Chapter 5

Active Blackhole Attack for AODV

5.1 Introduction

After studying the vulnerabilities that arise due to enhancement of protocols, we explore the effect of modifying the existing attacks on standard protocols. We mainly examine whether existing IDSs are capable of detecting such modified attacks. Standard AODV is vulnerable to a number of attacks namely, blackhole, wormhole, grayhole etc. [132]. In this contribution we concentrate on blackhole attack. In blackhole attack the malicious node sends falsified packets to the source node to make it believe that the best path to the destination is through the malicious node; once packets start moving to the destination node through the malicious node, it drops all the traffic leading to communication disruption between source and destination. A number of IDSs have been proposed to address blackhole attacks [10]. However, most of the schemes are host based leading to load on the battery of the sensor nodes. So the attacker can launch the attack multiple times, in spite of getting detected, but effecting the lifetime (power) of the sensor nodes. Secondly, most of these IDSs assume the most rudimentary form of blackhole attack, which may not always be the case. Thirdly, accuracy of these schemes is based on the time of arrival of the packets. This leads to lots of false alarms because delay in network at many times can
be due of genuine reasons like congestion etc. In this work, we first develop a modified version of the blackhole attack called "active blackhole attack", and show that it can easily defy existing IDSs for AODV. In other words, we study the effect of the proposed active blackhole attack on AODV with a traditional IDS. Following that, we design a network based IDS for this active attack and then show that accuracy of the proposed IDS scheme is much higher than the ones reported in the literature. Finally, NS-2 experimental results show the efficacy of the proposed IDS scheme in terms of better throughput, end-to-end delay, lifetime etc.

The chapter is organized as follows. Section 5.2 describes a survey on the reactive protocol AODV. In the same section, we illustrate the blackhole attack on AODV. Proposed active blackhole attack is presented in Section 5.3. Section 5.4 discusses the proposed IDS which can detect the active blackhole attack on AODV. Section 5.5 presents the performance evaluation of AODV under traditional blackhole attack and proposes the active blackhole attack. This section also compares the effect of active blackhole attack on AODV with and without using the proposed IDS. Section 5.6 concludes this chapter.

5.2 Preliminaries and background

5.2.1 Ad-hoc On-demand Distance Vector (AODV)

In Chapter 3, the routing protocol AODV was introduced with an example. In this chapter we re-look AODV in the context of freshness of packets in terms of Sequence Number (SN) \[15, 133\]. SN plays an important role in AODV to create a loop free route. SN serves as time stamps. They allow nodes to compare how fresh their information is in comparison to other nodes. A higher SN signifies a fresher route. Source Sequence Number (Src\_SN) or originator SN is the current SN to be used in the route discovery using Route Request (RREQ) packets. Similarly, Destination Sequence Number (Dst\_SN) is the
latest SN received in the past by the originator for any route towards the destination. If the source node is unaware about this $\text{Dst.SN}$ then it sets zero (unknown) as its value. Source (Destination) SN is increased when the source node initiates RREQ (when the destination node replies with Route Reply (RREP) packet).

![Figure 5.1: Example of an AODV scenario](image)

Figure 5.1 illustrates the flow of the RREQ and RREP messages in a scenario wherein a node $S$ wants to find a route to a node $D$. In the figure, the IP address of source node $S$ is denoted as $\text{Src.IP}$ and the IP address of destination $D$ is denoted as $\text{Dst.IP}$. Node $S$ broadcasts RREQ messages ($a_1$, $a_3$, $a_5$), which reach to the 1-hop neighbors of the node $S$. Node $I_1$ then re-broadcasts the request packet ($a_2$). Packet $a_1$ originates from $S$, so IP address of immediate source ($\text{Im.Src.IP}$) of $a_1$ is also $S$. $\text{Im.Src.IP}$ of packet $a_2$ is $I_1$. Since $a_1$ and $a_2$ both are route request packets and immediate destination address ($\text{Im.Dst.IP}$) of broadcasted packet is always 255, so $\text{Im.Dst.IP}$ of $a_1$ and $a_2$ are 255. RREQ packet $a_1$ is generated by source node $S$, so Hop Count (HC) of packet $a_1$ is 0. Since $I_1$ re-broadcasts $a_1$ as $a_2$, HC of $a_2$ is 1. $\text{Src.IP}$ and $\text{Dst.IP}$ of these two RREQ packets are $S$ and $D$, respectively. Request ID ($\text{RREQ.ID}$) of the RREQ packet is assumed to be 71 and $\text{Src.SN}$ is assumed
as 2897. As we assume that in the past no path to D has been established, so $Dst_{SN}$ is 0 (Unknown). Another intermediate node $I_4$ receives the RREQ message (a2) and this intermediate node has a valid route to the destination node, so $I_4$ need not re-broadcast the RREQ packet. The node $I_4$ can send a RREP packet to the source node on behalf of the destination.

Table 5.1: Request-reply flow of route $SI_1I_4I_6D$

<table>
<thead>
<tr>
<th>Attributes of Packet</th>
<th>RREQ</th>
<th>RREP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packet</td>
<td>a1</td>
<td>a2</td>
</tr>
<tr>
<td>$Im_{Src_IP}$</td>
<td>S</td>
<td>$I_1$</td>
</tr>
<tr>
<td>$Im_{Dst_IP}$</td>
<td>255</td>
<td>255</td>
</tr>
<tr>
<td>HC</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$Dst_IP$</td>
<td>$D$</td>
<td>$D$</td>
</tr>
<tr>
<td>$Dst_{SN}$</td>
<td>0 (Unknown)</td>
<td>3012</td>
</tr>
<tr>
<td>$Src_IP$</td>
<td>$S$</td>
<td>$S$</td>
</tr>
<tr>
<td>$Src_{SN}$</td>
<td>2897</td>
<td></td>
</tr>
<tr>
<td>RREQ_ID</td>
<td>71</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.1 illustrates this scenario. In this case, intermediate node $I_4$ generates a RREP packet b1 with $Src_{IP} S$, $Dst_{IP} D$ and HC 2. It may be noted that $I_4$ is an intermediate node that has an information about the route to the destination. $I_4$ also knows that the number of hops to the destination is 2. So, HC of RREP (b1) sent by $I_4$ is $0 + 2 = 2$. The reply packet b1 is unicast by $I_4$ to $I_2$, so $Im_{Src\_IP}$ of b1 is $I_4$ and $Im_{Dst\_IP}$ is $I_2$. Intermediate node $I_2$ receives the RREP (b1) and uncasts it to the originator $S$ as b2. In this case, instead of the destination node D, $I_4$ sets the SN for this RREP packet. The value of $Dst_{SN}$ of the RREP packet is fixed by $I_4$ as 3012 (assumed). Similarly, another path is found as $SI_2D$. Table 5.2 presents the details of the request-reply flow for this route. RREQ packet a3 is
broadcasted by the originator S. Intermediate node I₂ receives this request packet but I₂ has no active route to the destination D, so I₂ again broadcasts the RREQ packet as a₄. After receiving the RREQ, destination node D transmits a RREP b₃ towards the source node S; this RREP has the SN as 3077 (assumed).

<table>
<thead>
<tr>
<th>Attributes of Packet</th>
<th>RREQ</th>
<th>RREP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packet</td>
<td>a₃</td>
<td>b₃</td>
</tr>
<tr>
<td>Im_Src_IP</td>
<td>S</td>
<td>I₂</td>
</tr>
<tr>
<td>Im_Dst_IP</td>
<td>255</td>
<td>D</td>
</tr>
<tr>
<td>HC</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Dst_IP</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>Dst_SN</td>
<td>0 (Unknown)</td>
<td>3077</td>
</tr>
<tr>
<td>Src_IP</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Src_SN</td>
<td>2897</td>
<td></td>
</tr>
<tr>
<td>RREQ_ID</td>
<td>71</td>
<td></td>
</tr>
</tbody>
</table>

When source node S broadcasts the request packet, it is also received by another intermediate node I₃. I₃ receives the request packet as a₅ and re-broadcasts it, since I₃ does not have any fresh enough route to the destination D. Another intermediate node I₅ receives the RREQ packet a₆, broadcasted by node I₃. Node I₅ has a valid route to destination D, so I₅ initiates a RREP packet with the Dst_SN as 3156 (assumed). Since, I₅ is an intermediate node that has information about the route to the destination, I₅ also has the knowledge that number of hops to the destination is 1. So, HC of RREP (b₅) sent by I₅ is 0 + 1 = 1. Detailed information of RREQ and RREP packets of the route SI₃I₅D is described in Table 5.3.

From Figure 5.1, it is observed that the RREQ packets a₇ and a₈ are discarded by the
intermediate nodes \( I_2 \) and \( I_4 \), respectively, because these two nodes have already processed the RREQ packets with the same \( RREQ_ID \), i.e., 71. From Table 5.4 we try to conclude how SN and HC play an important role to select the best path between the originator and the destination.

<table>
<thead>
<tr>
<th>Attributes of Packet</th>
<th>RREQ</th>
<th>RREP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Im_Src_IP</td>
<td>S</td>
<td>I_3</td>
</tr>
<tr>
<td>Im_Dst_IP</td>
<td>255</td>
<td>255</td>
</tr>
<tr>
<td>HC</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Dst_IP</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>Dst_SN</td>
<td>0 (Unknown)</td>
<td>3156</td>
</tr>
<tr>
<td>Src_IP</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Src_SN</td>
<td>2897</td>
<td></td>
</tr>
<tr>
<td>RREQ_ID</td>
<td>71</td>
<td></td>
</tr>
</tbody>
</table>

We can observe from Table 5.4, that \( Dst\_SN \) of route \( SI_1I_4I_6D \) is 3012, \( Dst\_SN \) of route \( SI_2D \) is 3077 and \( Dst\_SN \) of route \( SI_3I_5D \) is 3156.

<table>
<thead>
<tr>
<th>Possible routes</th>
<th>Src_IP</th>
<th>Dst_IP</th>
<th>Dst_SN</th>
<th>HC</th>
</tr>
</thead>
<tbody>
<tr>
<td>( SI_1I_4I_6D )</td>
<td>S</td>
<td>D</td>
<td>3012</td>
<td>3</td>
</tr>
<tr>
<td>( SI_2D )</td>
<td>S</td>
<td>D</td>
<td>3077</td>
<td>1</td>
</tr>
<tr>
<td>( SI_3I_5D )</td>
<td>S</td>
<td>D</td>
<td>3156</td>
<td>2</td>
</tr>
</tbody>
</table>

\( SI_3I_5D \) is the route with the highest \( Dst\_SN \) (i.e., 3156) compared with the other two routes. So, the route \( SI_3I_5D \) is chosen by the source node \( S \) to communicate with the destination \( D \). It may be noted that, \( HC \) of route \( SI_2D \) (i.e., 1) is less than that of the
route $SI_3D$. However, destination SN always gets priority in the case of route discovery procedure because freshness of a route is decided only from SN.

### 5.2.2 Blackhole attack on AODV

Blackhole is one of the common Denial-of-Service (DoS) attacks in WSNs [132]. In this attack, the malicious node advertises itself as having the shortest path to the destination node. The destination SN ($Dst_{SN}$) determines the freshness of routing information in AODV. When the malicious node receives a RREQ message, it sends a false RREP packet to a source node to pose itself as the destination node or an immediate neighbor to the actual destination node.

![Figure 5.2: Example scenario of blackhole attack on AODV](image)

Also, the attacker must generate its RREP with high $Dst_{SN}$ and less HC. It may be noted that by declaring high SN the malicious node makes a false impression that the route passing through it is the latest. Similarly, lower HC implies that the path through the malicious node is the shortest. After receiving a number of RREP messages, a source node chooses a path with greatest destination SN and lowest HC to construct a route.
(giving priority to SN). Then the source node establishes a path through the adversary and it would forward all of its data packets to this node, which originally were intended for the genuine destination. After that, the malicious node drops data packets and never forwards them to the genuine destination. As a result, the source and the destination nodes are unable to communicate with each other, resulting in blackhole attack.

Figure 5.2 illustrates a traditional blackhole attack on AODV using an example. In this figure, source node S tries to establish a path between the nodes S and D. The originator S broadcasts a RREQ. When this request packet is received by other intermediates nodes, like I₁, I₂, I₃ they will re-broadcast this RREQ packets. But when the destination node D gets the request packet, it unicasts a RREP packet to the source node S. Even the other two intermediate nodes I₄ and I₅ transmit reply packets to the originator node S, since they have valid routes to the destination node D. When destination node initiates a RREP packet it sets a $Dst\_SN$. Intermediate nodes I₄ and I₅ also allocate some $Dst\_SN$ when they unicast RREP packets on behalf of the destination node. If there is a malicious node M among these intermediates nodes it sends a RREP with arbitrary high $Dst\_SN$ (which is expected to be higher than those sent by I₄, I₅ and D). Let attacker node M assign the value of $Dst\_SN$ as 3157, which is higher compared to 3012, 3077 and 3156. Node M also sets its HC as 1. Then source node considers the route through this attacker node as a fresh one and wants to establish a path through the node M. After establishing the route, S sends all the data packets to the destination D through the node M. The attacker node M drops data packets sent by S and as a result a successful blackhole attack is launched by malicious node M.

5.3 Proposed active blackhole attack for AODV

In this section, we discuss the proposed active blackhole attack. In this model, attackers are quite intelligent. Instead of sending a RREP with arbitrary high SN, the attacker node
captures all RREPs for a given RREQ. Following that the attacker node increments the sequence number of its RREP slightly, such that it is just higher compared to the SN sent by the normal nodes for the corresponding RREQ. This basic idea is that, anomaly IDS cannot detect the attack because the increase in SNs in the RREPs sent by the attacker nodes are slightly above the SNs in the RREPs sent by the normal nodes. In other words, the increase of the SN by malicious nodes are not high enough which can be detected by any statistical based anomaly IDS. So the attack crafts a false RREP with a bit higher SN (compared to the SN of captured RREPs) and low hop count. This falsified RREP is sent to source or nearest neighbor by unicast.

**Algorithm 5**: Create a RREQ table (\( Tab_{\text{AttackRREQ}} \)) for attacker

**Input**: RREQ packets

**Output**: \( Tab_{\text{AttackRREQ}} \)

1. Receive RREQ packet.

2. IF \( Tab_{\text{AttackRREQ}} \) is full
   
   DELETE oldest record.

   ELSE
   
   ADD \( RREQ\_ID^{RREQ} \), \( Src\_IP^{RREQ} \), \( Dst\_IP^{RREQ} \), \( Im\_Src\_IP^{RREQ} \), \( Src\_SN^{RREQ} \) in \( Tab_{\text{AttackRREQ}} \).

3. HALT

The proposed attack scheme is implemented in three parts. In first part, the attacker creates the attacker RREQ table (\( Tab_{\text{AttackRREQ}} \)). Algorithm 5 illustrates the necessary steps to create an attacker table. When attacker node receives a RREQ packet, it stores the request ID (\( RREQ\_ID^{RREQ} \)), IP address of the source node (\( Src\_IP^{RREQ} \)), IP address of the destination node (\( Dst\_IP^{RREQ} \)), IP address of the immediate source node (\( Im\_Src\_IP^{RREQ} \)) and the source sequence number (\( Src\_SN^{RREQ} \)) of that RREQ packet in the table, \( Tab_{\text{AttackRREQ}} \).
The second part of the scheme is described by another algorithm (i.e., Algorithm 6). In this phase of the scheme, attacker node generates another table (i.e., $Tab_{AttackRREP}$). When the malicious node receives a RREP packet, it starts a timer clock and waits for a certain interval of time.

**Algorithm 6:** Create a RREP table ($Tab_{AttackRREP}$) for attacker

**Input:** RREP packets, $Tab_{AttackRREQ}$

**Output:** $Tab_{AttackRREP}$

1. Repeat step 2 and 3 after every $\Delta t$
2. Receive RREP packet
3. IF $Tab_{AttackRREP}$ is full
   DELETE oldest record.
   ELSE
   IF $Src_{IP}^{RREQ} = Src_{IP}^{RREP}$ and $Dst_{IP}^{RREQ} = Dst_{IP}^{RREP}$
   ADD $Src_{IP}^{RREP}$, $Dst_{IP}^{RREP}$, $Im_{Src_{IP}}^{RREP}$, $Im_{Dst_{IP}}^{RREP}$, $Dst_{SN}^{RREP}$ and $HC^{RREP}$ in $Tab_{AttackRREP}$.
4. HALT

When attacker node receives RREP packets within this time span, it checks if the value of IP address of source node ($Src_{IP}^{RREP}$) and IP address of destination node ($Dst_{IP}^{RREP}$) are same as the values of $Src_{IP}^{RREQ}$ and $Dst_{IP}^{RREQ}$, respectively, of any entry in $Tab_{AttackRREQ}$. If it matches then the malicious node saves the values of originator IP address ($Src_{IP}^{RREP}$), destination IP address ($Dst_{IP}^{RREP}$), IP address of immediate source node ($Im_{Src_{IP}}^{RREP}$), IP address of immediate destination ($Im_{Dst_{IP}}^{RREP}$), destination sequence number ($Dst_{SN}^{RREP}$) and hop count ($HC^{RREP}$) of the received RREP packet in $Tab_{AttackRREP}$.

Algorithm 7, states the working procedure of the proposed active blackhole attack. This is the third part of the scheme. $Tab_{AttackRREQ}$ and $Tab_{AttackRREP}$ are used as
input elements to Algorithm 7. If no RREP packets are received within the time interval, then Tab\_AttackRREP will be empty.

**Algorithm 7**: Procedure of active blackhole attack

**Input**: Tab\_AttackRREQ, Tab\_AttackRREP

**Output**: Active Blackhole Attack

1. IF Tab\_AttackRREP is empty

CREATE RREP packet with randomly high $\text{Dst\_SN}^{\text{Attack}}$ and keep $\text{HC}^{\text{Attack}}$ to 1.

ELSE

SELECT $\text{High\_Dst\_SN}^{\text{RREP}} := \text{Highest \_\text{Dst\_SN}^{\text{RREP}}} \text{ and } \text{Min\_HC}^{\text{RREP}} := \text{Minimum \_\text{HC}^{\text{RREP}}} \text{ from Tab\_AttackRREP}$.

2. $\text{Src\_IP}^{\text{Attack}} := \text{Src\_IP}^{\text{RREQ}},$

$\text{Dst\_IP}^{\text{Attack}} := \text{Dst\_IP}^{\text{RREQ}},$

$\text{Im\_Dst\_IP}^{\text{Attack}} := \text{Im\_Dst\_IP}^{\text{RREP}},$

$\text{Im\_Src\_IP}^{\text{Attack}} := \text{Im\_Src\_IP}^{\text{RREP}},$

$\text{Dst\_SN}^{\text{Attack}} := \text{High\_Dst\_SN}^{\text{RREP}} + s$

IF $\text{Min\_HC}^{\text{RREP}} = 1$

$\text{HC}^{\text{Attack}} := 1.$

ELSE

$\text{HC}^{\text{Attack}} := \text{Min\_HC}^{\text{RREP}} - 1.$

3. CREATE RREP \{$\text{Src\_IP}^{\text{Attack}}, \text{Dst\_IP}^{\text{Attack}}, \text{Im\_Dst\_IP}^{\text{Attack}}, \text{Im\_Src\_IP}^{\text{Attack}}, \text{Dst\_SN}^{\text{Attack}}, \text{HC}^{\text{Attack}}$\}

4. Unicast the RREP to the $\text{Im\_Src\_IP}^{\text{RREP}}$ to establish a route through the attacker

5. STOP

When attacker node finds that the attacker RREP table is empty, it generates a RREP packet with a random destination sequence number ($\text{Dst\_SN}^{\text{Attack}}$) and $\text{HC}^{\text{Attack}} = 1$. But if the attacker table contains RREP packets, then the malicious node calculates the highest destination sequence number from all $\text{Dst\_SN}^{\text{RREP}}$, stored in the table Tab\_AttackRREP and assign the value in $\text{High\_Dst\_SN}^{\text{RREP}}$. After getting the value of $\text{High\_Dst\_SN}^{\text{RREP}}$ attacker node initiates a RREP packet with the destination sequence number ($\text{Dst\_SN}^{\text{Attack}}$) set
as $High_{Dest\_SN^{RREP}} + s$, where $s$ is a constant and the value of $s$ can be chosen based on requirements. Similarly, value of hop count ($HC^{Attack}$) is set as 1 if $Min_{HC^{RREP}}$ is 1, otherwise it is set as $Min_{HC^{RREP}} - 1$. Then the RREP is unicast to the next intermediate node (i.e., $Im\_Src\_IP^{RREP}$).

Figure 5.2 illustrates the scenario of the active blackhole attack. Let the malicious node $M$ receive a RREQ packet broadcasted by source node $S$ for destination node $D$. Table 5.5 shows all the RREQ packets, received by the attacker node $M$. The attacker node $M$ gets the RREQ packet $a_9$ first and starts the timer, initialized with current time $t_1$ and waits up to time $t_2$.

$t_2 - t_1 = \Delta t$, where $\Delta t$ is waiting time interval.

From Table 5.5, value of $t_1$ is 1.2 mins, i.e., 72 sec and the value of $t_2$ is assumed as 82 sec (for a scenario).

Therefore, $\Delta t = t_2 - t_1 = 10$ sec.

The node $M$ receives another two RREQ packets $a_{10}$ and $a_{11}$ with $RREQ\_ID$ as 71. Thus, both the packets are discarded by $M$. Attacker node $M$ collects all RREPs within the time interval $\Delta t$. If the source IP ($Src\_IP^{RREP}$) and destination IP ($Dest\_IP^{RREP}$) of the received RREP packet matches with the source IP ($Src\_IP^{RREQ}$) and destination IP ($Dest\_IP^{RREQ}$) of any entry of Table 5.5, attacker node stores the details of the RREP packet in the Table 5.6.

Table 5.5: Tab_AttackRREQ : RREQ table for malicious node $M$

<table>
<thead>
<tr>
<th>No.</th>
<th>Packet</th>
<th>RREQ_ID</th>
<th>Src_IP</th>
<th>Dst_IP</th>
<th>Im_Src_IP</th>
<th>Src_SN</th>
<th>Rcv_Time (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>a9</td>
<td>71</td>
<td>S</td>
<td>D</td>
<td>S</td>
<td>2897</td>
<td>1.2</td>
</tr>
</tbody>
</table>

The malicious node $M$ can sniff only the RREP packets $b_2$, $b_4$ and $b_6$, because these reply packets are within its range. Thus, it keeps the values of $Src\_IP^{RREP}$, $Dest\_IP^{RREP}$, $Im\_Src\_IP^{RREP}$, $Im\_Dest\_IP^{RREP}$, $HC^{RREP}$ and $Rcv\_Time$ of those packets in Table 5.6. The attacker node $M$ stops recording when clk reaches $t_2$. It may be noted that RREPs are unicast by broadcast and packets are not encrypted. So all nodes receive these RREPs.
Genuine nodes only process those unicast packets which are destined for them. But malicious node processes all packets, even unicast ones and not destined to it.

**Table 5.6: **Tab\_AttackRREP \(:\) RREP table for the malicious node M

<table>
<thead>
<tr>
<th>No.</th>
<th>Packet</th>
<th>Src IP</th>
<th>Dst IP</th>
<th>Im_Src IP</th>
<th>Im_Dst IP</th>
<th>Dst_SN</th>
<th>HC</th>
<th>Rcv_Time (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>b4</td>
<td>S</td>
<td>D</td>
<td>I_2</td>
<td>S</td>
<td>3077</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>2</td>
<td>b2</td>
<td>S</td>
<td>D</td>
<td>I_1</td>
<td>S</td>
<td>3012</td>
<td>3</td>
<td>1.32</td>
</tr>
<tr>
<td>3</td>
<td>b6</td>
<td>S</td>
<td>D</td>
<td>I_3</td>
<td>S</td>
<td>3156</td>
<td>2</td>
<td>1.325</td>
</tr>
</tbody>
</table>

Malicious node M stores three RREP packets whose $\text{Src\_IP}^{\text{RREP}}$ is $S$ and $\text{Dst\_IP}^{\text{RREP}}$ is $D$. After the completion of $\Delta t$ time interval, M finds the highest destination sequence number ($\text{High\_Dst\_SN}^{\text{RREP}}$) from the destination sequence number of the RREP packets b2, b4 and b6. The attacker table shows that 3156 is the value of $\text{High\_Dst\_SN}^{\text{RREP}}$ in a particular time interval $\Delta t$. Then M will reset its timer clk and create a falsified RREP with destination sequence number ($\text{Dst\_SN}^{\text{Attack}}$) 3157 and $\text{HC}^{\text{Attack}}$ 1, since the $\text{Min\_HC}^{\text{RREP}}$ obtained from Table 5.6 is 1. This falsified packet is sent as unicast to the source node. Now, 3157 is the highest destination sequence number for all the RREPs for a given RREQ. Thus, the originator assumes that the path through the malicious node is fresher than other paths and HC is also less. Therefore, the path through the malicious node is selected leading to blackhole attack. It may be noted that most of the existing IDSs cannot detect this attack because the sequence number of the falsified RREP is not arbitrarily high in comparison of other genuine RREPs.

### 5.4 Proposed IDS to detect the active blackhole attack

In this section an IDS is proposed to detect the active blackhole attack on AODV routing protocol. A sensor network is nothing but a collection of sensor nodes. Sensor Monitors (SMs) are special kind of sensors which observe request-reply flow, incoming and outgoing
from the cluster and try to detect malicious packets. In other words, the IDS is executed in the SM. The whole network is partitioned into some clusters and each cluster will be monitored by a SM.

**Algorithm 8:** Create a RREQ table (TabSM.RREQ) for sensor monitor

**Input:** RREQ packets  
**Output:** TabSM.RREQ

1. Receive a RREQ packet.
2. IF TabSM.RREQ is full
   DELETE oldest record.
ELSE
   ADD RREQ_ID^{RREQ}, Src_IP^{RREQ}, Dst_IP^{RREQ}, Src_SN^{RREQ}, Im_Src_IP^{RREQ} and Rcv_Time^{RREQ} in TabSM.RREQ.
3. HALT

This detection mechanism is presented in three parts. In the first part of the scheme, the SM prepares a table and collects some data for its later usage. When a monitor observes a new RREQ packet, it will start a timer and also store the details of the RREQ packet in the table TabSM.RREQ. This clock is initialized with the time T1. Algorithm 8 prepares this RREQ table for SM. It stores the values of RREQ_ID^{RREQ}, Src_IP^{RREQ}, Dst_IP^{RREQ} and Src_SN^{RREQ}, Im_Src_IP^{RREQ} of the received RREQ packet. It also saves the receive time (Rcv_Time^{RREQ}) of the RREQ packet.

Now the sensor monitor waits for RREP packets. SM accepts all those reply packets whose Src_IP^{RREP} and Dst_IP^{RREP} are similar with Src_IP^{RREQ} and Dst_IP^{RREQ}, respectively and saves all these RREPs with time stamp in sensor monitor RREP table, i.e., TabSM.RREP. This table keeps the values of Src_IP^{RREP}, Dst_IP^{RREP}, Im_Src_IP^{RREP}, Im_Dst_IP^{RREP}, Dst_SN^{RREP} and HC^{RREP} of RREP packets. So the table, TabSM.RREP collects all the RREP packets within a certain interval of time, say ΔT (where ΔT = T2 -
T1). This is the second part of the scheme and this part of the scheme is implemented by Algorithm 9.

**Algorithm 9:** Create a RREP table (TabSM,RREP) for sensor monitor

**Input:** RREP packets, TabSM,RREQ

**Output:** TabSM,RREP

1. Repeat step 2 and 3 after every $\Delta T$
2. Receive RREP packet
3. IF TabSM,RREP is full
   
   DELETE oldest record.

   ELSE

   IF Src_IP$^{RREQ}$ = Src_IP$^{RREP}$ and Dst_IP$^{RREQ}$ = Dst_IP$^{RREP}$

   ADD Rcv_Time$^{RREP}$, Src_IP$^{RREP}$, Dst_IP$^{RREP}$, Im_Src_IP$^{RREP}$, Im_Dst_IP$^{RREP}$, Dst_SN$^{RREP}$ and HC$^{RREP}$ in TabSM,RREP.

4. HALT

Algorithm 10 describes the detection strategy of the SM. After a time interval $\Delta T$, if SM obtains an empty TabSM,RREP, then it exits from the procedure. Otherwise SM calculates the highest destination sequence number (High,Dst_SN$^{RREP}$) from the Dst_SN$^{RREP}$ of those RREPs, which are captured by SM within the interval $\Delta T$. If the reply packet with highest destination sequence number is the last entry of the sensor monitor table, i.e., the RREP with the highest sequence number has arrived last, it implies that there is a possibility that the malicious node has observed all RREPs and then sent the falsified reply with highest sequence number. Thus, SM marks this Im_Src_IP$^{RREP}$ as a malicious node and it saves this Im_Src_IP$^{RREP}$ in the list of Malicious_IP.

To confirm this suspected node as an attacker node, SM applies active probing technique. SM generates a RREQ for a non-existing destination node. After a certain time duration, SM creates a false RREP message in response of its RREQ. As this destination
does not exist in this network, so any RREP packet should not come for this RREQ. If the IP under question (from the Malicious IP list) is that of an attacker node, it sends an RREP packet for this RREQ with a bit higher SN and low HC.

**Algorithm 10: Active blackhole attack detection procedure**

**Input:** TabSM.RREP  
**Output:** Active blackhole attack detection  

1. IF TabSM.RREP is empty  
   STOP  
ELSE
   (a) SELECT High Dst SN$^{RREP}$ := Highest Dst SN$^{RREP}$ and Min HC$^{RREP}$ := Minimum HC$^{RREP}$ from TabSM.RREP.  
(b) IF RREP with the value of High Dst Seq$^{RREP}$ and Min HC$^{RREP}$, is the last entry of Tab.SM, i.e., Rcv_Time is highest  
   Add to Malicious_IP = Src_Node$^{RREP}$ of last entry of TabSM.RREP  
   ELSE  
   STOP  
(c) Broadcast RREQ probe for non-existing Dst_IP  
(d) Generates false RREP for RREQ prob  
(e) IF get RREP from Malicious_IP  
   SET Blocked_IP = Malicious_IP  
   ELSE  
   STOP  

The attacker node is unaware about this destination node as it is non-existing, but still says that it has a path to the destination; this as per the philosophy of the attacker node to send RREP for any RREQ, whether it has a path to the destination or not. Then SM can easily detect this adversary and move this IP from the list of Malicious_IP to the list of Blocked_IP. The SM will inform all nodes over a secured channel about the IPs in the list Blocked_IP. It may be noted that as communication over secured channel is power hungry,
we use this communication only for sending the IPs in Blocked\_IP, which is very sparse event.

Figure 5.3, presents an example of a simple WSN scenario to illustrate the scheme. In this figure, SM\(_1\), SM\(_2\), SM\(_3\), SM\(_4\) all are the SMs. Assume that in a certain instance S, I\(_2\), I\(_3\), I\(_5\) are situated within the range of monitor SM\(_2\). Malicious node M also belongs to the range of SM\(_2\). S tries to establish a route between S and D. Thus, originator S broadcasts a RREQ message with RREQ\_ID as 71 and when SM\(_2\) gets a RREQ packet a1 at time 1.182mins, it starts a timer. SM\(_2\) receives a number of RREQ packets, i.e., a1, a2, a3, a4, a5, a6, a9 etc.
with $RREQ_{ID}$ as 71, shown in Table 5.7. The monitor SM$_2$ also stores the details of the RREP packets, having $Src_{IP}^{RREP}$ as S and $Dst_{IP}^{RREP}$ as D, in Table 5.8.

<table>
<thead>
<tr>
<th>Attributes of Packet</th>
<th>RREQ 1</th>
<th>RREQ 2</th>
<th>RREQ 3</th>
<th>RREQ 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rcv_Time</td>
<td>1.182</td>
<td>1.2</td>
<td>1.225</td>
<td>1.195</td>
</tr>
<tr>
<td>Packet</td>
<td>a1</td>
<td>a3</td>
<td>a4</td>
<td>a5</td>
</tr>
<tr>
<td>Im_Src_IP</td>
<td>S</td>
<td>S</td>
<td>I$_2$</td>
<td>S</td>
</tr>
<tr>
<td>HC</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Dst_IP</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>Src_IP</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Src_SN</td>
<td>2897</td>
<td>2897</td>
<td>2897</td>
<td>2897</td>
</tr>
<tr>
<td>RREQ_ID</td>
<td>71</td>
<td>71</td>
<td>71</td>
<td>71</td>
</tr>
</tbody>
</table>

The SM stops its timer at the time 1.432mins.

Thus, in this example $T_1 = 1.182$mins and $T_2 = 1.432$mins

Therefore, $\Delta T = T_2 - T_1 = 15$sec

After 15sec, SM$_2$ checks its table and finds four RREP packets. Receive time of three RREP packets $b_2$, $b_4$, $b_6$ are 1.32mins, 1.3mins and 1.325mins, respectively.

<table>
<thead>
<tr>
<th>No.</th>
<th>Packet</th>
<th>Src_IP</th>
<th>Dst_IP</th>
<th>Im_Src_IP</th>
<th>Im_Dst_IP</th>
<th>Dst_SN</th>
<th>HC</th>
<th>Rcv_Time (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>b4</td>
<td>S</td>
<td>D</td>
<td>I$_2$</td>
<td>S</td>
<td>3077</td>
<td>1</td>
<td>1.3</td>
</tr>
<tr>
<td>2</td>
<td>b2</td>
<td>S</td>
<td>D</td>
<td>I$_1$</td>
<td>S</td>
<td>3012</td>
<td>3</td>
<td>1.32</td>
</tr>
<tr>
<td>3</td>
<td>b6</td>
<td>S</td>
<td>D</td>
<td>I$_3$</td>
<td>S</td>
<td>3156</td>
<td>2</td>
<td>1.325</td>
</tr>
<tr>
<td>4</td>
<td>b6</td>
<td>S</td>
<td>D</td>
<td>M</td>
<td>S</td>
<td>3157</td>
<td>1</td>
<td>1.41</td>
</tr>
</tbody>
</table>

Last one is received at time 1.4 mins and $Dst_{SN}$ of this last RREP packet is also higher than other three RREPs. Then the monitor SM$_2$ suspects the node M as a malicious one.
and includes the $Im\_Src\_IP^{RREP}$ of this message in the list Malicious_IP. Now monitor sends a probe RREQ to a non-existing destination node. After a time interval SM$_2$ generates a false RREP for that RREQ. No other genuine RREP packet will arrive because that request packet was broadcasted for a non-existing destination node. But node M sends a reply packet with a destination sequence number just higher than the one sent in the RREP by SM$_2$. Now SM$_2$ can easily identify the node as an attacker and save this IP as a Blocked_IP. After getting the information all other sensors of the network would block this IP and discard all the data packets coming from that malicious node.

### 5.5 Performance evaluation

The performance evaluation of the proposed IDS has been studied in this section based on simulation using NS-2 using the parameters shown in Table 5.9. As shown in the table, for simulation we consider an area of 1000 m * 1000 m and 5-100 nodes randomly deployed in this area.

<table>
<thead>
<tr>
<th>No.</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Simulator</td>
<td>NS-2(Version 2.34)</td>
</tr>
<tr>
<td>2</td>
<td>Operating System</td>
<td>Linux(Redhat 5)</td>
</tr>
<tr>
<td>3</td>
<td>Channel Type</td>
<td>Channel/Wireless Channel</td>
</tr>
<tr>
<td>4</td>
<td>Traffic Model</td>
<td>Constant Bit Rate (CBR)</td>
</tr>
<tr>
<td>5</td>
<td>Source type</td>
<td>UDP</td>
</tr>
<tr>
<td>6</td>
<td>Area (m* m)</td>
<td>1000 * 1000 (initially)</td>
</tr>
<tr>
<td>7</td>
<td>Number of sensor node</td>
<td>5-100</td>
</tr>
<tr>
<td>8</td>
<td>Simulation Time</td>
<td>500 s</td>
</tr>
<tr>
<td>9</td>
<td>Routing Protocol</td>
<td>AODV</td>
</tr>
<tr>
<td>10</td>
<td>Antenna</td>
<td>Omni-directional</td>
</tr>
</tbody>
</table>

The initial battery power of each node is set to 20 Joules. Here the rate of packet generation is set to 1 packet/s. Performance of protocols, IDSs, etc. in WSNs are generally
measured in terms of the following parameters.

- **Packet Delivery Ratio (PDR):** PDR is the ratio of the number of packets delivered to the number of packets generated by the source.

- **End-to-End Delay:** End-to-End delay is defined as the average transit time of a packet, i.e., the time taken for a packet to reach the destination from the source.

- **Throughput:** Throughput is computed as the amount of data transferred divided by the data transfer time.

![Comparison of Packet Delivery Ratio](image)

**Figure 5.4:** PDR in AODV under normal condition and active blackhole attack

For evaluating the proposed IDS against active blackhole attack we have not considered any additional parameter. It may be noted that in any blackhole attack the adversary nodes drop all the received packets thereby effecting mainly the three parameters mentioned above.

PDR and throughput are directly effected by blackhole attack while end-to-end delay is increased in an indirect manner. Due to dropping of packets by an adversary node, the
corresponding routing path is considered damaged and new path needs to be discovered, leading to more delay.

![Comparison of Packet Delivery Ratio](image)

**Figure 5.5:** PDR in AODV under active blackhole attack with and without traditional IDS

![Comparison of Packet Delivery Ratio](image)

**Figure 5.6:** PDR in AODV under simple blackhole attack with and without traditional IDS

First we report our experimental results for PDR. Figure 5.4 illustrates the PDR of AODV under normal condition and the proposed active blackhole attack versus the num-
ber of nodes. It can be observed that PDR can be greatly compromised by the active blackhole attack. The reason is obvious, because the blackhole attack attempts to divert all routing paths through adversary nodes that drop all packets.

![Comparison of Packet Delivery Ratio](image1)

**Figure 5.7:** PDR in AODV under active blackhole attack with and without the proposed IDS

![Comparison of End to End Delay](image2)

**Figure 5.8:** Delay in AODV under normal condition and active blackhole attack

Then in Figure 5.5 we present the PDR of AODV under active blackhole attack, with
and without the traditional IDS. It may be noted that impact of active blackhole attack on PDR is same with and without the traditional IDS.

![Comparison of End to End Delay](image)

**Figure 5.9:** Delay in AODV under active blackhole attack with and without traditional IDS

![Comparison of End to End Delay](image)

**Figure 5.10:** Delay in AODV under simple blackhole attack with and without traditional IDS

In other words, the drop in PDR caused by *active* blackhole attack cannot be circumvented by the traditional IDSs developed mainly for blackhole attacks in AODV [132].
However, it may be noted that simple blackhole attack on AODV can be easily detected by traditional IDSs [10].

![Comparison of End to End Delay](image1)

**Figure 5.11:** Delay in AODV under active blackhole attack with and without the proposed IDS

![Comparison of Aggregate Throughput](image2)

**Figure 5.12:** Throughput in AODV under normal condition and active blackhole attack

Figure 5.6 illustrates this fact as the PDR of AODV under simple blackhole attack with the traditional IDS is significantly improved compared to the case without the traditional
IDS. As already mentioned, most of the traditional IDSs detect blackhole attacks by determining if values of some of the parameters in RREQ/RREP packets are substantially different from the base values e.g., the SN being too high, HC being too low etc.

![Comparison of Aggregate Throughput](image1)

**Figure 5.13:** Throughput in AODV under active blackhole attack with and without traditional IDS

![Comparison of Aggregate Throughput](image2)

**Figure 5.14:** Throughput in AODV under simple blackhole attack with and without traditional IDS

Active blackhole attack defies such IDSs by changing the parameters incrementally
just required to make the attack successfully and hence traditional IDSs cannot detect it.

Finally we execute the proposed IDS in presence of the active blackhole attack. Figure 5.7 illustrates the PDR of AODV under active blackhole attack, with and without the proposed IDS in execution. Results show that when the proposed IDS is executed the PDR of AODV under active blackhole attack is improved substantially. In other words, the proposed IDS nullifies the effect of the active version of the blackhole attack on AODV in terms of PDR, which could not be achieved by the traditional IDS.

![Comparison of Aggregate Throughput](image)

**Figure 5.15:** Throughput in AODV under active blackhole attack with and without the proposed IDS

Similar trends were found for the other two parameters i.e., end-to-end delay and throughput. Figure 5.8 through Figure 5.15 illustrate this fact. Finally we compared the detection rate and accuracy of the proposed IDS versus the traditional IDS on the active blackhole on AODV. We executed the traditional IDS [10] to detect the active blackhole attack under the network configuration given in Table 5.9. The detection rate was found to be lower than 10% and accuracy around 90%. The major reason for lower detection rate is, in [10] determining attack is based on looking for very high $SN$ in the RREP packets (that is sent by attacker). As already discussed, in active blackhole attack, $SN$ sent by
malicious node is based on looking at genuine RREPs and so the SN sent is just a bit higher. In the proposed IDS, detection rate and accuracy, both are 100%. The reason for 100% detection rate and accuracy is because attack detection is based on a response from attacker which under normal case should not arrive. Further, accuracy of the traditional IDS [10] is less than 100% because threshold regions need to be set for comparison with parameters measured from AODV related packets. Setting the regions too broad lead to fall in accuracy and too stringent regions result in missing attack detection. So even with the best efforts thresholds cannot be set that result in full accuracy and capturing all attack instances. As such thresholds are not required in the proposed IDS, 100% accuracy and detection rate could be achieved.

5.6 Conclusion

In this chapter we demonstrated that traditional routing protocols are secure to simple attacks because appropriate IDSs have been designed. AODV was considered as the routing protocol, blackhole was taken as the attack and a statistical system [10] was the IDS. Following that an intelligent version of the basic blackhole attack called active blackhole was proposed. It was then demonstrated analytically and using NS-2 simulations that the IDS [10] could not secure AODV from the active blackhole attack. To summarise, enhanced attacks cannot be handled by existing IDSs and for such cases new IDSs are required.

In the next chapter we make a similar study on the well accepted proactive protocol “OLSR”. We develop a modified version of the common wormhole attack termed as “camouflaging wormhole attack” and illustrate the weakness of the traditions IDSs for this attack on OLSR. Finally a suitable IDS is also designed for camouflaging wormhole attack.