IJoFCS (2010) 1, 22-29

e-forensic PRESS

The International Journal of
FORENSIC COMPUTER SCIENCE

www.IJoFCS.org

Double Umpiring System for Ad Hoc Wireless Mobile Network Security

A. Kathirvel(1) and P.Sivaraman

(1) Department of Computer Science and Engineering, Karpaga Vinayaga College of Engineering and Technology, Chennai, Tamilnadu, India. Email: ayyakathir@gmail.com

Abstract - Protecting the network layer from malicious attacks is an important and challenging issue in both wired and wireless networks. Protection becomes even more challenging in the case of MANET. In this paper we propose a solution based on a double umpiring system (DUS) to provide security for routing and data forwarding operations. In our system each node in the path from source to destination performs both packet forwarding and umpiring. In the umpiring role, each node in the path closely monitors the behavior of the succeeding node and immediately flags the offending node if any misbehavior is noticed. The double umpiring system is sufficiently general to be applied to any networking protocol. As a demonstration, we implemented the double umpiring system as a modification of the popular AODV protocol. Simulations demonstrated that even when 40% of the nodes were malicious and the maximum speed was 20 m/s, DUS throughput was increased by 161.25% and communication overhead increased by 18.5%. This is a vast improvement over the performance of the conventional AODV protocol, with only a nominal increase in overhead.

Keywords - Ad Hoc Networks, AODV, DUS, and Malicious node

1. Introduction

A mobile ad hoc network (MANET) is a self-created, self-organized, and self-administering set of nodes connected via wireless links without the aid of fixed infrastructure or a centralized administrator. Each node moves and operates in a distributed peer-to-peer mode, generating independent data and acting as a router to provide multi-hop communication. MANET is ideally suited for critical applications in civilian and military environments such as disaster response and battlefield conditions where security is an important factor.

In this paper we address the problem of securing network layer operations from malicious nodes. Malicious nodes may disrupt routing algorithms by transmitting a false hop count, they may drop packets, or they may route packets over unintended routes. Our work rests on the foundations of the previously described SCAN system [1].

In SCAN, mobile ad hoc network protection is based on (i) local collaboration in which neighboring nodes collectively monitor each other, and (ii) information cross-validation with each node monitoring neighbors by cross-checking overheard transmissions. Each node monitors the routing and packet-forwarding behavior of its neighbors and independently detects the existence of nearby malicious nodes. This is made possible by the wireless nature of the medium, which enables packet reception by all nodes within transmission limits. Cross-checking employs a modified AODV protocol containing a new field (*next_ hop*) in the routing messages, so that each node can correlate the overheard packets accordingly.

While each node monitors it neighbors independently, several neighboring nodes must in the neighborhood collaborate to convict a malicious node. An agreement between a minimum of k neighboring nodes is required to convict a malicious node. Once a node is convicted the network reacts by depriving it of access. In SCAN each node must possess a valid token in order to interact with other nodes. The tokens are produced using asymmetric key cryptography to prevent forgery. A group of nodes (minimum of k) may collaboratively sign a token, but no single node can do so. Each node must have its token renewed periodically. A node expressing continuous good behavior requires token renewal at less frequent intervals than a fresh entrant node.

Our double umpiring system has been strongly influenced by the above schemes. In our system the active nodes have dual roles just as in *watchdog* [2], and we also exploit the promiscuous hearing functionality used by both SCAN and *watchdog [2]*. We achieve avoidance of malicious nodes using a system of tokens similar to the system employed in SCAN. The tokens contain a nodeID and a status bit. The nodeID is considered to be immutable. The status bit of all participating nodes is initially set to 0, indicating a “green flag” with freedom to participate in all network operations. Nodes cannot change their own status bit.

When an umpiring node finds that a neighboring node is misbehaving it sends an M-Error message to the source and the malicious node’s status bit is set to 1 (red flag) using an M-Flag message. Nodes with a “red flag” are prevented from participating in the network.

Our objective in designing the security system was to keep the overhead as low as possible while optimizing throughput. We did not use the encryption or key algorithms found in SCAN. We found that token issue and renewal and broadcasts to announce convictions created very large communication overheads that degraded energy performance, which has been completely overlooked in SCAN implementations. There is no token renewal feature in our system. Instead, all nodes are preissued with green tokens, and they continue to enjoy this status until any immediate ancestor node in umpiring mode detects misbehavior and sends M-Error and M-Flag messages to set a red flag.

Similar to SCAN, we have used the “*next_hop*” field in our AODV implementation in order to facilitate cogent promiscuous hearing. Our umpiring system can detect any false reporting of hop count during the route reply process RREP. In *watchdog,* detection of malicious action is performed by a single node, while in SCAN it is done by a set of ‘k’ neighbors. In our double umpiring system, the value of ‘k’ is two.

In our system designated predecessor nodes carry out both detection and conviction while operating as umpires. The performance of each node is monitored by two umpires, hence the term ‘*Double Umpire System (DUS)*’.

The remainder of this paper is organized as follows: Section 2 discusses related work. Section 3 provides an overview of models and assumptions. Section 4 discusses double umpire system models. Section 5 describes an implementation of TUS. Sections 6 and 7 present simulation results and analysis, and Section 8 provides conclusions.

2. Related Works

The Key Distribution Center (KDC) architecture is widely applied in wired networks because of its many merits, including efficient key management (key generation, storage, distribution, and updating). The lack of a Trusted Third Party (TTP) key management scheme is a serious problem in ad hoc networks [7][9].

Kong et al. [8] describe a system for providing ubiquitous services to mobile hosts. In their design they distribute certification authority functions through a threshold secret sharing mechanism, in which each entity holds a secret share and multiple entities in a local neighborhood jointly provide complete services. No single entity in the network knows or holds the complete system secret (e.g. a signing key) [11][12]. Instead, each entity holds a share of the secret key [14][15][16][17]. Multiple entities (k) in a one-hop network locality can jointly provide complete security service in the same manner as provided by a single omnipresent certification authority [18][19].

Yong et al. [10][13] propose a novel cryptography system for ad hoc network security employing a new digital signature algorithm for an identity authentication and key agreement scheme. Their scheme has no central administrator and is capable of withstanding man-in-the-middle and Byzantine mode conspiracy attacks.

Hubaux et al. [16] [17] surveyed threats and possible solutions for ad hoc network security. They extended the idea of public key infrastructure using a system similar to Pretty Good Policy (PGP) in the sense that public key certificates are issued by the users. However, they did not rely on directories for certificate distribution. Two distribution algorithms were presented.

The above schemes only try to protect the system and make no attempt at quarantining attackers [20 - 25]. The twin systems *watchdog* and *pathrater* [2] not only detect mischievous nodes but also prevent their further participation in the network [26][28][29].

SCAN [1] acts in a similar but more comprehensive manner in which not only packet dropping but also other misbehaviors such as reporting misleading hop counts are detected [26 - 38]. Our DUS is an extension of the above two systems.

3. Models and Assumptions

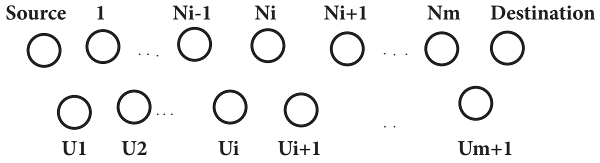
Several assumptions were made in the design of the double umpiring system:

1. The wireless ad hoc network nodes are free to move about or remain still at their discretion.
2. Nodes may fail at any time.
3. There exists a bi-directional communication link between any pair of nodes, which is a requirement for reliable transmission in most wireless MAC layer protocols including IEEE 802.11.
4. The source and the destination node are not malicious.
5. Wireless interfaces support a promiscuous mode of operation.

Promiscuous hearing means that any node A may overhear messages transmitted between a second node B and a third node C as long as B and C are within the communications range of A. Most of the existing IEEE 802.11 based wireless cards support promiscuous operations to improve routing protocol performance.

4. Double Umpiring System Model

In the umpiring system each node is issued a token at inception. The token consists of NodeID and status fields. The NodeID is assumed to be unique and deemed to be beyond manipulation; the status is a single bit flag. The status bit is initially preset to zero, indicating a green flag and conferring upon the node the freedom to participate in all network activities.



Ni-1, Ni, Ni, Ni+1.. Nm intermediate nodes in the active path.
Ui, Ui+1,.. Um+1 corresponding umpires

For Ni umpires Ui and Ui+1

Fig. 1 Double Umpiring System

In order to participate in a network activity such as a Route Request RREQ, a node must announce its token. If the status bit is "1" (indicating a "red flag") the protocol does not permit the node to participate in network activity.

In the double umpiring system (DUS) (Fig. 1) N_{i-1}, N_i, and N_{i+1} are nodes in the active path and U_{i-1}, U_i, and U_{i+1} are corresponding umpires. Each intermediate node in the active path is monitored by two umpires. Thus umpire U_i monitors the behavior of two nodes N_{i-1} and N_i. U_{i+1} monitor N_i and N_{i+1}, and so on. In order to enable this U_i is selected such that it is within the communication range of both N_{i-1} and N_i. Further adjacent umpires may communicate with each other. There are 'm' intermediate nodes and (m+1) umpires.

When a node N_i is found to be misbehaving (dropping packets or changing Hop_count or sequence number), umpire nodes U_i and U_{i+1} send an M-ERROR message to the source and use an M-Flag message to set the status bit of N_i to "1", indicating a red flag.

In our system there is no change in the token – it may be used for the full lifetime of the node if the node continues to behave correctly. At the first offense the status of the guilty node is set to 1, preventing further participation in the network.

We assume that no node can alter its own status bit. Only the designated umpire along the forward or reverse path under consideration may change the status bit. It is also assumed that a node cannot misleadingly transmit its NodeID or status bit.

5. Implementation of DUS

We chose to implement DUS over a traditional AODV protocol, but the principle is applicable to other routing protocols as well. In order to enable such umpiring cross verification, we modified the famous AODV routing protocol, adding a new field (next_hop) in the routing messages so that a node may correlate overheard packets accordingly. Our implementation of DUS is based on three important algorithms. Algorithm 1 describes the DUS route request procedure, while Algorithms 2 and 3 outline the route reply and packet forwarding procedures.

Algorithm 1: While sending a DUS RREQ packet

```
1: for each DUS RREQ packet (P) sent do
2:  if each node status is green flag then
3:   broadcast RREQ
4:   nodeprevhop ← nodecurrenthop [node address]
5:   neighhop1 ← prevhop[node address]
6:   neighhop2 ← nexthop[node address]
7:   repeat steps from 2 to 6 until packet reaches the
         destination node
8:  else
9:     drop umpire RREQ packet (P) sent
10:   endif
11: endfor
```

In the double umpiring system, two umpiring nodes are used to convict the malicious node. All of the nodes participate in both packets forwarding and umpiring. For node N_i, U_i and U_{i+1} will be umpires for the forward and reverse operations. The route reply process (RREP) is described in Algorithm 2.

Algorithm 2: While sending a DUS RREP packet

```
1: for each DUS RREP packet (P) sent do
2:   if node status is green flag then
3:     set designated umpires
4:     neighhop1 ← prevhop [node address]
5:     neighhop2 ← nexthop [node address]
6:     nodenexthop ← nodeprevhop [node address]
7:     unicast RREP to previous node
8:     repeat the steps from step 2 to step 7 until packet
         reaches the source node
9:     if neighhop1 and neighhop2 are equal to
         nodenexthopcount then
10:      process RREP as specified in the standard
protocol
11:    end if
12:  endif
13: endfor
```

Algorithm 3: While sending a DUS data packet

```
for each DUS DATA packet (P) sent do
  if node status is green flag then
    send packet to next forwarded node
    it tampered with the payload or header of the
      currently sent packet
    nodenexthop ← nodecurrentpacketheader
    neighhop1 ← nodecurrentpacketheader
    neighhop2 ← nodecurrentpacketheader
      this header information is kept until the next
      packet is forwarded to the node
  else
    nodenextnode has dropped the packet and
      is a malicious node
    neighhop1 and neighhop2 are umpire nodes
      for next immediate forwarded node
    if neighhop1← nodecurrentpacketheader
      and neighhop2← nodecurrentpacketheader
        is not equal to  prevhop ←
        currentpacketheader
      it has been marked as malicious node
      it broadcast MERR packet to 1-hop or 2-hop
        node distance
      nodenextnode status is marked as red flag
      Umpire nodes send link error message to
        source node
      process the RERR message as specified in
        the standard protocol
    endif
  endif
endfor
```

6.Simulations and Results

We used a simulation model based on QualNet 4.5 in our evaluation [3][4][30]. Our performance evaluations were based on simulations of 100 wireless mobile nodes forming an ad hoc network over a rectangular (1500 X 600 m) flat space. The MAC layer protocol used in the simulations was the Distributed Coordination Function (DCF) of IEEE 802.11 [5]. The performance setting parameters [6] [27] are listed in Table 1.

Before the simulation we randomly selected a certain fraction of the network population ranging from 0% to 40% to be malicious nodes. We considered only two attacks – modifying the hop count and dropping packets. The source and destination of each flow did not change during the simulation run.

6.1 Throughput

In the world of MANET, packet delivery ratio has been accepted as a standard measure of throughput. This is the ratio between the numbers of packets received by the destinations to the number of packets sent by the sources. In Table 2 the packet delivery ratios are presented for DUS in networks containing 0–40% malicious nodes, with node mobility varying between 0 and 20 m/s.

Simulation Time	1500 seconds
Propagation model	Two-ray Ground Reflection
Transmission range	250 m
Bandwidth	2Mbps
Movem ent model	Random way point
Maximum speed	0 - 20 m/s
Pause time	0 seconds
Traffic type	CBR (UDP)
Payload size	512 bytes
Number of flows	10/20

Table 1 Parameter Settings

Mobility (m/s)	Percentage of Malicious nodes				
	0 %	10 %	20 %	30 %	40 %
0	98.28	84.96	78.92	72.84	65.08
5	96.28	77.33	70.17	64.18	48.56
10	95.10	76.16	67.78	61.11	47.36
15	94.12	75.42	66.54	60.88	43.73
20	93.73	73.18	65.96	59.18	41.99

Table 2 Packet delivery ratios for DUS

The following conclusions may be drawn from the information in table 2:

1. The packet delivery ratio decreases as the mobility and percentage of malicious nodes increase.
2. In the case of 10% malicious nodes, the packet delivery ratio dropped from 84.96% when the nodes were stationary to 73.18%, when the nodes were moving at 20 m/s. In the presence of 30% and 40% malicious nodes with 20 m/s mobility, the corresponding values are 59.18 and 41.99%.
3. The DUS has low throughput because it has very high security.
4. In general, the packet delivery ratio decreases as the security increases.

From the above results we conclude that SCAN offers a substantial improvement over DUS, from the standpoint of throughput.

6.2 Failure to deduct (False Negative) Probability

False negative probability may be defined as the number of malicious nodes left undetected divided by the total number of malicious nodes.

From table 3 the following conclusions may be drawn:

1. In general false negative probability increases as the percentage of malicious nodes increases.
2. In the case of 20% malicious nodes when the nodes are moving at 20 m/s, the false negative probability is high because the number of malicious nodes in a particular area is high.

Mobility (M/s)	Percentage of Malicious nodes				
	0%	10 %	20 %	30 %	40 %
0	0	0.1111	0.1816	0.1852	0.1951
5	0	0.0916	0.1544	0.1513	0.1502
10	0	0.0601	0.0584	0.0749	0.0742
15	0	0.0716	0.0938	0.0988	0.0973
20	0	0.0833	0.1154	0.0918	0.1012

Table 3. False negatives for DUS

From the above results we conclude that DUS has a lower false negative probability than SCAN.

6.3 False Accusation (False Positives) Probability

The false positive probability increased with increasing mobility, varying from 0 to 0.0911. The results are similar to the patterns obtained for SCAN [2].

Mobility (M/s)	Percentage of Malicious nodes				
	0%	10 %	20 %	30 %	40 %
0	0	0	0	0	0
5	0	0.0101	0.0102	0.0116	0.0221
10	0	0.0201	0.0401	0.0471	0.0572
15	0	0.0322	0.0612	0.0692	0.0714
20	0	0.0618	0.0698	0.0748	0.0911

Table 4. False positives for DUS

The lowest false positive probability was obtained with DUS, and innocent node conviction occurred least frequently with DUS.

6.4 Communication Overhead

Communication overhead may be evaluated based on the number of control message transmissions such as RREQ, RREP, and RERR in the case of plain AODV and additionally M_ERROR and M-Flag messages in the double umpiring system. RREQ messages are disseminated to the entire network, whereas RREP messages are unicasts. We present the communication overhead details in table 5.

Mobility (M/s)	Percentage of Malicious nodes				
	0 %	10 %	20 %	30 %	40 %
0	9046	11406	13753	15305	16756
5	9618	12173	14824	16160	17608
10	10025	12989	15617	17046	18515
15	10576	13911	16516	17936	19685
20	11998	14520	17221	18642	20473

Table 5. Communication overhead for DUS

Communication overhead increased with mobility. DUS had a lower communication overhead than SCAN.

7. Analysis of Results

The plain AODV, SCAN and DUS results are presented in table 6.

Mobil (M/s)	Malicious node = 30 %		
	Plain AODV	SCAN	DUS
0	70.44	90	72.84
5	45.18	85	64.18
10	37.89	83	61.11
15	32.55	81	60.88
20	32.07	80	59.18

Table 6. Throughput for plain AODV, SCAN, and DUS.

SCAN provided much higher output compared to plain AODV or DUS. The increase in communication overhead ranged from 8.26% (plain AODV, 0 m/s mobility) to 15.91% (plain AODV 20 m/s mobility). The corresponding values for SCAN ranged from 11 to 28%.

Clearly the greater number of umpires involved in detection in DUS decreases the probabilities for false negatives and false positives, and DUS provides better rounding up of malicious nodes at a very low cost in communication overhead.

8. Conclusions

A double umpiring system for mobile ad hoc network security was proposed. We presented experimental results for plain AODV, SCAN, and DUS systems. The results of simulations indicate that even if 40% of the nodes are malicious and the maximum speed is 20 m/s, DUS throughput is increased by 161.25% while communication overhead is increased by only 18.5%. This is a vast improvement over the performance of the conventional AODV protocol with only nominal additional overhead. We envisage that our system may be profitably used in civilian situations where nodes are invariably lean and energy starved. Further research work is in progress.

Acknowledgements

We express our thanks to Mrs. Meenakshi Annamalai, the Director, and Dr. Illangovan, the Principal, Karpaga Vinayaga College of Engineering and Technology, Chennai, Tamilnadu, India, for the encouraging environment provided.

References

[1] Hao Yang, James Shu, Xiaoqiao Meng and Songwu Lu, "SCAN: Self-Organized Network-Layer Security in Mobile ad hoc networks", IEEE Journals on selected areas in communications, vol. 24, No. 2, February 2006.

[2] Sergio Marti, T.J. Giuli, Kevin Lai and Mary Baker, "Mitigating routing misbehavior in mobile ad hoc networks", in proc. ACM MobiCom, 2000, pp- 255-265.

[3] L. Bajaj, M. Takai, R. Ahuja, R. Bagrodia, and M. Gerla, "Glomosim: A scalable network simulation environment. Technical Report 990027, 1999.

[4] Scalable Networks Technologies: QualNet simulator version 4.5. http://www.scalable-networks.com

[5] IEEE 802.11. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, August, 1999.

[6] A.Kathirvel and R.Srinivasan, "Performance Enhancement on demand routing protocol in mobile ad hoc networks", in Proc. Second National Conference, PSG tech, 2006.

[7] Marianne A. Azer, Sherif M. El-Kassas, and Magdy S. El-Soudani, "Certification and revocation schemes in ad hoc networks survey and challenges, in proc. IEEE ICSNC 2007.

[8] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, "Providing robust and ubiquitous security support for MANET", in Proc. IEEE ICNP, 2001, pp. 251-260.

[9] Lei Feng-Yu, Cui Guo-Hua, and Liao Xiao-Ding, "Ad hoc Networks security mechanism based on CPK", in proc. IEEE ICCISW, 2007, pp. 522 – 525.

[10] Pi Jian Yong, Liu Xin Song, Wu Ai, Liu Dan, "A Novel Cryptography for Ad Hoc Network Security", in Proc. IEEE 2006, pp. 1448 -1451.

[11] Michael Hauspie, and Isabelle Simplot-Ryl, "Enhancing nodes cooperation in ad hoc networks", in proc. IEEE 2007, pp. 130 – 137.

[12] S. Capkun, L. Buttyan and J. Hubaux, "Self-organized public-key management for mobile ad hoc networks", IEEE Trans. Mobile Computing, vol. 2, No. 1, pp. 52-64, January, 2003.

[13] Pi Jian Yong, Liu Xin Song, Wu Ai, Liu Dan, "A Novel Cryptography for Ad Hoc Network Security", in Proc. IEEE 2006, pp. 1448 -1451.

[14] J. Hubaux, L. Buttyan, and S. Capkun, "The quest for security in Mobile ad hoc networks", in Proc. ACM MobiHoc, 2001, pp. 146-155.

[15] William Stallings, "Cryptography and network Security principles and Practices", Pearson Education, First edition, 2007.

[16] Y. Zhang and W. Lee, "Intrusion detection in wireless ad hoc networks", in Proc. ACM MobiCom, 2000, pp. 275-283.

[17] J. Hubaux, I. Buttyan and S. Capkun, "The quest for security in mobile ad hoc networks", in Proc. ACM MobiHoc 2001, pp. 251-260.

[18] S. Capkun, J.Hubaux, and L. Buttyan, "Mobility helps security in ad hoc networks", in Proc. ACM MobiCom, 2003, pp 46-56.

[19] Y. Hu, D. Johnson, and A. Perrig, "SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks", in Proc. IEEE WMCSA, June 2002, pp. 3-13.

[20] Y. Hu, A. Perrig, and D. Johnson, "Ariadne: A Secure on-demand routing for ad hoc networks", in Proc. ACM MobiCom, 2002, pp. 12-23.

[21] P. Papadimitratos and Z. Haas, "Secure routing for mobile ad hoc networks", in Proc. CNDS, 2002, pp. 193-204.

[22] K. Sanzgiri, B. Dahill, B. Levine, C. Shields, and E. Royer, "A secure protocol for ad hoc networks," in Proc. IEEEICNP, 2002, pp. 78-89.

[23] M. Zapata and N. Asokan, "Securing ad hoc routing protocols", in Proc. ACM Wise, 2002, pp.1-10.

[24] Azeddine Attir, Farid Nait Abdesselem, Brahim Bensaou, and Jalel Ben-Othman, "Logical Wormhole Prevention in Optimized Link State Routing Protocol", in proc IEEE GLOBECOM 2007, pp. 1011 – 1016.

[25] Nidal Nasser and Yunfeng Chen, "Enhanced Intrusion Detection System for discovering malicious nodes in mobile ad hoc networks", in proc. IEEE ICC, 2007, pp. 1154- 1159.

[26] Kathirvel A. and Srinivasan R. (2009a), 'A Study on Salvaging Route Reply for AODV Protocol in the Presence of Malicious Nodes', International Journal of Engineering and Technology, Vol. 1, No. 2, pp. 151-155.

[27] Kathirvel A. and Srinivasan R. (2009b), 'Performance Analysis of Propagation Model using Wireless Mobile Ad hoc Network Routing Protocols', International Journal of Wireless Communication, September 2009.

[28] Kathirvel A. and Srinivasan R. (2009c), 'Single Umpiring System for Security of Mobile Ad Hoc Networks', Journal of Advances in Wireless Mobile Communication, Vol. 2, No. 2, pp. 141-152.

[29] Kathirvel A. and Srinivasan R. (2009d), 'Triple Umpiring System for Security of Mobile Ad Hoc Networks', International Journal of Engineering and Information technology, Vol. 1, No. 2, pp 95-100.

[30] Kathirvel A. and Srinivasan R. (2009e), 'Global Mobile Information System Simulator in Fedora Linux', ACM on line Computer Communication Review, 2009.

[31] Kathirvel A. and Srinivasan R. (2010a), 'Enhanced Self Umpiring System for Security using Salvaging Route Reply', International Journal of Computer Theory and Engineering, Vol. 2, No. 1, 2010, pp. 129 – 134.

[32] Kathirvel A. and Srinivasan R. (2010b), 'Enhanced Triple Umpiring System for Security and Performance Improvement of Wireless MANETs', International Journal of Communication Networks and Information Security, Vol. 2, No. 2, 2010, pp. 77 – 84.

[33] Kathirvel A. and Srinivasan R. (2010c), 'Self_USS: A Self Umpiring System for Security in Mobile Ad-Hoc Network', International Journal of Engineering and Technology, Singapore, Vol. 2, No. 2, 2010, pp. 196 – 203.

[34] Kathirvel A. and Srinivasan R. (2010d), 'A System of Umpires for Security of Wireless Mobile Ad Hoc Network', International Arab Journal of e Technology , Vol. 1, No. 4, 2010, pp 129 – 134.

[35] A.Kathirvel, and Dr. Rajabushanam, "Survey of Wireless Manet Application in battlefield operations", International Journal of Advanced Computer Science and Application, Vol. 2, No. 1, January 2011.

[36] A.Kathirvel, and Dr. R. Srinivasan, "ETUS: An Enhanced Triple Umpiring System for Security and Performance Improvement of Mobile Ad Hoc Networks", International Journal of Network Management, John Wiley & Sons, 2011. (Online version published)

[37] A.Kathirvel, and Dr. R. Srinivasan, "ETUS: An Enhanced Triple Umpiring System for Security and Robustness of Mobile Ad Hoc Networks", International Journal of Communication Networks and Distributed Systems, Inderscience, 2011. (Online version published)

Dr. A.Kathirvel was born in 1976 in Erode, Tamilnadu, India. He received his B.E. degree from the University of Madras, Chennai, in 1998, his M.E. degree from the same University in 2002 and Ph.D from Anna University, Chennai, in 2010. He is currently Professor and Head of the Department at Karpaga Vinayaga College of Engineering and Technology, Chennai, Tamilnadu, India in the Department of computer science and Engineering. He is a member of the ISTE, ACM, IETF, IACSIT and IAENG. His research interests are protocol development for wireless ad hoc networks and security in ad hoc networks.