ISSN: 0974-2115
www.jchps.com Journal of Chemical and Pharmaceutical Sciences

Enhancement of distance vector routing protocol through the detection of malicious routers

R Sabitha*
Department of IT, Jeppiaar Engineering College, Chennai
***Corresponding author: E-Mail: sabisam73@gmail.com**

ABSTRACT

The world of telecommunication has been changing rapidly over the last few decades. The Internet has replaced circuit switching by packet switching. The Internet has become the foundation for world-wide digital communication. Many critical applications depend on the enormous infrastructure of the Internet for their functionalities. As the Internet changed its face over the years, the network security also had to follow suit. The presence of hackers, eavesdroppers and malicious routers leads to many instances wherein network infrastructure has been compromised. There has been a significant amount of research on securing the information rather than on securing the Internet infrastructure. Thus, Internet infrastructure security is an important issue and it requires more research attention. Among different network threats, the routing table poisoning attack is the most devasting and least researched topic which needs immediate research attention. In this paper, An efficient method is developed for faulty update detection of routing table in distance vector protocols. The proposed method designates a trusted router which has the capability of finding the wrong updates in the routing table and is able to detect the harmful router.

KEY WORDS: Link State Protocols, Distance Vector Protocols, Routing Information, Secure Routing Protocols, Routing Security.

1. INTRODUCTION

The Internet has been witnessing enormous growth over the last several years. Until now, the main research focus has been on improving the performance and scalability of the Internet. Although performance and scalability have their place in Internet research, the enormity of the Internet has forced the research community to look at its dependability aspects. The Internet, like any other product, is prone to failures, and researchers have started to realize the importance of dependable communication in order to tolerate device failures (e.g., link and node failures) and to overcome the presence of malicious users or “hackers”.

A growing number of attacks, such as hacking and viruses, are being perpetrated by people who fully take advantage of openness and anonymity of the Internet. Network attacks lead to loss of money, time, products, reputation, lives and sensitive information. An attacker could simply target the infrastructure instead of launching an attack against the connection between any two secure communication ends (Panagiotis Papadimitratos, 2002). There have been instances wherein the network infrastructures such as routers and servers have been compromised by malicious adversaries. Attacks on the Internet infrastructure can lead to enormous destruction, since different infrastructure components of the Internet have implicit trust relationships with each other (Anirban Chakrabarti, 2002).

The majority of work on routing protocols for the Internet has proceeded in two main directions: distance vector protocols (e.g. RIP (Malkin, 1988) and link state protocols (e.g. OSPF (May, 1994)). In distance vector protocols, each node sends the distance (in hops) to its neighbor nodes. In case of link state protocols, each node periodically floods the state of its links to all the nodes in the network. Distance vector protocols are less robust than the link vector protocols. This is because each router computes the routes based on the computation done in the other routes in the network. Distance vector protocols can be subjected to: (i) Link Attacks, where an adversary gets access to a link in the network and changes the distance vector update passing through the link and (ii) Router Attacks, where a malicious router (source or intermediate) changes the distance vector information. It is to be noted that such attacks are also possible in case of link state protocols. However, the attacks are much more lethal in case of distance vector because of the implicit trust relationship among the routers. This paper concentrates on router attacks in distance vector protocols.

Related Work: The solution proposed for detecting distance vector attacks can be broadly classified into three categories.

- Routing Information Techniques: In this type of techniques (Zang, 1998) digital signatures are used to detect malicious distance vector updates in case of link attacks. However, these schemes are unable to detect router attacks.
- Intrusion Detection Techniques: These techniques (Kirk, 1998) are used to detect the anomalous behavior in the routers, assuming that intrusion detection devices are available in the network.
- Routing Protocol Techniques: In this type of techniques, detection capability is built into the routing protocol itself. In Cisco White Papers (2000), several techniques have been mentioned to detect bad /malicious routers. Though the techniques are able to prevent looping, malicious distance vector updates

cannot be detected using these techniques. One of the methods of validating the integrity of the distance vector update, in presence of router attacks, is by using a technique called the "Consistency Check" (CC) (Bradely, 1997). In this technique, each router, in addition to the hop length information, also sends the predecessor information to its neighbors. This paper adopts the principle of routing protocol techniques.

Routing Table Attacks: The security of the networking infrastructure becomes a major concern as the dependence on it grows rapidly. So, there is an increased attempt to compromise the infrastructure. One of the most critical infrastructure security issues involves securing routing infrastructure (Dijiang Huang, 2005). In particular, the routing operation is a highly visible target that must be protected from a variety of attacks. Routing table threat is a challenging problem, because security was not introduced into the routing protocols. Routing tables forms an important element of a router and any malicious modifications to the routing table would cause great damages to the entire network. Moreover, the injection of false routing information can easily degrade network performance, or even cause denial of service for a large number of hosts and networks over a long period of time.

In today's Internet, routing protocols (e.g., RIP v2, EIGRP, BGP and OSPF) form the heart of the network infrastructure. Until recently, the security of these protocols was not fully emphasized. However, there is a growing awareness of the potential consequences of attacks aimed at the infrastructure; particularly at the routing protocols (Vetter, 1997). These protocols contain few security mechanisms for safeguarding the network infrastructure. Those that exist are often incomplete.

Distance Vector Routing Protocols: Distance vector routing protocol is one of the important routing protocols. In this protocol, distance can be hops or a combination of metrics calculated to represent a distance value. The IP distance vector routing protocols still in use today are: RIPv1, RIPv2 and IGRP. The key characteristics of the various distance vector routing protocols mentioned above are given in **Table 1.**

Table.1. Key characteristics of the various distance vector routing protocols

Characteristic	Routing Protocol		
	RIP v1	**RIP v2**	**IGRP**
Route updates	Broadcast	Multicast	Broadcast
Update timer	30 seconds	30 seconds	90 seconds
Hold-down timer	180 seconds	180 seconds	280 seconds
Metric	Hop length	Hop length	Bandwidth + delay
Load balancing	Equal-cost multipath routing	Equal-cost multipath routing	Equal-cost and unequal-cost multipath routing

The principle behind the protocol is very simple. Each router in an internetwork maintains the distance from itself to every known destination in a distance vector table (i.e. routing table). Distance vector tables consist of a series of destinations (vectors) and costs (distances) to reach them and define the lowest costs to destination at the time of transmission. The distances in the tables are computed from information provided by neighbor routers. Each router transmits its own distance vector table across the shared network.

The chief advantage of the distance vector is that it is very easy to implement. There are also the following significant disadvantages:

- The instability caused by old routes persisting in an internetwork
- The long convergence time on large networks
- The limit to the size of an internetwork imposed by maximum hop lengths
- Distance vector tables are always transmitted even if their contents have not changed.

2. METHODS OF PROPOSED SYSTEM

Methodology adopted in our proposed work: An enhanced approach is designed to prevent the vulnerabilities in the distance vector routing protocols. Any method, which is used to validate the router data in distance vector routing protocol should provide authenticity and integrity to the routing updates and maximize the detection of malicious updates. There also should be some mechanism incorporated in the routers to check the validity of the routing updates that is sent by the neighboring router thereby detecting the malicious routers.

Assumptions in our method: A number of assumptions are made in enhancing the security mechanisms for distance vector routing protocols. As any other secure routing protocol, our method requires the existence of key establishment mechanisms such as pair-wise shared key or a public key infrastructure (Housley, 2002). We consider a symmetric key system, since encryption/decryption operation in a public key system takes several orders of magnitude longer than that in a symmetric key system.

The assumptions in the method are:

- One router in the network is designated as a trusted router. This router cannot be compromised. The purpose of adding the trusted router prevents internal attackers from propagating false updates.

- The trusted router shares a different secret key with every other router in the network. This secret key is used to authenticate whether the routing update that is sent by a router belongs to the network.
- The trusted router possesses the full topology of the network and maintains all possible shortest routes from any point in the network to any other point in the network.
- To accomplish the above requirements, trusted router is provided with high memory capacity.
- Every router of the network shares a different secret key with every neighbor router. This secret key is used to authenticate routing updates exchanged between each pair of neighbor routers.

Methodology adopted: It is assumed that, there are two possible attackers in the network. One is referred as external attacker and the other as internal attacker. The proposed method uses an enhanced approach to secure distance vector routing protocols which provides protections from both internal and external attackers. The main idea is to designate a trusted router that involves in checking the validity of the routing update messages.

The process of validating the routing update and updating the routing table of all neighbors can be affected in three methods. In what follows, we describe and present a brief analysis of the overhead incurred in all the three methods. The following notations are used in the analysis.

R_i - Transmitting router
R_j - Any neighbor of R_i
R_T - Trusted router
s - Packet size
h' - Average hop length
nd_i - Number of neighbors of R_i – (Node degree)
nd' - Average node degree of the network
f - Validation flag
TTO - Total Transmission Overhead

Description of the validation flag: Validation flag is sent by the trusted router indicating whether the routing update is valid or invalid. The flag consists of three fields as shown in **Table.1**.Source address is the address of the trusted router. Destination address is the address of the router that receives the flag. Signal is a 1 byte field that contains the information about the validation of the routing update.

Table.2. Fields in the Flag

Source address	Destination address	Signal
6 bytes	6 bytes	1 byte

Method I – R_i to all R_j

1. Router R_i transmits the routing update to all R_j.
2. All R_j send the update to R_T.
3. R_T in turn sends the flag to all R_j after validating the update.

The scenario is shown in **Fig. 1**

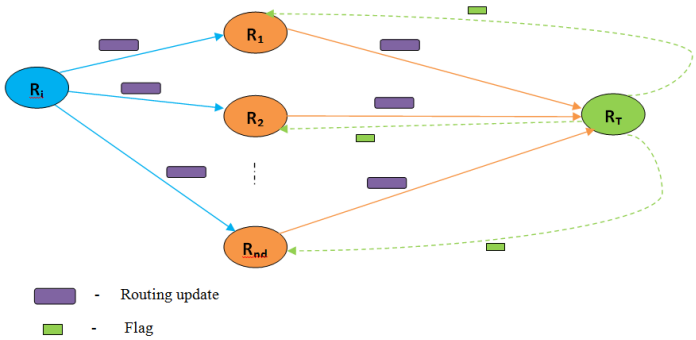

Figure.1. R_i to all R_j

Analysis: The overhead in transmitting the update from R_i to all R_j is = nd'× s.

The overhead in sending the update from all R_j to R_T is = nd'× h'× s.

The overhead in sending the flag from R_T to all R_j is = nd'× f.

Hence the total transmission overhead (TTO) is

$$TTO = nd' \times s + nd' \times h' \times s + nd' \times f$$
$$= nd' \times ((h'+1) \times s + f)$$

Because f << s, and s is constant, in big-O notation, we have

$$TTO = O(nd' \times h')$$

Method II – R_i to R_T

- Router R_i transmits the routing update to R_T.

- R_T validates the update.
- R_T transmits the new update to all R_j.

The scenario is shown in Fig. 2.

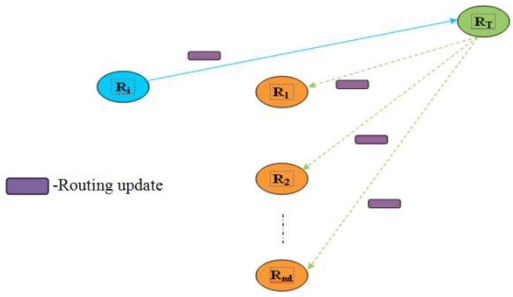

Figure.2. R_i to R_T

Analysis: The overhead in transmitting the update from R_i to R_T is = h' × s.
The overhead in sending the update from R_T to all R_j is = nd'× h'× s.
Hence the total transmission overhead (TTO) is

$$TTO = h' \times s + nd' \times h' \times s$$
$$= h' \times s \times (nd'+1)$$

Because s is constant, in big-O notation, we have

$$TTO = O(nd' \times h')$$

Method III – R_i to all

- Router R_i transmits the update to all R_j and R_T.
- R_T validates the update.
- R_T sends the flag to all R_j.

The scenario is shown in Fig. 3.

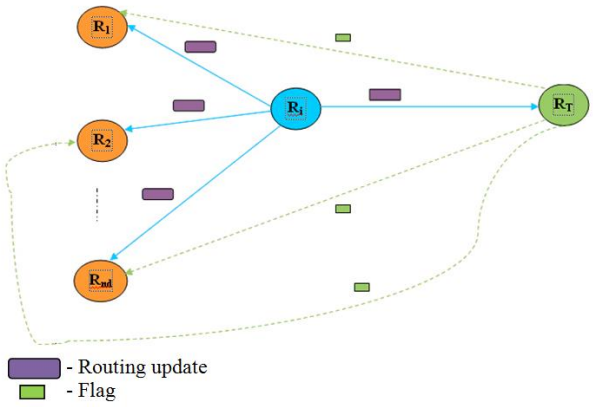

Figure.3. R_i to all R_j and R_T

Analysis: The overhead in transmitting the update from R_i to all R_j is = nd'× s.
The overhead in sending the update from R_i to R_T is = h'× s.
The overhead in sending the flag from R_T to all R_j is = nd'× f.
Hence the total transmission overhead (TTO) is

$$TTO = nd' \times s + h' \times s + nd' \times f$$
$$= s \times ((h' + nd') + nd' \times f)$$

Because f << s, and s is constant, in big-O notation, we have

$$TTO = O(h' + nd')$$

Comparing the complexity of all the above three methods, it is inferred that the method III is the most economical and it is followed in our simulation.

When a router R_i wants to send a routing update to its neighbors, it sends the routing update to all its neighbors and also to the trusted router. The neighbor routers waits for the validate signal from the trusted router. The trusted router checks the validity of the routing update by our proposed algorithm and sends a flag. The flag contains a positive signal if the update is valid. The positive signal is represented by a group of bits 00000000. The neighbor routers of R_i updates the routing table after receiving the positive signal.

The trusted router, on finding the routing update to be malicious raises an alarm to the administrator. It provides the information about the malicious update and the router that sends the update. It also inserts a negative

signal in the flag and sends the flag to all the neighbors of R_i. The negative signal is represented by a group of bits 00000001.

In the proposed work, some fields are added to the routing table which enables authentication of the neighbor routers, prevent the replay attacks and validate the integrity of the update. By adding those fields, it is possible to prevent both external attackers and internal attackers. The format of the routing message used in the proposed method is shown in Table. 3.Table.4 shows the fields in the routing table. The reason for adding the new fields in the routing message and in the routing updates and their significance are listed below.

Table.3. Routing message format

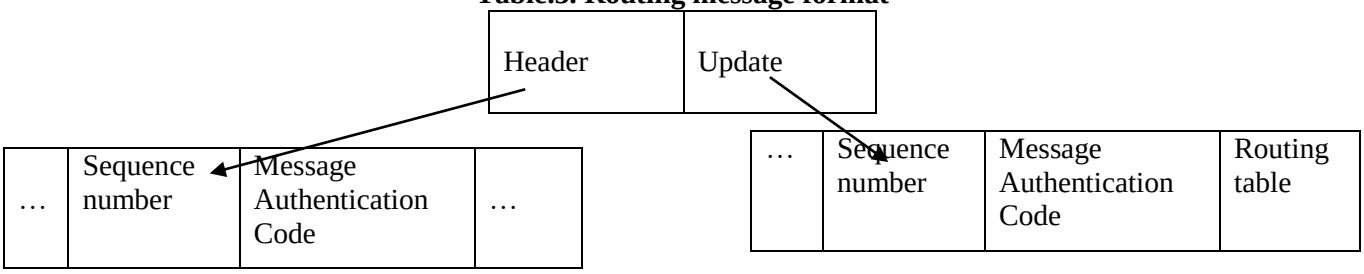

Table.4. Fields in the routing table

Destination id	Next hop	Hop length	Antecedent router	Path sum

A shared secret key authentication mechanism is used to authenticate routing messages and updates exchanged between neighbor routers. In this mechanism, when a router R_i sends routing message or an update to its neighbor R_j, it adds:

- A sequence number to protect against replay of previous routing messages or routing updates.
- Message authentication code (MAC) which is the result of a hash function to authenticate the routing messages and the updates.

When R_j receives the routing message from its neighbor R_i, it examines the type of the message. If the message is a normal packet, it is transmitted to the destination. If the message is a routing update, it verifies the MAC of the received message. It also checks the sequence number. If all these checks are valid, this routing update is accepted and R_j waits for a validate signal from the trusted router. If authentication fails, the update is rejected. With this mechanism, it is more difficult for an unauthorized router to impersonate a legitimate router. If the message is a validate signal (positive) from the trusted router, R_j updates its routing table with the received update.

The update that is sent to the trusted router for validity includes MAC and sequence number of the update. The purpose of including the sequence number and MAC is to authenticate the update that is exchanged between the router and the trusted router. The above authentication mechanism helps the trusted router to authenticate the update messages that are sent by the routers in the network. Update part also includes another entry called routing table which is very important for a neighboring router. The neighbor router uses this table to update its own routing table.

To summarize, every router maintains the following information for each neighbor router:

- A shared secret key with its neighbor
- A sequence number of the last routing update received from the neighbor. This provides the freshness of routing update exchanged between neighbors.
- A routing table which contains an entry for each destination. This route entry contains the following fields and the significance of these fields is shown in **Table.5.**

Table.5. Significance of fields in the routing table

Field	**Signification**
Destination id	Identifier of a destination
Next hop	The next hop of this destination
Hop length	The number of hops to reach this destination
Antecedent router	The previous node of this destination
Path sum	Sum of all hop lengths of all paths that pass through or terminates in this destination

The significance of the fields in the routing table and the definitions of those fields are as follows:

Antecedent (a) of a node x in the distance vector tree is y, if x is a descendant of y. Therefore,

$$a(x) = y$$

Descendants (D) of node x in distance vector tree are defined as the children of x.

Hop length (hl) of a node x is defined as the number of hops in the shortest path from the root node to x in the distance vector tree.

Path sum (ps) of node x in the distance vector tree is defined as the sum of all path lengths passing through and terminating in x. Therefore,

$$ps(x) = hl(x) + \sum_{\forall j \in D_x} ps(j)$$

Routing update validation: In the proposed algorithm, the transmitting node is the router that transmits the distance vector update and the receiving node is the router that receives the distance vector update. Trusted router is the router that validates the routing update. The following notations are used in this algorithm to denote the transmitting router, the receiving router and the trusted router.

R_i - Transmitting router
R_j - Any neighbor of R_i
R_T - Trusted router
Upid - Routing update id
Transmit (pkt id) - A routine which transmits the present packet
Signal (Upid) - Positive signal from R_T to all R_j with the Upid
Update (Upid) - A routine which updates the routing table
Delay ( ) - Delay routine that waits for a period of time

The various steps involved in the proposed method are:

Transmitting router (R_i):

- Distance vector tree based on shortest path is computed.
- Antecedent information and the path sum for each destination based on the distance vector tree are calculated.

Antecedent and path sum for each destination is sent to all its neighbors.

Receiving router (R_j):

- Read the reception buffer
- If received message = nil,

go to delay ( )
Else if received message = packet,
Transmit (pkt id)
Else
If received message = routing update,
go to step 3
Else
If Received message = signal (Upid)
Update (Upid)
Else
go to step 1
Routing update is authenticated by validating the MAC of the message, thereby preventing router impersonation.
If authentication fails,
Reject the routing update.
Else
Check if the advertised routes need to be updated in its routing table.
If Yes,
The routing update is sent to the trusted router for checking the validity of the update.
Else
go to delay ( )
Trusted Router (R_T)
The trusted router checks the validity of the updates by performing the following steps in sequence:

- Tree is constructed based on the antecedent information present in the update. In tree construction step, for each node in the distance vector update, the node is added to the list of descendant nodes of the antecedent node.
- Hop length and path sum are updated based on the constructed tree.
- The calculated values and the received values are compared for checking the validity of the update.
- The net path sum (nps) of a node i is calculated by the formula

  $nps_i = ps_i - ps_i^r$, where
  ps_i is the calculated path sum and
  ps_i^r is the received path sum
- Trusted router identifies the update as malicious if nps (i) $\neq 0$ for some i = 1, 2,…, n.
- If the difference is non-zero,

(i) The routing update is detected as malicious. R_T sends a negative signal to all the routers in the network, indicating that the routing updates that are numbered by the corresponding sequence numbers are malicious updates.
(ii) It also sends an alarm to the network administrator, indicating that the router with the corresponding IP address is malicious.
Else
R_T sends signal (Upid) to the router indicating that it can update its routing table with the update.

The algorithm uses the authentication and validation mechanism to find the faulty updates in the routing table that is sent by a router. The validity of the update is checked with the help of a trusted router. This validity check prevents the routing updates from modification by internal attackers. The authentication scheme and the inclusion of the sequence number, that is used in our method prevents external attackers from replaying previous routing updates, injecting erroneous routing messages and masquerading as a legitimate router. This method also detects malicious routers and this is informed to the network administrator.

Specifications Used In Our Method: The specifications used in the method are given **Table 6.** The protocol taken for this method is distance vector routing protocol. The method can be applied for RIPv1 and RIP v2. It cannot be implemented in IGRP because IGRP uses combined metric of bandwidth and delay. This method is suitable for a protocol that uses the hop length as a routing metric. With slight changes, the same method can be used to detect malicious updates in IGRP.

Table.6.Specifications used in our method

Parameters	**Specifications**
Protocol used	Distance vector routing protocol
Performance measures	Detection probability, packet delivery ratio, routing overhead

The performance measures used to evaluate our method are:

Detection probability: It is the probability that a malicious update can be detected.

Packet delivery ratio: Packet delivery ratio is the ratio of number of packets that are received by the destination to the number of packets submitted to the network.

Routing overhead: It is the ratio of total number of routing related transmissions to the total number of packet transmissions.

The proposed method measures detection probability with respect to node degree of the malicious node and the number of entry pairs changed. It also measures the packet delivery ratio and the routing overhead with respect to the number of the malicious nodes.

3. SIMULATION RESULTS AND DISCUSSION

The system that is being simulated can lead to a better understanding of the system and also suggests strategies that improve the operation and efficiency of the system. It is also easy to detect the important variables that interact in the system and to understand the interrelation between them. The method adopts the class of routing protocol techniques. In this type of techniques, detection capability is built into the routing protocol itself. Several techniques have been developed to detect malicious routers. However, though the techniques are able to prevent looping, malicious distance vector updates cannot be detected using these techniques. One method of validating the integrity of the distance vector update, in the presence of router attacks, is by using a technique called the "Consistency Check" (CC).

In the CC technique, whenever a node receives the distance vector form its neighbor, it carries out the consistency check by tracing the path from each destination to the source. The check fails when an intermediate node sends a wrong update keeping the topology in mind. Therefore, in some cases, CC is unable to detect a wrong update. Moreover, the CC algorithm does not provide any guarantee in terms of its detection capability.

Another method of detecting the malicious update is the "Pivot Based Algorithm for Inconsistency Recovery" (PAIR).In PAIR (Anirban Chakrabarti, 2003), the average node degree is 4. Moreover, the detection probability falls significantly when the number of entry pairs changed is 4.

In this method, the detection probability is very high when compared to the CC method and the PAIR, even in the presence of malicious routers.

The performance of the proposed method is evaluated by carrying out an extensive simulation studies in NS 2.29. The network configuration used in the simulation is shown in Table 7.

This work implements a trusted router that checks the validity of the routing update by adding antecedent and path sum information instead of hop length information in the routing update. Our method is able to detect the malicious update, thereby identifying the malicious router. In this method, average node degree is 6.

Table 7. Network configuration used in simulation

Simulation parameters	Values	Simulation parameters	Values
Network topology	Random	Trusted router	1
Traffic	CBR	Link cost	1
Number of nodes	50	Packet size	512 bytes
Number of malicious routers	11	Simulation time	180 seconds

The detection probability of the three methods is shown in Fig.4. It is the probability that a malicious update can be detected. In Fig.4, the detection probability is varied with the node degree. In these experiments, 1 pair of entries (antecedent and hop length in case of CC and antecedent and path sum in case of PAIR and our method) are randomly changed in the distance vector update. It is shown that 98% of updates are detected in our method. In PAIR, 98% updates are detected, but as the node degree increases, the probability of detecting the malicious updates decreases. The detection probability of CC also reduces as the degree of the node increases. This is because, when the nodes form the leaves of the distance vector tree, even a change in one pair of entries is sufficient to make the update undetectable by CC. This method, however, does not suffer from this deficiency and thus able to detect nearly 98% of the malicious updates.

In Fig. 5, the detection probability is varied with the number of pairs of changed entries in the distance vector update. The entries are selected to minimize the detection probability. A change in antecedent of a node X in case of our method, results in change of path sum of all nodes from X to root of the distance vector tree.

On the other hand, a change in antecedent of a node Y in case of CC, results in change of path sum of all nodes which are descendants of Y. Inspecting the results, the proposed method is able to detect the bad router at a very higher rate, when the number of entries changed is low. The detection probability is above 90%, even when the number of entries changed is 6 or less than 6. After 6, the detection probability drops significantly and becomes less than 30%. In PAIR, the detection probability is above 90%, when the number of entries changed is 4. After 4, the detection probability drops significantly. In the CC method, as the number of entry pairs changed is increasing, the detection probability is low when compared with our proposed method. It is to be noted that the detection probability is zero, for traditional distance vector protocols.

Fig. 5 shows the packet delivery ratio. Packet delivery ratio is the ratio of number of packets that are received by the destination to the number of packets submitted to the network.

$$\text{Packet delivery ratio} = \frac{\text{Number of packets received}}{\text{Total number of packets submitted}}$$

It is inferred from Fig. 6, that packet delivery rate is more than 85 %, even when there are a higher number of malicious nodes in the proposed method. In PAIR method, the ratio is same as that of our method, when there is one malicious node. As the number of malicious node increases, the delivery ratio of the PAIR method decreases when compared with our method. In CC method, the delivery ratio is above 80% and drops as the number of malicious nodes increases.

Fig. 7 shows the routing overhead incurred in the three methods. Routing overhead is the ratio of total number of routing related transmissions to the total number of packet transmissions.

$$Routing\ overhead = \frac{\text{Total number of route transmissions}}{\text{Total number of packet transmissions}}$$

As shown in Fig. 7, the routing overhead is increased significantly when the network topology changes or there is a large number of malicious nodes in the network. In the output, it is clear that all the three methods provide some overhead. In the proposed method, the routing update contains an extra field. Though it consumes fewer bytes, in the terms of message overhead, this proposed method experiences a slight increased routing overhead when compared to the PAIR and CC method.

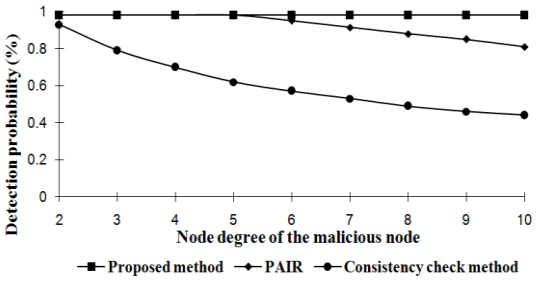

Figure.4. Detection probability versus node degree

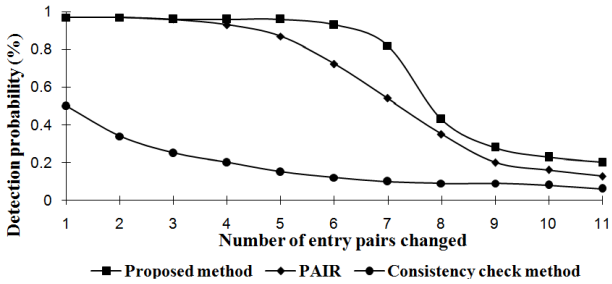

Figure.5. Detection probability versus number of entry pairs changed

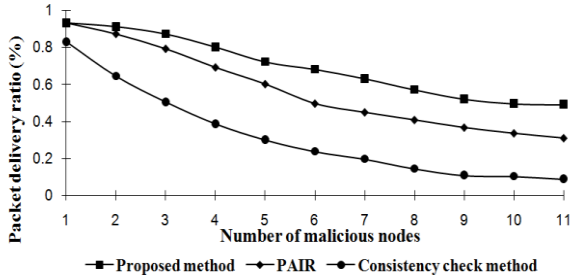

Figure.6. Packet delivery ratio

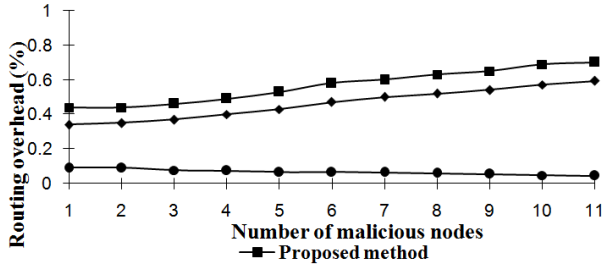

Figure.7. Routing overhead

The proposed method performs well when compared with the CC method and the PAIR method. It provides very high detection probability when compared with the CC method, though it has a little routing overhead. The method is also able to detect the malicious router which is not possible in other methods.

4. CONCLUSION

An efficient method is presented to detect the external as well as the internal attackers. The method presented measures, similar to existing proposals that protect routing transmissions across the networks from the masquerading router, unauthorized router, and subverted link classes of intruders. In addition, we proposed a new class of protection mechanism that involves sending the ancestor and path sum information along with the distance vector updates that protects the routing updates. It also introduces the concept of designating a trusted router that could finds out the malicious updates, thereby identifying the routers that send these malicious updates. It is believed that the assumption of designating a trusted router by the administrator is reasonable in wired corporate network. The case of trusted router failure can be detected by the underlying routing mechanisms.

The authentication mechanism used by the routers in this method is capable of preventing the routers from masquerading as a legitimate router. MAC is used to authenticate the routing messages which provide data origin authentication.

Moreover, it is shown from the simulation results that the proposed method achieves the following: (i) It requires fewer bytes of extra information per node in the distance vector packet. (ii) It is always able to detect malicious updates under certain well-defined conditions. (iii) Detection probability of our method is significantly higher than that of the existing methods.
(iv) It can identify the malicious routers. (v) It has low running time.

As a future work, fuzzy logic and genetic algorithms can be implemented to produce optimal results. It is aimed to expand the approach in the case of DSDV and AODV used in ad hoc networks, where the trust worthiness is a very serious issue in the absence of third trusted parties. It is planned to add more trusted routers without affecting any resources of the network in the case of increased population of the routers. Moreover, the routers are not able to recover from malicious updates. This may be considered in the future work. Future work also includes reduction of the routing overhead of this method.

REFERENCES

Anirban Chakrabarti and Manimaran G, An efficient algorithm for malicious update detection and recovery in distance vector protocols, Proc. of IEEE ICC, 2003, 1952-1956.

Anirban Chakrabarti and Manimaran G, Internet Infrastructure Security: Taxonomy, IEEE Network, Vol. 16(6), 2002, 13-21.

Bradely R, Smith, Shree Murthy, and Garcia-Luna-Aceves JJ, Securing Distance-Vector Protocols, in Proc. SNDSS, 1997.

Cisco White Papers, Strategies to Protect against Distributed Denial of Service Attacks (DDos), 2000.

Dijiang Huang, Qing Cao, Amit Sinha, Schniederjans M.J, Cory Beard, Lein Harn and Deep Medhi, Addressing intra domain network security issues through secure link-state routing protocol: A Network Architectural Framework, Commun. of ACM, 2005,1-14.

Housley R, Polk T, Ford W and Solo D, 'Internet x.509 public key infrastructure certificate and certificate revocation list CRL profile', RFC 3280, 2002.

Kirk A, Bradely S, Cheung B, Mukherjee and Ronald A, Olsson,Detecting Disruptive Routers A Distributed Network Monitoring Approach, in Proc. IEEE Symp. On Security and Privacy, 1998.

Panagiotis Papadimitratos and Zygmunt Haas J, Securing the Internet routing Infrastructure, IEEE Commun, 2002, 60-68.

Vetter B, Wang F and Wu S, An experimental study of insider attacks for the OSPF routing protocol', Proc. IEEE Int. Conf. on Network Protocols, 1997, 293-300.

Zang K, Efcient Protocols for Signing Routing Messages," in Proc. NDSS, 1998.