CHAPTER 5

NEIGHBOR CREDIT VALUE AODV ALGORITHM FOR A SECURE ROUTING IN MANET UNDER SELFISH NODE BEHAVIOR ATTACK

5.1 OVERVIEW

AODV is identified to be superior in the presence of attacks such as Selfish Behavior Attacks and blackhole attacks. NCV-AODV routing protocol is proposed to detect and prevent Selfish Behavior Attacks in MANET. An extensive analysis of this mechanism is executed against the Selfish Behavior Attack using AODV routing protocol and the results are depicted.

5.2 PROPOSED NCV-AODV

A simple credit based protocol named as NCV-AODV is proposed by altering the AODV routing protocol. The nodes in the network maintain a neighbor credit table and it would be updated periodically. The nodes credit would be increased in the table when a node forwards the packet to its neighbor. During this process, some nodes do not get the opportunity to forward the packet. Hence they may get a poor credit value because if a node does not forward the packet, the credit value will be decreased in the neighbor credit table. To rectify this problem, some dummy packets are sent by the nodes in the network to increase the credits in the neighbor credit table.
The credit value of the next hop node will be checked during the route resolving process and the forwarding activity will be carried out if the node has a minimum credit value. If not, it will leave the packet. The modifications done in an AODV to function as NCV-AODV are explained.

5.3 PROPOSED DETECTION MECHANISM IN NCV-AODV

The function named as AODV::recv is initiated at routing layer while receiving the packet. The credit value of the source node as well as the next hop would be updated whenever the data packet is received from a previous hop node which is shown in Figure 5.1.

```
OnReceivingAPacket(Pkt) {
    If (PT_AODV) {
       RecvAODV_PKT(Pkt, AODVPktType)
    } else{
        ....
        //Packet forwarding....
        IncreaseCredit(Pkt->src);
        IncreaseCredit(Pkt->PrevHop);
        Forward(Pkt)
    }
}
```

**Figure 5.1 Modifications in receive function of NCV-AODV**

The route lookup function is activated for finding the path to destination at routing layer as depicted in Figure 5.2. It explains that the credit value of the next hop node is considered during route resolving method. The route selected will be deactivated sometimes if a node forwards packet to a neighbor node which holds a credit value that is less than a threshold value. Hence the next potential route is determined in future.
Figure 5.2 Route Lookup of NCV-AODV-table function

Figure 5.3 depicts the On Selfish Behavior Detection Timer function included in AODV to initiate dummy data packets in a good manner. To avoid message overhead and flooding in the network, the dummy data packets were transmitted in a one hop broadcast with some long interval. Hence a few dummy data packets were sent by the genuine nodes to their neighbors to acquire finest credit value at their NCT. If a node does not send or receive any packet, then that node is categorized as malicious node. So, by implementing this mechanism, the genuine nodes are not considered as malicious node. The nodes which are participating actively in the communication would not use this timer function. By this way, additional overhead is avoided in the present research.

Figure 5.3 Selfish Behavior Detection Timer of NCV-AODV
Many cooperation enhancing techniques based on credit based mechanism would identify the node which does not get a chance to be cooperative in sending or receiving the packet as malicious node, but the proposed mechanism would avoid this false detection of nodes.

The function IncreaseCredit( ) is shown in Figure 5.4 which will award a credit to the node if it forwards the packet to its neighbor.

```
IncreaseCredit(NodeID) {
    NeighborCreditList(NodeID)++
}
```

**Figure 5.4 IncreaseCredit(NodeID) Function Added in NCV-AODV**

The Send Dummy Data function is shown in Figure 5.5. The timer function initiates the SendDummyData() function in a periodical manner. The nodes which remain idle would initiate the function to send data. Hence the overhead is prevented in this mechanism.

```
SendDummyData() {
    If (NodeIsIdleForSomeTime) {
        SendEmptyDataPacket()
    }
}
```

**Figure 5.5 SendDummyData() Function Added in NCV-AODV**

In AODV code, the malicious behavior is prevented in two positions. The malicious function of the nodes is prevented in two places of AODV code. The route lookup phase is the first place and RecvAODV packet is the next place. Figure 5.6 depicts the second phase to avoid the malicious node’s request in the network. The main function of second phase is that the
packet will be dropped if it is obtained from a neighbor node whose credit is less during routing process.

```c
RecvAODVPkt (Pkt, AODVPktType) {
    If(Detection) {
        If (GetCredit(Pkt->PrevHop) < CreditThreshold)
            // drop it if packet from low credit node
        Free(Pkt);
        Return();
    }
    switch(AODVPktType) {
        case TYPE_RREQ:
           RecvRequest(Pkt)
            break
        case RREP:
           RecvReply(Pkt);
            break
        case RERR:
           RecvError (Pkt);
            break
        case HELLO:
            } // end of Switch case
    } // end ofRecvAODVPacket
```

**Figure 5.6 Pseudocode for NCV-AODV:: recvAODV**

GetCredit(NodeID) is depicted in Figure 5.7. The objective of this function is to show the credit value of the node from neighbor credit list.

```c
GetCredit (NodeID) {
    Return
    (NeighborCreditList(NodeID))
}
```

**Figure 5.7 GetCredit(NodeID) of NCV-AODV**

In the present study, one of the AODV routing agent codes would perform as a normal AODV or malicious AODV or NCV-AODV. The reason behind this mechanism is that in a single node, more than one routing agents cannot be activated. So, one routing agent code that can function more than one way is designed for initialization parameters.
5.4 SIMULATION PARAMETERS

The simulation parameters are already represented in section 3.3. AODV states used in this simulation are NCV-AODV, normal AODV and AODV under Selfish Behavior Type I attack.

5.5 RESULTS AND DISCUSSION FOR NCV-AODV

The performance of the proposed mechanism is simulated by varying the nodes as 30, 40 and 50 and malicious node count is 5. The performance is analyzed for AODV without malicious node, AODV under Selfish Behavior Type I attack, NCV-AODV under Selfish Behavior Type I attack. The performance metrics considered in simulation are: packets maliciously dropped at Routing Layer, dropped packets at Application Layer, Throughput, PDF, End-to-End Delay and Routing Load.

5.5.1 Packets Maliciously Dropped at Routing Layer

Figure 5.8 shows the comparison of AODV without malicious nodes, AODV under Selfish Behavior Type I attack and NCV-AODV under Selfish Behavior Type I attack for the performance metric packets maliciously dropped at routing layer. AODV without malicious nodes does not affect the routing layer and the performance is good. Due to malicious activity, congestion occurs in AODV under Selfish Behavior Type I attack and so more packets are dropped maliciously at routing layer. NCV-AODV under Selfish Behavior Type I attack drops an average of 12 packets maliciously at routing layer. Even under the influence of Selfish Behavior Type I attack, the proposed mechanism with the help of neighbor credit value forwards the packet in a good manner and drops less number of packets at routing layer. By increasing the number of nodes, NCV-AODV performs like a normal AODV.
Figure 5.8 Packets Maliciously Dropped at Routing Layer in NCV-AODV

5.5.2 Packets Dropped at Application Layer

Figure 5.9 compares the AODV without malicious nodes, AODV under Selfish Behavior Type I attack and NCV-AODV under Selfish Behavior Type I attack for the performance metric dropped packets at application layer. AODV without malicious nodes drops less number of packets with the increase in nodes to 50, but AODV under Selfish Behavior Type I attack drops more packets at 30, and gradually decreases at node 50 due to the mobility of nodes. The proposed mechanism NCV-AODV under Selfish Behavior Type I attack delivers the packets at destination, and hence it drops less number of packets at application layer.
Figure 5.9 Packets Dropped at Application Layer in NCV-AODV

5.5.3 Throughput

Figure 5.10 depicts the throughput comparison for NCV-AODV under Selfish Behavior Type I attack, AODV without malicious nodes, AODV under Selfish Behavior Type I attack. It is observed from the figure that AODV without malicious nodes and NCV-AODV under Selfish Behavior Type I attack are good with the increase in the node to 50.

At routing layer and application layer, AODV under Selfish Behavior Type I attack drops more packets, and hence shows a poor performance in throughput. Thus, under the influence of the Selfish Behavior Type I attack, throughput is improved in the proposed mechanism.
5.5.4 Packet Delivery Fraction (PDF)

Figure 5.11 depicts that the PDF of NCV-AODV under Selfish Behavior Type I attack is 73.96% which is equal to that of AODV without malicious nodes that produce the PDF of 80%. The PDF of AODV under Selfish Behavior Type I attack is 67.13%, and hence the proposed mechanism shows 7% of improvement under Selfish Behavior Type I attack. The reason is that the genuine nodes are not marked as malicious node, and hence packets are delivered successfully at the destination.
5.5.5 End-to-End Delay

Figure 5.12 shows the comparison of end-to-end delay for AODV without malicious nodes, AODV under Selfish Behavior Type I attack and NCV-AODV under Selfish Behavior Type I attack. AODV without malicious nodes shows low end-to-end delay and the performance is good. In the case of AODV under Selfish Behavior Type I attack, due to selfish behavior of the node, links may be broken and this results in frequent rerouting during packet forwarding, and hence the end-to-end delay is high, whereas NCV-AODV under Selfish Behavior Type I attack resolves the malicious node activity with the help of neighbor credit value and so the end-to-end delay is reduced and the performance is equal with the increase in the number of nodes to 50.

![Average End to End Delay](image)

**Figure 5.12 Average End-to-End Delay of NCV-AODV**

5.5.6 Routing Load

Figure 5.13 shows the comparison of the routing load for AODV without malicious nodes, AODV under Selfish Behavior Type I attack and NCV-AODV under Selfish Behavior Type I attack. AODV without malicious nodes shows a less routing load because the routing path is not affected by
malicious activity and so the packets are delivered successfully. Similarly, AODV under Selfish Behavior Type I attack exhibits a low routing load as it drops more packets due to selfish behavior, and hence the performance is equal to that of AODV without malicious nodes. In the case of the proposed mechanism, more packets are generated and so the routing load is high and it shows positive enhancement in the network.

![Average Routing Load](image)

**Figure 5.13 Average Routing Load of NCV-AODV**

### 5.6 SUMMARY

The proposed mechanism NCV-AODV routing algorithm implements dummy data packets to identify the genuine node and make those nodes to get enough credit to involve in routing process. The drawback in the existing credit based mechanism is rectified in the proposed mechanism. Moreover, these dummy data packets are sent in an interval by the idle nodes, and hence the overhead is avoided. The simulation results show the efficiency, reliability, and data handling capability of the proposed mechanism by producing a high throughput and PDF even in the presence of high routing load. End-to-End delay is an important parameter in which the higher values of this parameter would affect the throughput largely, and hence
this metric should always be minimized. This minimized end-to-end delay is achieved in NCV-AODV and so link failures are avoided in the network. Thus, NCV-AODV under Selfish Behavior Type I attack outperforms the traditional AODV in all the performance metrics.