SELECTIVE WATCHDOG TECHNIQUE FOR INTRUSION DETECTION IN MOBILE AD-HOC NETWORK

DEEPIKA DUA AND ATUL MISHRA

Department of Computer Engineering, YMCA University of Science & Technology, Haryana, India

ABSTRACT

Mobile ad-hoc networks(MANET) is the collection of mobile nodes which are self organizing and are connected by wireless links where nodes which are not in the direct range communicate with each other relying on the intermediate nodes. As a result of trusting other nodes in the route, a malicious node can easily compromise the security of the network. A black-hole node is the malicious node which drops the entire packet coming to it and always shows the fresh route to the destination, even if the route to destination doesn't exist. This paper describes a scheme that will detect the intrusion in the network in the presence of black-hole node and its performance is compared with the previous technique. This novel technique helps to increase the network performance by reducing the overhead in the network.

KEYWORDS

MANET, Intrusion, Intrusion Detection system, Attacks

1. INTRODUCTION

Mobile ad-hoc network is formed by the collection of some mobile nodes which can act both as a sender as well as receiver for data communication. They are decentralized networks which are self organizing and self maintaining. There is no fixed infrastructure in the network, the topology changes dynamically [1]. As a result of continuously changing topology, there is no fixed boundary of the network. The nodes cooperate with each other to forward the data packet. In such a network where there is no well-defined boundary, open medium, nodes rely on one other to forward the data packet, firewalls cannot be applied for securing these networks. Intrusion detection system [2] is used in these networks to detect the misbehaviour in the network. Intrusion detection system acts as a second layer in mobile ad-hoc networks [3]. Intrusion detection system can be network based [4] or host based [5] on the basis of the audit data collection or it can be signature based, anomaly based or specification based on the basis of the detection technique [6].

In this paper, a scheme is proposed which detects the misbehaving nodes in the network in the presence of black-hole attack [7] [8] and reduces the network overhead.

Rest of the paper is organized as follows. In section 2, literature survey is presented. In section 3, scheme description is present, the methodology used is described. In section 4, simulation environment and results of the simulation are presented. And finally conclusion is presented in

DOI:10.5121/jgraphoc.2014.6302

section5.

2. RELATED WORK

Marti el al [9] proposed a scheme named Watchdog which is a reputation based scheme [10], in which after detecting the malicious node, information is propagated throughout the network so to avoid that node in future routes.

The watchdog scheme works in two parts-in the first part the watchdog detects the malicious node by promiscuously listening to its next neighbour's transmission. If a node doesn't forward the packet after a threshold, then watchdog declares that node as malicious. And then the path rater finds the new route to the destination excluding that malicious node. In this scheme malicious node is detected instead of malicious link there are six weaknesses that are mentioned by Marti [9]. They are 1)Receiver Collision problem 2)Ambiguous collision 3)Limited Transmission power 4) False misbehaviour 5) collusion 6) Partial Dropping.

Liu at al [10] proposed a scheme named TWOACK, which detects the misbehaving links in the ad-hoc network instead of misbehaving nodes. It is an acknowledgement based scheme in which every third node in the route from sender to receiver requires to send an acknowledgement packet to the first node down the reverse route.

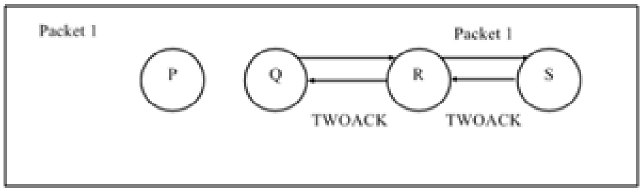


Figure 1.TWOACK

In figure 1, node Q sends a packet to node R which further forwards it to node S. When node S receives the packet and as it is third node in the path, it will send a TWOACK packet to the node Q acknowledging that it receives the packet successfully. All the nodes in the path work in the similar way.

It solves the receiver collision problem and limited power problem of the watchdog scheme. But due to the exchange of too many acknowledgement packets, this scheme consumes too much battery power and hence can degrade the network performance. It works on DSR (Dynamic source routing) protocol.

Sheltami et al. [11] proposed a scheme named Adaptive acknowledgment (AACK) which is based on TWOACK scheme. This scheme also works on DSR routing protocol. It is an advancement of the TWOACK scheme. It reduces the battery consumption by making the scheme a combination of end-to-end acknowledgement and TACK, which is similar to TWOACK. When the sender sends a data packet to destination, it waits for some time for the destination to acknowledge that data packet, but if the acknowledgement doesn't come within per-defined time, then it switches to TACK mode, where every third node sends the TACK packet to the nodes two hops away from it down the route.

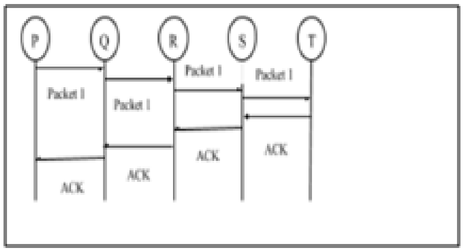


Figure 2: ACK scheme

Figure 2 shows the ack scheme in which source P waits for the destination T to acknowledge the data packet. If acknowledgment from T doesn't receive within specified time, then it switches to TACK mode by sending TACK packet.

Nidal Nasser and Yunfeng Chen [12] proposed an approach called Ex-Watchdog. It was basically an improvement over the Watchdog scheme proposed by Marti [9].Out of the six weaknesses mentioned in the Watchdog [9] scheme it solves the false misbehaviour problem. In this scheme, each node maintains a table having entries of source address, destination address, and the statistics of the packets received, forwarded and stored. If any node reports a node as being misbehaving, then instead of trusting that node immediately, a new route to destination is found excluding the reported malicious node and number of packets received is checked at the destination node. If it is equal to the number of packets sent, then it is a false misbehaviour report and whosoever generated is declared as malicious .Then, pathrater or routeguard cooperated with the routing protocol and update the rating of node in their corresponding tables. This scheme fails to detect the misbehaviour when the misbehaving node is in all the routes from source to destination.

Elhadi M. Shashuki, Nan Kang and Tarek R. Sheltami [13] proposed an approach called EAACK (Enhanced AACK) which solves receiver collisions problem, limited battery problem and false misbehaviour problem of the watchdog scheme. It is also an acknowledgement based scheme and to protect the acknowledgement packet from forging, this scheme makes use of digital signature. It is composed of three parts:-

ACK- It is an end-to-end acknowledgement as described in AACK scheme. Sender waits for the destination to acknowledge data packets but if the acknowledgement doesn't come within a specified time, then it switches to S-ACK mode.

S-ACK- In this mode, similar to TWOACK scheme, consecutive node works in a group i.e. every third node sends an S-ACK packet to its first node which is in the reverse directions. The difference between the S-ACK and TWOACK is that TWOACK immediately trusts the misbehaviour report and declares the node as malicious. But in this scheme, we switch to MRA mode to confirm the misbehaviour report.

MRA- It stands for misbehaviour report authentication. This mode cooperates with the routing protocol to find a new route to the destination which excludes the reported misbehaviour node. Destination is checked for the data packet using the new route. If the data packet is found at the destination, then it is a false misbehaviour report and the node which generated this report will be declared as malicious, else the misbehaviour report is trusted and the node would be declared as misbehaving.

3. PROPOSED APPROACH

We proposed an algorithm that detects the intrusion in the presence of black hole node in the network. The proposed technique is an improvement over the Watchdog technique[9].In Watchdog each node continuously hears its next node transmission but in the proposed selective Watchdog technique only when the acknowledgment would not be received ,then IDS would start.Morever,in watchdog[9] technique all nodes monitor their neighbours but in proposed selective watchdog technique ,network of nodes are divided into clusters and only nodes in the cluster which have value greater than threshold monitor their neighbours. The pseudo code for the black-hole attack is shown in the algorithm. The input parameters for the algorithm are set of all the nodes, a threshold value which gets updated dynamically, source node, destination node and all the nodes which send the route reply to the source node.

The algorithm works as follows:-

The source waits for the destination to send acknowledgement to it after every 10th packet. If source receives the acknowledgement, then there is no misbehaviour in the network and process continues as such. But if the destination fails to acknowledge the data packets for a time period, then IDS starts its functionality.

As in black-hole attack, there is a greater possibility that black-hole node will send the highest sequence number to the source in route reply. The proposed IDS algorithm maintains the list of all the nodes which send the route reply to the source with sequence number greater than the threshold value.

The IDS will be applied only on those nodes which are in the list maintained by ids.

Algorithm 1: Algorithm for detecting IDS

Input: Threshold_seq_no. Set_of_all_nodes; Set_of_nodes_who_sent_route_reply; source; destination.

```
Begin
If(pkt_received_by_dest==pkt_sent_by_source) then
            network does not shows any malicious behaviour
Else if(pkt_received_by_dest < certain percentage of pkt sent by source over the network)
{
  Then the network shows malicious behaviour and IDS is applied to detect malicious behaviour
For(int i=0; i<no._of_nodes_who_sent_route_reply; i++)
{
        If(seq_no[route_reply[Node]] > Threshold_seq_no) then
 List. add(next[Node])
 List. add(Node)
 List. add(prev[Node])
result=  Segment_watchdog(List);
 If(result==true)     then //(i.e. if malicious node is found)
                  Exit;
  ENDIF
   Else
        Continue
  EndElse
}
}
```

Algorithm: **Segmented_Watchdog (List)**

```
BEGIN
Result=false
malicious= Null
Node1= list. get(0)
 Node2= list. get(1)
Node3=list. get(2)
//Chk(Sent_pkt[Node2]);
If(sent_pkt[Node2] == Received_pkt_by_node2) THEN
     Monitor Node3
ENDIF
If(Sent_pkt_by_Node3==Received_pkt_by_Node3) THEN
 No malicious activity detected in this segment
        RETURN Result
END IF
Else if(Sent_pkt_by_Node1 < Received_pkt_by_Node1) THEN
            Malicious= Node1
             Result= True
             RETURN Result
END ELSEIF
 Else if(Sent_pkt_by_Node2 < Received_pkt_by_Node2) THEN
               Malicious= Node2
               Result=True
               RETURN Result
END ELSEIF
Else if(Sent_pkt_by_Node1 < Received_pkt_by_Node1) THEN
                Malicious= Node1
                 Result=True
             RETURN Result
END ELSEIF
END
```

For every node in the list, segment watchdog method gets called. In this method, the number of packets send and received by the node is checked. If number of send and received are equal, then its successor node in the route is checked else its predecessor node in the route is evaluated in the same way.

The Flowchart of the technique is shown in figure3

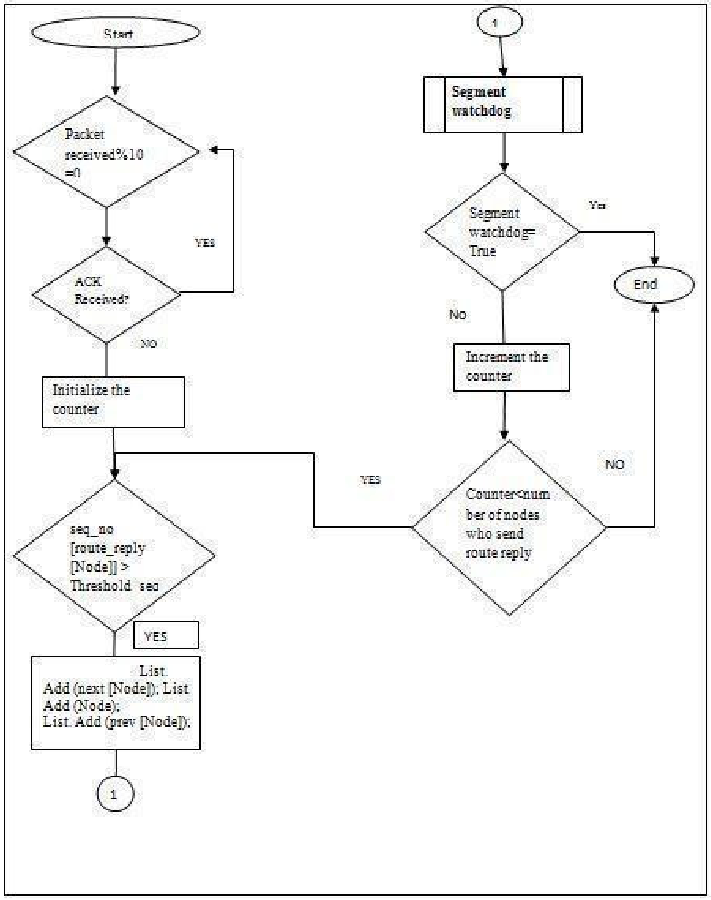

Start
Packet received%10 =0
YES
ACK Received?
NO
Initialize the counter
seq_no [route_reply [Node]] > Threshold seq
YES
List. Add (next [Node]); List. Add (Node); List. Add (prev [Node]);
1
1
Segment watchdog
Segment watchdog= True
Yes
End
No
Increment the counter
NO
YES
Counter<number of nodes who send route reply

Figure 3(a). Flowchart of proposed technique

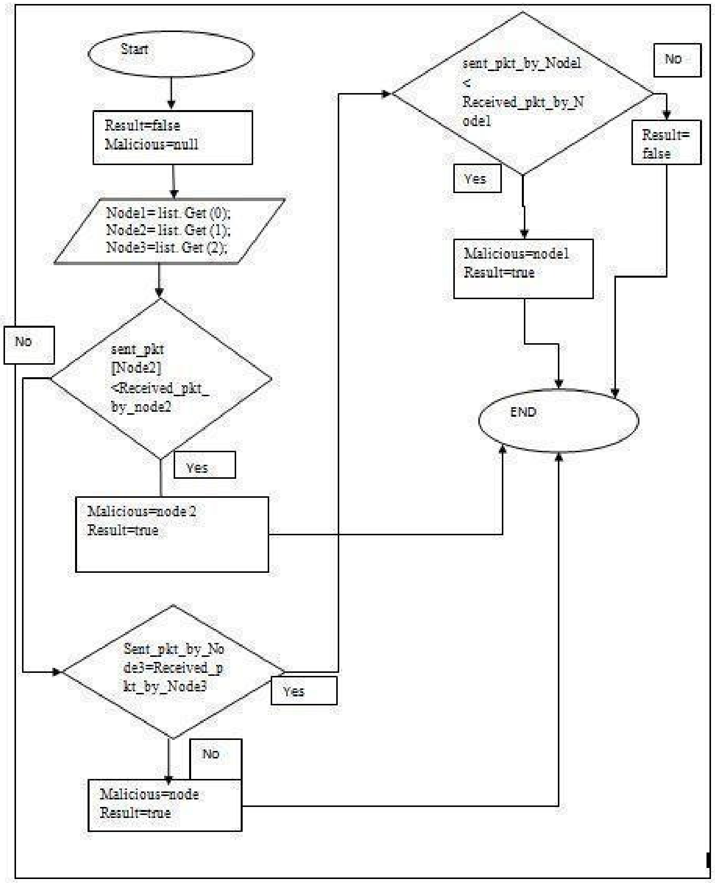


Figure 3(b). Flowchart of selective Watchdog

4. SIMULATION RESULTS

4.1. Assumptions

- We have assumed the bi-directionality in the links.
- Secondly, we have assumed that both the sender and receiver are trusted nodes, i.e. they are non-malicious.
- Duplicate MAC address doesn't exist.
- Lastly we have assumed that the nodes can overhear the transmission of their immediate neighbours.

4.2. Simulation Configuration

The Simulation is carried out using the tool Network Simulator 2 (NS-2) version 2.35 on Linux operating system Ubuntu version 12.10.The system runs on a laptop with Core 2 Duo T6500 processor with 4-GB RAM. For plotting graph, trace-graph version 202 is used.

- **Grid Size:** 500x500

- **Number of Nodes:** 10 of which 5 were communicating
- **Packet traffic:** CBR (Constant bit rate) on UDP
- **Packet Size:** 512B
- **Packet Interval:** 0.25
- **Routing Protocol:** AODV

4.3. Simulation Scenarios

To simulate our result we have taken two scenarios.

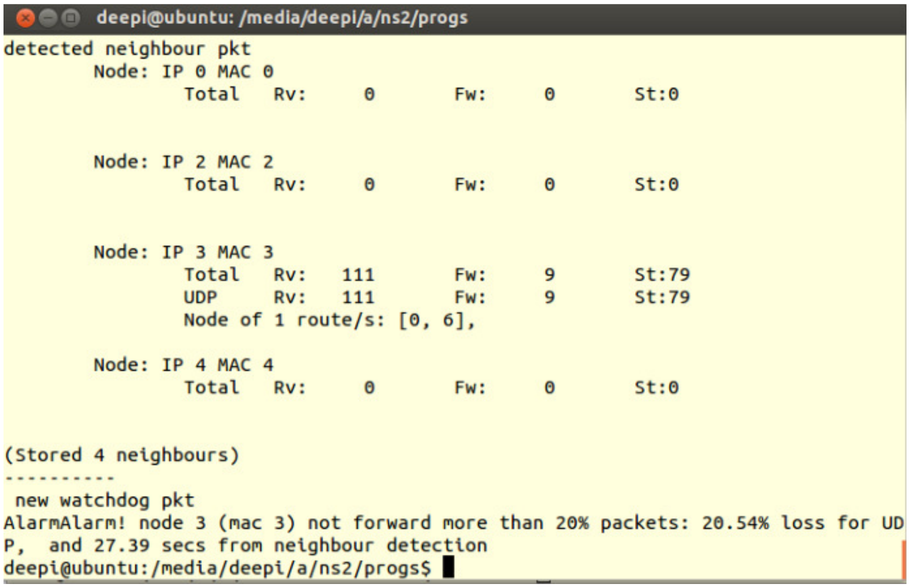
Scenario 1: In this scenario, Watchdog technique is implemented with one malicious node in path between source and destination.

Scenario 2: In this, Proposed Technique is implemented with same parameters taken in scenario 1.

4.4. Performance Evaluation

Scenario 1

In the first case, watchdog technique is implemented with a malicious node between the source and destination. Figure 4 shows the results that it detects the misbehaviour in the network of 10 nodes in 27.39 sec of neighbour detection.



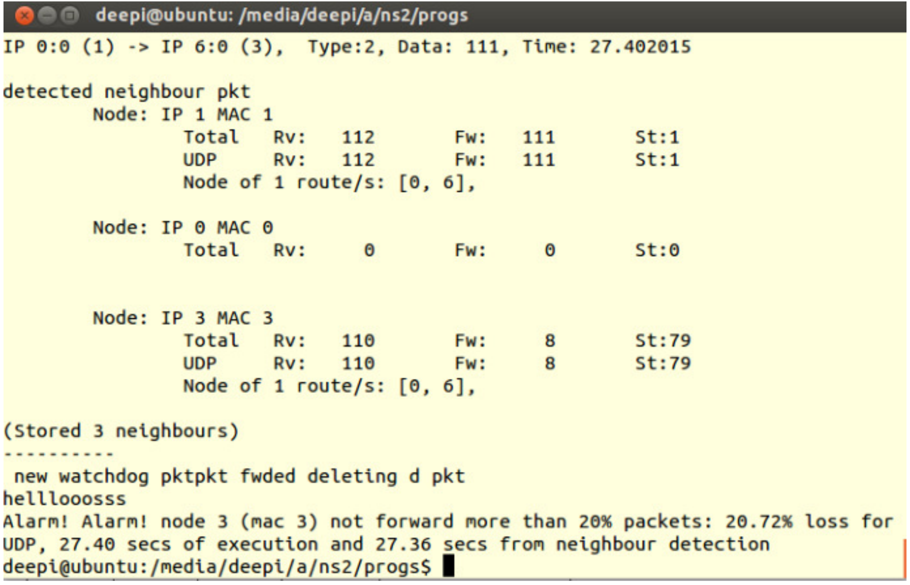
```
detected neighbour pkt
        Node: IP 0 MAC 0
                Total   Rv:     0       Fw:     0       St:0


        Node: IP 2 MAC 2
                Total   Rv:     0       Fw:     0       St:0


        Node: IP 3 MAC 3
                Total   Rv:   111       Fw:     9       St:79
                UDP     Rv:   111       Fw:     9       St:79
                Node of 1 route/s: [0, 6],

        Node: IP 4 MAC 4
                Total   Rv:     0       Fw:     0       St:0


(Stored 4 neighbours)
----------
 new watchdog pkt
AlarmAlarm! node 3 (mac 3) not forward more than 20% packets: 20.54% loss for UD
P,  and 27.39 secs from neighbour detection
deepi@ubuntu:/media/deepi/a/ns2/progs$ █
```

Figure 4: Screenshot of Watchdog Technique

Scenario 2

In this scenario, proposed Selective Watchdog technique is implemented and then results are compared with the results of scenario 1.

Figure 5 show that it took 27.36 sec for our scheme to detect the intrusion in the network.



```
IP 0:0 (1) -> IP 6:0 (3),  Type:2, Data: 111, Time: 27.402015

detected neighbour pkt
        Node: IP 1 MAC 1
                Total   Rv:   112       Fw:   111       St:1
                UDP     Rv:   112       Fw:   111       St:1
                Node of 1 route/s: [0, 6],

        Node: IP 0 MAC 0
                Total   Rv:     0       Fw:     0       St:0


        Node: IP 3 MAC 3
                Total   Rv:   110       Fw:     8       St:79
                UDP     Rv:   110       Fw:     8       St:79
                Node of 1 route/s: [0, 6],

(Stored 3 neighbours)
----------
 new watchdog pktpkt fwded deleting d pkt
hellloooss
Alarm! Alarm! node 3 (mac 3) not forward more than 20% packets: 20.72% loss for
UDP, 27.40 secs of execution and 27.36 secs from neighbour detection
deepi@ubuntu:/media/deepi/a/ns2/progs$ 
```

Figure 5: Screenshot of proposed technique

Figure 6 shows the graph of comparison between the proposed scheme and the watchdog scheme. From the graph, it is clearly shown that the proposed scheme performs better than the watchdog scheme in terms of detection time to detect the intrusion in the network.

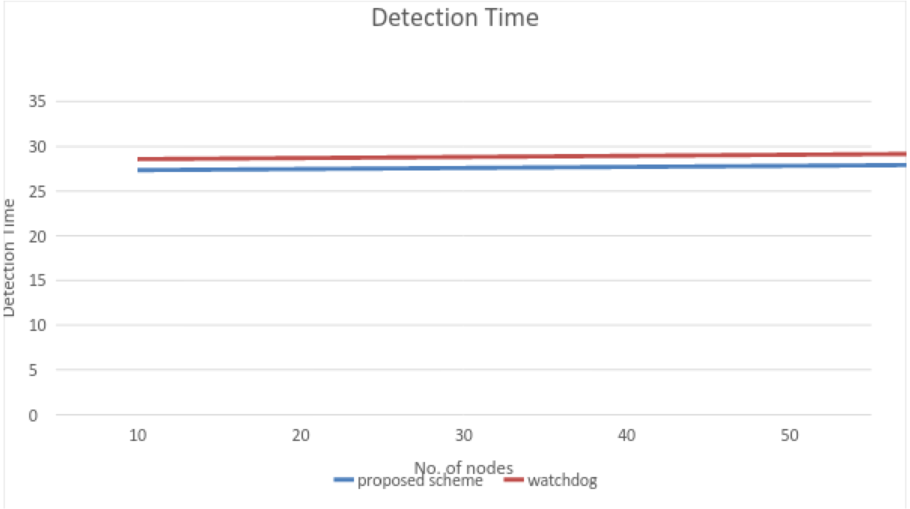


Figure6 Detection Time Comparison of proposed scheme and watchdog scheme

- **Quantitative Analysis**

For a network of n nodes, Watchdog scheme have n-2 promiscuous listening. As every node have to monitor its next neighbour except the source which is not monitored by any node and the destination which will not monitor any node?

For our proposed Selective Watchdog scheme, each cluster is of size say l, suppose we break the network of n nodes into K number of Clusters where K<<n i.e. n/l.

Let say a threshold value of T qualifies n/l*1/t, where t is a qualifier and its value will determine the number of clusters to be checked.

Promiscuous listening in a cluster of size l would be (l-2) in case both source and destination are included in the cluster and it would be l in other cases.

Total promiscuous listening in proposed study is l*(n\l-2) + 2*(l-2)

This formula calculates the number of promiscuous listening and it is for only one data packet. Varying the number of cluster size and number of nodes taken, we can get different number of promiscuous listening.

Table 1 shows the different values taken using the above formula. For n=12,l=3,the number of promiscuous listening in Watchdog technique is 10 and in proposed technique is 8.Similarly,for other values shown in table , number of promiscuous listening is calculated.

Table 1. Promiscuous listening with varying number of nodes and cluster size

	N=12	N=24	N=36
L=3	8	20	32
L=4	7	16	25
L=6	8	14	20
Watchdog Technique	10	22	34

Figure 7 shows the graph for number of promiscuous listening for the proposed approach and the watchdog technique, plotted using the data provided in the table 1. The graph is plotted by varying the number of nodes and the size of cluster taken for each case.

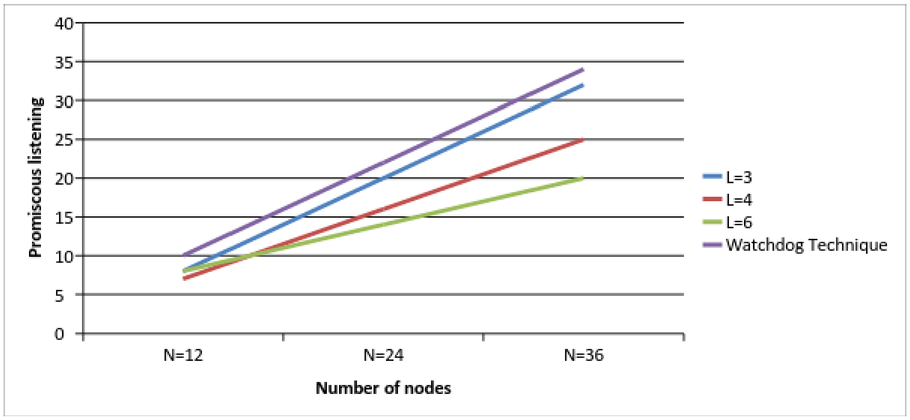


Figure 7 Number of nodes Vs Promiscuous Listening for proposed scheme

- **Experimental Analysis**

Number of nodes and size of cluster is varied and values are calculated by simulation. Table 2 shows the result of simulation.

Table 2. Experimental value of promiscuous listening

	N 12	N 24	N 36
L=3	238	580	900
L=4	234	536	779
L=6	220	448	589
Watchdog Technique	1109	2230	3689

From these values, the graph is plotted. Figure 8 show the graph plotted using the above values.

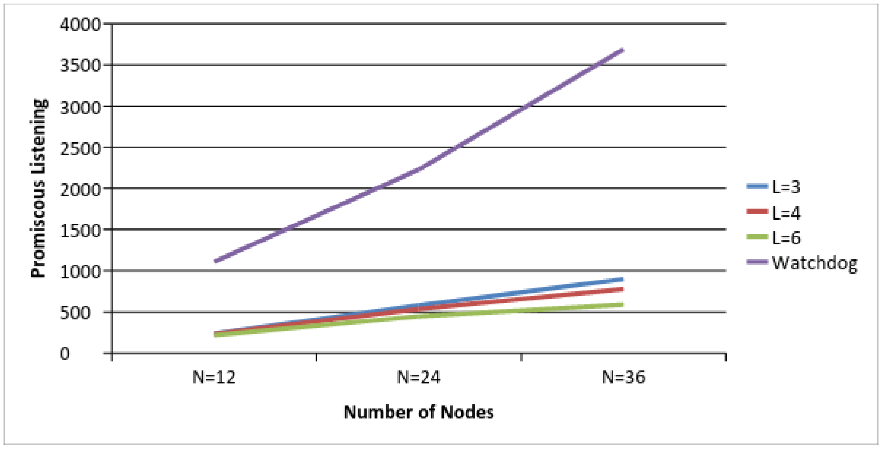
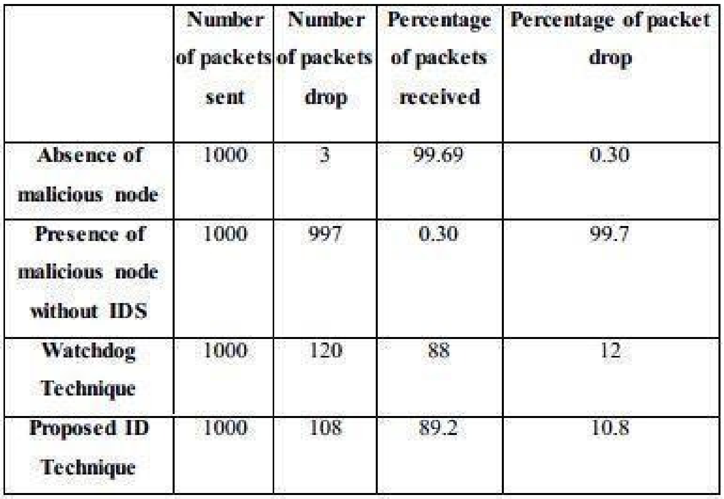


Figure 8 Number of nodes vs. Promiscuous listening

Table 3 shows the statistics of the number of packets sent, number of packets received and percentage of packets received and drop in all three scenarios i.e. in absence of malicious node, in its presence without IDS and with IDS. The statistics shows that with the presence of IDS in the network, the network performance gets improved.

Table 3: Statistics of simulation data



	Number of packets sent	Number of packets drop	Percentage of packets received	Percentage of packet drop
Absence of malicious node	1000	3	99.69	0.30
Presence of malicious node without IDS	1000	997	0.30	99.7
Watchdog Technique	1000	120	88	12
Proposed ID Technique	1000	108	89.2	10.8

5. CONCLUSION

Security is the major concern in the ad-hoc networks as nodes can be easily captured or compromised. Black-hole attack drops all the packets going through the malicious node. As a result network performance decrease drastically. The proposed scheme detects the intrusion in the presence of black-hole attack in the network and then routes the packets through secured path. The results obtained in various test scenarios suggest that proposed selective technique is better than conventional Watchdog technique in terms of time to detect the intrusion and number of promiscuous listening amongst the neighbours. The threshold reference removes many promiscuous listening as a big set of node lying under the value are not subjected to any detection related messages listening and associated networking cost. The graphs further show that making the cluster and starting the IDS only when acknowledgement not received further improves the network throughput as there would be further less network overhead. The proposed technique scales well with the increase in network size as shown in result graphs. A mathematical model capturing the costs discussed in the paper has been presented with results matching the experimental data.

REFERENCES

[1] T. Anantvalee and J. Wu, “A Survey on Intrusion Detection in Mobile Ad Hoc Networks,” in Wireless/Mobile Security. New York: Springer-Verlag, 2008

[2] Giovanni Vigna and Richard A. Kemmerer, “NetSTAT: A network-based intrusion detection system ,”, Journal of computer security,1999,pp.37-71.

[3] B. Sun, “Intrusion detection in mobile ad hoc networks,” Ph.D. dissertation, Texas A&M Univ., College Station, TX, 2004.

[4] David Wagner and Paolo Soto ,”Mimicry Attacks on Host-Based Intrusion Detection Systems”,in ACM, Nov.2002

[5] Giovanni Vigna and Richard A. Kemmerer, “NetSTAT: A network-based intrusion detection system,”, Journal of computer security,1999,pp.37-71.

[6] D. Sterne1, P. Balasubramanyam2, D. Carman1, B. Wilson1,R. Talpade3, C. Ko1,R. Balupari1, C-Y. Tseng2, T. Bowen3, K. Levitt2 and J. Rowe2 "A General Cooperative Intrusion Detection Architecture for MANETs.

[7] Arun Kumar. R, Abhishek M. K et.al. ,"A Review on Intrusion Detection Systems in MANET,",in International Journal of Engineering Science and Innovative Technology,vol.2,pp 609-618,March 2013

[8] Deepika Dua and Atul Mishra , "Intrusion Detection in Mobile Ad-hoc Network, " in International Journal of Advanced Research in Computer and Communication Engg. ,vol.2,no.2,pp.5691-5694.

[9] S. Marti, T.J.Giuli, K.Lai, and M.Baker,"Mitigating routing misbehavior in mobile ad hoc networks,"in Proc. 6Th Annu.Int. Conf. Mobile Comput. Netw., Boston, MA, 2000, pp.255-265.

[10] K.Liu, J.Deng, P.K.Varshney, and K.Balakrishnan , "An acknowledgment-based approach for the detection of routing misbehaviour in MANETs,"IEEE Trans Mobile Comput.,vol.6,no.5,pp.536-550 ,May 2007

[11] T.Sheltami et al. "AACK:Adaptive Acknowledgment Intrusion Detection for Manet with Node detection Enhancement, "in Proc. 24th IEEE International Conference on Advanced Information Networking and Applications,2010,pp 634-640

[12] N.Nasser and Y.Chen,"Enhanced Intrsuion detection system for discovering mailicious nodes in mobile ad hoc network,"in Proc.IEEE Int. Conf. Commun.,Glasgow,Scotland,Jun. 24-28 ,2007 , pp.1154-1159

[13] Elhadi M. Shashuki, Nan Kang and Tarek R. Sheltami"EAACK-A Secure Intrusion-Detection System for MANET's",in Proc. IEEE Transactions On Industrial Electronics,Vol 60,No.3,March 2013,pp 1089-1098

Authors

Deepika Dua[1] received her Mtech (Computer Engineering-networking) from YMCA University of Science and Technology, Faridabad in the year 2014.Her research interests include mobile ad-hoc networks.

Atul Mishra2 is working as an Associate Professor in the Department. of Computer Engineering, YMCA University of Science and Technology, Haryana, India. He holds a Masters Degree in Computer Science and Technology, from University of Roorkee (now IIT, Roorkee) and obtained his PhD in Computer Science and Engg. from MD University, Rohtak. He has about 16 years of work experience in the Optical Telecommunication Industry specializing in Optical Network Planning and Network Management tools development. His research interests include Cloud Computing, SOA, Network Management & Mobile Agents.