MULTICAST SECURITY ATTACKS AND ITS COUNTER MEASURES FOR PUMA PROTOCOL

Mr. A. Amuthan,  
Associate Professor,  
Pondicherry Engineering College,  
amuthan@pec.edu

D. Nagamani Abirami  
Pondicherry Engineering College,  
abiramimtech@gmail.com

Abstract

A Mobile Ad hoc NETwork (MANETs) is a continuously changing wireless network that can be created without any pre-existing infrastructure in which each node can operate as a router. MANETs has no clear line of control, so it is accessible to both legitimate network users and malicious attackers. The main challenge in MANETs is to design a secure solution which can protect the MANET from various kinds of security attacks. In this paper, we put a first step towards securing multicast routing protocol for ad hoc network. Specifically we examined the vulnerabilities of PUMA (Protocol for Unified Multicasting through Announcements) which is a representative of mesh based routing protocol. In this proposed work, we have presented a trust based approach where in the secure route is selected for the receivers not only based on current trust values of its neighbor nodes but also its past experience is considered for blackhole and wormhole attack.

1. Introduction

A Mobile Ad hoc Network (MANETs) [1] is a network consisting of a set of wireless node that can communicate with each other without any pre-existing infrastructure. It is an autonomous system in which mobile hosts connected by wireless links are free to move randomly and often act as routers at the same time. These type of networks have several advantages such as self reconfiguration and highly adaptable for mobile characteristics such as power level, bandwidth, etc. In MANETs, multicasting plays an important role for communicating one or more groups. Each node may join and leave the multicast group anytime. The main challenge in MANETs is to design a secure solution which can protect the MANETs from various kinds of security attacks.

In this paper we have summarized the protocol operation of PUMA, vulnerabilities present in PUMA, analyzed the possible attacks in the protocol and then presented the solution for the attacks.

2. Related Work

In [2], a trust based approach has proposed to secure DSR protocol. Secure route to a destination is calculated based on weighted average of the trust values of the nodes in the route with respect to the behavior experienced by the neighboring nodes and the number of nodes in the route. In [3], proposed a trust vector model for the routing protocols to detect malicious nodes in the network. Each node should evaluate its trust parameters by monitoring the neighbor nodes activities. In [4], uses trust based packet forwarding mechanism to detect and isolate the malicious nodes. Based on the behavior of the nodes a trust counter is decremented and incremented and also it maintains trust threshold when any node falls below the limit which will be marked as malicious.

3. Overview of PUMA

3.1 Features of PUMA

PUMA [5] is a reactive routing protocol which discovers route only when it is required. Its multicast connectivity is established and maintained by means of receiver initialization approach in which the receivers joins into the multicast group by using address of core node without the need for network-wide flooding of control or data packets from all the sources of the group. Each group has exactly has one special node which is called as core node in the group. PUMA’s uses the shared mesh based multicast topology for constructing routes to the members of the multicast group without depending upon any unicast routing protocol. Multicast group maintenance of PUMA is achieved by using the soft state approach where in
which the multicast group membership and its associated routes are refreshed periodically by flooding its Multicast Announcement (MA) packet.

3.2 Control Packet
PUMA uses a single control packet called Multicast Announcement (MA) to create and maintain its multicast topology in MANET.

<table>
<thead>
<tr>
<th>Mesh membership code</th>
<th>Distance to core</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Group ID</td>
</tr>
<tr>
<td></td>
<td>Core ID</td>
</tr>
<tr>
<td></td>
<td>Sequence Number</td>
</tr>
<tr>
<td></td>
<td>Parent</td>
</tr>
</tbody>
</table>

Figure 1. Multicast Announcement

Mesh membership code – this field is set to 1 when a node wants to join into the group; else it is unset.

Distance to core – hop count from the current node to core node.

Group ID – address of the group.

Core ID – address of the core node.

Sequence number – sequence number of the group.

Parent ID – address of the neighbor to reach the core.

3.3 Uses of MA
Multicast Announcements are used
i. To elect the core.
ii. Find out the sources outside a multicast group to unicast data packets towards the group.
iii. Join and leave the mesh of a group.
iv. Maintain the mesh of the group.

3.4 Core Election
PUMA chooses a core for each multicast group in the network. Each connected component has only one core. If one receiver joins the group before other receivers, then it becomes the core of the group. If several receivers join the group at the same time, then the one with highest ID becomes the core of the group.

When a receiver needs to join a multicast group, it first determines whether it has received a multicast announcement for that group. If the node has received, then it takes on the core specified in the announcement it has received, and it transmits the multicast announcements that specifies the same core for the group. Otherwise it assumes itself as the core of the group and starts transmitting multicast announcement periodically to its neighbors stating itself as the core of the group and a hop count of 0 distance to itself. Nodes propagate multicast announcements based on the best multicast announcement they receive from their neighbors.

A node that believes itself to the core of a group, it transmits multicast announcements periodically for that group. As the multicast announcement pass through the network, it establishes a connectivity list at every node in the network. Connectivity list is used to form a mesh structure and to route data packets from receivers to the core.

A node keep tracks of the data from all the multicast announcements it receives from its neighbors in the connectivity list. Fresher multicast announcements from a neighbor overwrite entries with lower sequence numbers for the same group. Hence all the nodes in a group store the recent information about a neighbor for each core in the group.

Each entry in the connectivity list also stores the time when it was received, and the neighbor from which it was received. The node then generates its own multicast announcement based on the best entry in the connectivity list.

While electing core, on receiving a multicast announcement with higher core ID, all entries in the connectivity list with a lower core ID are erased. Hence all suitable entries in the connectivity list at any point of time have the same core ID and sequence number. Among these suitable entries, the entries with a shortest distance to core be qualified as best entries, and the neighbors corresponding to these entries are called parents.

After selecting the best multicast announcement, the node generates the fields of its own multicast announcement in the following way:

- **Core ID**: The core ID in the best multicast announcement.
- **Group ID**: The group ID in the best multicast announcement.
- **Sequence number**: The sequence number in the best multicast announcement.
• **Distance to core:** One plus the distance to core in the best multicast announcement.
• **Parent:** The neighbor from which it received the best multicast announcement.
• **Mesh member:** A node sets its membership code field based on whether it is a mesh member or non-member.

After generating its own multicast announcement, it will broadcast to its entire neighbor.

![Diagram](image)

**Figure 2. Dissemination of Multicast Announcement**

**Table 1. Connectivity list at node 5**

<table>
<thead>
<tr>
<th>Neighbor</th>
<th>Multicast Announcement</th>
<th>Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Distance to core</td>
<td>Parent</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

Figure 2 illustrates the propagation of Multicast Announcements and Table 1 shows the building of connectivity lists. The solid arrows indicate the neighbor from which a node receives its best multicast announcement. Node 5 has three entries in its connectivity list for neighbors 10, 0, and 6. However it chooses the entry it receives from 10 as the best entry, because it has the shortest distance to core. Node 5 uses this entry to generate its own multicast announcement, which specifies Core ID = 10, Group ID = 224.0.0.1, Sequence Number = 79, Distance to Core = 2 and Parent = 10. When a node wants to send data packets to the group it forwards it to the node from which it received its best multicast announcement. If that link is broken then it tries its next best and so on. Hence each node in the network has one or more routes to the core. The multicast announcement sent by the core has distance to core set to zero and parent field set to INVALID-ADDRESS, because the core has no parents.

Multicast announcements are generated by the core for every three seconds. After receiving a multicast announcement with a highest sequence number, nodes wait for a short period (e.g. 100 ms) to collect multicast announcements from multiple neighbors before generating their own multicast announcement.

### 3.6 Multiple Groups

When multiple group exits, nodes collective all the fresh multicast announcements they receive, and broadcast them periodically for every multicast announcement interval. However, multicast announcement representing groups being received for the first time, resulting in a new core, or resulting in variations in the mesh membership code are forwarded immediately, without aggregation. This is to pass up delays in critical operations, when electing core node and establishing mesh structure.

### 3.7 Establishment and Maintenance

Initially only receivers consider themselves as mesh members and set the mesh member flag to true in the multicast announcements they send. Non-receivers consider themselves mesh members if they have atleast one mesh child in their connectivity list.

A neighbor in the connectivity list is a mesh child if:

a) its mesh member flag is set.
b) the distance to core of the neighbor is larger than the nodes own distance to core.
c) the multicast announcement corresponding to this entry was received in within a time period equal to two multicast announcement intervals. This condition is used to ensure that a neighbor is still in the neighborhood.
3.8 Forwarding Multicast Data Packets

This parent field of the connectivity list entry for a particular neighbor corresponds to the node from which the neighbor received its best multicast announcement. This field allows nodes that are non-members to forward multicast packets towards the mesh of a group. A node forwards a multicast data packet it receives from its neighbor if the parent for the neighbor is the node itself. Hence, multicast data packets move hop by hop, until they reach mesh members. The packets are then flooded within the mesh, and group members use a packet ID to detect and discard packet duplicates.

3.9 Vulnerabilities in PUMA

The major drawback of the PUMA protocol is the lack of security mechanisms to ensure that the packets have reached the destination. There is no acknowledgement procedure that is present and hence no delivery validation. A malicious node is capable of bringing about the routing attacks by altering critical fields in the Multicast Announcement packet.

- A node can impersonate another node by sending a MA with its address as the originating address.
- A node can change the hop count information of advertise that it has the shortest route to the destination by sending in a MA.
- A node that is in the radio range of other distant nodes can repeatedly put down MA’s and not forward them thus leading to failure in path discovery.
- A node can unnecessarily keep flooding the network with MA’s thereby powering down a sizeable portion of the network.
- A node can alter its destination sequence number to a node by just sending two or more MA packets.
- A node can alter the time field to pretend in order to route Multicast Announcement through the node.

4. Generation of attacks

4.1 Blackhole Attack

In our protocol, connection is established through an Multicast Announcements, where in each node maintains a connectivity list. After receiving the multicast announcement from its neighbor it will stores the field information in connectivity list. In addition to this, it also stores the timestamp when it was received and the neighbor from which it was received in the connectivity list.

A shortest path is chosen based on distance to the core and the timestamp in the connectivity list that reaches the core ID. If the node is an attacker, then it will sends the MA with highest timestamp and smallest distance to core. So, the legitimate node will choose the attacker node as the intermediate and it will send the data packets through it. The attacker node may send the packet with some modification or delay or it may drop the packets.

![Figure 3: Blackhole Attack](image-url)
In this figure 3, node 5 has received the multicast announcement from its neighbors 4, 0, 6 and it stores the entries in its connectivity list. But here node 4 is the malicious. Node 5 wants to send the data packet to its core it forward it to the node from which it received its best multicast announcement. Here node 4 and node 0 have the same distance to the core. The malicious node will forward the control packet earlier than other legitimate node. So, it chooses node 4 as the best entry because the time when it received is earlier than the node 0. Now node 5 will send the data packet to node 4 as the intermediate node.

4.2 WORMHOLE ATTACK
In our protocol, every node will broadcast the Multicast Announcement to its neighbors to form a mesh. In case if the node is malicious then the multicast announcement could be tunneled directly to nodes near the destination target node through the wormhole link. Moreover, this multicast announcement arrives earlier than the other nodes due to the high quality of the wormhole link. This attack prevents any other routes from being discovered. This will happen within the single group.

A wormhole consists of two attackers far from each other, and a communication tunnel between them. Each attacker captures all the routing packets in its immediate neighborhood and tunnels them to the other worm to be replayed. A attacker or worm captures all the routing packets from one group say A, has them replayed by its peer worm in another group say, and vice versa. Hence, nodes in both groups believe they are immediate neighbors. The network topology is corrupted and routing is compromised.

Table 2: Connectivity list at node 5

<table>
<thead>
<tr>
<th>Neighbor</th>
<th>Multicast Announcement</th>
<th>Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Distance to core</td>
<td>Parent</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 3: Connectivity list at node B

<table>
<thead>
<tr>
<th>Neighbor</th>
<th>Multicast Announcement</th>
<th>Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Distance to core</td>
<td>Parent</td>
</tr>
<tr>
<td>H</td>
<td>2</td>
<td>C</td>
</tr>
<tr>
<td>L</td>
<td>4</td>
<td>K</td>
</tr>
<tr>
<td>N</td>
<td>3</td>
<td>H</td>
</tr>
</tbody>
</table>

Figure 4. Wormhole Attack

Table 2: Connectivity list at node 5

Connectivity list at node 5
Core ID=10, Group ID=224.0.0.1, Seq No=79

Table 3: Connectivity list at node B

Core ID=C, Group ID=224.0.0.1, Seq No=79
In this figure 4, node B has received the multicast announcement from its neighbors Y, L, and K and it stores the entries in its connectivity list. But here node X and Y are two colluding wormhole node with high speed tunnel. Node B wants to send the data packet to its core it forward it to the node from which it received its best multicast announcement. Here the wormhole node Y has the shortest distance to the core. So it chooses node Y as the best entry and it will send the data packet through this malicious node. This malicious node will receive the data packets and tunnel it to its peer node (i.e.) wormhole 2. This wormhole 2 will drop the data packet and sometimes it will modify and send it to the core.

5. Solution for Blackhole and Wormhole Attack

In a network of multiple groups, a node can be a core or a member to a preexisted group. If it’s a core node then it would have received the data from its neighbor. If it’s a receiver node then it would have sent data to neighbors. Nodes(receivers) in such a scenario must maintain the details of how many data packets it has sent or forwarded for the core of the group and how many data packets it has received from a group. Based on these details it calculates a trust value.

In order to store the trust value, the connectivity list is maintained by each node should include three additional fields named Trust Value_1, Trust Value_2 and Trust Value_3 which are shortly called as TV1, TV2 and TV3.

A trust value is defined as the ratio of no of data packets sent to the no of data packets received. To identify the misbehavior and to use the trust value to present it accurately, the routing strategy (i.e) route selection is based on not only the (present) trust values and also on the average value of the past experiences. If given more weight to the history of past experiences of the node than its recent behavior, which can prevent a node with high trust value that suddenly starts to drop package will be identified quickly turning to malicious node to disrupt network activities.

\[
\text{Trust Value} = \frac{\text{Number of data packets sent}}{\text{Number of data packets received}}
\]

\[
\text{Average Trust Value} = \frac{\text{Number of data packets sent}}{\text{Number of data packets received}}
\]

\[
\frac{\text{TV1} + \text{TV2}}{2} = \text{TV3}
\]

Where,

TV1 and TV2 are the trust values of past experiences of the node

TV3 is the current trust value of the node

Whenever a receiver wants to send data to the core, it selects the routing path based on the details of the fields in the connectivity list. They are shortest distance and the time of the multicast packets received. To prevent from these attacks, we can also consider the trust value calculated in addition to the above mentioned fields. Hence before the data’s are being routed through the neighboring nodes to the source which has initiated the Multicast Announcement it has to consider the above parameters and has to choose the path to forward the data. Since each node is going to maintain the trust value of its immediate neighbors its efficient enough to mitigate the adverse activities of an attacker.

6. Simulation and Results Analysis

We take the black hole and wormhole attack scenario in the protocol. The performance of packet delivery ratio is evaluated by computer simulation using ns 2.34 by putting our secure trust based mechanism. We also watch the difference between the normal PUMA operation and PUMA under attacks, in which some nodes are made to play the role of attackers at different simulation time.

The nodes in the computer simulation move according to the Random Waypoint Algorithm. The scenario is defined with a set of parameters as follows.

Number of nodes: 50
Number of malicious nodes: 10-40
Simulation Area: Ns 2.34
Data rate: 11mb
Packet Size: 1000
Traffic Type: CBR
Simulation Duration: 250 sec
The computer simulation results are shown in the figure 5 and 6. It can be seen that the packet delivery ratio is increased two times when the network is attacked. The packet delivery ratio is increased 50% when there are attack behaviors in the network. The delivery ratio of PUMA with our mechanism is almost same as PUMA without attack.

6. CONCLUSION

A complete analysis is done on the PUMA protocol and the vulnerabilities of the protocol are identified such as black hole attack, wormhole attack and flooding attack. A trust based solution and dynamic threshold mechanism has been proposed to mitigate those attacks. Our future enhancement is to select the other multicast routing protocols which are susceptible to routing attacks and provide the secure solution to thwart against those security attacks.

REFERENCES


