Mohana , Dr.N.K. Srinath / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue4, July-August 2012, pp.1213-1219

Performance Analysis Of Secure And Trust Based Routing Algorithms For Mobile Ad-Hoc Network.

Mohana [1], Dr.N.K. Srinath [2]

[1] M.Tech student, Dept. of CSE, R.V. college of Engineering, Bangalore-59, India
[2]Professor and Head of Department, Dept. of CSE, R.V college of Engineering, Bangalore-59, India

ABSTRACT

Mobile Ad-hoc Networks (MANETs) comprises set of nodes connected by wireless links. The network is ad hoc because it does not rely on a preexisting infrastructure, such as routers in wired networks. Routing in MANETs is a challenging task due to dynamic topology and error prone shared environment.

Data is sent between nodes in a MANET by hopping through intermediate nodes, which must make decisions about where and how to route the data. MANET faces several problems because of node mobility, network traffic, network size, and the possibility of node faults. The efficiency and behavior of a MANET depends on how well information can be passed around and delivered.

In today's world the security vulnerabilities are increasing day by day. It is really difficult to route the packet with minimum packet loss. In this paper a new routing protocol is presented which would route the packets in a highly efficient way by introducing the concept of friend list, unauthenticated list and question mark list. The algorithm will avoid the malicious node by studying the network in an intelligent way. The proposed algorithm is also compared with other multipath routing algorithms namely Disjoint Multipath Routing (DMR), Trust based multipath routing (TMR), Message Trust based multipath Routing (MTMR) and the performance analysis proves that the proposed method will have better performance with respect to number of hops, route discovery time, and packet loss.

The performance metric considered for proposed work are number of malicious nodes detected, number of hops, route discovery time, packet loss, The simulation results show that FACES protocol works much better and provides more security than the other multipath routing protocols.

Keywords - Classes of Traffic, Data Rating, Dynamic Source Routing, Friend Rating, Net Rating, Route Discovery Time, Trust Level, Time To Live Period.

1. Disjoint Multipath Routing (DMR)

Initially secure connection has been established between source node to destination node. The DMR algorithm will find out the multiple routes from source to destination using DSR algorithm [1] [5]. After finding multiple routes, all the routes are sorted based on the route discovery time. Then it will choose the best four routes which are having minimum time delay. In this method the message is split into parts. This protocol takes advantage of the shortest path between the source node to destination node. Then it routes the four encrypted parts through four different routes. In this method to decrypt the original message all the encrypted parts are required. The security of this method lies in the fact that enemy node needs all the encrypted parts to decrypt the original message.

2. Trust based Multipath Routing (TMR)

TMR provides a method of message security using trust based multipath routing. In this approach, less trusted nodes are given lesser number of self encrypted parts of a message, thereby making it difficult for malicious nodes to gain access to the minimum information required to break through the encryption strategy [2]. Using trust levels, it makes multipath routing flexible enough to be usable in networks with "vital" nodes and absence of necessary redundancy. In addition, using trust levels, it avoids the non trusted nodes in the routes that may use brute force attacks and may decrypt messages if enough parts of the message are available to them.

Secure connection has been established between source node to destination node. The TMR algorithm will find out the multiple routes from source to destination using DSR algorithm. After finding multiple routes, all the routes are sorted based on the trust level. Then it will choose the best route which is having maximum trust level. In this method the message is split into parts. Then it routes the encrypted parts through best single route. The following table gives the description about

the trust levels and the trust levels are varied between -1 to 4 [2] [3].

Sl No	Trust Value	Meaning	Description
1	-1	Distrust	Completely untrustworthy.
2	0	Ignorance	Cannot make trust-related judgment about entity.
3	1	Minimal	Lowest possible trust.
4	2	Average	Mean trustworthiness.
5	3	Good	More trustworthy than most entities.
6	4	Complete	Completely trust this entity.

Table1: Trust levels.

3. MTMR Routing Algorithm

MTMR uses a trust assignment and updating strategy which can be used to identify and isolate malicious nodes. The MTMR algorithm will find the routes by using the DSR algorithm. If the route has been fined for the first time then the MTMR algorithm will have the all routes obtained from DSR with trust level of zero. The MTMR algorithm will then choose a route with minimum time delay. Then if the route contains the malicious node then the trust level of the node is decremented otherwise the trust levels of all the nodes in the best route will be incremented [4].

Unlike TMR MTMR routing algorithm does not assign random trust levels instead the trust levels are assigned only to those nodes which behave properly and deliver the packets successfully.

4. Friend Based Ad-hoc Routing (FACES)

This protocol is used to find out the secure route from source to destination. It will route the packets in a highly efficient way by introducing the concept of friend list, unauthenticated list and question mark list. The algorithm will avoid the malicious node by studying the network in an intelligent way. If any malicious node will come in the best route, it will detect that node and it will put it in the question mark list.

Source node will pick the intermediate node from the friend list. If the Friend List is empty then the node will obtain the unauthenticated list and pick one of the node as intermediate node. The value of TTL is decremented each time intermediate node is picked. Once the TTL becomes zero we have to use min hop routing so that the packet can be delivered to the destination at the faster rate.

The new routing algorithm will make use of following parameters

Question Mark List (QML): The list of nodes which are deemed suspicious by a particular node. This list is stored for each and every node in its data structure.

Unauthenticated List (UL): The list of nodes of which no security information is present.

Friend List (FL): This is the list of nodes which convey trust. Like the question mark list, a friend list is also stored for each node in its data structure. Friends are rated on a scale of 0 to 10.

FREQ: Friend Sharing Request, this is a control packet which is used to initiate friend sharing. A node receiving this packet replies with the nodes in its friend list, unauthenticated list and the question mark list.

DR: Data Rating, this is the rating given to nodes after they transmit some amount of data for the source node.

FR: Friend Rating, this is the rating computed when nodes share their friend lists.

NR: Net Rating, this rating is computed as a weighted mean of DR and FR.

OR: Obtained Rating, rating received during the friend sharing stage.

4.1 SHARE FRIEND STAGE ALGORITHM

This is the stage in which a node will exchange the friend list with other node in the network

The following figure (1) gives brief information about the share friend stage for various cases between two nodes namely A and B.

Friend initiator NodeA	Friend Giver NodeB	Common Nodes	Uncommon Nodes	Nothing
FL=[ ]	FL=[ ]			FR=FR+1 DR=DR+1 NR=NR+1
FL=[ ]	FL !=[ ]	FR=FRnodeB+1 NR=NRnodeB+1 DR=DRnodeB+1	FR=FR+1 DR=DR+1 NR=NR+1	
FL!=[]	FL=[]	FR=FRnodeA+1 NR=NRnodeA+1 DR=DRnodeA+1	FR=FR+1 DR=DR+1 NR=NR+1	
FL!=[]	FL!=[]	FR=FRnodeA+ FRnodeB +1 NR=NRnodeA+ FRnodeB +1 DR=DRnodeA+ FRnodeB +1	FR=FR+1 DR=DR+1 NR=NR+1	

Figure1:Friend sharing stage

Friend sharing is a periodic process which is chiefly responsible for the security of the algorithm. To

accomplish friend sharing we use the control packet *FREQ* (Friend sharing request). The node receiving the *FREQ* replies with the nodes in its friend list, unauthenticated list and the question mark list. The rules for friend sharing are as follows:

1. Any node can ask for a friend sharing request.
2. After friend sharing, challenges are initiated for those nodes which were not in the friend list.
3. If a node is already in the friend list the node updates its friend list.

Let us consider that the node *A* shares a list with node *B*. Then the friend sharing process is carried out as follows:

STEP 1: As the network is initialized each node starts the friend sharing process, which leads up to a challenge to start with the formation of friend list.

STEP 2: If node *B* is in the list of node *A*, then if a particular friend of node *B* is not present in the list of node *A*, node *A* includes it in its list and initializes the Friend Rating as the Obtained Rating from *B* and the Data Rating to Zero. Net Rating is calculated on the basis of pre determined weights.

STEP 3: if the node B is in the list of node A and if a Particular friend of node *B* is present in the list of node *A* then the Friend Rating is calculated.

Data Rating: The data rating is updated by a node for its friend on the basis of amount of data it transfers for it. The DR of a friend node varies according to the number of data packets transferred through it. The net DR is calculated as a moving average of the last five data ratings. Equation (1) describes the moving average relation between a data rating *i and* the previous five data ratings:

$$DR(i) = \frac{DR(i-1) + DR(i-2) + DR(i-3) + DR(i-4) + DR(i-5)}{5} \quad \ldots\ldots (1)$$

Friend Rating: During the *Friend Sharing* stage a node *A* asks for the friend list of node *B* and incorporates the rating of friends in the following way:
1. If the node *A* and node *B* have a common friend *C*, then node *A* obtains the rating of node *C* from the node *B* as:

$$obtained\ rating = \frac{Net\ rating\ of\ B\ in\ list\ of A * Net\ rating\ of\ C\ in\ list\ of\ B}{10} \quad \ldots\ldots (2)$$

The idea behind equation (2) is *to* incorporate the trust that node *A* has on node *B* while obtaining the rating of node *C* from it.

2. After the Friend Sharing Process has finished, each node adds up the *OR* from various nodes and divides them with the sum of the rating of those nodes from which it obtained the *OR*. In other words, the *FR* is the weighted average of the Net Ratings obtained during the Friend Sharing stage, where the weights are basically the Net rating of the friend responding to a particular Friend Sharing request.

Net Rating: The idea behind calculating DR and FR is to have two opinions in front of each node. This is done because malicious nodes can identify some nodes for which they would work properly while for some they would drop packets. The DR acts as the soul opinion of the host node and FR acts as the opinion of its friend nodes. The Net Rating (NR) would be a weighted mean of the two ratings as given in equation (3):

$$NR = \frac{W_1 * DR + W_2 * FR}{W_1 + W_2} \quad \ldots\ldots\ldots\ldots (3)$$

Where *W1* and *W2* would be the weights assigned to DR and FR respectively. The values of *W1*and *W2* are network dependant and can be learnt with experience.

4.2 Form Un-Authenticated List

The nodes will find out the nodes in the transmission range or within the coverage area and then it maintains a list of nodes which are reachable by the nodes on its data structure.

4.3 Sequential challenges

When a source discovers that the data was not transmitted properly it initiates a sequential challenge. To explain it let us consider the path as, $S \rightarrow A \rightarrow B \rightarrow C \rightarrow D$ where S and D are the source and the destination nodes and A, B, C are the intermediate nodes. On discovering data transmission problem after waiting for a back off interval the following process takes place.

Step1.The source challenges node **A**. If it is not able to complete the challenge it removes it from the friend list and places it in the question mark list. It then tries to route the data through the next best path.

Step 2. If **A** successfully completes the challenge, it challenges its neighbor and reports the result backward To **A.** Node **A** then takes action similar to Step 1.

Step 3. Similarly, if the node **B** is also authenticated, it challenges its neighbor node **C** and reports backwards to node **A**.

Step 4. If every node is authenticated then node **D** gets to know the result of the sequential authentication, when node **C** tries to authenticate it, it responds to it and sends a backward message attaching the number of data packets received by it. Since everyone is authenticated no one lies, and attaches the corresponding packets truthfully. The source thus comes to know where the packet drops has taken place.

We emphasize here that during the sequential challenge process no node would be able to detect whether it is a sequential challenge or a regular friend sharing process till it is authenticated. And thus a malicious node would be caught in the process and eliminated from the list of trusted node.

4.4 FRIEND ROUTING PROTOCOL

The Friend routing protocol will perform the following steps

Step 1.The Source Node will first find the set of intermediate nodes by doing a lookup in its Friend List.

Step 2.If the friend list is empty then the source node will look into the unauthenticated List.

Step 3. If the unauthenticated list is empty the friend list has no other choice of picking the node from question mark list.

Step 4. The source node will check whether it contains the destination node in its list if yes then the faces protocol will transmit the data directly to the intermediate node.

Step 5.The intermediate node will then become the source node (picked up during either the steps 1 2 and 3).

Step 6.The process repeats until the Time to live period expires or destination node is reached.

Step 7. If the TTL =0 then the current node will always pick a node which is closer to destination so that the destination can be reached at a faster rate.

5. Simulation Analysis

The simulations have been performed using IDE, java eclipse Galileo. Database used My SQL and some of the software packages used are Jdk 1.6, Jre6 and Jar files of Structs framework. Data is exported using Toad software and performance graphs are plotted using MATLAB.

Source Node	Destination Node	Coverage Area
5	45	30

Table 2: Inputs to Routing Algorithms

1. DMR Algorithm Output

Output of Stage1 for DMR Algorithm

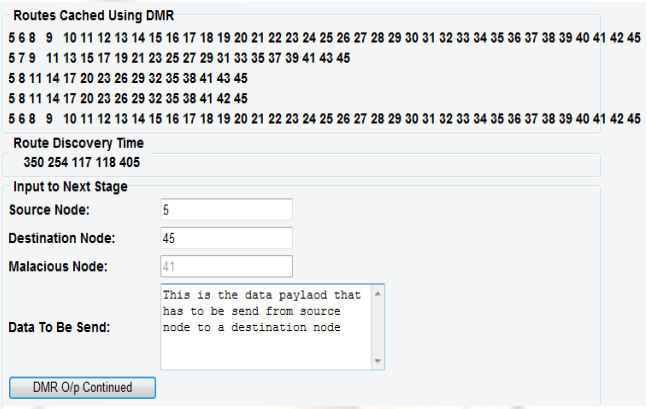


Figure 2: Multiple Routes Discovered using DSR, data payload to be send.

Figure 2 shows the multiple routes that have been discovered from source node to a destination node using DSR (Dynamic Source Routing) algorithm and there corresponding Route Discovery time as well. The user is also entering the data payload that has to be sent from source node to destination node.

Output of DMR Algorithm

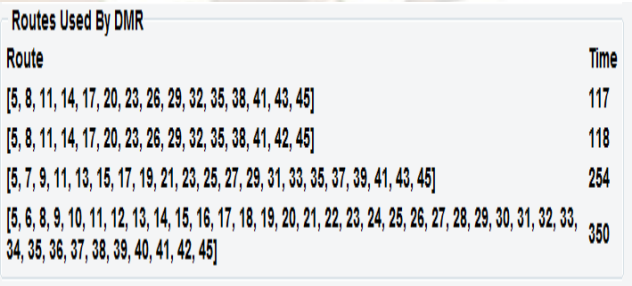
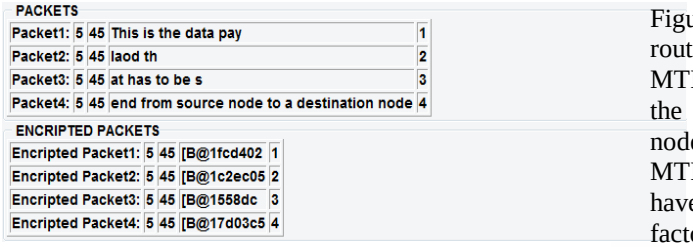


Figure3: Routes Chosen by DMR

Figure-3 shows the multiple routes that are discovered using DMR algorithm from source node to the destination node. DMR will select best four routes which is having a less route discovery time.



PACKETS

Packet1:	5	45	This is the data pay	1
Packet2:	5	45	laod th	2
Packet3:	5	45	at has to be s	3
Packet4:	5	45	end from source node to a destination node	4

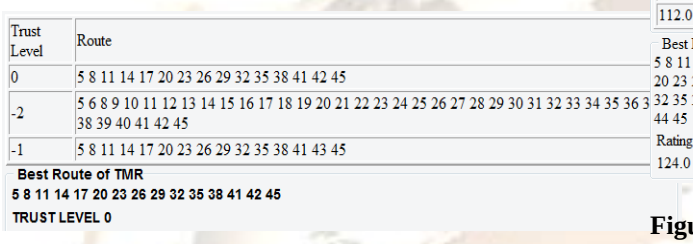
ENCRIPTED PACKETS

Encripted Packet1:	5	45	[B@1fcd402	1
Encripted Packet2:	5	45	[B@1c2ec05	2
Encripted Packet3:	5	45	[B@1558dc	3
Encripted Packet4:	5	45	[B@17d03c5	4

Figure4: Packet Formation Output

Figure4 shows the Packet Formed using the Triple Des Encryption for the data fragments. These data fragments would be sent over multiple independent routes from source node to destination node.

2. TMR Algorithm Output



Trust Level	Route
0	5 8 11 14 17 20 23 26 29 32 35 38 41 42 45
-2	5 6 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 38 39 40 41 42 45
-1	5 8 11 14 17 20 23 26 29 32 35 38 41 43 45

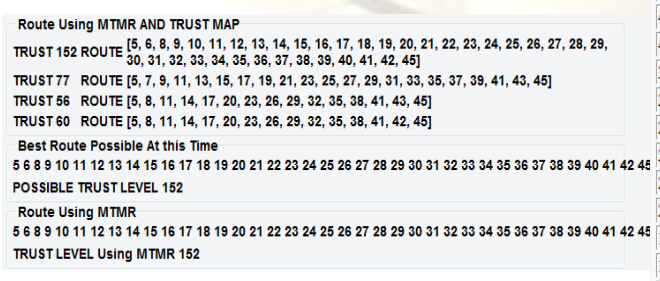
Best Route of TMR
5 8 11 14 17 20 23 26 29 32 35 38 41 42 45
TRUST LEVEL 0

Figure-5: TMR Algorithm Output

Figure-5 shows the TMR algorithm having multiple routes from source node to the destination node. The TMR algorithm will choose a route which is having the maximum Trust from Source Node to Destination node in the network. The encrypted data fragments will be sent in the single best route.

3. MTMR Algorithm Output

MTMR Routing Algorithm takes an additional parameter as input i.e threshold trust of a route.Threshold Trust=40



Route Using MTMR AND TRUST MAP
TRUST 152 ROUTE [5, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 45]
TRUST 77 ROUTE [5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45]
TRUST 56 ROUTE [5, 8, 11, 14, 17, 20, 23, 26, 29, 32, 35, 38, 41, 43, 45]
TRUST 60 ROUTE [5, 8, 11, 14, 17, 20, 23, 26, 29, 32, 35, 38, 41, 42, 45]

Best Route Possible At this Time
5 6 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 45
POSSIBLE TRUST LEVEL 152

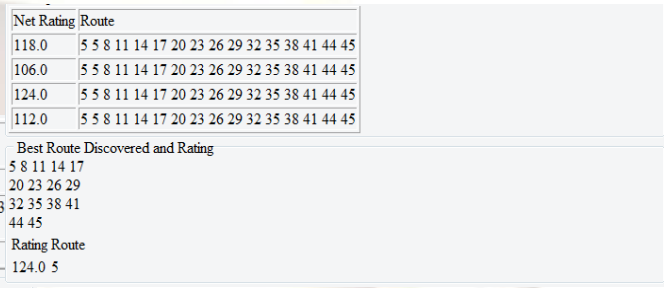
Route Using MTMR
5 6 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 45
TRUST LEVEL Using MTMR 152

Figure-6: M TMR Algorithm Output

Figure-6 shows the MTMR algorithm having multiple routes from source node to the destination node. The MTMR algorithm will choose a route which is having the maximum Trust from Source Node to Destination node in the network. The additional thing happening in MTMR is the nodes which are in the best route will have their corresponding trust levels incremented by a factor of 1.

4. Friend based Ad-hoc Routing output

The Friend will also take TTL has an additional input parameter as compared to other algorithms.



Net Rating	Route
118.0	5 5 8 11 14 17 20 23 26 29 32 35 38 41 44 45
106.0	5 5 8 11 14 17 20 23 26 29 32 35 38 41 44 45
124.0	5 5 8 11 14 17 20 23 26 29 32 35 38 41 44 45
112.0	5 5 8 11 14 17 20 23 26 29 32 35 38 41 44 45

Best Route Discovered and Rating
5 8 11 14 17
20 23 26 29
32 35 38 41
44 45
Rating Route
124.0 5

Figure-7: Friend Routing Algorithm Output

Figure-7 shows the Friend Based Routing Algorithm output .As seen from the figure the Friend Routing Protocol has discovered all the routes by picking based on the combination of friend rating, data rating and net rating. The Friend Routing has chosen the route which is having the maximum rating as the best route.

Share Friend Stage Output

NodeId	Friend Rating From Giver	Friend Rating From Initiator	Data Rating From Giver	Data Rating From Initiator	Net Rating From Giver	Net Rating From Initiator
6	1	1	1	1	1	1
9	1	1	1	1	1	1
12	3	3	3	3	6	6
32	4	0	3	0	6	0
35	4	4	3	3	6	6
38	4	0	3	0	6	0
45	4	0	3	0	6	0
41	4	0	3	0	6	0
44	4	0	3	0	6	0
30	10	10	11	11	65	65
25	10	18	11	19	65	101
26	10	10	15	11	108	65
29	10	10	15	11	108	65
23	10	10	27	23	270	211
20	10	10	27	23	270	211
17	10	10	27	23	270	211
14	10	10	27	23	270	211
11	10	10	29	25	302	240
8	10	10	29	25	302	240
5	10	10	41	33	520	369

Figure-8: Friend Sharing Stage Output

Figure 8 shows the output of Friend Sharing Stage As seen in the figure the friend list of Friend Stage Initiator is shown where the Node Id are friend node ids , Friend Rating from giver is rating allocated from node 5. Friend Rating from initiator is as per node 4. Similarly Data Rating and Net Rating are shared between two nodes Node 4 and Node5.

6. Performance Analysis of Algorithms

1. No of Hops

Figure 9 shows the number of hops taken from source to destination for all four algorithms DMR, TMR, MTMR and FACES. DMR will take more number of hops.TMR and MTMR will take almost equal number of hops. FACES will take less number of hops. From the figure conclude that FACES algorithm is the best w. r. t number of hops.

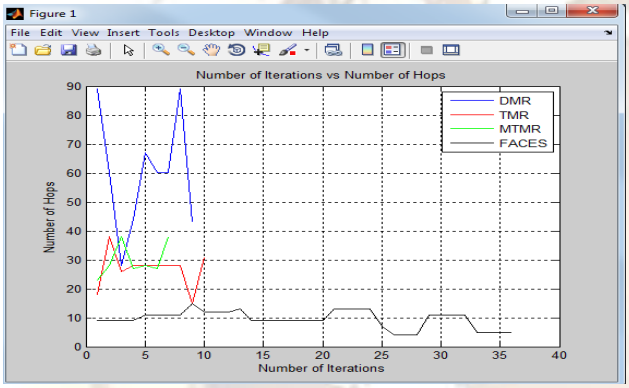


Figure-9: Number of hops

2. Route Discovery Time

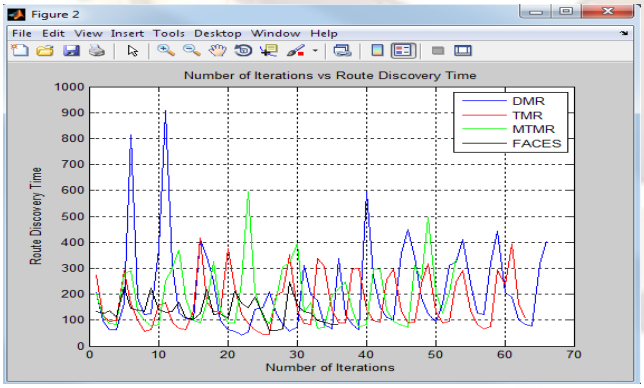


Figure-10: Route Discovery Time

Figure-10 shows the route discovery time taken for all the routes from source to destination for all four algorithms. DMR and MTMR are taking more time to establish path between source node to destination node. TMR is taking medium route discovery time. From the figure conclude that the route discovery time taken by FACES algorithm is less as compared to DMR, TMR and MTMR

3. Packet Loss

Figure 11 shows the packet loss taken for all the routes from source to destination for all four algorithms. DMR is having a more number of packet losses. Packet drop is minimal in FACES, as it will detect more malicious nodes and efficiently discards routes containing malicious nodes. But other multipath routing protocols drop a larger number of packets as they route through a greater number of nodes and thus increasing the chances of routing data through malicious nodes. From the figure conclude that FACES algorithm is the best when compared to DMR, TMR, and MTMR

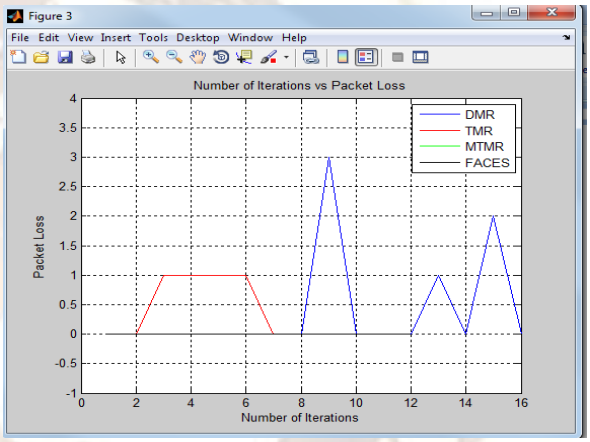


Figure-11: Packet Loss

7. Conclusion and Future work

Mobile Ad-hoc network (MANETs) due to its dynamic nature has many challenges. Some of the major challenges are number of malicious nodes detected, number of hops, route discovery time and packet loss.

Many Routing algorithms namely DMR, TMR, MTMR and FACES have their own way in order to establish the trust and transmit packet securely. But Friend based protocol proved to be best in terms of number of malicious nodes detected, number of hops, route discovery time and packet loss.

In future we plan to implement the existing secure routing protocols such as the ARIADNE and ARAN and compare them with the FACES protocol. The system handles only text as message for data packets does not handle multimedia data packets. In

future it can be enhanced to include multimedia packets.

References

[1] David B. Johnson David A. Maltz Josh Broch "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks" *Computer Science Department, Carnegie Mellon University*.

[2] Alfarez Abdul-Rahman & Stephen Hailes "A Distributed Trust Model" *Department of Computer Science, University College London, Gower Street, London WC1E 6BT, United Kingdom.*

[3] A.A. PIRZADA and C. McDONALD "Trust Establishment in Pure Ad-hoc Networks" *School of Computer Science & Software Engineering, The University of Western Australia, 35 Stirling Highway, Crawley, W.A. 6009, Australia,* Wireless Personal Communications (2006) *37: 139–163, DOI: 10.1007/s11277-006-1574-5 C _ Springer 2006*.

[4] Su Bing, Ma Zheng-Hua,Sun Yu-Qiang"A Trusted-Based Encryption Mechanism for Efficient Communication Over Wireless Network" Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08 4th International Conference, *Digital Object Identifier: 10.1109/WiCom.2008.1111*

[5] Al-Mekhlafi Z.G, Hassan R, "Evaluation study on routing information protocol and dynamic source routing in Ad-Hoc network", *Information Technology in Asia (CITA 11), 2011 7th International Conference, E-ISBN : 978-1-61284-130-4, INSPEC Accession Number: 12208218 Digital Object Identifier :10.1109/CITA.2011.5999535 Issue Date : 30 August 2011, pp 1-4.*

[6] L.Wang and N.-T. Zhang, "Locally forwarding management in ad-hoc networks," in *Proc. IEEE Int. Conf. Communications, Circuits and Systems and West Sino Expositions, Jun./Jul. 2002, pp. 160–164*.

[7] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Book Chapter inMobile Computing*, T. Imielinski andH. Korth, Eds. Dordrecht, The Netherlands: Kluwer, 1996, pp. 131–181.

[8] A. Wood and J. A. Stankovic, "A taxonomy for denial-of-service attacks in wireless sensor networks," in *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*. Boca Raton, FL: CRC, 2005, pp. 32:1–32:20.

[9] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Trans. Inform. Theory*, vol. IT-22, no. 6, pp. 644–654, 1976.

[10] M. S. Obaidat and N. Boudriga, *Security of e-Systems and Computer Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2007.

[11] K. Sanzgiri, B. N. Levine, C. Shields, B. Dahill, and E. M. Belding- Royer, "A secure routing protocol for ad hoc networks," in *Proc. 10th IEEE Int. Conf. Network Protocols (ICNP)*, Paris, France, Nov. 12–15, 2002, pp. 78–89.

[12] Y. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A secure on-demand routing protocol for ad hoc networks,"*Wireless Netw.*, vol. 11, no. 1–2, pp. 21–38, Jan. 2005.

[13] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proc. MobiCom 2000*, Boston, MA, Aug. 2000, pp. 255–265.