

Design of Cross Layered Security Architecture to Mitigate Misbehaving Nodes in Self-Defending Network

Neelavathy Pari.S

Assistant Professor

Department of Computer Technology

Madras Institute of Technology, Anna University

Chennai , India

Neela_pari@yahoo.com

Jayapriya Surendran

Department of Computer Technology

Madras Institute of Technology

Anna University

Chennai,India

priya7390@gmail.com

Abstract

Countermeasures for node misbehavior and selfishness are mandatory requirements in Mobile Adhoc Network (MANET). Selfishness that causes lack of node activity cannot be solved by classical security means that aim at verifying the correctness and integrity of an operation. In this paper, we suggest a new routing protocol Trusted Path Routing Protocol (TPRP) to enforce cooperation among the nodes of the MANET and to prevent selfish behavior. Each mobile node in the network maintains a data structure called Trust and Reputation Table (TRT) to keep track of other node's behavior. To decide whether a node is malicious or not involves decision making and hence is a problem of uncertainty. The best way to deal with uncertainty is by means of probability. So we make use of Bayesian Probability mathematical model to calculate trust whose value lies between 0 and 1. If the trust value goes below the *threshold trust*, then the node is termed as malicious and excluded from the network.

Key words:

Mobile Ad hoc Network, Cross layer protocol design, Trust based routing, Watchdog mechanism, Self defending network.

1. Introduction

The performance of MANET severely degrades in face of simple node misbehavior. Unlike networks using dedicated nodes to support basic functions like packet forwarding, routing and network management, in ad hoc networks those functions are carried out by all available nodes.

Node misbehavior that affects network operations may range from simple selfishness or lack of cooperation due to the need for power saving to active attacks aiming at denial of service(DoS) . *Selfish nodes* use the network but do not cooperate, saving battery life for their own communications. They do not intend to directly damage other nodes. Malicious nodes, on the other hand, aim at

damaging other nodes by causing network outage by partitioning while saving battery life is not a priority.

Because of increased vulnerability, ad hoc networks should take into account security problems as a basic requirement regardless of application scenarios. A network in which the countermeasures are integrated with the basic networking mechanism at the early stages of the design is termed as *self-defending* network. A basic requirement for keeping the network operational is to enforce ad hoc nodes' contribution to network operations despite the conflicting tendency of each node towards selfishness which is motivated by the scarcity of node power.

We propose a trust based mechanism in TPRP to enforce node cooperation and to detect and isolate misbehaving nodes based on trust value of the node. Each network entity in TPRP maintains other entities' behavior using a trust computation technique. The trust metric is computed based on data monitored by the local entity and some recommendation provided by other nodes involved in each operation.

The remainder of this paper is organized as follows: section 2 explains the trust computation technique, the cross layer design principle is presented in section 3, section 4 explains the various components of TPRP, section 5 illustrates the working of Trusted Path Routing Protocol and finally section 6 demonstrates the experimental results.

2. The Trust Concept

In this proposed protocol, by dynamically calculating the node trust values, the destination can be able to select the more trusted routes for route reply rather than selecting the shorter ones. Our protocol isolates the malicious nodes from participating in the network. So the potential damage caused by the malicious nodes are reduced. An additional data structure called Trust and Reputation Table(TRT) is

maintained by each network node.

2.1 Trust Factors

The main factors that affect the trust evaluation process of one node about other nodes are

- Direct Interactions
- Indirect Interactions
- Reputation

Trust is modeled using the following parameters:

- Packet forward rate/Successful transmissions(α)
- Packet drop rate/Unsuccessful transmissions(β)

Initially, when nodes are just deployed for the first time or when new nodes are introduced to the network, the presence of some optimistic nodes willing to take risks is required, as there is no evidence of node's past behavior. Initial trust value between nodes can be assigned based on the application and/or deployment environment, using one of the following methods:

- *All nodes are trustworthy.* This is the quickest method of establishing trust and building a functional network, but it is very risky, as malicious nodes can be given high trust value.
- *All nodes are considered to be untrustworthy.* It is a very slow method. Trust formation takes a very long time to be established, but on the other hand it is very robust and can be used for mission critical applications.
- *All nodes are neutral.* They are neither trustworthy nor untrustworthy This is in between in terms of establishing trust compared to other mentioned methods.

2.2 Trust Evolution

The evolution process is another important dynamic aspect of trust and can be regarded as iterating the process of trust formation as additional evidence becomes available. It is the process of updating trust level between the nodes. In order for other nodes in the network to receive updates regarding the trusted behaviors of nodes or even threats, a mechanism for trust reporting is necessary. The trust evolution process involves:

- Updating direct trust
- Updating indirect trust
- Updating total trust, based on the updated values of direct and indirect trust.

We use the term direct trust to talk about the trust calculated directly from a node's observation. Indirect trust is calculated from the recommendations given by other members of the network community. Reputation is obtained by using weight factor on both direct and indirect trust.

Node	Direct Trust(A)	Indirect Trust(B)	Combined Trust E(Reputation)

Table 1: Trust and Reputation Table (TRT)

2.3 The Beta Density Function

The reputation system is based on the beta probability density function which can be used to represent probability distributions of binary events. This provides a sound mathematical basis for combining direct and indirect trust and expressing reputation ratings. The mathematical analysis leading to the expression for posteriori probability estimates of binary events can be found in [2] and we will present only the results here.

The reason behind using the Beta distribution is that, it's being used widely in risk and decision analysis, due to its flexibility, and also it can be estimated easily. The beta family of probability density functions is a continuous family of functions indexed by the two parameters α and β . The beta distribution can be expressed using gamma function Γ as

$$f(p|\alpha,\beta) = \frac{\Gamma(\alpha + \beta) p^{\alpha-1} (1-p)^{\beta-1}}{\Gamma(\alpha) \Gamma(\beta)} \quad (1)$$

The probability expectation value of the beta distribution is given by

$$E_A(p) = \frac{\alpha_A}{\alpha_A + \beta_A} = P_A \quad (2)$$

$$E_B(p) = \frac{\alpha_B}{\alpha_B + \beta_B} = P_B \quad (3)$$

where α_A , β_A and α_B , β_B are number of packets successfully transmitted and number of packets dropped which is obtained by direct observation and recommendations. P_A and P_B are direct and indirect trust values respectively.

2.4 The Beta Reputation System

Reputation is obtained by using weight factor on both direct and indirect trust. The weight factor for direct and indirect trust is obtained using Beta reputation system.

$$\text{Reputation} = A * W_A + B * W_B \quad (4)$$

$$\text{where } W_A = \frac{\alpha_A + \beta_A}{K} \quad (5)$$

$$W_B = \frac{(\alpha_A + \beta_B)(\alpha_B - 1)}{A_B * K} \quad (6)$$

where $K = \alpha_A + \alpha_B + \beta_A + \beta_B - 2$

The reputation is embodied in the Beta model and carried by two parameters α_{ij} and β_{ij} where α_{ij} represents number of successful transactions and β_{ij} represents number of un successful transactions.

The reputation of node n_i maintained by node n_j is

$$R_{ij} = \text{Beta}(\alpha_{ij} + 1, \beta_{ij} + 1) \quad (7)$$

The combined trust is defined as the expected value of the reputation,

$$T_{ij} = E(R_{ij}) = E(\text{Beta}(\alpha_{ij} + 1, \beta_{ij} + 1)) = \frac{\alpha_{ij} + 1}{\alpha_{ij} + \beta_{ij} + 2} \quad (8)$$

Second hand information is presented to node n_i by another neighboring node n_k . Node n_i receives the reputation of node n_j by node n_k as R_{kj} , in the form of two parameters α_{kj} and β_{kj} . Using this new information, node n_i combines it with its current assessment R_{ij} to obtain new reputation R_{ij}^{new} .

$$R_{ij}^{\text{new}} = \text{Beta}(\alpha_{ij}^{\text{new}}, \beta_{ij}^{\text{new}}) \quad (9)$$

where

$$\alpha_{ij}^{\text{new}} = \alpha_{ij} + \frac{2 \alpha_{ik} \alpha_{kj}}{(\beta_{ik} + 2)(\alpha_{kj} + \beta_{kj} + 2)(2 * \alpha_{ik})} \quad (10)$$

$$\beta_{ij}^{\text{new}} = \beta_{ij} + \frac{2 \alpha_{ik} \beta_{kj}}{(\beta_{ik} + 2)(\alpha_{ki} + \beta_{kj} + 2)(2 * \alpha_{ik})} \quad (11)$$

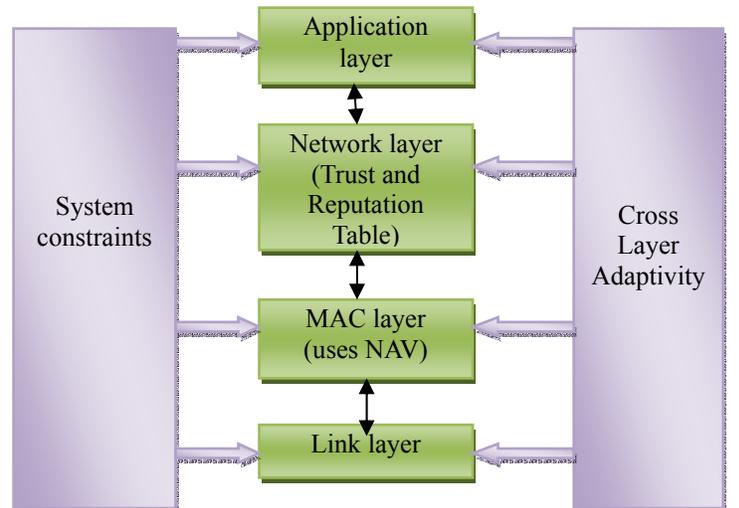
3. Cross layer design principle

One of the major components in the success of the Internet is the layered open system interconnection (OSI) architecture. The modularity achieved through layering leads to better understanding of the abstract functionality of the layers and thus enables better understanding of the overall system. This led to rapid growth in the development of number of applications that drive the Internet. The fact that the same lower layers may be reused for every application decreases the development cost of the application and enhances the utility of the network architecture. Layering simplifies network design and leads to robust scalable protocols in the Internet.

But layering suffers from sub-optimality and inflexibility. Layering is suboptimal because each layer has insufficient information about the network since it does not allow the sharing of information among the layers. The interface between the layers are static and independent of individual network constraints and applications. Layering is inflexible because the developer of a new application has to solely depend on the functionality of the lower layers. To achieve better results, the layers in a wireless network must coordinate and adapt with the change in the state of the wireless network. This is the motivation behind the cross layer paradigm for protocol design in wireless networks.

The cross layer design of protocol stack enables layers to exchange state information in order to adapt and optimize the performance of the network. The sharing of information enables each layer to have a global picture of the constraints and characteristics of the network, leads to better coordinations and enables them to take decisions that would jointly optimize the performance of the network. [9]

Figure 1 : Cross Layer Design Principle



3.1 Network Allocation Vector (NAV)

The Network Allocation Vector (NAV) is virtual carrier sensing mechanism used with wireless network protocols . The virtual carrier sensing is a logical abstraction which limits the need for physical carrier sensing at the air interface in order to save power. The MAC layer frame headers contain a *Duration Field* that specifies the transmission time required for the frame, in which time the medium will be busy. We make use of this NAV information in the MAC header in TPRP routing algorithm to achieve faster convergence when there is a packet loss in the wireless network. When the neighbor node fails to send packet, the node checks its NAV factor. If NAV is set to zero then we can conclude that the neighbor node is a selfish node else dropping of packet is due to collision.

4. Components of TPRP

4.1 Network Entity

The network entity corresponds to a mobile node. Each entity is enriched with a set of Trust and Reputation Table (TRT) and a Watchdog mechanism (WD). The TRT and the WD together constitute the trust based mechanism presented in this paper. These two components allow each entity to observe and classify each other entity that gets involved in the request/reply process, reflecting the cooperative behavior of the involved parts.

4.2 Trust and Reputation Table

The Trust and Reputation Table is defined as a data structure stored in each network entity. Each row of the table includes the reputation data pertaining to the node. Each row consists of four entities: the unique identifier of the entity, a collection of recent direct observations made on entity's behavior, a list of recent indirect observations provided by other entities and the value of combined trust which is calculated from the direct and the indirect trust.

4.3 The Watchdog Mechanism

The Watchdog Mechanism implements the validation phase and it is used to detect misbehaving nodes. Here, we assume wireless interfaces that support promiscuous mode operation. Promiscuous mode means that if a node A is within the range of node B, it can overhear communications to and from B even if those communications do not directly involve A. The Watchdog mechanism relies on this promiscuous mode operation. Every time a network entity transmits data to its next hop, it stores the expected result E_r in a temporary buffer and verifies if the Observed result O_r matches with the E_r stored in the buffer. If the next hop node forwards the data without performing any modification, then the Observed result (O_R) will match the Expected result (E_R) and the

Watchdog will remove the corresponding data from its temporary buffer and increment the number of successful transactions (α) by 1. On the other hand, if the Observed result did not match the Expected result, then the node is considered to be malicious and the Watchdog increments the number of unsuccessful transactions (β) by 1. If the watchdog is not able to detect any outgoing packets from its next hop node, it checks the NAV timer. If the timer is set, then it concludes that the packet drop is due to collision and the node is not termed malicious. If the timer is not set, it means that the medium is free and there is no possibility of collision and hence it terms the node as malicious.

5. The TPRP Protocol

In TPRP, when source node wants to communicate with another node (destination), and if no routing information is available, it initiates path discovery by sending the route request that contains source id, broadcast id, destination id and trust values of each neighbor and reliability of source node and, hop count. On accepting the route request the neighboring node calculates its reliability using trust values in the table and takes the following decision. If the trust value is below threshold trust, the node discards the route request. If the trust value is above the threshold, cumulative trust is found by adding the predecessor trust with its trust value.

If the node has already received the route request with same source address and same broadcast id and if the cumulative trust is less than the cumulative trust of current route request, the previous route request path is rejected and the current route request path is recorded. The route request is then forwarded to the intermediate nodes' neighbor which contains trust values of each neighbor and cumulative reliability. Each time when route request is forwarded from one node to another, hop count is incremented and that is also sent along with the route request. When two or more route request reaches the destination from the same source and same broadcast id and in different path, it selects the most reliable path by finding the average trust. Average trust is computed as follows:

$$\text{Average Trust} = \text{Cumulative trust} / \text{Number of hops.}$$

If average trust of one path is greater than other path, the destination sends the route reply in most trusted path. The source receives the new path and records the path for future use. When there is packet loss, we make use of the cross layer design to achieve faster convergence. When the neighbor node fails to send packet, the node checks its NAV factor. If NAV is set to zero then the neighbor node is a selfish node else we can predict that the dropping of packet is due to collision.

Pseudo code for Route Discovery in TPRP

```

Check the destination in intermediate node

Direct Trust = number of packets successfully transmitted
(A)           $\frac{\text{under direct observation } (\alpha_A)}{\text{total number of packets transmitted}}$ 

Indirect Trust= number of packets successfully transmitted
(B)           $\frac{\text{under indirect observation } (\alpha_B)}{\text{total number of packets transmitted}}$ 

Reputation =  A*WA + B*WB

If intermediate node is the destination
{
(i) Intermediate node generates RREP
(ii) Intermediate node increases the sequence number
(iii) When two or more route request reaches the
destination from same source and same broadcast id,
it selects most reliable path by finding the average
trust

Average Trust = Cumulative Trust / Number of hops

(iv) Destination sends route reply in the path whose
Average trust value is high
(v) Setup an acknowledgement message
}
Else if((destination node is unknown) and (Direct trust of
predecessor node >= 50%)or (Reputation of predecessor
node >=50%))
{
(i) Add trust value of predecessor node to cumulative
Trust value
(ii) Copy the RREQ message(increment hop count and
decrement TTL) and trust value
(iii) Rebroadcast it
}
Else if((intermediate node sequence number < RREQ
sequence number) or (destination not reachable)or
(predecessor node trust value or reputation < 50%))
{
Return NULL;
}
Else if (intermediate node has a route to destination)
{
(i) Copy the RREQ values of the sequence number,
hop count and lifetime to the RREP message.
(ii) Add source node to the precursor list of the
Destination
(iii) Setup an acknowledgement message.
}
}

```

6. Methodology of Evaluation

A. Simulation Environment:

For the simulations, we use NS-2 (v-2.34) network simulator. NS-2 provides faithful implementation of different network protocols. At the physical and data link layer, we used the IEEE 802.11 algorithm. The channel used is Wireless Channel with Two Ray Ground radio propagation model. At the network layer, we use AODV as the routing algorithm. Finally, UDP is used at the transport layer. All the data packets are CBR(constant bit rate) packets. The size of the packet is 512 bytes. The packet transmission rate is 0.2 Mbps.

The connection pattern is generated using *cbrgen* and the mobility model is generated using the *setdest utility*. *Setdest* generates random positions of the nodes in the network with specified mobility and pause time. The terrain area is 800m x 800m with number of nodes varying from minimum 10 to maximum 70 with chosen speed from 10 m/s to 70 m/s. The simulation parameters are summarized in table 2.

Each data point represents an average of ten runs. The same connection pattern and mobility model is used in simulations to maintain the uniformity across the protocols.

Table 2 Simulation Parameters

Parameter	Value
Simulator	Ns-2 (version 2.34)
Simulation Time	100 s
Number of nodes	10 to 80
Routing Protocol	AODV and TPRP
Traffic Model	CBR
Terrain Area	800m x 800m
Transmission Range	250 m
No of malicious node	1

B. Metrics used for Simulation

To analyze the performance of our solution, various contexts are created by varying the number of nodes and node mobility. The metrics used to evaluate the performance of these contexts are given below.

Packet Delivery Ratio: The ratio between the number of packets originated by the application layer CBR sources and the number of packets received by the CBR sink at the destination.

Average End-to-End Delay: This is the average delay between the sending of the data packet by the CBR source and its receipt at the corresponding CBR receiver. This includes all the delays caused during route acquisition, buffering and processing at intermediate nodes, retransmission delays at the MAC layer etc. It is measured in milliseconds.

Simulation Results and Analysis

To evaluate the packet delivery ratio, End-to-End Delay and Normalized Routing Overhead, simulation is done with nodes with the source node transmitting maximum 1000 packets to the destination node.

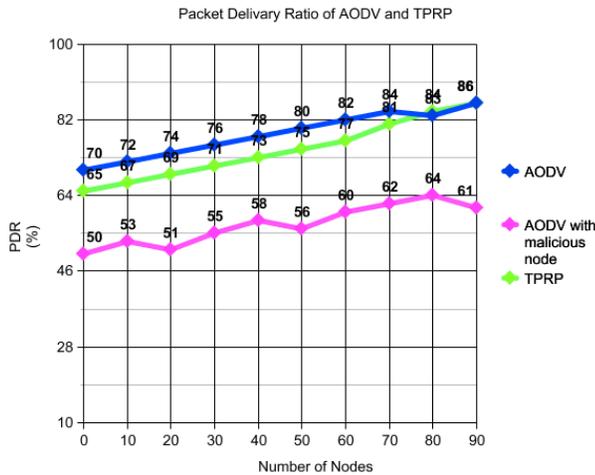


Figure 2

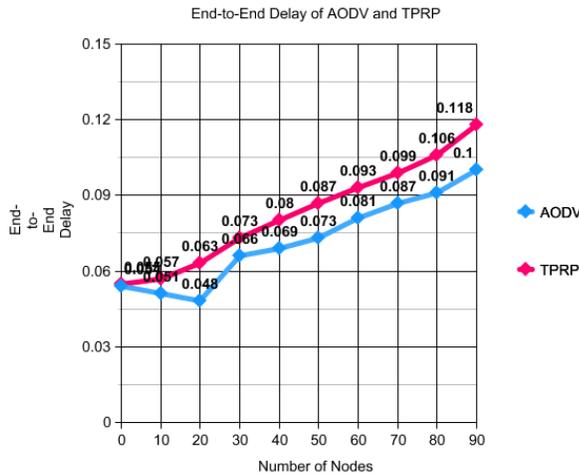


Figure 3

Figure 2 and 3: Effect of Network size

Conclusion and Future Work

With the fact that default AODV is susceptible to malicious attacks, in this research exercise, we attempt at investigating the existing solutions for their viability. Having justified the need for further improvements, we propose an algorithm to counter attacks by malicious

nodes in MANETs. From the experimental results, we

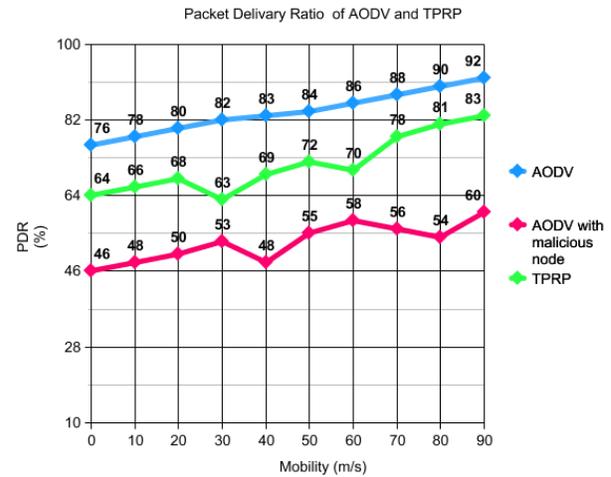


Figure 4

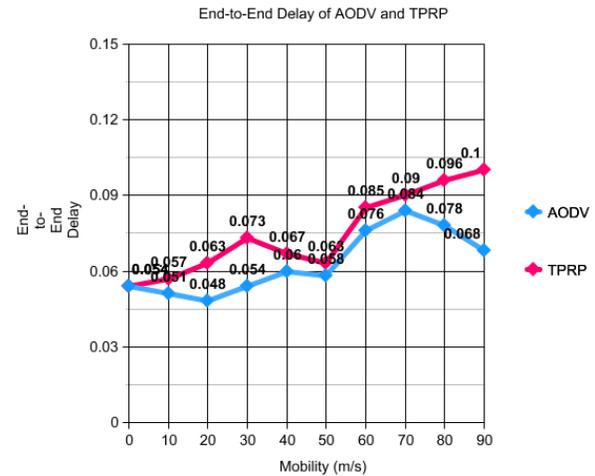


Figure 5

Figure 4 and 5 : Effect of Mobility

conclude that the proposed solution achieves a very good rise in PDR with acceptable rise in end-to-end delay. Moreover, the proposed algorithm does not entail any hidden overhead on either the intermediate nodes or the destination nodes. Thus, as compared to other approaches discussed in section II, we believe the proposed algorithm is simple and efficient in implementation.

We also emphasize that though the proposed trust and reputation based algorithm is implemented and simulated for the AODV routing algorithm, it can also be further trivially extended for use by any other routing algorithms, as well. As part of our future endeavor, we aim to include the mobile node's energy level as a component to isolate the selfish nodes from the network.

References

- [1] Yanwei Wu, Shaojie Tang, Ping Xu, Xiang-Yang Li “Dealing with Selfishness and Moral Hazard in Non Cooperative Wireless Networks,” IEEE Transactions on Mobile Computing, vol.9, no.3, 2010.
- [2] Statistical Inference by Casella and Berger, 1990 p.298.
- [3] Guanfeng Liang, Rachit Agarwal, Nitin Vaidya “When Watchdog meets Coding,” Proceedings of IEEE Infocomm,2010.
- [4] Feng Li, Jie Wu “Uncertainty Modeling and Reduction in MANETs,” IEEE Transactions on Mobile Computing”,vol 9,no.7,2010.
- [5] Li Shi Chang , Yang Hao –Lan , Zhu Qing – Sheng , “Research on MANET Security Architecture Design”, International Conference on SignalAcquisition and Processing,2010.
- [6] Sintayehu Dehnie and Stefano Tomasin , “Detection of Selfish Nodes in Networks Using Coop-MAC Protocol with ARQ”.IEEE Transactions on Wireless Communications, 2010.
- [7]Bounpadith Kannhavong, Hidehisa Nakayama, Yoshiaki Nemoto, And Nei Kato, “A Survey of Routing Attacks in Mobile Ad Hoc Networks”. IEEE Wireless Communications, October 2007.
- [8] J .Jaramillo and R .Srikant , “Darwin: Distributed and Adaptive Reputation Mechanism for Wireless Adhoc Networks,” Proceedings of ACM, Mobicom, 2007
- [9] Nabhendra Bisnik “Protocol Design for Wireless Ad hoc Networks-A Cross Layer Paradigm”
- [10] Hongmei Deng, Wei Li and Dharma P .Agrawal, “Routing Security in Wireless Ad-hoc Networks”, IEEE Communications Magazine, October 2002.
- [11]Yao Wang and Julita Vassileva, “Trust and Reputation Model in Peer-to-Peer Networks, Proceedings of IEEE Conference on P2P Computing, September, 2003
- [12]Kitae Nahm, Ahmed Helmy and C.C.Jay Kuo, “Cross Layer Interaction of TCP and Ad Hoc Routing Protocol In Multihop IEEE 802.11 Networks, IEEE Transaction on Mobile Computing,vol 7,no 4,April 2008.
- [13] Tao Wang and Shunzheng Yu, “Anomaly Detection in Wireless Mobile Ad Hoc Networks with Multi-Layer Observation Sequences”, Proceedings of the International Symposium on Intelligent Information Systems and Applications , IISA 2009.
- [14] Moitreyee Dasgupta S.Choudary and N.Chaki, International Journal of Computer Applications,2010.
- [15] L.Khoukhi and S.Cherkaoui, “Flexible QoS Routing Protocol for Mobile Ad Hoc Networks” Proceedings of the 11th IEEE International Conference on Telecommunication, ICT 2004.
- [16] Yonglin Ren and Azzedine Boukerche, “Modelling And Managing the Trust for Wireless and Mobile Ad Hoc Networks”, Proceedings of ICC, 2008.
- [17]Chu-Hsing Lin, Wei-Shen Lai, Yen-Lin Huang, Mei-Chun Chou, “Secure Routing Protocol with Malicious Nodes Detection for Ad Hoc Networks”, International Conference on Advanced Information Networking and Applications”, IEEE 2008.
- [18] Jie Li and Ruidong Li, “Future Trust Management Framework for Mobile Ad Hoc Networks”, IEEE 2008.
- [19] Satoshi Kurusawa, Hidehisa Nakayama, Nei Kato, Abbas Jamalipour and Yoshiaki Nemoto, “Detecting Blackhole Attack on AODV-based Mobile Ad Hoc Networks by Dynamic Learning Method” International Journal of Network Security, vol.5, no.3, Pp 338-346,November 2007.
- [20] Pitipatana Sakarindr and Nirwan Ansari, “Security Services in Group Communications over Wireless Infrastructure, Mobile Ad Hoc, and Wireless Sensor Networks”, IEEE Wireless Communications,2007.

