ISSN 2278 - 3091

International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol.2 , No.5, Pages :75-80 (2013)
Special Issue of ICCECT 2013 - Held during September 20, 2013, Bangalore, India

Network Intrusion Detection and Response System for Mobile Ad hoc Networks

Seema Tryambak Sawant[1], Sambre Nitin B.[2]
[1]K.I.T.'s College of Engineering, Kolhapur, India, seemasawant2678@yahoo.co.in
[2]K.I.T.'s College of Engineering, Kolhapur, India, sambrenitin@gmail.com

Abstract : As wireless network growing rapidly it becomes very important to have intrusion detection system in it. Mobile ad hoc networks are particularly vulnerable to denial of service attacks (DoS) due to their open decentralized architecture, highly dynamic topology and shared wireless medium in which they exist. We implemented the network intrusion detection system (NIDS) for mobile ad-hoc networks to indentify misbehaving nodes that agree to forward packets but fail to do so, and an response system that helps routing protocols to avoid these nodes. Simulation result shows increase in throughput.

Key words : MANET, ad hoc networks, AODV, network intrusion detection, response system.

INTRODUCTION

The advent of wireless communication and the proliferation of handheld devices has significantly advanced the growth of nomadic communications. The capability of these handheld mobile devices to self organize themselves on-the-fly in the absence of an infrastructure, and to extend their communications beyond their wireless radio range has potentially led to the development of Mobile Ad hoc Networks (MANET). Limited range wireless communication and high node mobility means that the nodes must cooperate with each other to provide essential networking, with the underlying network dynamically changing to ensure needs are continually met. Mobile devices in these networks are commonly referred to as nodes and are predominantly deployed in conditions that include emergency scenarios, such as earthquakes and other natural disasters, rescue operations and defense related applications, environmental monitoring, conferences etc. Furthermore, the self-organized, multi-hop and infrastructure-less features have evolved the MANET into being the basis for sensor networks , Vehicular Ad hoc Networks (VANET) , peer-to-peer wireless networks , pervasive networks and mesh networks.

However, the successful deployment of civilian and commercial MANET is still in its infancy stages, because the same features that support the development of MANET emerge as a hindrance for their deployment. In other words, these features give rise to a range of issues, such as, (a) broken and sporadic links that result from a mobility-induced dynamically changing topology , (b) insecure and promiscuous wireless communications, (c) self-organized, multi-hop and infrastructure-less features of MANET being reliant on the cooperation between the mobile nodes , and (d) slow advancement in battery technology causes battery power to be a constrained resource among the heterogeneous mobile nodes . Extensive research has been carried out to date to address these issues; the nuclei of this research focus on security Quality of Service (QoS) and reliability mobility management and topology control, network connectivity and routing , multicasting, power management, and localization and node auto-configuration.

MANETs by their very nature are more vulnerable to attack than wired net-works. The flexibility provided by the open broadcast medium and the cooperativeness of the mobile devices (which have generally different resource and computational capacities, and run usually on battery power) introduces new security risks. As part of rational risk management we must be able to identify these risks and take appropriate action. In some cases we may be able to design out particular risks cost-effectively. In other cases we may have to accept that vulnerabilities ex-ist and seek to take appropriate action when we believe someone is attacking us. As a result, intrusion detection is an indispensable part of security for MANETs.

BACKGROUND AND RELATED WORK

1) D. B. Johnson [3,7] introduces Dynamic Source Routing (DSR) uses source routing to deliver packets from one node in the network to some other node. DSR operates on two mechanisms: Route Discovery and Route maintenance. Route Discovery is used when the sender does not know the path to the destination. In this mechanism, the sender broadcasts a ROUTE REQUEST message. Each intermediate node adds its address in ROUTE REQUEST message and rebroadcast it, unless it has not rebroadcasted earlier. With this controlled broadcast, the ROUTE REQUEST will ultimately reaches the destination. The destination then sends a unicast ROUTE REPLY message in reverse direction. When the ROUTE REPLY packet reaches the source, it records the route contained in it and saves in its cache for the specific destination. For better performance, intermediate nodes also records this route information from the two route messages. Finally, Route Maintenance mechanism is used to notify source and potentially trigger new route discovery events when changes in the network topology invalidates a cached route. Ad-hoc On demand Distance Vector routing (AODV) [5] is another on-demand protocol. It has similar mechanism of ROUTE REQUEST and ROUTE REPLY as that in DSR. However, it does not rely on source routing, rather it makes use of routing tables at intermediate nodes. The nodes maintain routing table entries of all reachable nodes in the network. The route table is used to route data packets destined for a particular node and to respond to ROUTE REQUEST.

2) Marti et al. [2] introduced Pathrater, which chooses a path from source to destination based upon a simple rating algorithm, instead of the shortest path. The Pathrater run by each node in the network, combines knowledge of misbehaving nodes with link reliability data to pick the route most likely to be reliable. Each node maintains a rating for every other node it knows about in the network. It calculates a path metric by averaging the node ratings in the path. The pathrater assigns node ratings to node

ISSN 2278 - 3091

International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol.2 , No.5, Pages :75-80 (2013)
Special Issue of ICCECT 2013 - Held during September 20, 2013, Bangalore, India

according to the following algorithm. When a node in the network becomes known to the pathrater(through route discovery) the pathrater assigns it a neutral rating of 0.5. A node always rates itself with 1.0. The pathrater increments the rating of nodes on all actively used paths by 0.01 at periodic intervals of 200 ms. An actively used path is one on which the node has sent a packet within the previous rate increment interval. The maximum value of a neutral node can attain is 0.8. Nodes rating decremented by 0.05 when a link break is detected during packet forwarding and the node become unreachable. A lower bound rating of a neutral node is 0.0. The pathrater does not modify the ratings of nodes that are not currently in active use. A special high negative value -100 is assigned to nodes suspected misbehaving.

3) Bhargava et al. [8] proposed an intrusion detection and response model (IDRM) to enhance security in the Ad Hoc On Demand Distance Vector (AODV) routing protocol [3]. In this scheme, each node employs the IDRM that utilizes neighborhood information to detect misbehavior of its neighbors. When the misbehavior count for a node exceeds a predefined threshold, the information is sent out to other nodes as part of global response. The other nodes receive this information, check their local malcount for this malicious node, and add their results to the initiators response. In the intrusion response model (IRM), a node identifies that another node has been compromised when its Malcount increases beyond the threshold value for that allegedly compromised node. In such cases, it propagates this information to the entire network by transmitting a special type of packet called a MAL packet. If another node also suspects that the detected node is compromised, it reports its suspicion to the network and retransmits another special type of packet called REMAL. If two or more nodes report about a particular node, another special packet, called a PURGE packet, is transmitted to isolate the malicious node from the network. All nodes that have a route through the compromised node look for newer routes. All packets received from a compromised node are dropped. Some of the internal attacks include distributed false route request, DoS, impersonation, and compromise of a destination. A malicious node might send frequent unnecessary route requests. When the nodes in the network receive a number of route requests greater than a threshold count by a specific source for a destination in a particular time interval, the node is declared malicious. Although the Pathrater scheme provides major improvements to security in ad hoc networks, it still suffers from critical limitations and weaknesses. The Pathrater always categorizes nodes as either neutral or as malicious, depending on the rating. This means that the Pathraters tolerance scheme is typically exceedingly poor. A malicious node may possibly mislead the Pathrater. In that case what the node could do is behave well for stretched periods of time. During this time, its rating continuously improves until it reaches the maximum rating of 0.8. At this moment, it would start to misbehave, drop packets. If a new node is added to the network, it is treated by the Pathrater as a neutral node, no particular vigilance is given to that recently added node. Similar to the above weakness, old nodes that have previously been classified as malicious are allowed to directly rejoin the network after a long period of time and are classified as neutral.

A two-layered (first-layer of detection systems and second-layer of detection response /reaction systems) is required for the security of MANET routing protocols. Therefore the objective is to propose an intrusion detection system to detect nodes that agree to forward the packets, but fail to do so; and a response system which helps routing to avoid such nodes for AODV routing protocol.
Proposed system exploits many of the limitations and weaknesses suffered by Pathrater in order to produce a more effective and reliable intrusion detection and response system in mobile ad hoc networks. Following are the detailed objectives of the proposed work.

1. Creation and configuration of Ad hoc Network.
2. Use of AODV Routing Protocol in above Ad hoc network and Creation of traffic flow.
3. Implementing function to add malicious behaviour in a node and measure throughput for above [step 2] generated traffic.
4. Implementation of Intrusion Detection System and Response System which are discussed in theoretical analysis.
5. Use of above system in same network and measure throughput.
6. Analysis of the system based on throughputs and conclude.

WORKING OF AODV

It is an on-demand routing protocol, similar to DSR. Basically it is the integration of DSDV(hop by hop) and DSR (on demand). The routing table only maintains the routing information needed, instead of keeping the entire routing table (like DSR). The routing information is recorded into the routing table of the intermediate router along the path, so the data packet only contains the destination address (like DSDV). Routing table consist of- Destination address, Next hop address, Destination Sequence number and hop count. It consist of two phases route discovery and route maintainace.

Route Discovery:

When the route is needed, the source sends the RREQ packet in a controlled flooding manner throughout the network. Intermediate node checks its routing table. If with the routing information, reply to destination with the RREP packet otherwise, the intermediate forwards RREQ packet to its neighbors. Finally, the destination or some intermediate nodes will reply the routes to the source

Route Maintenance:

If a node is continuously sending packets via a route, it has to make sure that the route is held upright. As soonas a node detects problems with the current route, it has to find an alternative.
The following schemes can detect the link breakage-
-hop-by-hop MAC layer ACK
-Hello message
After detecting the link breakage, the upstream node will notify the source with an RERROR packet. Source will initialize a new route discovery stage and flood the RREQ packet. In ns-2.1b8a, the upstream node can directly flood the RREQ packet as well as notify source to eliminate the invalid route entry.

ISSN 2278 - 3091

International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol.2 , No.5, Pages :75-80 (2013)
Special Issue of ICCECT 2013 - Held during September 20, 2013, Bangalore, India

INTRUSION DETECTION AND RESPONSE SYSTEM

In this section, we describe the intrusion detection and response system. Intrusion detection system detects intrusion from malicious nodes and reports this information to the response system.

A. Intrusion Detection System:

The system detects the misbehaving nodes. Figure 1 illustrates how intrusion detection system works. Suppose there is a path from S to D through intermediate nodes A, B, and C. Node A can not transmit all the way to node C, but it can listen in on node B's traffic. Thus, when A transmit a packet for B to forward to C, A can often tell if B transmits the packet. If encryption is not performed separately for each link, which can be expensive, the A can also tell if B has tampered with the payload or the header.

When a node forwards a packet, intrusion detection system verifies that the next node in the path forwards the packet. The system does this by listening promiscuously to the next nodes transmissions. If the next node does not forward the packet then it is misbehaving. It can be implemented by maintaining a buffer of recently sent packets and comparing each overheard packets with packet in the buffer to see if there is match. If so, the packet in the buffer is removed since it has been forwarded on. If packet has remained in the buffer for longer than a certain timeout, system will increment failure tally for the node responsible for forwarding on packet. If the tally exceeds a certain threshold it determines that the node is misbehaving and sends a message to the source notifying it of the misbehaving node. This mechanism is illustrated in fig.1.

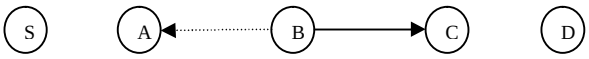

Fig1: When B forwards a packet from S towards D through C, A can overhear B's transmission and can verify that B has attempted to pass the packet to C. The solid line indicates the intended direction of the packet sent by B to C, while the dashed line indicates that A is within transmission range of B and can overhear the packet transfer.

This technique has advantage that it can detect misbehavior at the forwarding level and not just the link level. Its weaknesses [2] are that it might not detect a misbehaving node in the presence of 1) ambiguous collisions, 2) receiver collisions, 3) limited transmission power, 4) false misbehavior, 5)collusion, and 6) partial dropping.

Intrusion detection system detects intrusion from malicious nodes and reports this information to the response system. The situation of network partition is especially serious when intermediate node A reports all the nodes on the routing path from source S to destination D being malicious.

We consider the case when node A is on all the paths from S to D and A is a malicious node. Figure 2 shows the result when malicious node A reports all nodes, F and B, on the path from the source S to the destination D being malicious. In Figure 2, there are two paths from S to D after Route Discovery:

S -> A -> B -> D, and S -> A -> F -> D.

The situation of network partition is especially serious when intermediate node A reports all the nodes on the routing path from source S to destination D being malicious. We consider the case when node A is on all the paths from S to D and A is a malicious node. Figure 2 shows the result when malicious node A reports all nodes, F and B, on the path from the source S to the destination D being malicious.

In Figure 1, there are two paths from S to D after Route Discovery:

S -> A -> B -> D, and S -> A -> F -> D.

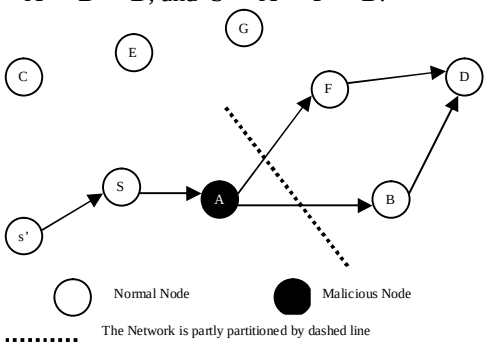

Figure 1: Malicious node A falsely reports all nodes on the path from source to destination as misbehaving in order to partition the network

If A reports B and F as misbehaving successively then S marks B and F as malicious. As a consequence, D, B and F will also mark B and F as malicious. And the network will be partitioned just as by the dashed line in Figure 2.

A. Intrusion Detection System:

It is implemented by maintaining a table that stores entry <source, destination, sum, path>. Whatever the current node is, the source, the destination or the intermediate node, it inserts such an entry into the table when sending, forwarding or receiving packets for the first time. The value of each field is:

Source: the address of source.

Destination: the address of destination.

sum: the total number of packets that the current node sends, forwards, or receives using the route path Path as source, intermediate node or destination respectively.

path: the route that is used for the communication between <source, destination>. The path is a list of nodes addresses.

When an intermediate node on a route path reports to the source that its next hop is malicious, the source will not immediately decrease the rating of the malicious node. Instead, it will send a message to the destination using an alternative path in the route table. The message contains <source, destination, sum, malicious_node_address>. Source, destination and sum are the same as the above. malicious_node_address is the address of the node being reported malicious. The source node then searches a path that has no malicious node in it from the routing table. If there is not such a path available, the source then launch a Route Discovery to find a new one. After finding a path, the source sends the message using the found path.

Upon receiving the message, destination node will search its own table to see if there is a match. If there is not

ISSN 2278 - 3091

International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol.2 , No.5, Pages :75-80 (2013)
Special Issue of ICCECT 2013 - Held during September 20, 2013, Bangalore, India

a matching entry in the table, it means the node is malicious and the destination node returns a message to the source confirming that the malicious node is really malicious. If there is, destination node then compares the sum field of the passing in message with the one found in the table. If the two sums equal, it means that the malicious node forwards all packets that the source sends thus it is not malicious. On the contrary, if the two sums are not equal, the node falsely report might be malicious.

Following is the pseudo code of maintaining the additional work of removing false misbehavior weakness.

```
Nodes sending, forwarding, or receiving
packets do:

//search if the entry exists in table
One_entry = search_entry();

If one_entry = NULL
    // add new entry to table
       add_new_entry();
else
   // update the sum of existed entry
      Update_sum();

The destination node verifies if the node is
malicious does:
One_entry = search_entry();

If one_entry = NULL
   return false;
else{
   sum_in_table = one_entry.getSum();
   sum_from_source= msg_from_source.getSum();
  if (sum_in_table == sum_from_source)
    return true;
  else
    return false;
}
```

B. Response System:

Similar to the traditional Pathrater, this is run by each and every node in the network. Each node stores a rating for all the nodes it knows about in the network. However it assigns ratings to nodes and calculates a path metric in a refined way. This system implements a classification that places each network node into one of five classes: *Fresh*, *Member*, *Unstable*, *Suspect* or *Malicious*. Each node is treated differently depending on its status and rating. This system tries to obtain the maximum amount of network throughput and the best performance possible. For the rest of this section, system is explained in more detail. The state machine diagram in Fig. 2 is a detailed representation of system operation. Fig. 3 provides an abstract algorithmic pseudo code of system.

When the network finds out about a new node through route discovery, systemd classifies it as Fresh. The network is on the whole "precautious" from this recently added node. The system assigns it a rating of 0. If the node behaves well, by participating in forwarding packets for example, then its rating is incremented by one. If the node misbehaves, then its rating is decremented by four. The node remains in this Fresh state for a short period of time t_f, which is measured in seconds. For the duration of this phase, a node is permitted to forward and receive packets, but not send its own packets. If after t_f, seconds the node's rating is positive or zero i.e. $rating \geq 0$, then its classification is changed to Member. If however, the node's rating was negative, i.e. $rating < 0$, then its classification is changed to *Suspect* instead. This method solves the "New node anonymity" predicament discussed in Section II. With such simple attentiveness the ad hoc network may possibly avoid a multiplicity of fatal attacks on the network or denial of service attacks.

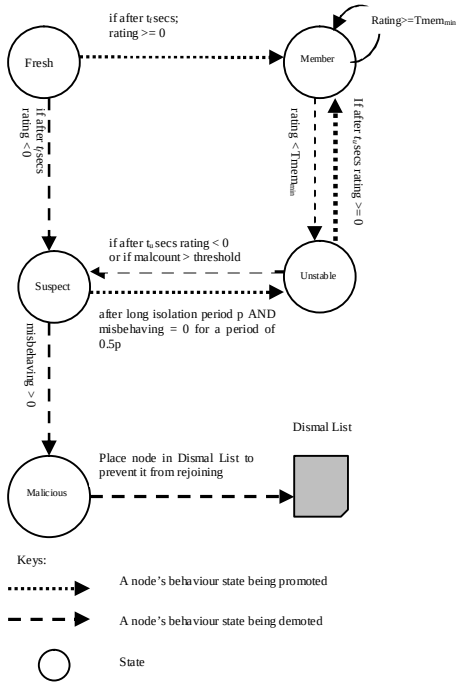

Fig. 2 State diagram for operation of Intrusion Detection System

The second classification category is *Member*. This is the ordinary and regular operation mode. Any node within this *Member* category is allowed to send, receive and forward packets; it is treated with more trust by the network, although as expected, Member nodes are monitored on a regular basis by the Watchdog. Whenever a node's state is changed to *Member*, this node's *rating* is reset to 0. A well behaved node, which contributes actively and positively to the network, is rewarded by having its rating incremented by one. The maximum value a *Member* node can attain is $Tmem_{max}$. If the networking Environment is known to be secure, with low probability of intrusion detection, then $Tmem_{max}$ could be assigned a large value. This would help prevent previously well behaved broken nodes from being isolated from the network. Otherwise, smaller values should be assigned to $Tmem_{max}$. Sergio Marti et al. [3] suggested a value of thirty for their Pathrater. They do however plan to experiment with different Pathrater variables in order to compute optimal values. If a node misbehaves, its rating is decremented by five. While calculating the path metric, any node with a rating below a specified threshold value $Tmem_{min}$ is considered to be disobedient and is therefore classified as Unstable. Once again $Tmem_{min}$ could be variable depending on the network's intrusion detection and response stringency.

Nodes tagged as *Unstable* by the system are allowed to operate semi-normally. They are permitted to forward and receive packets. However, they are not allowed to send

ISSN 2278 - 3091

International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol.2 , No.5, Pages :75-80 (2013)
Special Issue of ICCECT 2013 - Held during September 20, 2013, Bangalore, India

their own packets for an explicit period of time t_u. During this interval, the same Fresh node's rate-time race scenario takes place: The system assigns it a rating of 0. If the node behaves well, by participating in forwarding packets for example, then its rating is incremented by one. If the node misbehaves, then its rating is decremented by six. After t_u seconds elapse, the node's rate is examined; if its rating is positive or zero i.e. $rating \geq 0$, then its classification is changed back to Member. If however, the node's rating was negative, i.e. $rating < 0$, then its classification is changed to Suspect instead. One might argue that this method may possibly be exploited by the malicious node i.e. a node could keep misbehaving for elongated periods of time and once it is classified as *Unstable*, it temporarily well behaves until it is reclassified as *Member*. This scenario is avoided by introducing a *malcount* counter. Initially a node's *malcount* is reset to 0 and each time the node's status is changed from Member to Unstable, *malcount* is incremented by one. If *malcount* exceeds a certain threshold value then its status is reclassified as Suspect. This prevents nodes from repeatedly fluctuating between *Member* and *Unstable* states.

As mentioned in the above paragraph, a node is classified as a Suspect node by the system under two conditions: Either the node was in Unstable state and received a negative rating after t_u seconds had elapsed or the *malcount* counter exceeded the system's endurable threshold value. The intrusion detection system raises the "danger alert" whenever a Suspect node is encountered. The Suspect node is completely isolated from the network for a period of time *p*. It is not allowed to send, receive or forward packets. If a neighbour receives a packet from this spiteful node it just discards the packet. However, it could be preferable if this node was not permanently excluded from routing. Therefore, after the long timeout *p*, the node is reconnected and is immediately monitored for a reasonably extensive period of time (0.5*p* is a fine estimation). If the Watchdog stops reporting misbehaviour relating to that node, it changes the node's status to Unstable and resets its *malcount* to 0.

If the intrusion detection system continues to report misbehaviour of a *Suspect* node, then it is labelled as Malicious. The response system implements a very shrewd rating system and therefore once a node is labelled as Malicious, its behaviour cannot be tolerated any more and as a consequence this node is permanently banned from the ad hoc network. In order to ensure that this malicious node does not try to reconnect, its identification is added to the Dismal List. Table 1 provides a summary of the different node states discussed above.

Node state	Network's (as a whole) sensitivity to nodes in specified state.
Fresh	New nodes are mysterious to the network and should be treated with caution.
Member	Nodes are relatively trusted and are allowed to operate normally.
Unstable	Might be malfunctioning or misbehaving; allowed to operate semi-normally.
Suspect	Danger alert raised and node temporarily banned then closely monitored.
Malicious	Node's behavior is absolutely intolerable and therefore it is permanently banned.

Table 1 : Summary of different states

PERFORMANCE ANALYSIS

A simulation model for intrusion detection system has been developed in Network Simulator (NS-2) [6]. Our simulations take place in a network with 300m x 300m flat space filled with a scattering of 50 wireless mobile nodes. The nodes communicate using 10 constant bit rate (CBR) node-to-node connections with a data rate of 4 packets per second. All nodes move in random mode with speed varying from 0 meter/second to 3 meter/second.

while state is **Fresh** do:
 	
set *malcount* to 0
set *rating* to 0
set *timer* to 0 and start *timer*
 	
while *timer* is < t_f do:
if *misbehave* = false
rating = *rating* + 1
else
rating = *rating* - 4
 	
if *rating* is ≥ to 0
state = **Member**
else
state = **Suspect**
 	
Node in Fresh state may only receive and forward other packets but cannot send its own packets.
	while state is **Member** do:
	
	set *rating* to 0
	
	if *misbehave* = false
	& *rating* is < $Tmem_{max}$
	rating = *rating* + 1
	else
	rating = *rating* – 5
	
	if *rating* is < $Tmem_{min}$
	malcount = *malcount* + 1
	& *state* = **Unstable**
	
	Node in Member state has full privileges to send, receive and forward packets.

while state is **Suspect** do:
 	
set *timer* to 0 and start *timer*
while *timer* is < *p* do
completely isolate node
 	
set *rating* to 0
set *timer* to 0 and start *timer*
 	
while *timer* is < 0.5**p* do
if *misbehave* = true
rating = *rating* + 1
 	
it *rating* is > 0
state = **Malicious**
else
malcount = 0
state = **Unstable**
 	
Node in Suspect state isolated for long time p then monitored with 0 tolerance for 0.5p.
	while state is **Unstable** do:
	
	if *malcount* > *Threshold value*
	state = **Suspect**
	
	set *rating* to 0
	set *timer* to 0 and start *timer*
	
	while *timer* is < t_u do:
	if *misbehave* = false
	rating = *rating* + 1
	else
	rating = *rating* - 6
	
	if *rating* is ≥ to 0
	state = **Member**
	else
	state = **Suspect**
	
	Node in Unstable state may only receive and forward other packets but cannot send its own packets.

Fig.3 Pseudo Code of Intrusion Detection and Response System

In our simulations, the misbehaving nodes can damage the network performance especially by falsely reporting that other normal nodes as misbehaving. Of the 40 nodes in the simulated network, some variable percentages of the nodes misbehave. Throughput is the percentage of data transferred correctly from source to destination in a specified amount of time and the overhead is the additional AODV control packets needed to be send.

ISSN 2278 - 3091

International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol.2 , No.5, Pages :75-80 (2013)
Special Issue of ICCECT 2013 - Held during September 20, 2013, Bangalore, India

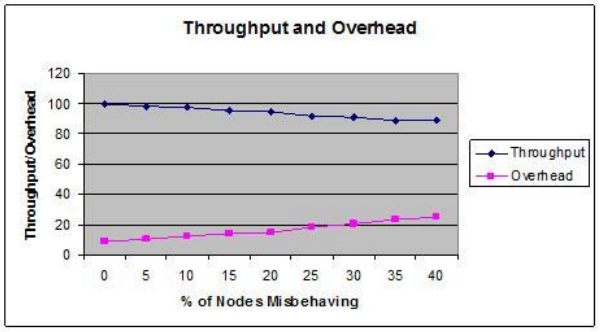

Fig. 4: Overall throughput and overhead

Initially due to dropping of packets the throughput was zero but after starting our system the system throughput increases. We vary the percentage of nodes misbehave. The throughput decrease and overhead increases as we increase the number of misbehaving nodes. We have got increase up to 88-90 % when 40 % nodes misbehave. Overhead incurred due to additional control packets is 15 % when 40 % nodes misbehave.

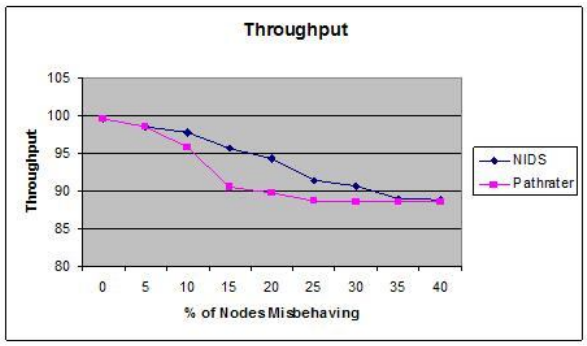

Fig. 5 Overall Throughput as a Function of Intensity of Node Misbehavior

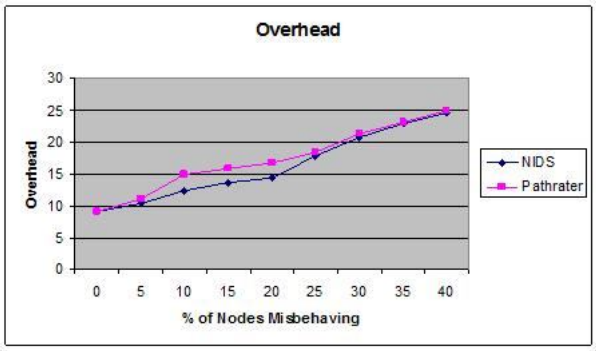

Fig. 6 Overall overhead as a Function of Intensity of Node Misbehavior

CONCLUSION

Ad hoc networks are an increasingly promising area of research with lots of practical applications. However, MANETs are extremely vulnerable to attacks due to their dynamically changing topology, absence of conventional security infrastructures and open medium of communication, which, unlike their wired counterparts, cannot be secure.

This system solves the problem of watchdog i.e. a malicious node can partition the network by falsely reporting other nodes as misbehaving as increase the throughput. This system also gives increased performance compared to pathrater due the node classification scheme. With IDS we have indetified the malicious node. With response system the corrective major (skipping the malicious node and taking another path) is taken. Increased throughput upto 88-90% when 40% of the nodes misbehave, which was zero due to dropping. Overhead incurred is 15% when 40% nodes misbehave. Throughput and overhead is better as cpmpared to pathrater. Thus AODV made secure with maximum throughput and minimum overhead.

REFERENCES

[1] Mishra, A. Nadkarni, K, and Patcha, A.,"Intrusion Detection in wireless Ad hoc Networks," *IEEE Wireless Communications*,Vol. 11, Feb. 2004, pp. 48-60

[2] S. Marti, T.J. Giuli, K. Lai, M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," in 6th International Conference on Mobile Computing and Networking, *MOBICOM'00*, Aug. 2000, , pp. 255-265

[3] D. B. Johnson and D. A. Maltz, *Dynamic Source Routing in Ad HocWireless Networks*, In Mobile Computing, Chapter 5, P153-181, Kluwer Academic Publishers, 1996.

[4] V. Park, and S. Corson, *Temporally-Ordered Routing Algorithm (TORA)*, Ver. 1, Internet draft, IETF, Aug 1998.

[5] A. Patwardhan, J. Parker, A. Joshi, M. Iorga and T. Karygiannis, *Secure Routing and Intrusion Detection in Ad Hoc Networks*, Proceedings of the 3rd International Conference on Pervasive Computing and Communications, Hawaii, March 2005.

[6] The NS2 Project, http://www.isi.edu/nsnam/ns/.

[7] Security Issues in MANETs, Abhishek Seth, 04329001, November 12, 2004.

[8] S. Bhargava and D. P. Agrawal, Security Enhancements in AODV Protocol for Wireless Ad Hoc Networks, VTC 2001 Fall, vol. 4, Oct.11, 2001, pp. 43-47 .

.