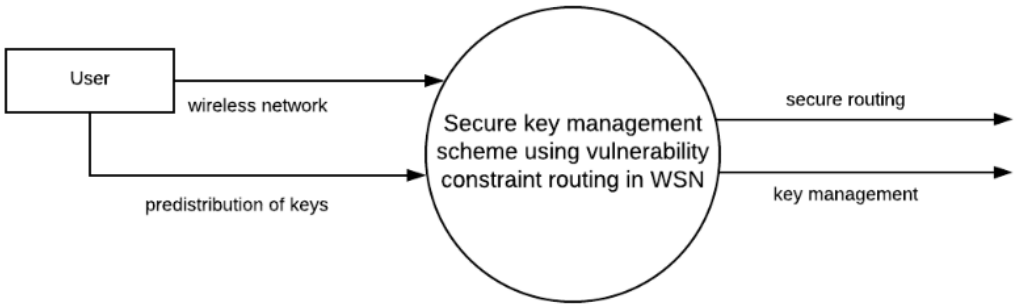

Figure 2- Level 0 DFD

2. Level 1 DFD

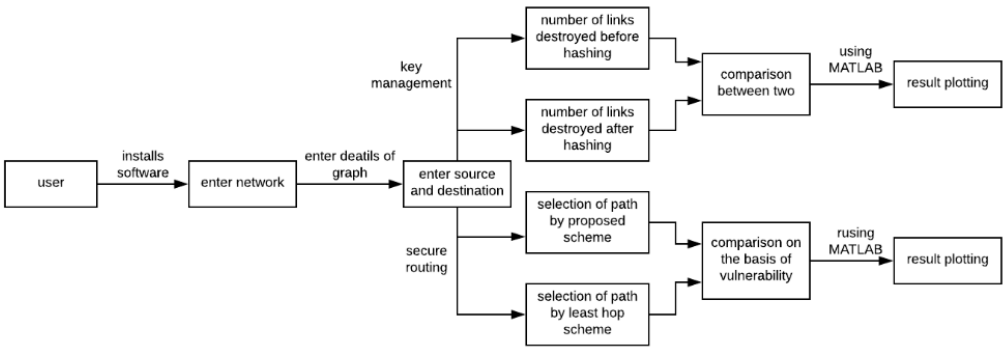

Figure 3- Level 1 DFD

3. Level 2 DFD

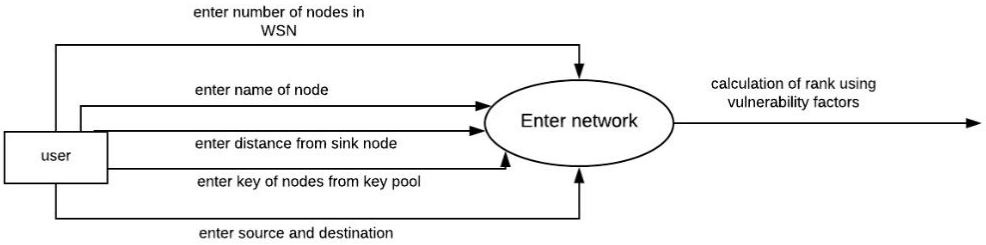

Figure 4- Input DFD

Figure 5- Key Management DFD

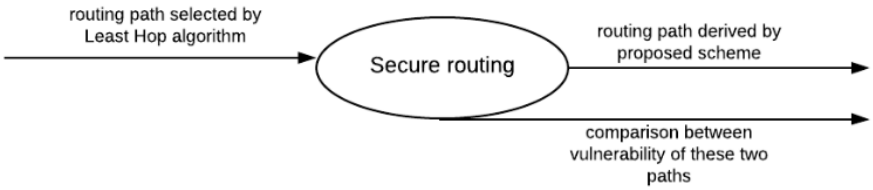

Figure 6- Routing DFD

7. RESULT

The results of the proposed scheme are shown below-

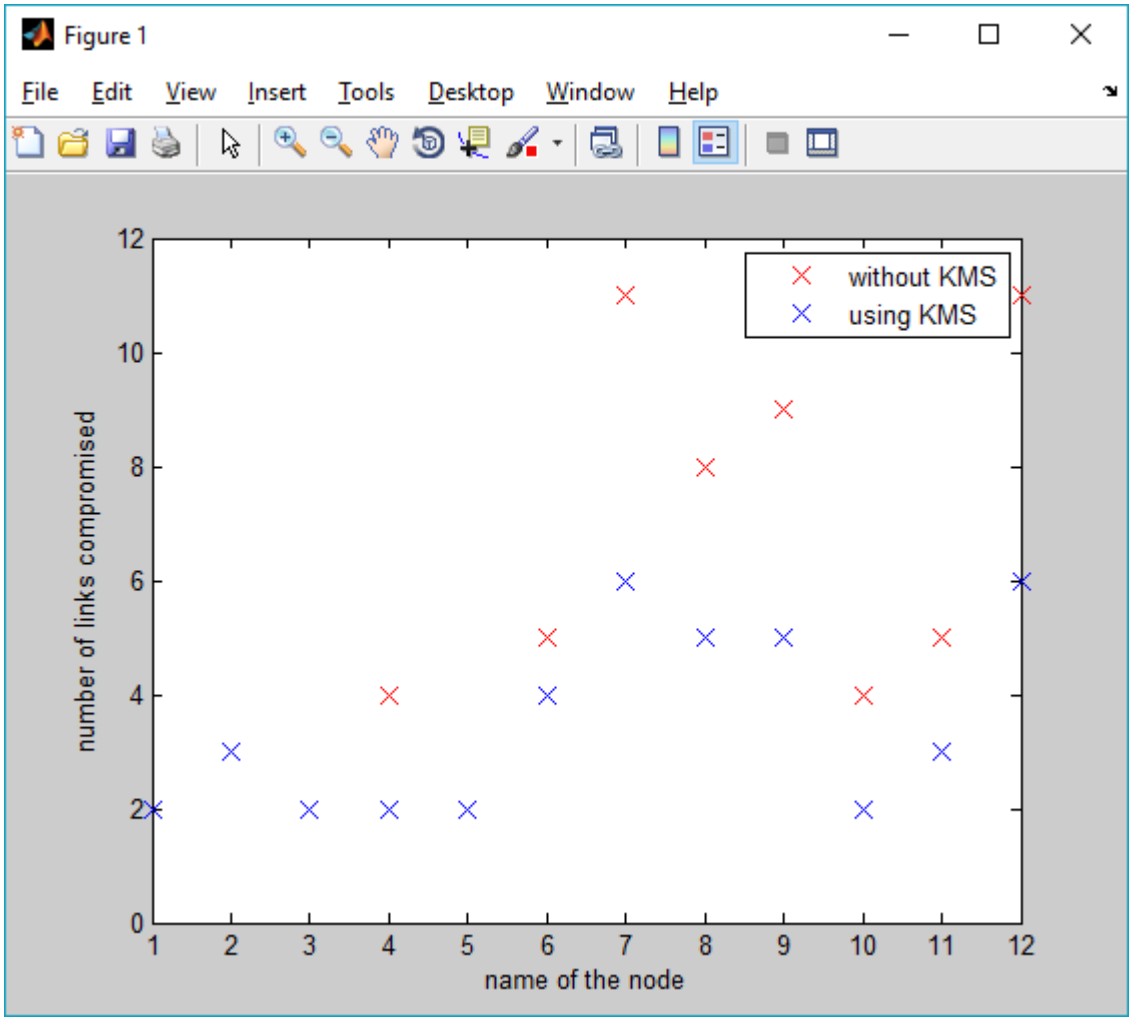

Figure 7- Results showing the number of links compromised with and without using KMS for each node

The above graph shows that after applying key management scheme using hashing the number of links compromised has been reduced for some nodes. It shows that security of the network has been increased in terms of links that can be attacked.

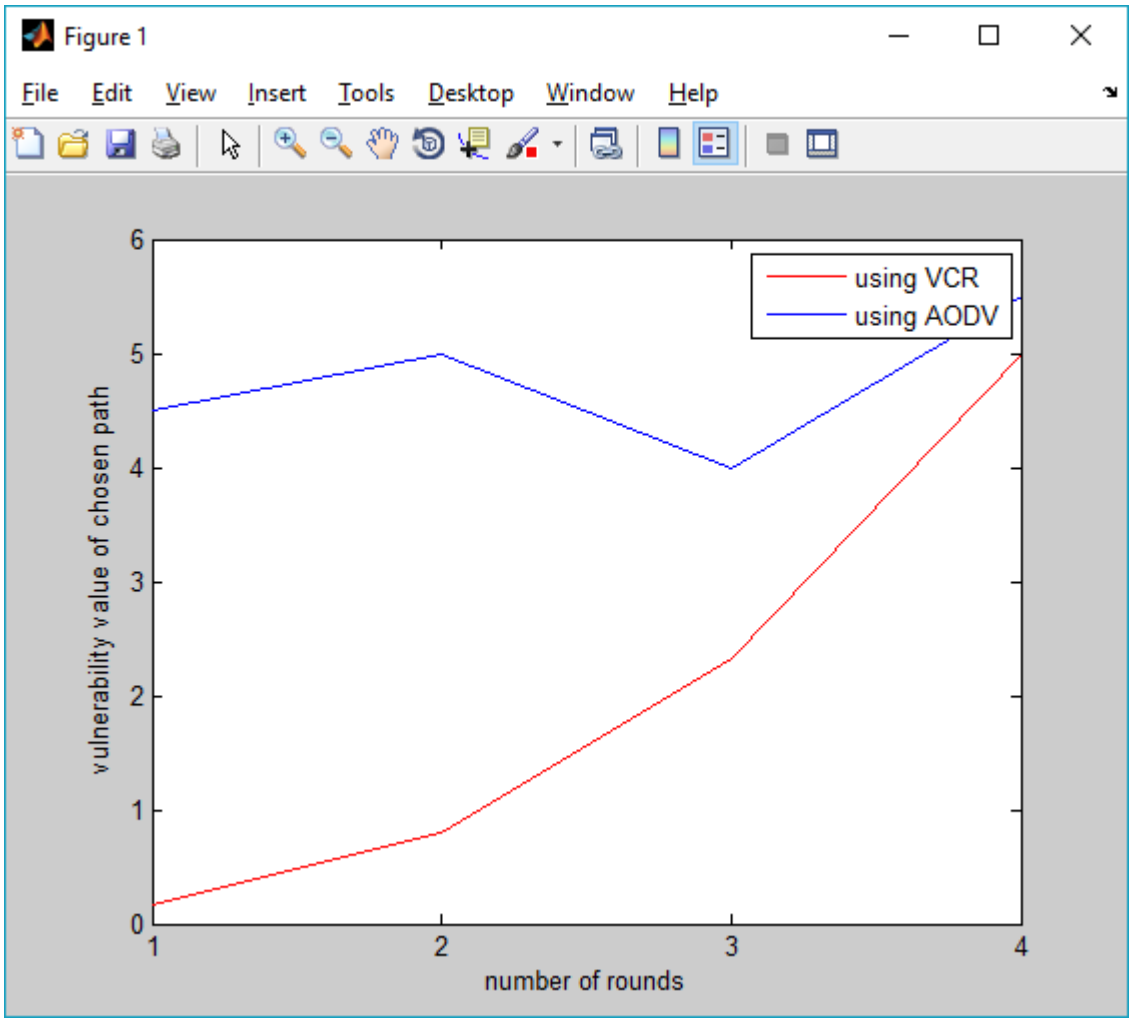

Figure 8- Results showing that VCR choses a safer path than AODV

The above graph shows the comparison between paths that are being chosen by using VCR scheme (our proposed scheme) and AODV routing (which uses least hop path). It shows our scheme is an enhanced version of AODV in terms of secured routing.

The following two figures are screenshot of the network simulation done in NS2 software. We have implemented our proposed routing scheme with modification of AODV in NS2 on a wireless network traffic. The communication in these nodes occur on the basis of their ranges.

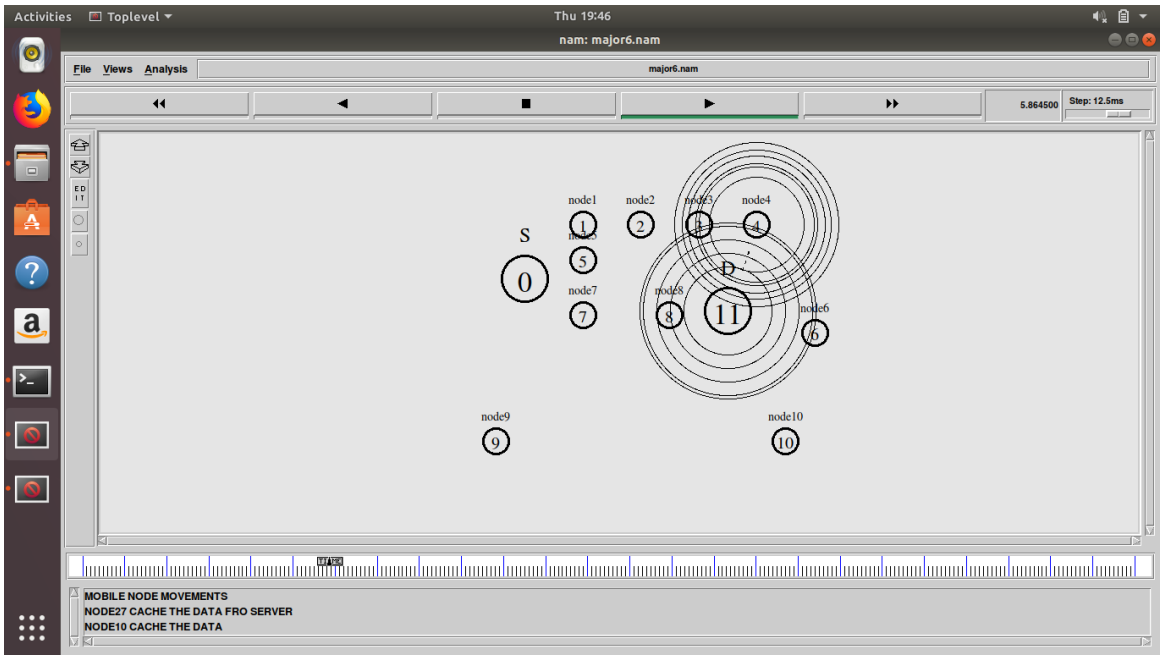

Figure 9- Simulation of network on NS2

This figure shows the initial network status before any of the most vulnerable node is captured.

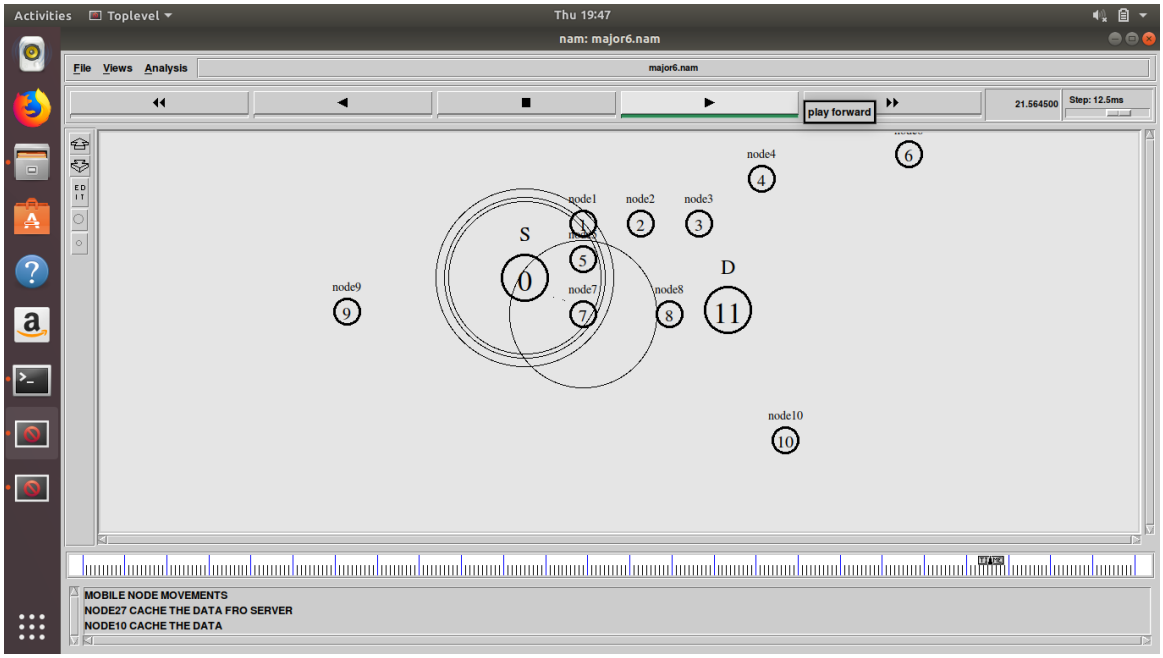

Figure 10- Simulation after most vulnerable node capture in last round

This figure shows the status of the network after the last round in which the node is captured and new path is chosen.

8. CONCLUSION

The proposed scheme using Vulnerability Constrained Routing (VCR) and Key Management Scheme (KMS) is providing a double layer security to the WSNs. With VCR each node is assigned a rank on the basis of its vulnerability computed using three factors- neighbor factor, gradient factor and path factor. The result shows that using such vulnerability constrained routing, a safer path is chosen in the WSN.

Another aspect of security that we have implemented is key management. If a link is compromised, then the number of links with same key that are compromised is minimum. That is, link compromising ratio is reduced making the network safer.

REFERNCES

[1] Akyildiz, I.F., W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks", IEEE Communications Magazine, vol.40, no.8, pp. 102-114, August 2002.

[2] He, X., Neidermeier, M., & Meer, H. (2013). Dynamic key management in wireless sensor network: A survey. Journal of Network and Computer Applications, 36, 612–622.

[3] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: A survey. Computer Networks, 38(4), 393–422.

[4]. Zhang, J., & Varadharajan, V. (2010). Wireless sensor network key management survey and taxonomy. Journal of Network and Computer Applications, 33(2), 63–75.

[5] Eschenauer, L., & Gligor, V. (2002). A key-management scheme for distributed sensor networks. In Proceedings of 9th ACM Conference on Computer and Communications Security (pp. 41–47).

[6]. Chan, H., Perrig, A., & Song, D. (2003). Random key predistribution schemes for sensor networks. In Proceedings of 2003 IEEE Symposium on Security and Privacy, California, USA (pp. 197–213).

[7]. Anita, E. M., Geetha, R., & Kannan, E. (2015). A novel hybrid key management scheme for establishing secure communication in wireless sensor networks. Wireless Personal Communications, 82(3), 1419–1433.

[8] Aneta e al., Obaidet, M. S., & Lin, C. (2012). A high efficient node capture attack algorithm in wireless sensor network based on route minimum key set. Security and Communication Networks, 6,
230–238.

[9] Eschenauer, L., & Gligor, V. (2002). A key-management scheme for distributed sensor networks. In Proceedings of 9th ACM Conference on Computer and Communications Security (pp. 41–47).

[10]. Anita, E. M., Geetha, R., & Kannan, E. (2015). A novel hybrid key management scheme for establishing secure communication in wireless sensor networks. Wireless Personal Communications, 82(3), 1419-1433.

[11] Jamal N. Al-Karaki, Ahmed E. Kamal, "Routing Techniques In Wireless Sensor Networks: A Survey", *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6-28, 2004.

[12] De, P., Liu, Y., & Das, S. (2006). Modeling node compromise spread in wireless sensor networks using epidemic theory. International Symposium on World of Wireless, Mobile and Multimedia Networks, IEEE Computer Society, Washington DC, USA (pp. 237–243).

[13] De, P., Liu, Y., & Das, S. (2009). Deployment-aware modeling of node compromise spread in wireless sensor networks using epidemic theory. ACM Transactions on Sensor Networks, 5(3), 1.33.

[14] Tague, P., Slater, D., Rogers, J., & Poovendran, R. (2008). Vulnerability of network traffic under node capture attacks using circuit theoretic analysis. In 27th Annual IEEE Conference on Computer Communications. INFOCOM 2008, IEEE (pp. 161–165).

[15] Eschenauer, L. and V.D.Gligor, "A key management scheme for distributed sensor networks", in Proceedings of the 9th ACM conference on Computer and communications security, Washington DC, USA, November 18–22, 2002, 41-47.

[16] Jamal N. Al-Karaki, Ahmed E. Kamal, "Routing Techniques In Wireless Sensor Networks: A Survey", *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6-28, 2004.

[17] Tague, P., Slater, D., Rogers, J., & Poovendran, R. (2009). Evaluating the vulnerability of network traffic using joint security and routing analysis. IEEE Transactions on Dependable and Secure Computing, 6, 111–123.

[18] Lin, C., & Wu, G. (2013). Enhancing the attacking efficiency of the node capture attack in WSN: A matrix approach. Journal of Supercomputing, 66(2), 989–1007.