Intrusion Detection System for Detecting Malicious Nodes in Mobile Ad hoc Networks

Yuvraj Singh* and Sanjay Kumar Jena

Department of Computer Science and Engineering
National Institute of Technology Rourkela, Odisha, India
{yuvraj1510*, skjenanitrkl}@gmail.com

Abstract. A mobile ad hoc network (MANET) is a self-configuring infrastructure less network of mobile devices connected by wireless links. In this network, a mobile node behaves as a host and a router at the same time. MANETs are highly vulnerable to attacks than wired networks due to their characteristics. Ad hoc network maximize the total network throughput by using all available nodes for routing and forwarding. Hence, a node can misbehave and fail to establish route or route the data due to its malicious activity to decrease the performance of ad hoc network. In this paper, we propose an intrusion detection system to detect the malicious nodes in MANETs. The propose detection algorithm is divided into two phases: Detection during route establishment and Detection during data forwarding. The detection effectiveness of the proposed algorithm is more than 80% and for some cases detection effectiveness may reach to 100%. The silent feature of propose scheme is its simplicity and effectiveness in detecting malicious nodes.

Keywords: MANET, security attacks, malicious nodes, wireless network.

1 Introduction

In the last few years, we have seen the rapid development of wireless communication technologies. Today wireless technologies are widely used across the globe to support the communication needs of a huge number of end users [1]. The cost of wireless devices and installing wireless networks in emerging market has significantly reduced and making them much more affordable to end users.

A mobile ad hoc network (MANET) is formed by a group of mobile wireless nodes often without the assistance of fixed network infrastructures. The mobile or portable devices are free to move at any direction and are part of the network only when they are within range [12]. Applications of Ad hoc network include military tactical operations, emergency services, instantaneous meeting room applications and sensor networks [13].

MANETs are highly vulnerable to attacks than wired networks due to the open medium, dynamically changing network topology, cooperative algorithms, lack of centralized monitoring and lack of a clear line of defense [8]. Most current ad hoc routing protocols cope well with the dynamically changing topology. However, they do not address the problems when misbehavior nodes present in the

ISSN 1865-0929
ISBN 978-3-642-24036-2
DOI 10.1007/978-3-642-24037-9_41

network [9]. A commonly observed misbehavior is packet dropping. These misbehaved nodes are very difficult to identify because we cannot tell that whether the packets are dropped intentionally by the misbehaved nodes or dropped due to the node having moved out of transmission range or other link error [2].

In this paper, we propose an intrusion detection mechanism that will operate in ad hoc network to detect the malicious nodes. The propose detection mechanism is divided into two phases: Detection during route establishment and Detection during data forwarding. In first phase, we use two timer Sense timer and Reward timer and a drop counter. In second phase, each node forwards the data packet to the next hop and ensures that next node handles the packet appropriately by receiving a certificate of packet received from the next hop.

2 Related Work

Many researchers have focused on developing efficient mechanism to secure the routing in MANETs. Various secure routing, intrusion detection and response mechanisms have been proposed. Zhang, Lee, and Huang proposed intrusion detection (ID) and response system [3, 14], each node is responsible for detecting signs of intrusion locally and independently, but neighboring nodes can collaboratively investigate in a broader range. Individual IDS agents are placed on each and every node. Kachirski and Guha proposed a multi-sensor intrusion detection system based on mobile agent technology [4]. The system can be divided into three main modules, each of which represents a mobile agent with certain functionality, i.e.: monitoring, decision-making and initiating a response. Sergio Marti [10] discusses two tools Watchdog and Pathrater for detecting and mitigating routing misbehavior. These two techniques improve the throughput of MANETs in presence of compromised nodes that agree to forward but failed to do so. Watchdog is used to detect and identify a malicious node, while the Pathrater performs the job of isolating that node. Every node in the network includes both a watchdog and a Pathrater.

3 Proposed Approach

In this section, we propose an algorithm for the detection of malicious nodes in the wireless ad hoc networks. The malicious node may be defined as a node which does not follow the exact behavior. Most of the attacks are accomplished by modifying a message or simply not to forward the message which it is supposed to forward [5]. While developing the algorithms we have taken some assumptions.

3.1 Assumptions

The assumptions are as follows:

- A malicious node either drops the packet, modify the packet or simply forward the packet.

ISSN 1865-0929
ISBN 978-3-642-24036-2
DOI 10.1007/978-3-642-24037-9_41

- Each node is having a public and private key pairs.
- A key management system that helps each node to access the public key of other nodes.
- A key distribution algorithm exists.
- The availability of one way hash function H () that creates the digest of the input message.

3.2 Proposed Algorithm

This algorithm has been designed to keep the concept in the mind that malicious node may drop the packet or modify the packet. As we have seen that many routing protocol for ad hoc networks have been proposed. We mount our algorithm over the Ad hoc On-demand Distance Vector (AODV) routing protocol. The proposed algorithm is divided into two phases: Detection in route establishment phase and Detection in data forwarding phase.

Detection in Route Establishment Phase

AODV routing protocol uses the control packets (e.g. RREQ, RREP) in the route establishment. Once a route has been established the data packet has to forward via established route. During this phase, each node is having two timers (Sense Timer and Reward Timer) and a counter (Drop Counter). Here is the brief description of these timers and counters.

- **Drop Counter:** This is used as a counter and updated at two places when a packet received by the node and forwarded by the node. For each incoming RREQ packet, Drop Counter is increased by one and for each outgoing RREQ packet Drop Counter is decreased by one.
- **Sense Timer:** This timer used as a detection period for a wireless node to identify whether a node forwards the received RREQ packet during this detection period or not. If a node does not forward the RREQ packet and the sense timer expires, the value of Drop Counter is increased by one.
- **Reward Timer:** As we know the RREQ is having broadcast nature. So a node may receive duplicate RREQ. This timer is used to reward some time to a node in which node could drop the duplicate RREQ without being penalized. Reward timer is only initiated when a valid RREQ packet is forwarded during the period of Sense Timer.

The steps of the algorithm are shown in Algorithm 1. According to this algorithm, when a node receives a RREQ packet from its neighbor then we check whether this is a duplicate RREQ or not. If this is a duplicate RREQ and $Reward\_Timer$ is pending for that node then we will not penalize the node otherwise we increment the value of $Drop\_Counter$. If this is a fresh RREQ then first we initialize the both timers to $CURRENT\_TIME$ and start the $Sense\_Timer$. Then we increment the value of $Drop\_Counter$ and calculate the $Time\_To\_Send$

Algorithm 1 Detection in Route Establishment Phase

```
Require Notations
$Boolean : isduplicateRREQ$ = FALSE
$CURRENT\_TIME$ : time in the system clock
$Sense\_Timer$ : the value of detection period for the node
$Reward\_Timer$ : the value of grace period for the node
$SENSE\_TIME$ : the duration of sense time
$REWARD\_TIME$ : the duration of reward time
$Drop\_Counter$ : 0
$Time\_To\_Send$ : total time taken by the node to forward the packet
$Threshold\_Value$ : a predetermined threshold value for detection
INPUT: A RREQ packet to node
OUTPUT: Detection Status of Node
for all control packets to this node do
    if the packet is neither from nor to this node itself then
        if request is duplicate RREQ then
            $isduplicateRREQ$ =TRUE
        end if
        Step 1
        if $isDuplicateRREQ$ =TRUE AND $Reward\_Timer$ is pending then
            message "Not a New Request" and skip all the next steps
        else
            $Drop\_Counter = Drop\_Counter + 1$
        end if
        Step 2
        Set the timers
        $Sense\_Timer = CURRENT\_TIME$
        $Reward\_Timer = CURRENT\_TIME$
        Step 3
        Start the sense timer such that
        $Sense\_Timer = CURRENT\_TIME + SENSE\_TIME$
        $Drop\_Counter = Drop\_Counter + 1$
        Calculate $Time\_To\_Send$ for this packet
        if $Time\_To\_Send > Sense\_Timer$ then
            $Drop\_Counter = Drop\_Counter + 1$
        else
            Start the reward timer such that
            $Reward\_Timer = CURRENT\_TIME + REWARD\_TIME$
            $Drop\_Counter = Drop\_Counter$ - 1
        end if
    end if
    if $Drop\_Counter > Threshold\_Value$ then
        "Mark the Node as Malicious" and stop
    end if
end for
```

i. e. total time taken by the node to forward the packet. Now compare the value of $Time\_To\_Send$ with the $Sense\_Timer$. If the value of $Time\_To\_Send$ is greater than the $Sense\_Timer$ then we increment the value of $Drop\_Counter$ otherwise we start the $Reward\_Timer$ and decrease the $Drop\_Counter$. Finally, we compare the value of $Drop\_Counter$, if it is greater that a predetermined $Threshold\_Value$ then we mark node as a malicious node. Using this algorithm we can detect the nodes which are acting as maliciously during route establishment phase.

Detection in Data Forwarding Phase

In AODV protocol, after the route establishment phase a route from sender to destination has been established. The sender has all the information about the path and hops which is followed by data packet. During this phase, when a node forwards the data packet to next hop then node will receive a certificate of packet received from its next hop. This certificate represents that the node has forwarded the data packet correctly. If a node in the path does not able to produce a valid certificate then node is detected as malicious. Here is the description of certificate of packet received.

- **Certificate of Packet Received:** When a node receives the data packet from its previous hop then it generate a certificate of packet received and send to previous hop. For example suppose node A forwards a message M to node B. Then B generates a certificate C_{AB} i. e. node A has sent the data packet to node B. This certificate is generated as:

$$C_{AB} = [H(M)]_{PR_B}$$

where,

C_{AB} = Certificate received by node A from node B
M = Data Packet
H () = One way hash function
PR_B = Private Key of node B
PU_A = Public Key of node A

Algorithm 2 **Detection in Data Forwarding Phase**

```
Require Notations
$CURRENT\_TIME$ : time in the system clock
$Sense\_Timer$ : the value of detection period for the node
$SENSE\_TIME$ : the duration of sense time
$Drop\_Counter$ : 0
$Time\_To\_Receive$ : total time to receive the certificate
$Threshold\_Value$ : a predetermined threshold value for detection

INPUT: A DATA packet to node
OUTPUT: Detection Status of Node
for all all data packets to this node do
    if the packet is neither from nor to this node itself then
        Step 1
        $Sense\_Timer = CURRENT\_TIME$
        Step 2
        Start the sense timer such that
        $Sense\_Timer = CURRENT\_TIME + SENSE\_TIME$
        $Drop\_Counter = Drop\_Counter + 1$
        Step 3
        Generate a certificate for the previous hope consider to node A such that
        $C_{AB} = [H(M)]_{PR_B}$
        and send it to node A.
        Step 4
        A will calculate the $Time\_To\_Receive$ for this certificate
        if $Time\_To\_Receive > Sense\_Timer$ then
            "Discard the certificate" and
            $Drop\_Counter = Drop\_Counter + 1$
        else
            Node A will verify the certificate with node's public key
            if certificate is valid then
                $Drop\_Counter = Drop\_Counter$ - 1
            else
                $Drop\_Counter = Drop\_Counter + 1$
            end if
        end if
    end if
    if $Drop\_Counter > Threshold\_Value$ then
        "Mark the Node as Malicious" and stop
    end if
end for
```

ISSN 1865-0929
ISBN 978-3-642-24036-2
DOI 10.1007/978-3-642-24037-9_41

The steps of the algorithm are shown in Algorithm 2. According to this algorithm, Let us consider a data transfer from node A to node B. When a route establishes between source and destination the data transfer takes place. When node B receives a data packet from node A then we initialize the *Sense_Timer* to *CURRENT_TIME*, start the *Sense_Timer* and increment the value of *Drop_Counter* for node B by 1. Now node B generates a certificate for node A such as $C_{AB} = [H(M)]_{PR_B}$ and send it to node A. After that node A will calculate the *Time_To_Receive* for this certificate and compare with the *Sense_Timer* of node B. If it is greater, then node A discards the certificate and increment the value of *Drop_Counter* for node B by 1. Otherwise node A verify the certificate with the help of node B's public key. If it is a valid certificate then we decrease the value of *Drop_Counter* for node B by 1 else increase the value of *Drop_Counter* for node B by 1. Finally, we compare the value of *Drop_Counter*, if it is greater that a predetermined threshold value then we mark node as a malicious node. Using this algorithm we can detect the nodes which are acting as maliciously during data forwarding phase.

4 Simulations

In this section, we discuss about our simulator, simulation parameters and performance metrics.

4.1 Simulation Scenario

We simulate our proposed algorithm using Network Simulator version 2.34. We modify the AODV protocol in ns-2 to enable some nodes to be configured as misbehaving. The misbehavior here is define as either drop the packets or not to forward the packet in the specified time interval. The following table shows the simulation parameters.

Table 1. Simulation Parameters

S. No.	Simulation Parameters	Values
1	Simulator Used	Network Simulator (version 2.34)
2	Number of Nodes	100
3	No. of malicious nodes	10, 20, 30, 40, 50
4	Routing Protocol	AODV
5	Area Size	1900m×1900m
6	MAC	802.11
7	Simulation Time	200Secs
8	Traffic Source	CBR
9	Packet Size	512
10	Propagation Model	Two ray ground model
11	Speed	10m/s
12	Pause Time	2sec

ISSN 1865-0929
ISBN 978-3-642-24036-2
DOI 10.1007/978-3-642-24037-9_41

4.2 Performance Metrics

In this section, we discuss about the performance parameter which are used to measure the performance of the proposed algorithms. Some of them are as follows:

- **Detection Effectiveness:** This measures the performance of algorithm. This is measured as total number of detected nodes divided by the total number of malicious nodes in the network.

$$\text{Detection Effectiveness} = \frac{Detected\_nodes}{Total\_malicious\_nodes} \times 100$$

- **False Positive:** This is measured as total number of good behaving nodes but detected as malicious divided by the total number of good behaving nodes.

$$\text{False Positive} = \frac{Good\_behaving\_detected\_nodes}{Total\_good\_behaving\_nodes} \times 100$$

- **False Negative:** This is measured as total number of malicious nodes which are not detected divided by the total number of malicious nodes.

$$\text{False Negative} = \frac{Malicious\_Undetected\_nodes}{Total\_malicious\_nodes} \times 100$$

5 Results

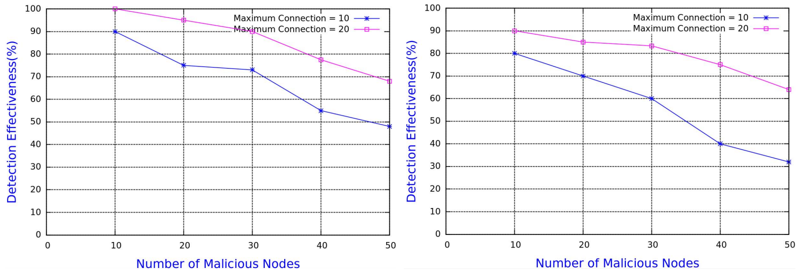
In this section, we discuss about the results of simulation and evaluate the performance of the proposed algorithm. The descriptions of the results are as follows.

- **Detection Effectiveness:** Table 2 and Figure 1 show the detection effectiveness of the proposed algorithm. In this table we have shown that the detection effectiveness is high if the network is highly connected. As number of malicious nodes increase then also the detecting effectiveness is around 70% for threshold value 20. If there are less number of malicious nodes in the network the detection effectiveness may reach to 100%. The following table and graphs describes the detection effectiveness of the proposed algorithm.

Table 2. Detection Effectiveness (%)

No. of Malicious Nodes	Maximum Connection=10		Maximum Connection=20	
	Threshold(20)	**Threshold(30)**	**Threshold(20)**	**Threshold(30)**
10	90	80	100	90
20	75	70	95	85
30	73.33	60	90	83.33
40	55	40	77.5	75
50	48	32	68	64

ISSN 1865-0929
ISBN 978-3-642-24036-2
DOI 10.1007/978-3-642-24037-9_41



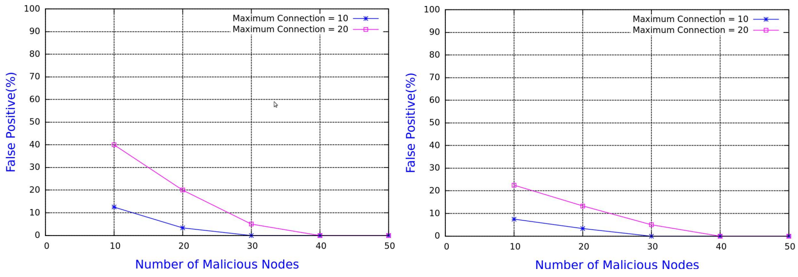
(a) Threshold=20, Max Nodes=100 (b) Threshold=30, Max Nodes=100

Fig. 1. Detection Effectiveness vs Number of Malicious Nodes

- **False Positive:** Table 3 and Figure 2 show the false positive of the proposed algorithm. In this table we have shown that we are reducing the false positive as the number of malicious nodes increase. As the number of malicious nodes increase and network is higly connected then the percentage of false positive is reaching to 0. After increase the threshold value the maximum percentage of false positive is 22 and minimum percentage reaches to 0. The following table and graphs describes the false positive of the proposed algorithm.

Table 3. False Positive (%)

No. of Malicious Nodes	Maximum Connection=10		Maximum Connection=20	
	Threshold(3)	**Threshold(5)**	**Threshold(3)**	**Threshold(5)**
10	12.5	7.5	40	22.5
20	3.33	3.33	20	13.33
30	0	0	5	5
40	0	0	0	0
50	0	0	0	0



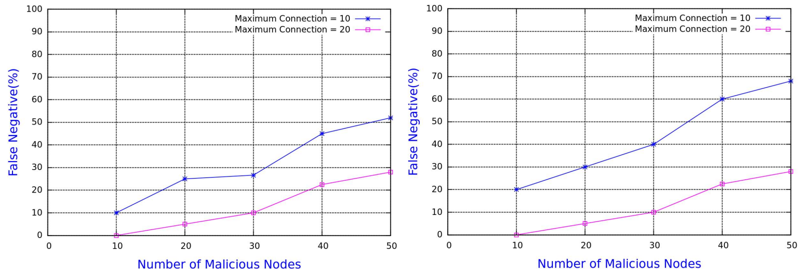
(a) Threshold=3, Max Nodes=100 (b) Threshold=5, Max Nodes=100

Fig. 2. False Positive vs Number of Malicious Nodes

- **False Negative:** Table 4 and Figure 3 show the false negative of the proposed algorithm. In this table we have shown that the percentage of false negative is below 30 for the higly connected network. As the number of malicious nodes increase and network is higly connected then the percentage of false negative is reaching to maximum 28. If the malcious nodes are less in the nework then the percentage of false negative may reach to 0. The following table and graphs describes the false positive of the proposed algorithm.

Table 4. False Negative (%)

No. of Malicious Nodes	Maximum Connection=10		Maximum Connection=20	
	Threshold(20)	**Threshold(30)**	**Threshold(20)**	**Threshold(30)**
10	10	20	0	0
20	25	30	5	5
30	26.66	40	10	10
40	45	60	22.5	22.5
50	52	68	28	24



(a) Threshold=20, Max Nodes=100 (b) Threshold=30, Max Nodes=100

Fig. 3. False Negative vs Number of Malicious Nodes

6 Conclusion

We implement packet dropping attack and an attack in which a node refuse to forward the packet within a specified interval with AODV routing protocol. The proposed algorithm has been analysed with different parameters such as connectivity of the networks and number of malicious nodes with different threshold values. The detection effectiveness of the proposed algorithm is more than 80% and for some cases detection effectiveness may reach to 100% and false positives are below 20% for different number of malicious nodes and threshold values. Thus, our experiment shows very predicting results on detecting malicious nodes. The silent feature of propose scheme is its simplicity and effectiveness in detecting malicious nodes. In the future, we would like to extend this scheme to detect other type of attacks such as application layer attack, denial of service, manipulation of network traffic and so on.

Acknowledgement

The authors would like to thank Ministry of Communication and Information Technology, Department of Information Technology, Government of India for their support and financial assistance.

References

1. Anjum, F., Mouchtaris, P.: Security for Wireless Ad Hoc Networks. WILEY, 2nd edn. (2007)
2. A.Rajaram, Palaniswami, D.S.: Malicious node detection system for mobile ad hoc networks. (IJCSIT) International Journal of Computer Science and Information Technologies 1(2), 77–85 (2010), http://dx.doi.org/10.1016/j.adhoc.2005.11.005
3. Huang, Y., Lee, W.: A cooperative intrusion detection system for ad hoc networks. In: SASN. pp. 135–147 (2003), http://doi.acm.org/10.1145/986858.986877
4. Kachirski, O., Guha, R.K.: Effective intrusion detection using multiple sensors in wireless ad hoc networks. In: HICSS. p. 57 (2003), http://computer.org/proceedings/hicss/1874/track2/187420057aabs.htm
5. Komninos, N., Vergados, D., Douligeris, C.: Detecting unauthorized and compromised nodes in mobile ad hoc networks. Ad Hoc Networks 5(3), 289–298 (2007), http://dx.doi.org/10.1016/j.adhoc.2005.11.005
6. Li, W., Joshi, A.: Security issues in mobile ad hoc networks. In: Ad Hoc Networks. pp. 1–23 (2008)
7. Mahmoud, A., Sheltami, T., Mahmoud, A., Shakshuki, E., Mouftah, H.: Aack: Adaptive acknowledgment intrusion detection for manet with node detection enhancement. In: 24th IEEE International Conference on Advanced Information Networking and Applications. pp. 634–640. IEEE Computer Society (2010)
8. Mandala, S., Ngadi, M.A., Abdullah, A.: A survey on manet intrusion detection. International Journal of Computer Science and Security 2(1), 417–432 (2007)
9. Marchang, N., Datta, R.: Collaborative techniques for intrusion detection in mobile ad-hoc networks. Ad Hoc Networks 6(4), 508–523 (2008), http://dx.doi.org/10.1016/j.adhoc.2007.04.003
10. Marti, S., Giuli, T.J., Lai, K., Baker, M.: Mitigating routing misbehavior in mobile ad hoc networks. In: MOBICOM. pp. 255–265 (2000), http://doi.acm.org/10.1145/345910.345955
11. Mitrokotsa, A., Mavropodi, R., Douligeris, C.: Intrusion detection of packet dropping attacks in mobile ad hoc networks. International Conference on Intelligent Systems And Computing: Theory And Applications pp. 111–118 (2006)
12. Murthy, C.S.R., Manoj, B.S.: Ad Hoc Wireless Networks: Architectures And Protocols. Pearson Education India (2008)
13. Sterne, D.F., Balasubramanyam, P., Carman, D., Wilson, B., Talpade, R., Ko, C., Balupari, R., Tseng, C.Y., Bowen, T.F., Levitt, K.N., Rowe, J.: A general cooperative intrusion detection architecture for manets. In: IWIA. pp. 57–70 (2005)
14. Zhang, Y., Lee, W.: Intrusion detection in wireless ad-hoc networks. In: MOBICOM. pp. 275–283 (2000), http://doi.acm.org/10.1145/345910.345958

ISSN 1865-0929
ISBN 978-3-642-24036-2
DOI 10.1007/978-3-642-24037-9_41