PATH-HOP BASED SECURE AODV TO DETECT BLACK-HOLE AND GRAY-HOLE ATTACKS IN MANET

Dr. S.V. Vasantha*,[1] Dr. A. Damodaram,[2] Dr. S. Rama Krishna[3]

[1]Assoc. Professor of CSE Department, Keshav Memorial Institute of Technology, Telangana, India
[2] Professor of CSE Department, Director of School of Information Technology, JNTUH, Telangana, India
[3]Assoc. Professor of CSE Department, Bapatla Engineering College, Andhra Pradesh, India
Correspondence: *Dr. S.V. Vasantha, Email: s.v.vasantha@gmail.com
Present Address: Keshav Memorial Institute of Technology, Telangana, India

ABSTRACT
In this modern communication world, a network can be built on the fly using mobile nodes having networking capability. Such network without any pre-existing support for communication is referred as Mobile Ad Hoc Network (i.e. MANET). It can be deployed in essential areas such as military and civilian environments, disaster management circumstances etc., because of its ad hoc nature. But it lacks central supervision system monitoring legitimacy of the nodes joining and leaving the network. So, any adversary can join spontaneously launching an attack and then silently disappear from the network. Black-hole and Gray-hole are the most possible common attacks, in which a node instead of relaying packets consumes them creating a hole in the network. Network performance is severely degraded due to these attacks. In this paper we propose a Path-Hop based Secure AODV protocol, which utilizes multiple paths for relaying data. It detects and discards malicious path(s) involving Black-hole or Gray-hole node during data forwarding phase. It also ensures reliable data transmission between the communicating nodes. Simulation results show that improved performance over other contemporary methods in terms of PDR with minimum NRL while addressing all kinds of Black-hole and Gray-hole attacks.

Keywords: MANET, AODV, Black hole attack, Gray hole attack, Security

1 Introduction

A collection of mobile-nodes interconnected to one and another using wireless links constitute a Mobile Ad Hoc Network (i.e. MANET). A mobile-node to span its transmission coverage region uses multi-hop transmission. So, when it requires to communicate with a far-away receiver then, it merely depends on the nodes along the multi-hop path to relay packets [1]. Thus, intermediate-nodes substitute themselves as routers. Its main applications such as military, emergency rescue operations, etc., are feasible due to its infrastructure-less nature [2] which facilitates the nodes with spontaneous and quick communication [3].

Here nodes are in agreement for cooperation so that they relay other’s packets. A fundamental design assumption of MANET’s routing algorithms is that all of its nodes works honestly and cooperatively [4]. Hence, it works properly only when nodes cooperate [5]. Categorization of routing protocols depending on its strategy is a) proactive b) reactive and c) hybrid. In the first category, every node continuously maintains routing information reflecting the changes in network topology. Coming to the second category, the route is searched upon its necessity. AODV and DSR are most often used reactive protocols but these are not designed to deal with security threats [6] and the third category is a blend of first two categories so, it takes benefits of the both. Any node can come to join or can leave from the network whenever they wish as a result it exhibits dynamic topology [7]. Network is open to legitimate users along with adversaries [8]. Due to its peculiar behaviour and vulnerabilities, it faces routing security challenges [9], [10]. Few major attacks that we confront which disturb the usual behaviour and detrimentally affect the network performance are Black-hole, Gray-hole, Worm-hole, Selfish-node misbehaving, Sybil, impersonation, etc., [11]. Out of these Black-hole and Gray-hole come under a perilous variety of Denial-of-Service attacks which happen at the Network-layer of the MANET [12]. They occur from a forwarding intermediate-node that compromise to drop packets so, it is also named as packet-drop attack [7].

2 Basic AODV Protocol Working

Ad hoc On-Demand Distance Vector (i.e. AODV) is an algorithm that enables multi-hop routing. Here nodes when they need quickly obtain routes and maintain them on active communication only [13]. It uses 3 main control packets to find and maintain paths, they are a) RREQ (i.e. Route Request), RREP (i.e. Route Reply) and RERR (i.e. Route Error). Suppose in the Figure1 source-node N_1 needs a route to destination-node N_8 then it broadcasts RREQ into the network. Intermediate-nodes not having path to the destination simply broadcasts RREQ further. If any intermediate-node having a latest path such as N_7

or the destination node N_8 itself receives RREQ then it unicasts RREP on the reverse path which is learned from RREQ propagation. Here nodes N_4, N_6 and N_7 receive duplicate RREQs which are rejected by tracking its unique IDs. Forward path for relaying data is setup by RREP on its way. Source-node N_1 receives two RREPs i) N_1-N_2-N_6-N_8 and ii) N_1-N_3-N_5-N_7-N_8. Suppose first reply is most recent and shortest path then it is selected to transmit data. As nodes are subject to mobile they affect the topology resulting in broken links. RERR is used to pass this information to other nodes [14].

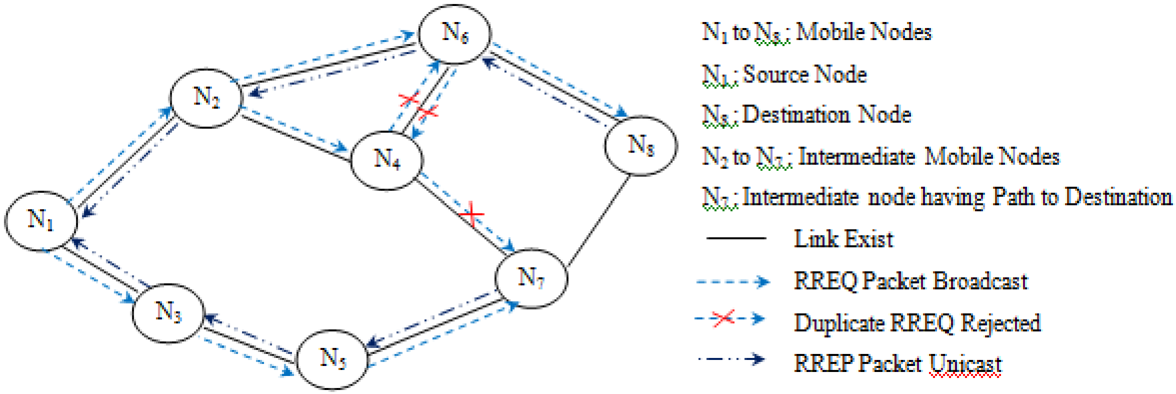


Figure 1. Basic AODV Protocol operation

3 Black-hole and Gray-hole Attacks

Categorization of packet dropping attacks on the basis of adversary's behaviour and number of adversaries involved is specified below. MANET's performance is affected by the following mentioned one or more types of attacks.

3.1 Single Black-hole attack

Here one node out of the set of mobile-nodes becomes an attacker. It falsely announces saying that it has the shortest and most up-to-date path to the requesting destination-node. If source-node gets attracted and uses it to send data then it completely consumes all the received packets. Such an attack is named as Single Black-hole attack [15].

3.2 Single Gray-hole Attack

Here the adversary is similar to the previous attack, which lures the source-node using its false routing announcement or behaves well while searching for routes thereafter transforms itself to malicious [16] and it is dissimilar in nature of dropping packets because it selects packets to be dropped. Thereby it moves back-and-forth from normal and malicious, so as to escape itself from identification. Because of this uncommon behaviour it comes in 3 forms they are listed below.

i) Type 1: It consumes packets of some selective source-node(s) or destination-node(s).
ii) Type 2: It consumes packets throughout certain selected time period.
iii) Type 3: It consumes packets randomly [17].

It also exhibits hybrid form of these 3 attacks creating a complicated issue to deal with [18]. If a single adversary executes this attack, it is named as Single Gray-hole attack.

3.3 Cooperative Black-hole/Gray-hole Attacks

In this attack, attacker works together with its neighbour(s). Usually source-node collects suspicious information for a node from its neighbour(s) to judge its behaviour. Here adversary's neighbour upon enquiry gives false reply saying that its adjacent-node is behaving perfectly normal and hides the happening malicious action from others [19].

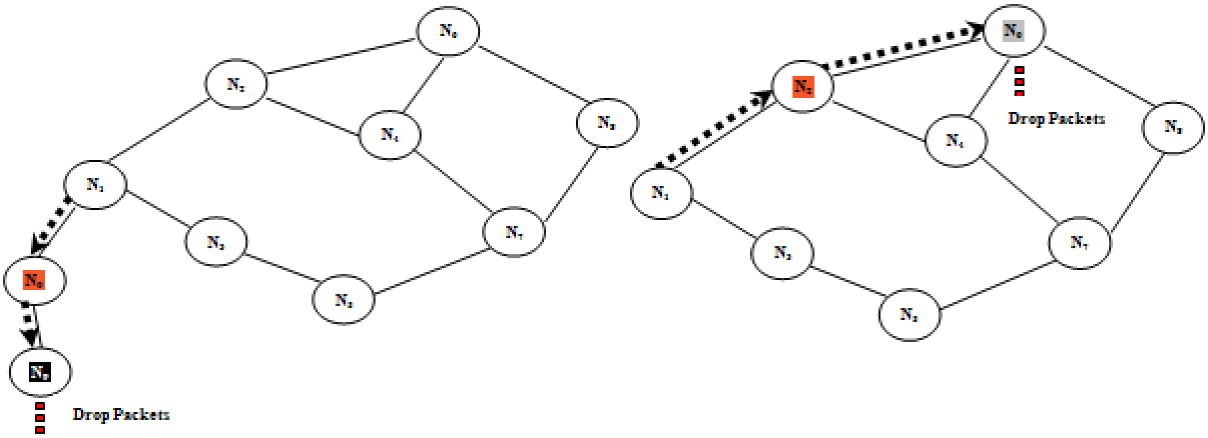
3.4 Multiple Black-hole/Gray-hole Attacks

When more than 1 Black-hole/Gray-hole nodes are there in a MANET then it is named as Multiple Black-hole/Gray-hole attack [19], [20]. Many such nodes are present so that one or the other can be on the selected active path.

4 Black-hole and Gray-hole attacks in AODV Protocol

Source-node to find a route for the destination-node send RREQs by broadcasting them. Attackers respond with falsely fabricated RREP containing highest value for DSN and lowest value for hop count indicating that it is the best out of the rest in terms of distance and freshness. After receiving this fabricated reply source-node selects it to start its communication task as it is the best route. Consequently the adversary succeeds in consuming arrived packets on it. If total packets are consumed by the attacker then it creates Black-hole attack [21] instead if it drops selectively then it creates Gray-hole attack. Attack becomes worst if an attacker and its next-hop neighbour collude. When source-node or any other node asks about the attacker's behaviour then colluding neighbour gives wrong information to support the attacker in consuming packets [14]. Figure 2(i) depicts Black-hole attack, here adversary N_9 is not actually having route to the RREQ specified destination but has succeeded in creating an attack as its falsely claimed route got selected. And if node N_0 colludes with it to hide ongoing malicious activity from other benevolent nodes then leads to Cooperative Black-hole attack. In the Figure 2(ii) Adversary N_6

has route so, initially it works well but later when data is to be relayed it becomes malicious resulting in one or more types of Gray-hole attacks specified in Section 3.2. And if node N_2 colludes with it to hide ongoing malicious activity from other nodes then leads to Cooperative Gray-hole attack.



i) Single or Cooperative Black-hole Attack ii) Single or Cooperative Gray-hole Attack

Figure 2. Black-hole and Gray-hole Attacks in AODV

5 Related Work

Nidhi & Lokesh [22] solution to AODV, prevents Black-hole attack. Their detection technique maintains a timer based trust value for each of its neighbours. It sets timer to detect drop in a packet accordingly its value is updated and this information is propagated to all other nodes. No further communication is granted with a neighbor whose trust value gets less than min_trust value. But it cannot handle Cooperative attacks and also it does not provide rapid detection for selective drop of packets. Authors of the paper [23] provide Secure AODV, which counts arrived RREPs. It simply ignores the 1st RREP since they have observed that malicious RREP reaches first. Sudheer & Nitika [24] detection solution uses IDS-agent to find out a suspicious node that sent highest DSN. Such a node is then blacklisted. Papers [23] and [24] cannot give a right solution when multiple Black-holes exist because even 2nd reply is also malicious.

In [25] paper, authors come up with a mechanism which uses 2nd shortest route instead of 1st for sending data. It sends 1st packet with Message Digest (MD) along with its delivery time. These fields are used to detect modified MD and also to find out packet drops. As it uses MD, here packet drops are only identified after the complete message is transferred that will be too late to find. In [26] paper, authors modified AODV to address Black-hole attacks of single and cooperative types. Intermediate-node sends gratuitous RREP to destination-node for proving its RREP validity. Source-node then cross verifies with destination-node. It also addresses collaborative attack by checking REPLYCONFIRM replies from intermediate-nodes with destination. But it fails when REPLYCONFIRM is forged by an attacker. Vimal & Rakesh [27] come up with an enhanced AODV solution to identify Black-holes. Each arrived RREP's DSN at source-node is compared with a fixed threshold value. When received DSN is greater than the threshold then it confirms that, RREP is from a Black-hole else from a legitimate one. It cannot handle Gray-holes. Threshold value defined will be incorrect when more number of malicious RREPs received. Proposed method of [28] observes promiscuously the node's packet dropping nature in the network and then it is compared with a calculated threshold value to fix on whether it is Black-hole or not. But it needs a time span over which node's behaviour is observed to decide on attacker and it cannot detect Cooperative attacks since neighbours may collude with attacker.

The solution given by authors in [29] includes hash function, encryption, decryption and nonce values exchange to authenticate a node in presence of black-holes. Proposed method in [14] is designed in view of Military purpose. It first broadcasts fake RREQ so as to catch all Black-holes. Next to catch cooperating-nodes, it asks neighbours of caught Black-holes for its forwarding behaviour. Methods in [29], [14] cannot work when Gray-holes exist. Author in [30] presented EDRI based detection and elimination of Cooperative Black-holes. Data and ordinal control packets are proposed and EDRI table to catch malicious-nodes but it fails if intermediate-nodes collude. Approach given in [31] is based on multipath transmission and permuted ACK. It uses several shortest paths for data forwarding. And their ACKs are permuted i.e. sent on other paths. Due to loss of single packet, two permuted ACKs are lost. Using received permuted ACKs source-node detects adversary affected routes. Consequently, this information is passed to destination. It then deletes its next-hop related to that route. But this solution requires, ACKs equal to that of data packets and extra fields to be included in each packet.

6 Problem Statement

Most of the detection processes designed are not covering all kinds of packet drop attacks under various networking scenarios. One of the sturdy and all-inclusive detection mechanisms to counter various kinds of Black and Gray-hole attacks is the Invincible AODV [33]. It initially uses Bulwark AODV mechanism [32] to prevent the malicious-nodes early during route discovering process and then starts data transmission on the selected optimal path. If any malicious-node has bypassed the first level of prevention process or toggles its behaviour from benevolent to malevolent while relaying data then it identifies the attacking-node promptly and isolates it from the network creating a deterrent environment while assuring reliable data transfer, as it monitors each and every packet reception. But it requires one ACK to be sent for each data packet reception, if this requirement is not satisfied then detection and recovery of lost packets will be delayed thereby average PDR and Throughput degrades. In presence of multiple alternate paths, a better solution can be designed by sending data packets on all available valid paths. When multiple paths are active a rapid detection and recovery mechanism can be designed in which a single ACK acknowledges multiple data packets.

Hence a detection technique is proposed as an optimization that can be employed when multiple alternate paths exists between the communicating nodes otherwise FCS-AODV the basic version of Invincible AODV is used when there exists a single valid path for data relay to ensure reliable data transmission against the packet-drop attacks.

7 Proposed Method

The proposed solution Path-Hop based Secure AODV (PHS-AODV) is a simple defensive mechanism used in data forwarding phase of basic AODV protocol against various kinds of Black and Gray-hole attacks, it is employed in case of multiple valid paths exist between the communicating nodes. It as well assures reliable transmission of data packets.

Proposed method utilizes available multiple disjoint paths as shown in the Figure 3, thereby malicious paths are easily and promptly identified with the help of ACK on other alternate valid path. Even recovery process is also uncomplicated as it merely discards detected malicious paths and resumes its data transmission with other valid paths without any delay. Moreover, it makes intermediate-nodes free from the detection process. Cooperative attacks are discouraged by this technique since malicious activity detection is not depending on the intermediate-node's response. To handle transmission of data on multiple paths, source-node and destination-node processes are improved which are explained in the following sections.

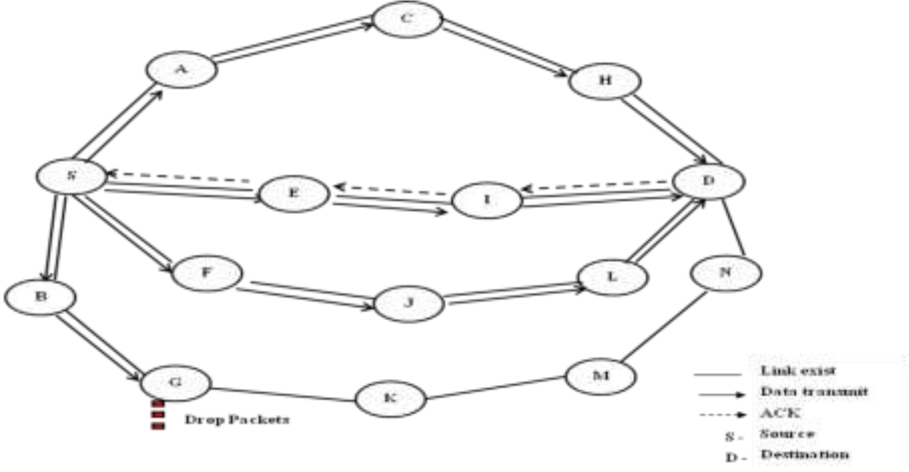


Figure 3. Data Forwarding based on Multiple Disjoint Paths

7.1 Improved Source-Node Process

It selects optimal paths from the collected multiple totally-disjoint paths (i.e. node disjoint) to forward data. In this method data packets are not sent simultaneously, instead they are sent one after the other on different paths as if they are sent on a single path thus there is no need of extra buffer space. Source-node's process is updated accordingly, which includes processing related to selection of optimal and totally-disjoint paths, Path-Hop based Data forwarding and Response based on Authenticated ACK.

7.1.1 Optimal Totally-Disjoint Paths Selection

Normally in the basic protocol when a node receive more than one RREQs from the same source then, it forwards only the first received RREQ and the subsequent RREQs are discarded since they have the same RREQ ID as shown in Figure 4. Thus, at the destination-node each RREQ message arrives from a distinct node. And the proposed system is designed in such a way that destination-node will respond to each RREQ/ENQ packet arrived from the distinct node but not from the same node and send replies for the first 10 RREQ/ENQ packets so that paths are not too long at the same time to have a good number of alternate paths as shown in Figure 5 and Figure 6. Here Enquiry packet (ENQ) is sent by an intermediate-node to find current destination sequence number (CDSN) of the destination-node. In response destination sends Enquiry Reply packet (EREP) containing CDSN. Intermediate-node updates CDSN field and prepares RREP packet [32].

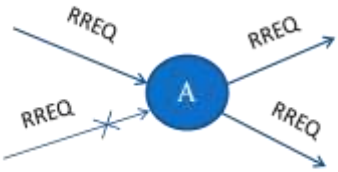


Figure 4. Intermediate-Node Rejecting Duplicate RREQs

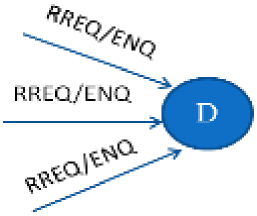


Figure 5. Destination-node Accepting RREQ/ENQ on Distinct Next-Hops

Coming to RREP processing in normal scenario, when a node receive more than one RREPs, it will forward the first received RREP after that it forwards any RREP which is having highest DSN or with same DSN and having a least hop count. So, source-node may receive many RREPs from the same next-hop node, but it discards the previously received RREPs from the same next-hop as they might be old or longest path when compared to the recently received one. Thus, it updates with the last received RREP. In our system, when a node receives RREP with a DSN value less than its value then that RREP is forwarded to handle malicious replies, which is then updated at the source-node. Thus, source-node holds only one RREP from the same next-hop that is, the one which is received recently. These improved RREQ and RREP processes are explained in the Bulwark AODV mechanism [32].

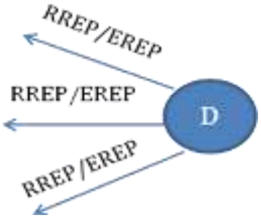


Figure 6. Destination-node Responding with RREP/EREP on Distinct Next-Hops

An intermediate-node may get many RREQs or RREPs but it updates with only one of them and at the source-node only one RREP from its same next-hop is updated similarly at the destination-node, it responds to only one RREQ/ENQ from the same neighbour. Therefore source-node will have multiple safe totally-disjoint paths forming a valid set of RREPs. The proposed Path-Hop scheme is used to find out the optimal paths from this set and it proceeds with data relaying process if multiple optimal paths exist otherwise if there is exists only one optimal path or one RREP then it calls FCS-AODV [33].

To obtain optimal paths in the Path-Hop based scheme, initially valid RREPs are sorted in ascending order based on hop_count and then the first ‘x’ paths are selected whose hop_count < (2 × min_hop) since longer paths incurs more delay degrading the performance and even data packets may not arrive in sequence.

7.1.2 Data Forwarding using Path-Hopping Mechanism

Source-node send data packets on the selected ‘x’ paths using round-robin scheduling i.e., first packet is sent on the 1st path then next packet is sent on the 2nd path so on till xth packet is sent on the xth path and after that next packet is sent again on the 1st path in a round-robin fashion as if they were sent serially on a single path facilitating multiple active paths. Thus, the receiver has no requirement for extra buffer space. Initially, it sets sender_window to ‘x’ after that it sets to 2x which is not further increased. Source-node records the sent data packet’s sequence numbers with respect to each next-hop which is depicted in the Figure 7 so that source-node has information about which packets are sent on which next-hop which is further used for tracking the malicious paths.

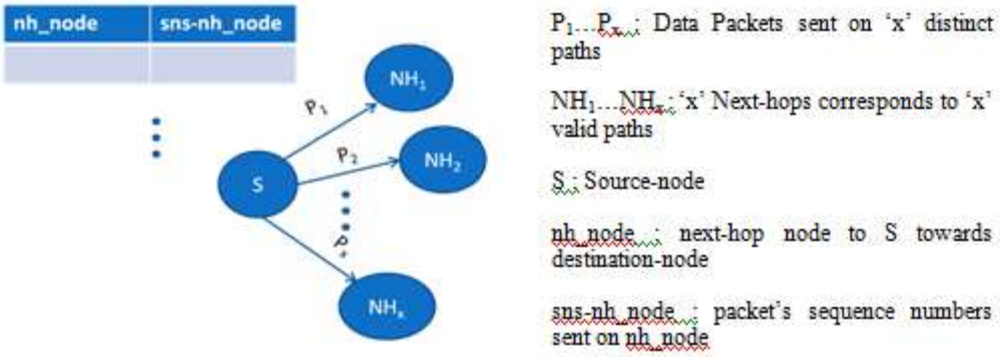

nh_node
sns-nh_node
NH1
NH2
NHx
S
P1
P2
Px
P1…Px : Data Packets sent on 'x' distinct paths
NH1…NHx : 'x' Next-hops corresponds to 'x' valid paths
S : Source-node
nh_node : next-hop node to S towards destination-node
sns-nh_node : packet's sequence numbers sent on nh_node

Figure 7. Packet Transmission using Path-Hopping Scheme

It delays next round by t_d which is expressed in the Equation (3). Here t_1 represents the 1st valid path's transit time for a packet from source-node to destination-node that is given by Equation (1). Similarly t_x represents the transit time of a packet on last valid path that is given by Equation (2). So, the difference t_d gives the time delay required to be introduced between the rounds i.e. inter round delay to match transmission delays of first shortest and last longest paths so that packets arrive in order which is necessary in this scheme otherwise destination-node may confuse that missing packet is lost.

$$t_1 = (RTT_{VP1})/2 \qquad (1)$$

$$t_x = (RTT_{VPx})/2 \qquad (2)$$

$$t_d = t_x - t_1 \qquad (3)$$

where VP_1: 1st valid path, VP_x: last valid path

7.1.3. Response based on Authenticated ACK

As source-node initially sends 'x' packets so, it starts ACK_Timer after transmitting the x^{th} packet whereas for subsequent 2x packets transmission, it starts ACK_Timer after sending the $2x^{th}$ packet whose value is updated using first ACK received on the last longest path. Source-node expects to receive ACK before timer expires, if it has not received then it checks its route cache for more than one valid RREPs (i.e. totally-disjoint paths) if available then it proceeds with the Path-Hop scheme else if exactly one RREP exists then it proceeds with Invincible AODV otherwise when no RREP exists then it goes for discovering new routes.

When source-node receives ACK it checks for validity using is_forge_ACK procedure explained in Algorithm 2 which is based on ACK with Nonce processing technique [33], if it is a forged one then it is rejected simply and source-node waits till another valid ACK arrives or ACK_Timer goes off otherwise it verifies the sequence number(s) received in the ACK, if it is the next expected packet's sequence number then clears all the previously recorded sequence numbers and proceed with the transmission of next 2x packets else if any packet(s) missing is found and no congestion or no link failure notification arrived then it simply discards malicious paths corresponding to the missing packets and reschedules the lost packets with the highest priority and continues transmission on the remaining valid paths without any delay where as in the normal scenario which involves transmission on a single path, once a malicious-node is detected it is discarded and then new route is found that may be malicious. Thereby, it provides a rapid detection, simple response and speedy recovery mechanism than a single path based data forwarding mechanism with the use of several active paths. Before proceeding for further transmission of data packets, it checks for the number of available optimal paths, if it is more than 1 then continues with Path-Hop scheme otherwise shifts to FCS-AODV. Improved source-node process is expressed in Algorithm 1.

Algorithm 1: Improved Source-Node Process

Notations

x: Number of optimal totally-disjoint paths
z: Total number of valid totally-disjoint paths
IN_1: Source-node
m_sqno: missing packet's sequence number
n_sqno: next to be sent packet's sequence number
IN_n: Destination-node
t_d : inter-round delay
P_k: k^{th} Packet where $1 \leq k \leq m$
min_hop: minimum hop-count
nh_node: next-hop node to IN_1 to reach IN_n
sns-nh_node: packet's sequence-numbers sent on nh_node

```
$IN_1$ holds valid totally-disjoint RREPs
```

```
    // obtain multiple optimal totally-disjoint paths
2:  if z > 1 then
3:      $IN_1$ sort z RREPs based on hop_count
4:      $IN_1$ selects first x paths whose hop_count < 2 × min_hop
5:      if x >1 then
            // Initially set Sender window to x later increased to 2x
6:          $IN_1$ sends $P_k$ on x paths using roundRobin (x)
7:          $IN_1$ stores sns-nh_node across nh_node
8:          $IN_1$ delays next round by $t_d$
9:          if $ACK_k$ received then
10:             flag ← is_forge_ACK($ACK_k$ )
11:             if flag = 1 then
12:                 discard forged ACK
13:                 goto 9
14:             else if flag = 0 then
15:                 if $ACK_k$ with n_sqno arrived then
16:                     delete all sns-nh_node upto n_sqno-1
17:                     goto 6
18:                 else if $ACK_k$ with m_sqno(s) arrived then
19:                     if $IN_1$ with no ECN and with no RERR then
                            // discard malicious paths
20:                         delete nh_node(s) entries with m_sqno(s)
21:                         delete all sns-nh_node upto m_sqno-1
22:                         reschedule lost $P_k$ with highest priority
23:                         goto 5
24:                     end if
25:                 end if
26:             end if
27:         else if no ACK found then
28:             if z ≥ 1then
29:                 goto 4
30:                 else
31:                 start Route Discovery process
32:             end if
33:         end if
34:     else
35:         FCS-AODV ( )
36:     end if
37: else
38:     FCS-AODV ( )
39: end if
```

~~Algorithm 2: is_forge_ACK Module~~

```
Notations
PN : Prime-Number
$N_k$ : $ACK_k$'s nonce
NPN : Next Prime-Number
Input : $ACK_k$ with $N_k$

    // First packet is given with random sequence number
    // X is initialized to first packet's sequence number
1:  X ← $P_1$_Rand_Seqno
    // c is initialized to 0
2:  c ← 0
3:  if c = 0 then
        // First ACK with nonce received
        PN ← $N_k - X$
        if isPrime (PN) then
            c ← c + 1
            return 0
        else
            return 1
        end if
    else
        NPN ← nextPrime (PN)
        if $N_k - X$ = NPN then
            PN ← NPN
            return 0
        else
            return 1
        end if
    end if
```

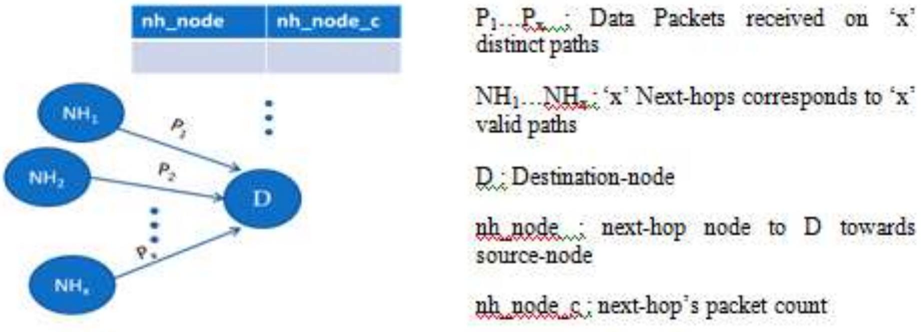
7.2 Improved Destination-Node Process

Usually in basic AODV protocol, destination is designed to receive data from the source node on a single path where as in the proposed system it receives data packets on multiple disjoint paths. Thus, destination collects packets from several next-hop nodes one after the other. Destination-node's process is improved to include Multi-Path based Data Receiving process and Generation and Transmission of Authenticated ACK. The following sections discuss about these mechanisms.

7.2.1 Multi-Path based Data Receiving

Destination-node when it receives first two packets from the same next-hop with the same source-node address then it proceeds with FCS-AODV otherwise if first 'x' packets arrived from 'x' distinct next-hops from the same source-node then, it maintains next-hop and its associated packet count information as depicted in the Figure 8.

Initially it sets next-hop's packet count to zero across each next-hop for the 1st 'x' packets received, next it sends only one ACK for all the first 'x' packets received. When subsequent packets are received then, next-hop's packet count is incremented for each packet received on that next-hop and when 2x packets are received by the receiver i.e., when packet count reaches to 2 on the last next-hop then it sends only one ACK for all the 2x packets received.



$P_1 \ldots P_x$: Data Packets received on 'x' distinct paths

$NH_1 \ldots NH_x$: 'x' Next-hops corresponds to 'x' valid paths

D : Destination-node

nh_node : next-hop node to D towards source-node

nh_node_c : next-hop's packet count

Figure 8. Packet Reception using Path-Hopping Scheme

7.2.2 Generation and Transmission of Authenticated ACK

ACK is generated when inter-packet arrival time exceeds the predefined threshold value which is expressed in Equation (7), here the first path's number of hops is represented as 'n' (i.e. min_hop) and last path's maximum number of hops allowed are given by 2n-1. Suppose in the worst case, i^{th} packet P_i is sent on the 1^{st} path and next packet P_{i+1} is sent on the last path and P_{i+1} is delayed by one hop to the previous packet P_i (i.e. one node traversal time) at the source-node then these two packets differ by n hops which is given from the Equations (4) and (5). Thus, the maximum number of hops a packet is delayed by is 'n'. Therefore to arrive at ACK generation time ACK_i_Genr_Time, the average packet traversal time for one hop (i.e. NODE_TRAVERSAL_TIME) is multiplied by hop_diff which is expressed by the Equation (6).

$$\text{hop\_diff} = [(2 \times n)-1] - (n-1) \qquad (4)$$

$$\Rightarrow \text{hop\_diff} = n \qquad (5)$$

$$ACK_i\text{\_Genr\_Time} = \text{hop\_diff} \times \text{NODE\_TRAVERSAL\_TIME} \qquad (6)$$

$$\Rightarrow ACK_i\text{\_Genr\_Time} = n \times \text{NODE\_TRAVERSAL\_TIME} \qquad (7)$$

The first ACK is generated when all the first 'x' packets are received or inter-packet arrival time exceeds ACK_i_Genr_Time. Destination-node includes either sequence number of next packet to be sent or sequence number(s) of the missing packets in ACK_1 which is sent on the last longest path so that source-node learns and updates its ACK_Timer to the maximum value taken for the worst case. It is also accompanied with a nonce value, which is produced using nonce procedure explained in the Algorithm 4 to prove its validity.

Subsequent ACK is generated when last next-hop's packet count reaches 2 or a packet is missing or inter-packet arrival time exceeds ACK_i_Genr_Time. Before sending an ACK, it deletes next-hop entries corresponding to the missing packets if any, next reset all next-hop's packet count to zero and then sends ACK with either sequence number of next packet to be sent or sequence number(s) of the missing packets along with the next nonce produced using nonce module, it is sent on the first shortest path out of 'x' valid paths so that source-node gets informed quickly about the last 2x packets transmission. The improved destination-node process is explained in the Algorithm 3.

Algorithm 3: Improved Destination-Node Process

Notations

IN_1: Source-node

IN_n: Destination-node

N_k: ACK_k' s nonce where $1 \leq k \leq m$

P_k: k^{th} Packet where $1 \leq k \leq m$

x: number of optimal disjoint paths

nh_node: next-hop node to IN_n to reach IN_1

P_j: First x data packets where $1 \leq j \leq x$

nh_node_c: nh_node's packets count

n_sqno: next to be sent packet's sequence-number

m_sqno: missing packet's sequence-number

```
if $IN_n$ receive $P_k$ from same $IN_1$ on $x > 1$ paths then
    if $P_j$ received on x paths then
        for each path of x do
            store nh_node, nh_node_c ← 0
        end for
        $N_1$ ← nonce ( )
        if missing packets in $P_j$ found then
            send $ACK_1$ with $N_1$, m_sqno(s) on last nh_node in the list
        else
            send $ACK_1$ with $N_1$, n_sqno on last nh_node in the list
```

```
        end if
    else
        // subsequent packets received
        nh_node_c ← nh_node_c +1
        $N_k$ ← nonce ( )
        if $P_k$ is missing then
            delete nh_node entries for missing $P_k$
            reset all nh_node_c entries to 0
            send $ACK_k$ with $N_k$, m_sqno(s) on $1^{st}$ nh_node in the list
        else
           $IN_n$ waits till two $P_k$s received on last path or till timeout
           reset all nh_node_c entries to 0
           send $ACK_k$ with $N_k$, n_sqno on $1^{st}$ nh_node in the list
        end if
    end if
else if $x = 1$ then
    FCS-AODV ( )
end if
```

Algorithm 4: Nonce Module

Notations

PN: Prime-Number

X: P_1's sequence-number
NPN: Next Prime-Number
IRP: Initial Random Prime-Number
N_k: ACK_k's nonce

```
$X \leftarrow P_1$_Rand_Seqno
// c initialized to 0
$c \leftarrow 0$
if $c = 0$ then
    $IRP \leftarrow$ rand_Prime ( )
    $N_1 \leftarrow X + IRP$
    $PN \leftarrow IRP$
    $c \leftarrow c + 1$
    return $N_1$
else
    $NPN \leftarrow$ next_Prime (PN)
    $N_k \leftarrow X + NPN$
    $PN \leftarrow NPN$
    return $N_k$
end if
```

8 Simulation Environment and Results

This section gives explanation about how the simulation environment is configured and how the C++ libraries of NS-2.35 are used to implement the proposed PHS-AODV. It shows performance improvement of PHS-AODV over basic AODV protocol against Black and Gray-hole attacks. It also gives comparative analysis of the proposed system with other contemporary solutions.

8.1 Simulation Setup

All the simulation experiments related to our proposed security method is conducted on the system with an Intel Core i7 processor which is running at 2.67 GHz, 4 GB RAM and it runs Ubuntu 14.04. Network Simulator NS-2.35, a well known open source simulator for MANETs is used to conduct simulations of our proposed mechanism. The basic AODV's C++ libraries of the NS-2.35 along with our improved AODV C++ libraries are used. The components used for simulating and evaluating the performance of our proposed method are Tool Command Language(TCL), Network Animator(NAM) and Tracegraph along with AWK programming. To setup the experimental simulation environment, parameters mentioned in the Table 1 are configured. Each of the simulation experiments is conducted over 5 times under various network topologies and then its average value is considered to demonstrate the result.

Table 1. Simulation Parameters

Parameter	**Value**
Channel-type :	Channel/WirelessChannel
Radio-propagation model :	Propagation/TwoRayGround
Network-interface type :	Phy/WirelessPhy
MAC-type :	Mac/802_11
Interface-queue type :	Queue/DropTail/PriQueue
Link-layer type :	LL
Antenna-model :	Antenna/OmniAntenna
Routing-protocol :	AODV
Number of mobile-nodes :	10 to 100

X-dimension of topography :	850
Y-dimension of topography :	850
Time of simulation :	100 sec to 500 sec
Traffic-Model :	CBR
Packet-size :	512 bytes

8.2 Implementation Details

The necessary C++ files of basic AODV of NS-2.35 are modified to create B-AODV and G-AODV folders in the place of existing AODV folder to perform Black-hole and Gray-hole attacks respectively. After that we have modified the C++ files of basic AODV according to the proposed method to create PHS-AODV folder.

8.3 Performance Metrics

The below described metrics are employed for evaluating the performance of our proposed technique.

Packet Delivery Ratio (PDR) : It is given by the formula expressed in the Equation (8). It is used to measure how reliable the transmission is.

$$PDR = \left(\frac{Total\ no.of\ data\ packets\ received}{Total\ no.of\ data\ packets\ sent}\right) \times 100 \tag{8}$$

Average Throughput : It is given by the formula expressed in the Equation (9). Its units are kilobits per second (kbps).

$$Average\ Throughput\ = \frac{Total\ no.\ of\ data\ packets\ received}{Total\ simulation\ time} \tag{9}$$

Average End-to-end Delay : End-to-end delay (i.e. packet_delay) gives the total delay occurred in transit of a packet from its source-node to destination-node which is given by the Equation (10). Average End-to-end Delay is the average of all n packet_delays where n is the total number of data packets sent, which is given by the Equation (12).

$$packet\_delay\ =\ packet\_received\_time - packet\_sent\_time \tag{10}$$

$$Total\_delay\ =\ \sum_{i=1}^{n} packet\_delay_i \tag{11}$$

$$Average\ e2e\ delay\ =\ Total\_delay\ /\ n \tag{12}$$

Normalized Routing Load (NRL) : It is given by the formula in Equation (13). It specifies about the overhead of control packets used for routing the data packets over entire transmission period.

$$NRL = \frac{Total\ no.of\ routing\ packets\ used}{Total\ no.of\ data\ packets\ received} \tag{13}$$

8.3 Results Discussion

The proposed PHS-AODV is compared with the regular AODV (i.e. AODV in absence of malicious-nodes), AODV in presence of Black-holes and AODV in presence of Gray-holes with respect to the Average Throughput, PDR and NRL which are depicted in the Figure 9, Figure 10 and Figure 11 respectively. Table 2 shows that the performance of PHS-AODV is better than the basic AODV protocol in presence of malicious-nodes in terms of PDR, NRL and Average Throughput.

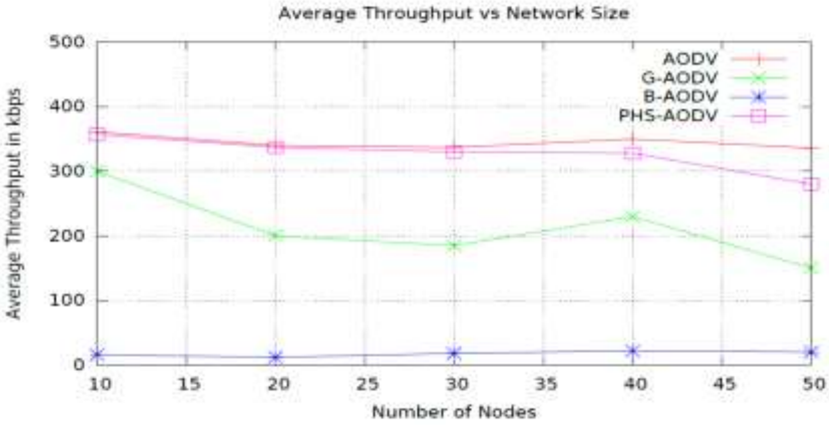


Figure 9. Average Throughput Vs Number of Mobile-Nodes of PHS-AODV

Next it is compared with the Invincible AODV with respect to the PDR, NRL and Average Throughput which are depicted in the Figure 12, Figure 13 and Figure 14 respectively. Average PDR of the Invincible AODV is increased to

94.14 % in case of FCS-AODV and 92.58 % in case of EFCS-AODV and when PHS-AODV is applied as an alternate to Invincible AODV in presence of multiple optimal active paths, it has boosted remarkably to 96.58 % because it provides rapid detection and recovery. In PHS-AODV, when the receiver notices a missing packet, it informs the source-node immediately through ACK on the other valid path whereas in case of Invincible AODV, source-node waits till the timer goes off. On the other hand, it also looses the pipelined packets sent before the expiry of ACK timer dissimilar to PHS-AODV, in which subsequent packets are sent on the other paths using Round-robin scheduling. When malicious activity is found, the PHS-AODV simply continues with the other active paths which are readily available. Therefore it provides rapid detection and recovery strategy.

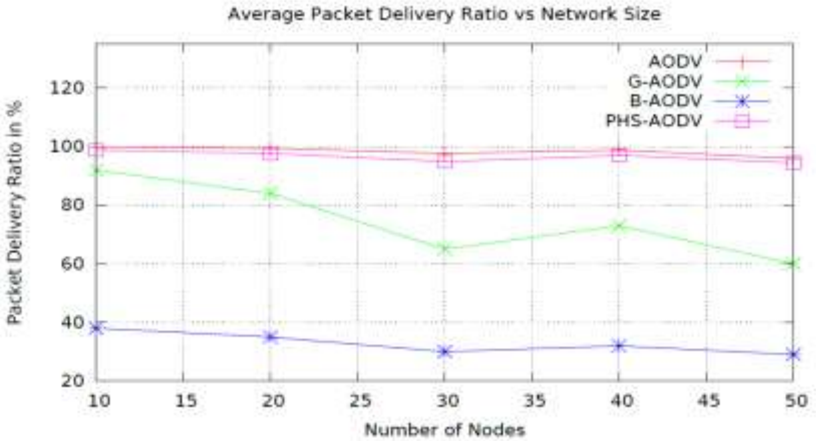


Figure 10. PDR Vs Number of Mobile-Nodes of PHS-AODV

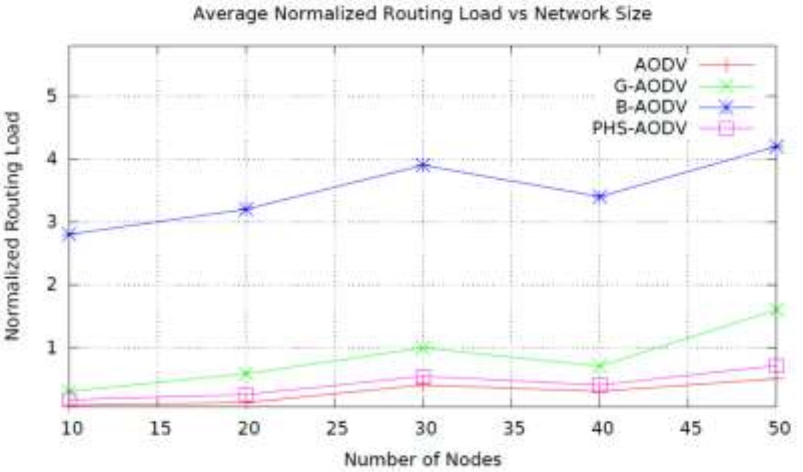


Figure 11. NRL Vs Number of Mobile-Nodes of PHS-AODV

Average Throughput of the Invincible AODV is increased to 335.8 kbps in case of FCS-AODV and 332.6 kbps in case of EFCS-AODV and when PHS-AODV is applied, it has slightly decreased to 326.4 kbps on average when compared to the Invincible AODV because it uses multiple paths for forwarding packets, which may be of different lengths. In the best case, PHS-AODV performs better than the Invincible AODV in which all the optimal disjoint paths are of the same length (i.e. having same number of hops).

Table 2. Performance Improvement of PHS-AODV over Basic-AODV

	B-AODV(with Black-hole-attack)	G-AODV(with Gray-hole-attack)	PHS-AODV	**Improvement in PHS-AODV over B-AODV**	**Improvement in PHS-AODV over G-AODV**
PDR	32.8 %	74.8 %	96.58 %	**63.78 %**	**21.78 %**

Average-Throughput	17.6 kbps	213 kbps	326.4 kbps	**308.8 kbps**	**113.4 kbps**
NRL	3.5	0.83	0.413	**3.087**	**0.417**

The average NRL of the Invincible AODV is slightly increased to 0.54 in case of FCS-AODV and 0.68 in case of EFCS-AODV and when PHS-AODV is applied it has significantly reduced to 0.413 because it has no requirement to find a new path when malicious path is detected.

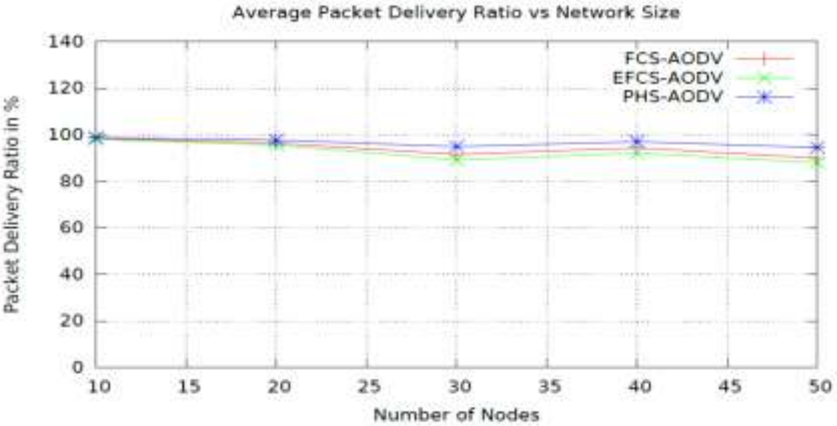


Figure 12. PDR of PHS-AODV Vs Invincible AODV

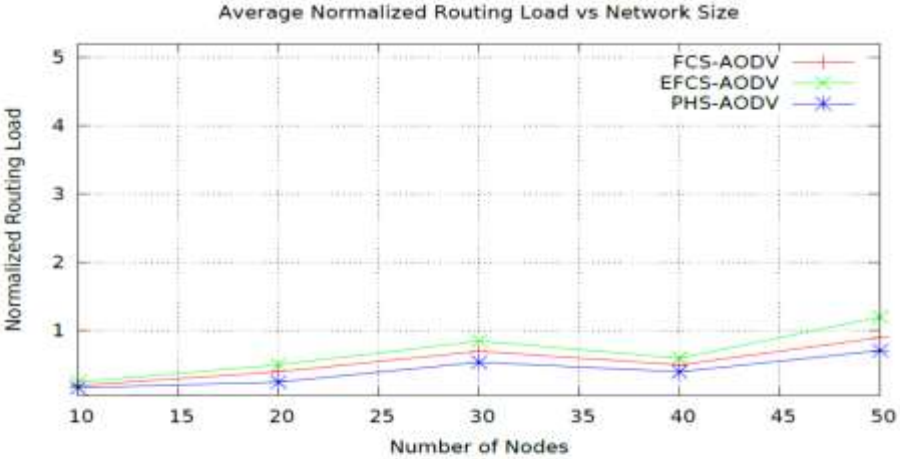


Figure 13. NRL of PHS-AODV Vs Invincible AODV

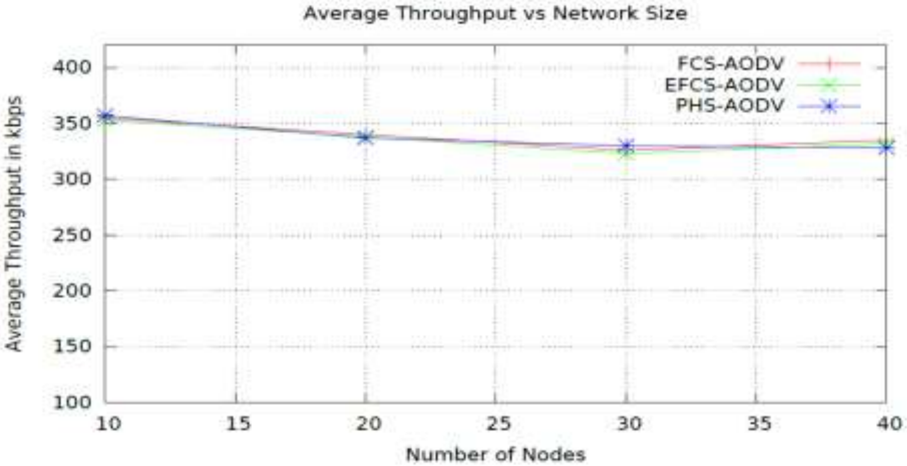


Figure 14. Average Throughput of PHS-AODV Vs Invincible AODV

8.3.1 Comparative Analysis

Average PDR in presence of Single and Cooperative Black-hole attacks for the PHS-AODV is about 93 % whereas, Yang et al. [13] method N-SAODV with Single Attack is 63.25 % and N-SAODV with Cooperative Attack is 61.75 % and coming to the Gray-holes case, N-SAODV with Single Attack is 85.5 % and N-SAODV with Cooperative Attack is 78.25 % whereas PHS-AODV is about 96.35 % which are shown in Figure 15 and Figure 16 respectively, it shows that there is a significant improvement in terms of PDR over N-SAODV.

In presence of Gray-holes ranging from 1 to 5, the average PDR of the PHS-AODV is around 97.58 % whereas Cluster-AODV by Katawkar & Nitin [11] is 94.26 % which is lesser than our proposed method, the same is expressed in the Figure 17. In presence of malicious-nodes varied from 1 to 5, the average PDR of GAODV by Sanjay et al. [7] is 90.2 % whereas PHS-AODV is about 95.64 %, which shows a significant improvement and the same is depicted in the Figure 18.

Therefore the results show that, there is a remarkable improvement in terms of PDR when the proposed PHS-AODV is applied when compared other solutions.

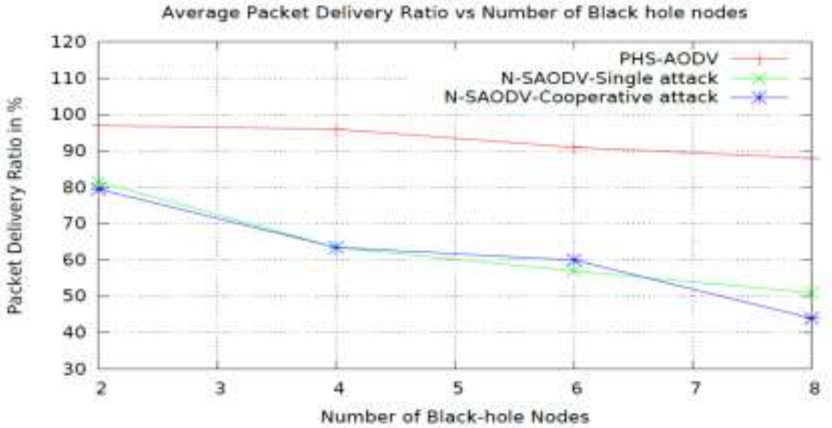


Figure 15. PDR of PHS-AODV Vs N-SAODV in presence of Blackholes

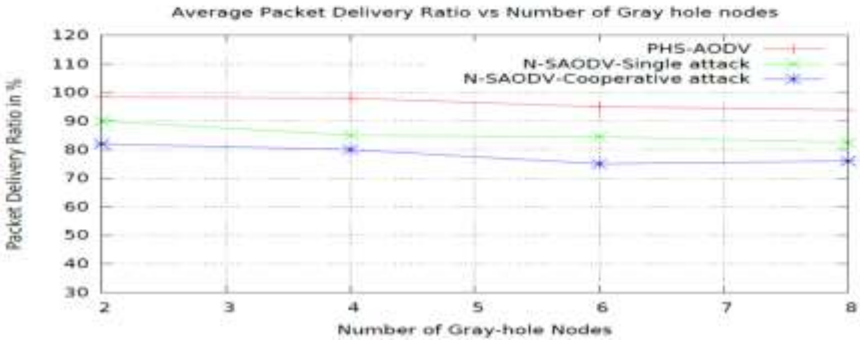


Figure 16. PDR of PHS-AODV Vs N-SAODV in presence of Grayholes

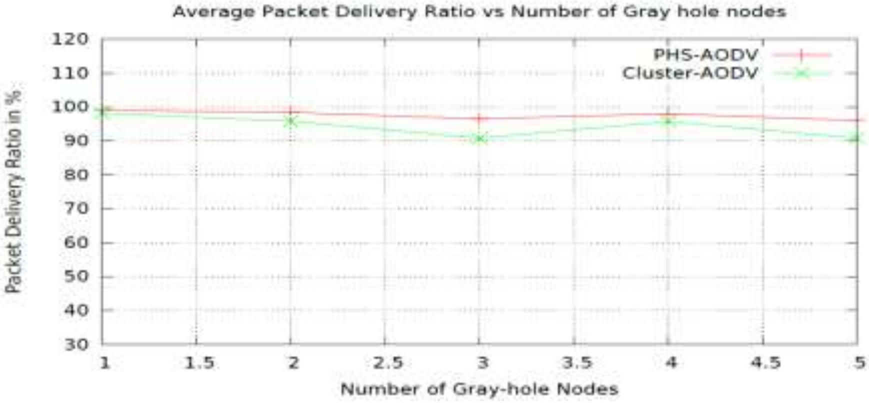


Figure 17. PDR of PHS-AODV and Cluster-AODV

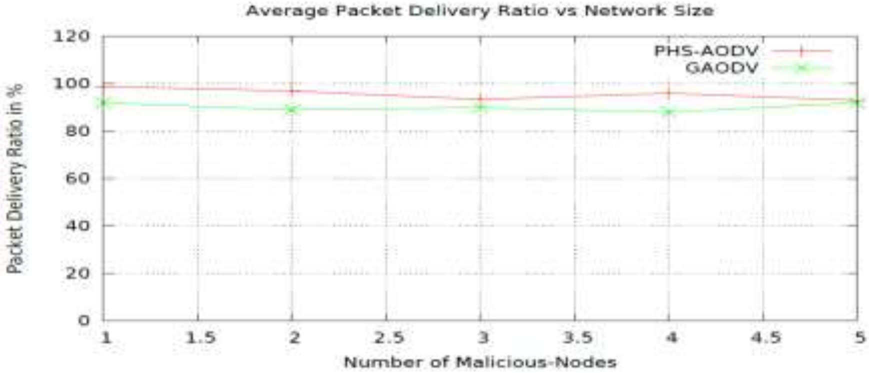


Figure 18. PDR of PHS-AODV Vs GAODV

9 Conclusion

The proposed PHS-AODV is an alternate approach to Invincible AODV in presence of multiple optimal disjoint paths. It identifies and discards Black/Gray-hole involving routes. It provides a simple defensive mechanism as it does not need intermediate-nodes involvement in the detection process thereby discouraging cooperative attacks. It also provides rapid detection and recovery due to the existence of multiple active valid paths. Simulation results show that the proposed PHS-AODV with minimum routing overhead provides outstanding PDR when compared to the Invincible AODV and other contemporary solutions.

10 References

1. Mahmoud, Xiaodong Lin, Xuemin shen. “Secure and Reliable Routing Protocols for Heterogeneous Multihop Wireless Networks”. Parallel and Distributed Systems, IEEE Transactions, Vol. 26, No. 4, 2013, Pg. 1140 – 1153.

2. Jian-Ming Chang, Po-Chun Tsou, Isaac Woungang, Han-Chieh Chao, Chin-Feng Lai. “Defending Against Collaborative Attacks by Malicious Nodes in MANETs: A Cooperative Bait Detection Approach”. IEEE System Journal, Vol. 9, No. 1, 2015, Pg. 65-75.
3. N. Chaubey, A. Aggarwal, S. Gandhi, K. A. Jani. "Performance Analysis of TSDRP and AODV Routing Protocol under Black Hole Attacks in MANETs by Varying Network Size". IEEE Fifth International Conference on Advanced Computing & Communication Technologies, 2015, Pg. 320-324.
4. Soufiene Djahel, Farid Nait-abdesselam, Zonghua Zhang. “Mitigating Packet Dropping Problem in Mobile Ad Hoc Networks: Proposals and Challenges”. IEEE Communications Surveys & Tutorials, Vol. 13, No. 4, Fourth Quarter 2011, Pg. 658 - 672.
5. H. Xia, Z. Jia, L. Ju, Y. Zhu. “Trust management model for mobile ad hoc network based on analytic hierarchy process and fuzzy theory”. Wireless Sensor Systems, IET, Vol. 1, No. 4, 2011, Pg. 248–266.
6. Deshmukh SR, Chatur PN. “Secure routing to avoid black hole affected routes in MANET”. IEEE Symposium on Colossal Data Analysis and Networking, 2016, Pg. 1-4.
7. Rathiga P, Sathappan S. “Hybrid detection of black hole and gray hole attacks in MANET”. IEEE International Conference on Computational Systems and Information Systems for Sustainable Solutions, 2016, Pg. 135-140.
8. U-H. Syed, AI Umar, F Khurshid. "Avoidance of Black hole affected routes in AODV-based MANET". IEEE International Conference on Open Source Systems and Technologies, 2014, Pg. 182-185.
9. Ruo Jun Cai, Peter Han Joo Chong. “A Neighborhood Connectivity-based Trust Scheme to Identify Active Black Hole Attacks”. IEEE International Conference on Communication Systems, 2014, Pg. 543-548.
10. Jin-Hee Cho, Ananthram Swami, Ing-Ray Chen. “A Survey on Trust Management for Mobile Ad Hoc Networks”. IEEE Communications Surveys & Tutorials, Vol. 13, No. 4, Fourth Quarter 2011, Pg. 562-583.
11. Umang S, Reddy BVR, Hoda MN. “Enhanced Intrusion Detection System for Malicious Node Detection in Ad Hoc Routing Protocols using Minimal Energy Consumption”. IET Communications, Vol. 4. No. 17, 2010, Pg. 2084–2094.
12. G.S. Mamatha, S.C. Sharma. “Network Layer Attacks and Defense Mechanisms in MANETS- A Survey”. International Journal of Computer Applications (0975 – 8887), Vol. 9, No. 9, 2010, Pg. 12-17.
13. C. Perkins, E. Belding-Royer, S. Das. “Ad-hoc on-demand distance vector (AODV) routing.” 2003, Request for Comments: http://www.ietf.org/rfc/rfc3561.txt.
14. Bikramjeet Singh, Dasari Srikanth, C.R. Suthikshn Kumar. “Mitigating effects of Black hole Attack in Mobile Ad-hoc Networks: Military Perspective”. IEEE International Conference on Engineering and Technology, 2016, Pg. 810 – 814.
15. Mohammad Taqi Soleimani, Mahboubeh Kahvand. “Defending Packet Dropping Attacks Based on Dynamic Trust Model in Wireless Ad Hoc Networks”. 17th IEEE Mediterranean Electrotechnical Conference, 2014, Pg. 362-366.
16. Bindra, G. S., Kapoor, A., Narang, A., Agrawal, A. “Detection and removal of co-operative blackhole and grayhole attacks in MANETs”. IEEE International Conference on System Engineering and Technology, 2012, Pg. 1–5.
17. Sharma B. “A distributed cooperative approach to detect gray hole attack in MANET”. ACM, In Proceeding of WCI. 2015, http://dx.doi.org/10.1145/2791405.2791433.
18. Usha G, Bose S. “Impact of gray hole attack on ad hoc networks”. IEEE International Conference on Information Communication and Embedded Systems, 2013, Pg. 404-409.
19. Arathy K S, Sminesh C N. “A Novel Approach for Detection of Single and Collaborative Black Hole Attacks in MANET”. Elsevier, Global Colloquium in Recent Advancement and Effectual Researches in Engineering, Science and Technology, Vol. 25, 2016, Pg. 264–271.
20. Sathish M, Arumugam K, Neelavathy Pari S, Harikrishnan VS. “Detection of single and collaborative black hole attack in MANET”. IEEE conference on WiSPNET, 2016, Pg. 2040-2044.
21. Anuj Rai, Rajeev Patel, R.K. Kapoor, D. S. Karaulia. “Enhancement in Security of AODV Protocol against Black-hole Attack in MANET”. ACM, International Conference on Information and Communication Technology for Competitive Strategies, 2014. 10.1145/2677855.2677946.

22. Nidhi Choudhary, Lokesh Tharani. “Preventing Black Hole Attack in AODV using Timer-Based Detection Mechanism”. IEEE International Conference on Signal Processing and Communication Engineering Systems, 2015, Pg. 1-4.
23. Ashish Kumar Jain , Vrinda Tokekar. “Mitigating the Effects of Black hole Attacks on AODV Routing Protocol in Mobile Ad Hoc Networks”. IEEE International Conference on Pervasive Computing, 2015. 10.1109/PERVASIVE.2015.7087174.
24. Sudheer Kumar, Nitika Vats Doohan. “A modified approach for recognition and eradication of extenuation of gray-hole attack in MANET using AODV routing protocol”. IEEE Symposium on Colossal Data Analysis and Networking, 2016, Pg. 1 – 5.

25. Hizbullah Khattak, Nizamuddin. “A Hybrid Approach for Preventing Black and Gray Hole Attacks in MANET”. 2013 IEEE International Conference on Digital Information Management, 2013, Pg. 55 – 57.
26. Sanjay K. Dhurandher, Isaac Woungang, Raveena Mathur, Prashant Khurana. “GAODV: A Modified AODV against single and collaborative Black Hole attacks in MANETs”. IEEE 27th International Conference on Advanced Information Networking and Applications Workshops, 2013, Pg. 357-362.
27. Vimal Kumar , Rakesh Kumar. “An Adaptive Approach for Detection of Blackhole Attack in Mobile Ad hoc Network”. International Conference on Intelligent Computing, Communication & Convergence, ScienceDirect, Procedia Computer Science 48, 2015, Pg. 472 – 479.
28. Ayesha Siddiqua, Kotari Sridevi, Arshad Ahmad Khan Mohammed. “Preventing Black Hole Attacks in MANETs Using Secure Knowledge Algorithm”. IEEE International Conference on Signal Processing and Communication Engineering Systems, 2015, Pg. 421-425.
29. Usha M K, A.S. Poornima. “Node-To-Node Authentication Protocol to Prevent Black Hole Attack in AODV”. IEEE International Conference on Wireless Communications, Signal Processing and Networking, 2016, Pg. 133 – 136.
30. A. Dorri. "An EDRI-based approach for detecting and eliminating cooperative black hole nodes in MANET". Springer Wireless Network, 2016. 10.1007/s11276-016-1251-x.
31. Dhaval Dave, Pranav Dave. “An effective Black hole attack detection mechanism using Permutation Based Acknowledgement in MANET”. IEEE International conference on advances in computing, communications and informatics, 2014, Pg. 1690–1696.
32. S.V. Vasantha, Dr. A. Damodaram. "Bulwark AODV against Black hole and Gray hole attacks in MANET". IEEE International Conference on Computational Intelligence and Computing Research. 2015, Pg. 1-5.
33. S.V. Vasantha, Dr. A. Damodaram. “Invincible AODV to Detect Black hole and Gray hole attacks in Mobile Ad hoc Networks”. International Journal of Communication Systems - Wiley Online Library, Vol. 31, No. 6, 2018, Pg. 1-19. 10.1002/dac.3518.