CHAPTER 4

NON CRYPTOGRAPHIC SOLUTION

The on demand routing protocols for mobile AdHoc networks use network-wide broadcasts of control packets to learn routes. This feature can be exploited by malicious nodes to launch highly leveraged denial of service attacks in AdHoc networks. The privacy issues and non-cryptography based security solutions can be under the research community work mainly seen from data mining and machine learning area. Techniques from statistics, combinatorics, data mining can be helpful in privacy of AdHoc network area if they are applied (Aggarwal and Philip Yu 2008). Marchette (2003) has identified denial of service attacks as a basic area in which statistics can play an important role in network security. Brutch and Ko (2003) presented a brief survey of current research in intrusion detection systems for AdHoc networks and reviewed current research efforts to detect attacks against the AdHoc routing infrastructure, and illustrated the challenges to intrusion detection.

4.1 INTRUSION DETECTION SYSTEM

The main function of intrusion detection systems is to classify network traffic. Based on classification rules, systems are classified into two types, namely misuse detection based on “signature” and anomaly detection based on “pattern”. A misuse detection system looks at the packets in network traffic and compares the data sources for a “signature” that correspond to attack data, and issues an alert if any suspicious activity has been identified. Although the idea of using a signature for detection is simple, it is understood
that the development of such a system is labor and time intensive, and is consequently one-step behind the attacks. Anomaly detection identifies behaviour deviations from the normal or anomaly-free behaviour patterns. Theoretically, such a system may be able to detect some new attacks but it is difficult in practice because it is hard to define an anomaly-free behaviour pattern. In general, statistical anomaly detection can be categorized into two aspects: threshold based detection and profile-based detection. The threshold method evaluates the frequency of anomalous events occurring over a specified interval and, by itself, has higher false positive and false negative rates. By contrast, the profile-based method focuses on analyzing current or historical user behaviour and detects any outlier values based on a series of measures such as mean, medium, standard deviation, or interval estimates of various user activity-related parameters and variables (Yun Wang 2009).

4.2 PROFILE BASED INTRUSION DETECTION SYSTEM FOR BLACK HOLE ATTACK

Though there are many solutions, in literature, to avoid black hole nodes, they are based either on the sequence number in the RREP or on the time of arrival of RREP to the source. But, there is a need for a solution against black hole node, considering both sequence number and the RREP arrival time. This chapter proposes two routing protocols Secure AdHoc On Demand Distance Vector Routing protocol against Black hole (SBAODV) and Secure AdHoc On Demand Multipath Distance Vector Routing protocol against Black hole (SBAOMDV) based on the well known routing protocols AODV and AOMDV respectively. These two protocols use profile based statistical intrusion detection technique to detect the black hole node. These protocols analyzes sequence number in the current RREP and detects any outlier values based on the measures of mean of difference in sequence numbers of previous RREPs.
The proposed protocol (SBAODV) enhances the conventional AODV with two additional modules. The modules are:

1. Caching mechanism to collect RREPs
2. Dynamic Threshold updation.

The correct RREP selection procedure in the conventional AODV is used along with the Dynamic Threshold Updation process.

4.3 CACHING MECHANISM

In these protocols, the caching mechanism at the source node is introduced to collect the RREPs for a particular RREQ (Nital et al 2010) arrived within a particular period of time, namely WAIT-TIME. In the original AODV protocol, by default, the source node accepts the first fresh enough RREP coming to it. After broadcasting a RREQ, a node waits for a RREP. If a RREP is not received within NET_TRAVERSAL_TIME milliseconds, the node will try again to discover a route by broadcasting another RREQ. The NET_TRAVERSAL_TIME in original AODV is calculated using the equation (4.1).

\[
\text{NET_TRAVERSAL_TIME} = 2 \times \text{NODE_TRAVERSAL_TIME} \times \text{NET_DIAMETER} \quad (4.1)
\]

The default value of NODE_TRAVERSAL_TIME is 40 milliseconds and NET_DIAMETER is 35. Thus the default value of NET_TRAVERSAL_TIME is 2800 milliseconds.

In this proposal, the caching mechanism stores all the RREPs in the newly created table “Cache_RREP_Tab” until the time, WAIT_TIME after receiving the first RREP. Based on the heuristics, WAIT_TIME is set to be half the value of NET_TRAVERSAL_TIME (NTT) milliseconds.
Table 4.1 shows the number of RREPs received and the latency in the route discovery process with the variation in the WAIT_TIME, when total number of nodes is 50.

**Table 4.1 Observation with Variable WAIT_TIME**

<table>
<thead>
<tr>
<th>S.No</th>
<th>WAIT_TIME (ms)</th>
<th>Average number of RREPs received</th>
<th>Routing latency/Route discovery</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NTT</td>
<td>16.3</td>
<td>NTT</td>
</tr>
<tr>
<td>2</td>
<td>NTT / 1.25</td>
<td>14.9</td>
<td>NTT / 1.25</td>
</tr>
<tr>
<td>3</td>
<td>NTT / 1.50</td>
<td>11.6</td>
<td>NTT / 1.50</td>
</tr>
<tr>
<td>4</td>
<td>NTT / 1.75</td>
<td>9.2</td>
<td>NTT / 1.75</td>
</tr>
<tr>
<td>5</td>
<td>NTT / 2</td>
<td>7.1</td>
<td>NTT / 2</td>
</tr>
<tr>
<td>6</td>
<td>NTT / 3</td>
<td>3.2</td>
<td>NTT / 3</td>
</tr>
<tr>
<td>7</td>
<td>NTT / 4</td>
<td>1.05</td>
<td>NTT / 4</td>
</tr>
</tbody>
</table>

Thus, with the presence of five black holes, the worst case scenario is all the five black holes are sending fake RREPs for a particular RREQ. So, the source node must require a maximum of six RREPs for a particular RREQ to find out the black hole node. From the above table, it is observed that, when the WAIT_TIME is set as NTT/2, the average number of RREP in “Cache_RREP_Tab” is 7.1 and the routing latency is equal to NTT/2. For WAIT_TIME is equal to NTT/1.75, an average 9.2 RREPs were received, but the delay in route discovery is NTT/1.75, which is higher than the previous one. So, setting NTT/2 for WAIT_TIME is the best choice. It does so, by calling the method viz. the `Pre_ReceiveReply()` method.

The pseudo code for collecting multiple RREPs within the period WAIT_TIME is given below:
Pre_ReceiveReply (Packet P) {
    t_0 = get(current time value)
    t_1 = t_0 + WAIT_TIME
    while(CURRENT_TIME <= t_1) {
        Store Dest_Seq_No in RREP and NODE_ID in Cache_RREP_Tab table
    }
    while (Cache_RREP_Tab is not empty) {
        -----------------------------
        Goto ReceiveReply()
        -----------------------------
    }
}

Where   Dest_Seq_No --- Destination Sequence Number
        Src_Seq_No ----- Source Sequence Number

Thus, the source node after receiving first RREP control message
waits for WAIT_TIME. For this time, the source node will save all the
coming RREP control messages in Cache_RREP_Tab table.

4.4 RREP SELECTION AND DYNAMIC THRESHOLD
    UPDATION

For the RREPs selected from Pre_RecieveReply( ), the source node
compares the sequence number in the RREP with the threshold value, where
this threshold value is dynamically updated in a particular time interval.
As the value of the sequence number in RREP from a black hole node is
found to be higher than the threshold value, the node is suspected to be
malicious and it adds the node to the black list. After detecting the node as a
black hole node, the source node will send a new control packet to its
neighbours to indicate the presence of the black hole node and its identity.
Further, if any node receives the RREP packet, it looks over the list, if the
reply is from the blacklisted node, then no processing is done for the same. It
simply ignores the node and does not receive reply from that node again. In this way, the malicious node is isolated from the network.

The continuous replies from the malicious node are blocked, which results in less Routing overhead. The threshold value is dynamically updated (Payal Raj and Prashant Swadas 2009) using the data collected in the particular time interval. The threshold value is the average of the difference of destination sequence number in each time slot between the sequence number in the routing table and the RREP packet. The formula for the threshold at each time slot is given in equation (4.2).

Let \( S_{rt} \) - Destination sequence number in the routing table.

\( S_{rp} \) - Destination sequence number in the RREP Packet.

\( dS_i \) - Difference\((S_{rt}, S_{rp})\) at a time slot ‘i’

After \( N \) time slots, the threshold \( Th \) will be

\[
Th = \frac{1}{N} \sum_{i=1}^{N} dS_i
\]

(4.2)

The time interval to update the threshold value is as soon as a node receives a RREP packet. As a node receives a RREP for the first time, it gets the updated value of the threshold. So, the proposed routing protocol not only detects the black hole attack, but tries to prevent it further, by updating the threshold value which reflects the real changing environment. And other nodes are also updated about the malicious act, and they react to it by isolating the malicious node from the network.

The method of detecting the Black hole nodes based on the sequence number in the proposed SBAODV and SBAOMDV are same and it is given below:
At Source Node:

ReceiveReply (Packet P)
{
    if (the source of the RREP is not in Black List)
    {
        if (P has an entry in Route Table)
        {
            if(Dest_Seq_No in RREP > Threshold)
            {
                Node is suspected as a Black hole
                Discard RREP
                Add the node in the black list
                Update threshold value using equation (5.1)
                    }
            else
            {
                if(Dest_Seq_No in RREP > Dest_Seq_No in the Routing table)
                {
                    Update entry of P in routing table
                    Unicast data packets to the route specified in RREP
                }
                else
                {
                    Discard RREP
                }            }
        }
        else
        {
            if ((Dest_Seq_No in RREP >= Source_Seq_No)
                && (Dest_Seq_No in RREP < Threshold))
            {
                Make entry of P in routing table
            }
            else
            {
                Node is suspected as a Black hole
                Discard RREP
                Add the node in the black list
            }            }
        }
    else
    {
        Discard RREP // reply is from black hole node
    }    } // End of ReceiveRREP
Thus, in all situations, the black hole will be identified and isolated for further route update process. Thus, the effect of black hole nodes is reduced throughout the MANET. Figure 4.1 shows the flow chart for the entire route discovery process in SBAODV.

Figure 4.1 Flow Chart – Route Discovery Process
4.5 SUMMARY

In this chapter, two protocols SBAODV and SBAOMDV, independent of any cryptographic standard algorithms, were proposed. These two protocols will overcome the effect of black hole attack only. These solutions depend only on the change in sequence number to detect the black hole node. A clever malicious node can rescue itself from the mitigation technique with slightly higher sequence number that is not detected by this solution. The next chapter discusses one such proposal, which uses modified hashed message authentication code.