



GLOBAL JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY: E
NETWORK, WEB & SECURITY

Volume 17 Issue 4 Version 1.0 Year 2017

Type: Double Blind Peer Reviewed International Research Journal

Publisher: Global Journals Inc. (USA)

Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Survey: Trust-based Approaches to Solve Routing Issues in MANET

By Pawan Makhija & Neha Sharma

Shri G.S. Institute of Technology and Science

Abstract- A mobile ad hoc network is a wireless network. The ability to work without any central controlling authority without any requirement of established infrastructure makes it need of the present scenario. This dynamicity comes with a downside of security. Since the nodes may act maliciously and pose threat to the working condition of the MANET. Trust approaches are well suited in these situations. Here we discuss TRUST; the meaning, characteristics and different schemes.

Keywords: MANET, trust, mobile node, black hole attack.

GJCST-E Classification: E.3



Strictly as per the compliance and regulations of:



Survey: Trust-based Approaches to Solve Routing Issues in MANET

Pawan Makhija ^α & Neha Sharma ^σ

Abstract- A mobile ad hoc network is a wireless network. The ability to work without any central controlling authority without any requirement of established infrastructure makes it need of the present scenario. This dynamicity comes with a downside of security. Since the nodes may act maliciously and pose threat to the working condition of the MANET. Trust approaches are well suited in these situations. Here we discuss TRUST; the meaning, characteristics and different schemes.

Keywords: MANET, trust, mobile node, black hole attack.

I. INTRODUCTION

Since the invention of computer networks there has been a constant demand for three things: increased connectivity, increased communication speed, and increased storage capacity. It is clear that if a user cannot connect to the infrastructure the speed and the storage demands become irrelevant. Recently, wireless devices (hereafter referred as nodes) have become the preferred mean to access the networked infrastructure. In this paper, we focus on trust based approaches designed for ad-hoc networks, which, by definition are networks that do not have a previously deployed infrastructure. Here nodes collaborate to forward packets from a source to a destination [2]. However, since nodes are resource constrained they may behave selfishly. If the number of selfish nodes grows considerably the performance of the network may degrade to the point where nodes can only communicate with nodes located within its transmission range. A large numbers of trust-based approaches have been proposed to deal with nodes' malicious behaviours [5, 6, 7, 8, 9, 10, 11, 12, 13].

II. TRUST CHARACTERISTICS

Trust is useful in environments where the participants need to depend on each other to achieve a goal. In ad-hoc networks nodes need to rely on routers to forward the packets from the transmitting to the receiving nodes. However with current technology restrictions (memory, power, etc.) one or more routing devices may choose to behave selfishly. To achieve a reliable cooperation trust must be established.

Author α: Assistant Professor, Department of Information technology, Shri G.S. Institute of technology and science, Indore(M.P).
e-mail: pawanmakhijaacro@gmail.com

Author σ: Assistant Professor, Department of Information technology, Shri G.S. Institute of technology and science, Indore(M.P).
e-mail: ne3haa@gmail.com

Trust is context sensitive: Trust should not be calculated over a node as an entity, but rather as a set of actions that a node can perform. e.g. Instead of expressing trust of a node say 'A' as "node A is trusted", in an ad-hoc networks use "node A trusts node B to forward a packet to the required destination, but Node B may or may not be making copies of a received packet".

Trust is subjective: Trust, as defined in psychology, is not only evaluated based on the behavior of the entity, but also on how the evaluator perceives the behavior. The variations in point of view depend on the intrinsic characteristics of the evaluator such as how easily the evaluator trusts others, and the expectation of trust. In trust-based approaches for ad-hoc networks all the intrinsic believes and requirements of ad-hoc nodes are reflected in a threshold value.

Trust is not reciprocal: When a node trusts another node, the trusted node is not required to trust the trusting node in return. For example: node A may trust node B to forward its packets, but this does not necessarily mean that node B trusts node A to forward its packets. This is because the expectations of trust may be different in every node.

Trust may not be transitive: Transitivity is a logical relationship in which if it holds, for an operator (op), that $A \text{ op } B$, and $B \text{ op } C$ hold then $A \text{ op } C$ must be true. Most authors argue that this property does not apply to trust-based approaches. Therefore the fact that node A trusts node B, and node B trusts node C does not necessarily imply that node A trusts node C.

Trust is dynamic: Due to the quick and unpredictable change in a node's behavior and to the unpredictability of a node becoming compromised trust should be updated frequently.

Trust is a measure of uncertainty: Trust is most frequently represented as the probability that an agent will perform an action. When the probability is different from 1 or 0 then the subject has no way of knowing for sure whether the agent will perform the action.

III. TRUST METHODOLOGY

We here present a variety of approaches based on Trust methodology. These approaches are applied on MANETs. As MANET provides open for all environments to all the aspiring nodes, the need of secure route is even more required. This survey paper is

aimed to provide different ideologies of Trust in MANET.

a) *Trusted AODV [5]*

In this scheme, AODV protocol is modified implementing node trust and route trust. Two new control packets are added to AODV protocol i.e. trust request packet(TREQ) and trust reply packet(TREP) and routing table is modified by adding one new field: route trust. The RREP packet of AODV is also modified by extending two new fields: neighbour list and route trust.

i. *Calculation of Node Trust*

All the nodes maintain neighbour table to keep information of frequently changing node and node trust value. Node trust value is evaluated using neighbour's collective opinion. The node trust value (NTV) of a node i is calculated by the following formulae:

$$NTV = [NNT(1) + NNT(2) + NNT(3) + \dots + NNT(n)]/n$$

where NNT is the neighbour node trust value about the i node and n is the no of neighbour in the neighbour list.

ii. *Calculation of Route Trust*

Every node calculates route trust for each route in the routing table at some regular interval. Destination node in each entry in the routing table generates R_ACK packet and send back in reverse path. The nodes that receive R_ACK calculate the route trust value using the value in the no_of_packets_received field of R_ACK packet and the value of no_of_packets_sent field in the routing table. Route trust value is calculated by the following formulae: **Route trust = (no of packets send by source - no of packets received by destination)**. The route with route trust value 0 is the perfect one. If the route trust value is equal to the no of packets sent the route is rejected.

iii. *Route Discovery*

In route discovery phase when a node has packets to send it broadcasts RREQ packets. When all RREQ reaches to the destination, it sends RREP packets. After receiving the RREP packets, source node selects three RREP packets that have high route trust value. Then the source node generates the TREQ packets and sends it to all neighbours in the neighbour list of that RREP packet. After receiving the TREQ packet, all neighbours replies with TREP packet to the source node. Then the source node calculates the node trust of the nodes. Next, the source node arrange the RREP packets in the ascending order based on node trust value and selects the first RREP packet and hence that path is selected for communication.

b) *Cooperation of Nodes: Fairness in Dynamic Ad-hoc Networks [6]*

The main idea of CONFIDANT protocol is to identify non-cooperative nodes. A node selects a route based on trust relationships which is built up from experienced routing and packet forwarding behavior of other nodes. Each node monitors the behavior of all

neighbor nodes. When any misbehaving node is found, alarm messages are sent to all other nodes in the network. As a result, all nodes in the network will be able to avoid that misbehaving node while selecting a route. The components of CONFIDANT protocol works as follows:-

i. *The Monitor*

This component watches the behavior of nodes during the routing procedure. If any node misbehaves, then the monitor module detects that misbehaving node and immediately calls reputation system.

ii. *The Trust Manager*

The trust manager handles ALARM messages. When any misbehaving node is found ALARM messages are sent to all other nodes to inform about that node. The trust manager maintain alarm table and trust table for checking the trustworthiness of alarm. The rating function assigns greater weights for own experience and smaller for other nodes opinion about that detected node. The rating of a node is updated when sufficient proof of the nodes maliciousness is found. If the rating falls below threshold value path manager module is called.

iii. *The Reputation System*

The reputation system maintains the rating of nodes in a table which has 2 fields node id and their ratings. The ratings are done according to the type of nodes behavior detected. The rating of a node is updated when sufficient proof of the nodes maliciousness is found. If the rating falls below threshold value path manager module is called.

iv. *The Path Manager*

The path manager manages the routing path according to ratings of the nodes. The path containing malicious nodes are deleted by this module. If any route request comes from malicious node path manager takes appropriate action like ignore request or don't reply etc.

c) *Friendship Based AODV (FrAODV) [7]*

In Friendship based AODV is based on AODV, there are two evaluation algorithms to evaluate forward and reverse path between source and destination. In this scheme, it is assumed that each node has identity can't be forged by any other malicious node and no of malicious node is less than the no of good nodes. In this proposed scheme every node has a list of friends with friendship values. The range of friendship values is 0 to 100. More the friendship values means more trustable. The two algorithms for establishing path are described as follows:

i. *RvEvaluate Algorithm*

This algorithm sets up reverse path from destination to source. After broadcasting RREQ packet the two things can happen: -

Case-1: The receiving node can be destination node itself. If so it checks the friendship value of the node from which it receives the RREQ packet, as every node maintains a friendship list along with friendship value of the neighbor nodes. If the node is not a friend the node rejects the RREQ packet. Otherwise it calculates the friendship value of the route to originator from destination and then compares the current routes friendship value with the existing route's friendship values. The reverse route's friendship value (RvFrRte) is the sum of friendship values of all nodes in that path and it is calculated as follows:

$$RvFrRte = \sum_{i=1}^n \frac{PrFrHpi}{h}$$

where PrFrHpi is friendship value of that node from which the current node receives RREQ packet and h is the no. of hops between source and destination. If the friendship value of the new route is less than the existing route the new route is rejected otherwise it is registered as a friendly route.

Case-2: If the receiving node is intermediate one, it first checks the friendship value of the node from which it receives the RREQ packet and next neighbor node. If one of these two nodes is not in friend list, the intermediate node rejects the RREQ packet. Otherwise it calculates the friendship value of the route to originator from destination using the previously mentioned formulae and compares it with the existing route's friendship value. If the friendship value of the new route is less than the existing route the new route is rejected otherwise the reverse path is established from current node to the previous node.

ii. *FwEvaluate Algorithm*

This algorithm sets up the forward path i.e. from source to destination during RREP forwarding. There are following two cases when any node receives that packet:

Case-1: If the node receiving the RREP packet is sender node itself, it checks the friendship list and the friendship value of the node from which it receives the RREP packet i.e. the next node. If the next node is not a friend, rejects the RREQ packet. Otherwise it calculates the friendship value of forward route to destination and then compares it with the existing route's friendship value. If the friendship value of the new route is less than the existing route the new route is rejected otherwise it is registered as a friendly forward route. If there is not any existing route the new route is included as a friendly route. The forward path's friendship value is formulated as:

$$FwFrRte = \sum_{i=1}^n \frac{FwFrHpi}{h}$$

Where FwFrHpi is friendship value of that node from which the current node receives RREP packet and h is the no. hops between source and destination.

Case-2: If the node is an intermediate node then it checks the friendship value of the node from which it receives the RREP packet and previous node. If one of these nodes is not friend, rejects the RREP packet. Otherwise it calculates the friendship value of the route to destination in the same way and compares it with the existing forward route's friendship value. If the friendship value of the new route is less than the existing route the new route is rejected otherwise the forward path is established from current node to the next node. In this way after establishing friendly path from source to destination the sender sends data packet along that path.

d) *Secure Routing using Trust (SRT) [8]*

In this paper, a secure routing using trust level is proposed. This scheme is based on node transition probability (NTP) and AODV. This scheme develops a new algorithm to secure NTP protocol. A trust rate (T rate) is calculated as a parameter. When a node has data packet to send, it first floods control frame (beacon) in search of secure and reliable route. After broadcasting the first beacon trust rate is evaluated as:

$$Trate = \frac{(r - t)}{r}$$

Where r = no of beacons received by a node, t =no of beacons send by a node. This T rate value divides the nodes of the network into 3 categories: Ally list (level2), Associate list(level1), Acquaintance list (level0).

Ally list: The nodes of the ally list send highly secured information.

Associate list: The nodes of this list send medium secured information.

Acquaintance list: The nodes of this list send the information that does not require any security.

An additional field "level" is there in neighbor table. When a node has data to send it just checks its neighbor table, if the destination is available it just sends data packets. If not, it searches for a node which has route to destination in its same level. If no suitable node is not found it goes to next lower level and so on. If any node in the same level is not found trust is compromised by choosing a neighbor in the next lower level using the following formulae:

Trust compromise= n (associate) + 2*n (acquaintance)

Where n (associate) is the no of nodes in associate list and n (acquaintance) is the no of nodes in acquaintance list. When all the nodes including destination node are in the same level with the source node trust compromise will be very low because trust rate is very high as it is better to forward control packets in the same level than to forward the packets to the another level. In this way after finding secure route the data packets are sent to the destination.

e) *Trusted AOMDV [9]*

AOMDV is a multipath routing protocol. In the paper, a trust mechanism is employed with soft encryption methodology in AOMDV protocol. This Trusted AOMDV protocol has the following steps:

i. *Degree of Secrecy for Path /Message*

Degree of secrecy of a path implies how much degree of security level required for a path to transfer packets. The path trust value (Tp) is the minimum trust value among all nodes along the path p depending upon the path trust value there are three classifications:
 - If $Tp \geq 8$ implies class A paths. All the class A paths have degree of secrecy ≥ 8 . $Tp \geq 5$ implies class B paths. All the class B paths have degree of secrecy ≥ 5 . $Tp \geq 3$ implies class C paths. All the class C paths have degree of secrecy ≥ 3 . This classification is also applied for data packets. Class A data only is transferred to class A category path. It is same for other categories.

ii. *Message Encryption*

The message is divided into three parts and then encrypted using soft-encryption methodology to secure the message. It is encrypted in the following way:

$$a' = aXORc; b' = bXORc; c' = aXORbXORc$$

iii. *Message Routing*

Before routing the encrypted messages a secure trusted path is established using the following trust mechanisms:-

The trust mechanism of this scheme depends on the monitoring of packets and node's behavior. It is assumed here that when a node sends packets it will monitor its neighbor node to which it sends its packet and determines node's trust value depending on its behavior. If the neighbor node sends the packets correctly node's trust will increase, otherwise it is decreased. The trust value of a node (Tn) is calculated as:

$$Tn = Wd * Td + Wr * Tr$$

where Wd is the weight assigned to direct trust Td, Wr is the weight assigned to recommendation trust Tr. Again Direct trust is calculated as:

$Td = Td + c.Ts$, if no. of successful packet transmission time is high and

$Td = Td - c.Tf$, if the no. of packet transmission failed time is high.

where Ts is the aggregate successful transfer time, Tf is the aggregate failure transfer time and c is the predefined constant value. Ts is incremented by 1 for every successful transfer of packet, otherwise Tf is incremented by 1. The trust table values determined through hello message transmission. When a node receives hello message it first check trust table contained in hello packet and find some common nodes it has. If any node common node is found that wants to participate in forwarding packets the trust recommendation (Tr) is calculated by the formulae:-

$$Tt = \sum_{x=0}^n 0.1 * Td(A \rightarrow X) * td(X \rightarrow D) / n$$

where Td(A → X) implies source A's trust on intermediate node X and Td(X → D) implies X's trust on destination D and n is the no. of hop.

In the routing process, source broadcasts RREQ packet. When an intermediate node receives the first RREQ packet it checks the path list and hop count and updates its reverse route table and sets up reverse path. When duplicate request packet arrives at node it checks the hop count of that packet, if it has lesser hop count than the previous one, record of the previously received packet is replaced by the new one in the reverse route table. After receiving request packet destination node generates reply packet (RREP) and sends back to the sender. When an intermediate node receives RREP packet, it compares the trust value in RREP packet with the node's trust value from which it receives the RREP packet. If the node's trust value is less than the one in RREP packet, the trust value in RREP packet is replaced by that node's trust value. In this way, finally when RREP packet reaches to the source node, it gets the trust value from the RREP packet and set it as a trust value of that path. After receiving all the RREP packets and the path trust values, it sorts the paths based on the trust values. Then it breaks Computer Science & Information Technology the message in three parts and encrypts it in the previously mentioned way and starts sending it to the appropriate path according to the data degree of secrecy. After route discovery, if the appropriate path is not found, routing process will be restarted.

f) *Friend Based Ad Hoc Routing using Challenges to Establish Security [10]*

This algorithm achieves security in ad hoc network by sending challenges and sharing friend lists. In this scheme, there are different list of nodes: **Question mark List, Unauthenticated List, Friend List**. The rating of friends ranges from 0 to 10. This algorithm has four steps: challenging neighbor, friends rating, sharing friends and route through friends. FACES is a hybrid protocol as the routing of data is on demand where

as challenging and sharing occurs periodically. When the network is initialized, the nodes are not familiar with each other. So after initializing the network the nodes challenge each other to find the friend nodes. The challenging mechanism works as – suppose node A challenges its neighbor B. A first performs share Friend list with B by sending FREQ packet to B. After receiving FREQ packet from A, B replies by sending its all three list to A. After getting replies A picks one node (let C) from B's list to which it can reach by own. Then send a challenge packet to C directly and through node B. When C receives challenge packet it replies node A and node B in turns replies to node A. then node A compares these two results if it matches node A add B in its friend list otherwise in question mark list.

Friends are rated in this scheme using three parameters: Data rating (DR), friend rating (FR), net rating (NR). Initially the nodes only have friend List, nodes of which perform a successful challenge. The sharing of friend list takes place periodically. Let node B sends its friend list to node A during the friend sharing stage, then node A picks those nodes that are not in its own list from friend list of B and includes those nodes in its own list and the rating of those nodes, which is obtained from B set as FR of those nodes. The data rating (DR) of those nodes is set to zero. Then the net rating (NR) of node is calculated as:

$$NR = \frac{W1 * DR + w2 * FR}{w1 + w2}$$

where w1 and w2 are the weight that is network dependent.

If the friend of B is already in the list of A i.e. if the nodes A and B have common nodes (let C) then A obtains rating of C from B and calculate obtain rating as:

$$OR = (\text{net rating of B in list of A} * \text{net rating of C in list of B}) / 10$$

FR of node C is obtained by adding all OR from various neighbor nodes and divides the value by the sum of ratings of those various nodes. The data rating is calculated on the basis of data transfer by a node. DR is calculated as: $DR = 10 * (1 - e^{-\lambda x})$, where x is no of forwarded data packets and λ is a factor by which data packets are related to rating. The routing of data takes place when any node has data to end. It broadcasts route request message including no of data it wants to send. After receiving route reply messages, it finds the best route depending on the net rating value of nodes, to the destination from its friend list.

g) *Trust Based Security Protocol Routing [11]*

In this protocol a trust mechanism is employed in DSR protocol. An extra data structure is maintained

by every node that is Neighbor's Trust Counter Table (NTT) which is used to keep track of no. of sent packets by a node using a forward counter (FC) and also stores the trust counter(TC) corresponding to node. Initially a node can completely trust its neighbor or fully distrust its neighbor as the nodes don't have any information about its neighbor nodes reliability. When any node needs to send data it broadcasts RREQ packets. Each time a node (let nk) receives packet from another node (let ni), node nk increments the FC of ni as:

$$FC_{ni} = FC_{ni} + 1; i = 1, 2, \dots$$

then, this new FC_{ni} value is stored in NTT of node nk. After receiving all RREQ packets, destination node makes a MAC on the no of packets it received (Prec) using the shared key between the sender and destination. Then the destination node attaches that MAC and also the accumulated path from the RREQ after digitally signed it, in the RREP packet and sends back in the reverse path to the destination. The intermediate nodes of that path determines Success ratio as: - $SC_{ni} = FC_{ni} / Prec$, where Prec is the no. of packets received at destination. This SC_{ni} is appended in RREP packet. The intermediate nodes in reverse path check the validity of the RREP packet by verifying digital signature of destination. If it is valid, the intermediate node signs the packet and forwards it to the next, otherwise the packet is dropped. When source node finally gets the reply it first verifies the first node id in RREP packet. If it is its neighbor, then all other intermediate nodes' digital signature is verified. If the verifications of all the nodes are successful then the trust counter is incremented for all the nodes as: $Tc_i = Tc_i + \delta 1$, if the verification is failed the trust counter value is decremented by 1: $Tc_i = Tc_i - \delta 1$. where $\delta 1$ is the small fractional value. The source node also checks the success ratio of all other nodes and compares it with the minimum threshold value (SR_{min}), if the SR_{ni} of a node is less than the SR_{min} the trust counter is decremented by another step value $\delta 2$ again, otherwise it is incremented. Another comparison is made by comparing trust counter with a minimum threshold. If trust counter is less than the trust threshold value the node is marked as malicious. This mechanism is applied to all the other routes and a route with no or least malicious node is selected. In this way, a trusted and authenticated route is found for secure routing.

h) *Trust Based DSR [12]*

This protocol is proposed to improve the security of the existing DSR protocol. The trust based secure route is established in this scheme. In DSR the shortest route is selected which may not be secure. There are some malicious nodes in the network that replies to the route request packet with shorter hop count (black hole) so that the source will select that path, and routing process is disrupted. The following

components are used in this newly proposed protocol: Initialiser, Upgrader, Administrator, Monitor, and Router. In this scheme, there is a separate administrator to maintain the trust values of all other nodes. An acknowledgement module is there which is used to keep track of all received acknowledgements and trust values of nodes are adjusted. Every node has trust value which depends on its interaction with its neighbor. Trust unit of this scheme comprises of three modules: - Initialiser module assigns low trust values to the unknown nodes in initial stage. If the route contains some known and unknown nodes, then it assigns trust of those known nodes as the initial trust value of the unknown nodes. Upgrader module upgrades the trust value of a node based on experiences of that node in a particular situation. When a node receives any reply from its neighbor the trust value of neighbor node is updated. If any reply is not received by a node the trust value of the neighbor node is decreased. Trust value is evaluated as: $T = \tanh[(\Delta+W) * Te]$ where T is the updated trust, Te is existing trust, W is a weight i.e. 1 for acknowledgements and 0.5 for data packets forwarded and received, Δ is +1 for positive and 0 for negative experiences. Positive experience means acknowledgement is received within the time frame and otherwise it is considered as the negative experience. Administrator module keeps the trust information of all the known nodes and also has some methods to query this trust information. The monitor module monitors the received acknowledgments to adjust trust values of nodes. The router module selects the route to forward packets based on nodes trust values. Monitor module uses two routing strategy: In the first routing strategy, the route is rated based on the average value of all nodes along that path. The route which gets highest rating is selected for routing. In the second routing strategy, the average of all nodes trust value is divided by no of nodes to get shorter path. The route which gets high value is selected.

i) *GAODV : Against single Black Hole and Collaborative Black Hole [13]*

The AODV protocol has a provision of sending a gratuitous RREP packet to the destination node. Whenever an intermediate node has a route towards destination, in addition to sending the RREP to the source, it also unicasts a gratuitous RREP to the destination node. In our protocol the gratuitous RREP is conceptualized and simulated as the CONFIRM packet. Thus, a CONFIRM packet is unicasted/routed by the RREP to the destination. Note that it can be sent only if the RREP has a route towards destination. It is only after the receipt of CONFIRM will the destination await for packets from the source. In order to facilitate cross checking by the source (of the route claimed by the RREP), the source unicasts a CHCKCNFRM to the destination. Upon CHCKCNFRMs receipt the destination

replies by broadcasting a REPLYCONFIRM to the source, only if it received a CONFIRM and a CHCKCNFRM. Since a black hole does not possess a route towards the destination, it fails to send the CONFIRM, thus reply to the CHCKCNFRM is never generated by the destination. This leads the source to conclude that the RREP sending node was the black hole one.

IV. CONCLUSION

MANETs are vulnerable to different types of attacks due to its infra-structure less network. Different trust based approaches are proposed to prevent such types of attacks and to improve Quality of Services (QoS). These trust based approaches try to give a secure node in routing path by implementing trust mechanism in the existing routing protocols. In this paper, firstly we have given a brief idea on several types of characteristics that Trust possesses in itself and then different Trust schemes.

We have seen that there are different methods in which Trust can be applied. But there is a possibility to develop an approach that can be standardised to attain QoS as well as minimizing the several attacks. Trust mechanism can be applied in various environments like in hybrid environments. We can also develop some rules in the protocol on the basis of which the actions are taken to detect the nodes that are authenticated but perform malicious behaviour without dropping packets and also authenticate the nodes to prevent attacks. So we can work on these approaches to develop a new trust based protocol for standardisation.

REFERENCES RÉFÉRENCES REFERENCIAS

1. G. Aggelou (2004) Mobile Ad Hoc Networks, Mcgraw-Hill.
2. MANET Working Group : <http://datatracker.ietf.org/wg/ace/documents/>
3. S. Corson, J.Macker. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. IETF RFC2501, 1999.
4. Jin-Hee Cho, Ananthram Swami, and Ing-Ray Chen "A Survey on Trust Management for Mobile Ad Hoc Networks" IEEE Communications Surveys & Tutorials, VOL. 13, NO. 4, FOURTH QUARTER 2011.
5. M. Pushpa, (2009) "Trust Based Secure Routing In Aodv Routing Protocol", International Conference On Internet Multimedia Services Architecture And Applications (Imsaa), Usa: Ieee Press, 1-6.
6. S. Buchegger, J. L. Boudec, (2002) "Performance Analysis Of Confidant Protocol", Mobihoc'02, Epfl Lausanne, Switzerland, Pp226-236.
7. Essia, T., Razak, A., Khokhar, R.S., Samian, N.: Trust-Based Routing Mechanism In Manet: Design And Implementation. Springer, 18 June 2011.

8. Edua Elizabeth, N., Radha, S., Priyadarshini, S., Jayasree, S., Naga Swathi, K.: Srt-Secure Routing using Trust Levels in Manets. *European Journal of Scientific Research*, Issn 1450-216x Vol. 75, No. 3 (2012), Pp. 409-422.
9. Huang, J., Woungang, I., Chao, H., Obidant, M., Chi, T., Dhurandher, S.K.: Multi-Path Trust-Based Secure Aomdv Routing In Ad Hoc Networks. *Ieee* 2011.
10. Dhurandher, S.K., Obidant, M.S., Verma, K., Gupta, P., Dhuradar, P.: Faces: Friendship-Based Hoc Routing using Challenges to Establish Security InManets Systems. *Ieee System Journal*, Vol.5, No. 2, June 2011.
11. Sharma, S., Mishra, R., Kaur, I.: New Trust Based Security Approach For Ad-Hoc Networks. *Ieee* (2010).
12. Bhalaji, N., Mukherjee, D., Banerjee, N., Shanmugam, A.: Direct Trust Estimated on Demand Protocol For Secured Routing In MobileAd-Hoc Networks. *International Journal of Computer Science & Security*, Vol. 1, Issue (5).
13. GAODV: A Modified AODV against single and collaborative Black Hole attacks in MANETs: Dhurandher S.K.; Woungang I.; Mathur R.; Khurana P. *Advanced Information Networking and Applications Workshops (WAINA)*, 2013.